Chapter 3

Biological sequence analysis

The steps outlined in the Figure1.1 are discussed in this chapter. Each step contains many sub steps, and there may have different approaches known to tackle the same problem. The methodologies that are used in this thesis for deriving results are described in details.

3.1 Identifying Close and Remote Homologues to the Query

Nature is a tinkerer and not an inventor. New sequences are adapted from pre-existing sequences rather than invented *de novo* (Jacob, 1977). This is very fortunate for computational sequence analysis, since discovery of sequence homology (recognition of significant similarity) to a known protein or family of proteins often provides the first clues about the function of a query sequence (Altschul *et al*., 1990). When the homologue is encountered the information about structure and/function can be transferred to query sequence by *homology*. Homologous proteins are defined as one that shares clear evolutionary relationship (or a common ancestor) with each other, while remote homologues are one in which the evolutionary relationships can not be detected at the first glance (e.g., using sequence similarity) due to divergent evolution. Following flowchart (Figure 3.1) summarizes ways of identification of functional and structural similarity.

Well-curated databases assume prime importance as sequence analysis is totally based upon the quality of the databases. An error in database may progress by repeated copying of annotations between similar sequences. Some of the important publicly available sequence and structure databases are listed below.

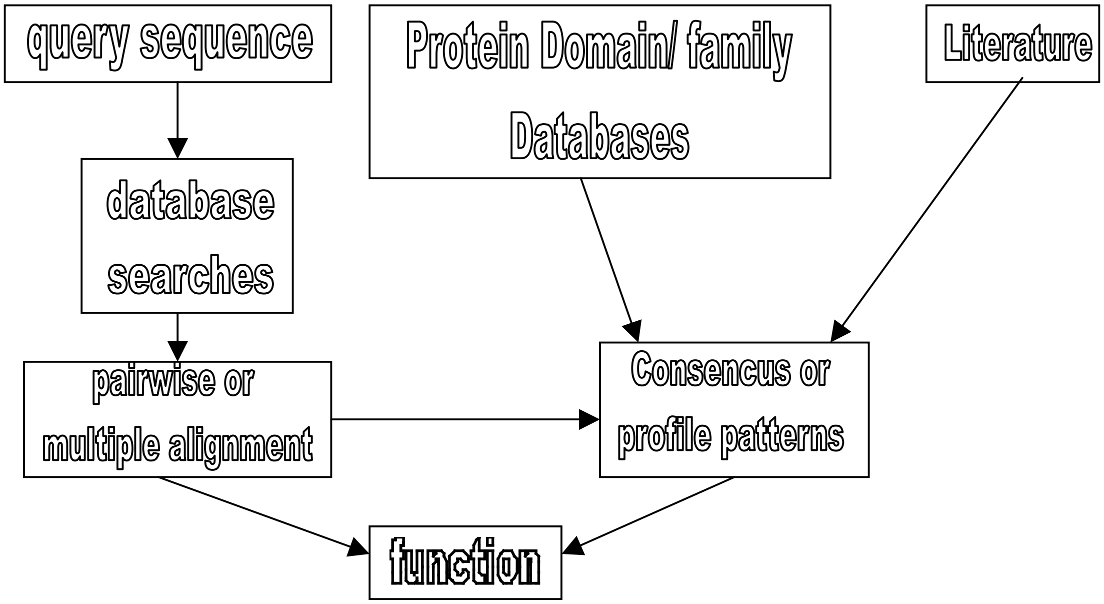
query sequence
Protein Domain/ family Databases
Literature
database searches
pairwise or multiple alignment
Consencus or profile patterns
function

Figure 3.1 Identification of close and remote homolouges of query sequence by searching databases of deposited sequences and profiles. Plese see text for clear discussion.

3.1.1 Protein Sequence Databases

➢ SWISS-PROT	http://www.expasy.ch/sprot
➢ TrEMBL	http://www.expasy.ch/sprot
➢ PIR	http://pir.georgetown.edu
➢ Entrz protein (NRDB)	http://www.ncbi.nlm.nih.gov:80/entrez/
➢ OWL	http://www.leeds.ac.uk/bmb/owl/owl.htm
➢ GenPept	http://www.ncifcrf.gov/pub/genpept/

3.1.1.1 SWISS-PROT (Bairoch and Apweiler, 2000)

The SWISS-PROT database distinguishes itself from other protein sequence databases by three distinct criteria:

Annotation: In SWISS-PROT, as in most other sequence databases, two classes of data can be distinguished. First are the core data and the annotation. For each sequence entry the core data consists of the sequence data; the citation information (bibliographical references) and the taxonomic data (description of the biological source of the protein). The annotation consists of the description of the following items: Function(s) of the protein, post-translational modification(s), domains, secondary structure, quaternary structure, similarities to other proteins, disease(s) associated with deficiencies in the protein, sequence conflicts and variants, etc. Systematic recourse both to publications other than those reporting the core data and to subject referees represents a unique and beneficial feature of SWISS- PROT.

Minimal redundancy: In SWISS-PROT all possible data are merged so as to minimize the redundancy of the database. If conflicts exist between various sequencing reports, they are indicated in the feature table of the corresponding entry.

Integration with other databases: Users of biomolecular databases are provided with a degree of integration between the three types of sequence-related databases (nucleic acid sequences, protein sequences and protein tertiary structures) as well as with specialized data collections. SWISS-PROT is currently cross-referenced with 30 different databases. Cross-references are provided in the form of pointers to information related to SWISS-PROT entries and found in data collections other than SWISS-PROT.

3.1.1.2 TrEMBL (Bairoch and Apweiler, 2000)

It consists of entries in SWISS-PROT-like format derived from the translation of all coding sequences (CDSs) in the EMBL Nucleotide Sequence Database, except the CDSs already included in SWISS-PROT.

3.1.1.3 PIR (Barkar *et al*., 2000)

The Protein Information Resource, is the most comprehensive and expertly annotated protein sequence database in the public domain, aiming to provide timely and high quality annotation and promote database interoperability. PIR employs rule-based and classification-driven procedures based on controlled vocabulary and standard nomenclature and include status tags to distinguish experimentally determined from predicted protein features. The database contains about 200000 non-redundant protein sequences, which are classified into families and superfamilies and their domains and motifs identified. Entries are extensively cross-referenced to other sequence, classification, genome, structure and activity databases. The PIR web site features search engines that use sequence similarity and database annotation to facilitate the analysis and functional identification of proteins.

3.1.1.4 Entrez (on NCBI)

It is a search and retrieval system have been compiled from sources, including SwissProt, PIR, PRF, PDB, and translations from annotated coding regions in GenBank and RefSeq.

3.1.2 Protein Structure Databases

➢ PDB	http://rcsb.org/pdb
➢ NRL3D	http://pir.georgetown.edu/pirwww/dbinfo/nrl3d.html
➢ MODBASE	http://pipe.rockefeller.edu/modbase/index.shtml

3.1.3 Protein Family/ Domain Databases

➢ PFAM	http://www.sanger.ac.uk/Pfam/
➢ PRODOM	http://protein.toulouse.inra.fr/prodom.html
➢ PROSITE	http://www.expasy.ch/sprot/prosite.html

➢ BLOCKS	http://www.blocks.fhcrc.org/
➢ SMART	http://smart.embl-heidelberg.de/
➢ DOMO	http://www.infobiogen.fr/~gracy/domo
➢ PRINTS	http://www.biochem.ucl.ac.uk/bsm/dbbbrowser/PRINTS/ PRINTS.html
➢ PROFILESCAN	http://www.isrec.isb-sib.ch/software/ PFSCAN_form.html

3.1.3.1 PFAM (Bateman *et al.*., 2000)

Pfam is a database of protein domain families. Pfam contains curated multiple sequence alignments for each family, as well as profile hidden Markov models (profile HMMs) for finding these domains in new sequences. Pfam contains functional annotation, literature references and database links for each family. There are two multiple alignments for each Pfam family, the seed alignment that contains a relatively small number of representative members of the family and the full alignment that contains all members in the database that can be detected. All alignments use sequences taken from pfamseq, which is a non-redundant protein set composed of SWISS-PROT and TrEMBL. The profile HMM is built from the seed alignment using the HMMER package (Durbin *et al.*, 1998), which is then used to search the pfamseq sequence database. All the matches found above the curated thresholds are aligned using the profile HMM to make the full alignment. The Pfam WWW servers can present the domain architecture of a protein graphically as 'beads on a string' with a color-coded and hyperlinked bead for each domain. For a fine-grained analysis of the evolution of domain architectures, a Java tool displays the graphical domain schematics of each sequence connected in an evolutionary tree.

3.1.3.2 PRODOM (Corpet *et al.*, 2000)

The rapid growth of primary sequence databases makes it more and more difficult to comprehend the ever increasing diversity of known proteins. One major underlying difficulty is that many proteins exhibit a combinatorial arrangement of domains, which makes it desirable to develop databases and tools to describe proteins at an intermediary level of structure, in terms of domain arrangements. The ProDom database was designed with this explicit purpose, with particular emphasis on the user interface. Domains are detected in an automatic process that uses sequence similarities between homologous domains of SWISS-PROT and TrEMBL (Bairoch and Apweiler, 2000) sequences using PSI-BLAST (Altschul *et al*., 1997). ProDom `domains' thus essentially reflect protein subsequences conserved in various proteins. To increase the number of these expert-validated families, the curated part of Pfam (Bateman *et al*., 2000) is used: the seed alignments of Pfam-A families were added to the list of 21 ProDom expert-validated multiple alignments and used to build new ProDom families with the PSI-BLAST program. An interactive graphical interface is available to allow for easy navigation between schematic domain arrangements, multiple alignments, phylogenetic trees, SWISS-PROT entries, PROSITE patterns (Hoffman *et al*., 1999), Pfam-A families and 3-D structures in the PDB (Bernstein *et al*., 1977; Berman *et al*., 2000). Alignments and trees can be reduced or developed to facilitate the analysis of sequence relationships within large domain families. New sequences can be searched against ProDom and aligned with existing domain families, and modeled on the basis of homologous domains in the PDB.

3.1.3.3 PROSITE (Hoffman *et al*., 1999)

PROSITE is a database of protein families and domains. It is based on the observation that, while there is a huge number of different proteins, most of them can be grouped, on the basis of similarities in their sequences, into a limited number of families. Proteins or protein domains belonging to a particular family generally share functional attributes and are derived from a common ancestor. It is apparent, when studying protein sequence families, that some regions have been better conserved than others during evolution. These regions are generally important for the function of a protein and/or for the maintenance of its three- dimensional structure. By analyzing the constant and variable properties of such groups of similar sequences, it is possible to derive a signature for a protein family or domain, which distinguishes its members from all other unrelated proteins. A biologically significant patterns and profiles formulated in such a way that with appropriate computational tools it can help to determine to which known family of protein (if any) a new sequence belongs, or which known domain(s) it contains. PRINTS (Attwood *et al*., 1999) is a compendium of protein fingerprints. A fingerprint is a group of conserved motifs used to characterize a protein family. Fingerprints can encode protein folds and functionalities more flexibly and powerfully than can single motifs: the database thus provides a useful adjunct to PROSITE.

3.1.4 For Text Based searches in Sequence Databases

➢ Entrez at NCBI	http://www.ncbi.nlm.nih.gov/Entrz/
➢ SRS at EBI	http://srs.ebi.ac.uk/
➢ WWW-QUERY	http://pbil.univ-lyon1.fr/
➢ ExPASy	http://www.expasy.ch/sprot/
➢ DBGET	http://www.genome.ad.jp/

3.1.5 Sequence Alignment and Detection of Similarity

The concept of alignment is crucial for understanding the sequence searching procedures from known databases. The most basic sequence analysis task is to ask if two sequences are related and this is usually done by first aligning the sequences (or parts of them) and then deciding whether that alignment is more likely to have occurred because the sequences are related or just by chance. When sequences are compared, in essence, we are looking for the evidence that they have diverged from a common ancestor by a process of mutation and selection. We look for a series of individual characters or character patterns in the same order in the sequences. The key issues are: (1) what sorts of alignment should be considered; (2) the scoring system used to rank the alignment; (3) the algorithm used to find optimal (or good) scoring alignments; and (4) the statistical methods used to evaluate the significance of an alignment score. The basic mutational processes that are considered are *substitutions*, which change residues in sequences and *insertions* and *deletions*, which add or remove residues. Insertions and deletions are together referred to as *gaps*. The total score assigned to an alignment will be a sum of terms for each aligned pair of residues, provisions for substitutions, plus terms for each gap.

The Scoring System

Since the proteins, in the course of evolution accommodates the substitutions and gaps. It is important to use appropriate substitution matrices while doing sequence alignments. The matrices are simply the prediction of tolerable amino acid changes that might occur to a sequence during the course of evolution. Two major families of matrices are available: (1) Point Accepted Mutation or PAM matrices (Dayhoff *et al*., 1983) and (2) Blocks Amino Acid Substitution Matrices (Henikoff and Henikoff, 1992). The matrices are discussed in detail while talking about statistics of sequence similarity scores. The probability of occurrence of a gap depends upon its length. Thus, when computing an alignment, penalties (P) associated with gaps are often estimated using a linear or "affine" model such as

$$P = \alpha + \beta\phi$$

Where, ϕ is the length of the gap, α the gap opening penalty, and β is the gap extension penalty. The gap opening penalties are higher than the gap extension penalties.

Alignment of Pairs of Sequences

There are two types of sequence alignment, global and local. In global alignment, an attempt is made to align the entire sequence, using as many characters as possible, up to both ends of each sequence. Sequences that are quite similar and approximately the same length are suitable for global alignment. In local alignment, stretches of sequence with the highest density of matches are aligned, thus generating one or more islands of matches or subalignments in the aligned sequences. Local alignments are more suitable for aligning sequences that are similar along some of their lengths but dissimilar in others, sequences that differ in lengths, or sequences that share a conserved region or domain. It is also the most sensitive way to detect similarity when comparing two highly divergent sequences.

Alignment of pairs of sequences can be performed using: (1) Dot matrix analysis (Gibbs and McIntyre, 1970); (2) Dynamic programming algorithms for global (Needleman and Wunsch, 1970), and local alignment (Smith and Waterman, 1981); and (3) Word or k-tuple methods, as used by BLAST (Altschul *et al*., 1990).

Unless the sequences are known to be very much alike, the **dot matrix** method should be used first. This method displays any possible sequence alignments as diagonals on the matrix. Dot matrix analysis can readily reveal the presence of gaps and direct and inverted repeats that are more difficult to find by other methods.

The dynamic programming method, first used for global alignment of the sequences (Needleman and Wunsch, 1970) and subsequently for local alignment (Smith and Waterman, 1981) provides one or more alignments of the sequences. An alignment is generated by starting at the ends of the two sequences and attempting to match all possible pairs of characters between the sequences and following a scoring scheme (as described before; using substitution matrix and gap penalties) for matches, mismatches and gaps. This procedure generates a matrix of numbers that represents all possible alignments between two sequences. The highest set of sequential scores in the matrix defines an optimal alignment. The dynamic programming is guaranteed in a mathematical sense to provide the optimal or highest scoring alignment for user defined variables, including the substitution matrix and gap penalties.

3.1.5.1 Global Alignment: Needleman-Wunsch Algorithm:

As the name suggests Needleman and Wunsch suggested the first global alignment algorithm in 1970. A more efficient version of the algorithm was introduced by Gotoh in 1982. The later version is described here.

A matrix F indexed by *i* and *j*, is constructed. Where *i* and *j* are index for each sequence. The value F (i, j) is the score of the best alignment the initial segment $\chi_{1\dots i}$ of χ up to χ_i and initial segment $\gamma_{1\dots j}$ of γ up to γ_{j}. F(i, j) is built recursively. One can start by initializing F (0,0) = 0. Then proceed to fill the matrix from top left to bottom right (or from bottom right to top left). If F $(i-1, j-1)$, F $(i-1, j)$ and F $(i, j-1)$ are known, it is possible to calculate F (i, j). There are three possible ways that the best score F(i, j) of an alignment up to χ_i, γ_j could be obtained: χ_i could be aligned to γ_j in which case F$(i, j) =$ F$(i-1, j-1) +$ s (χ_i, γ_j); or χ_i is aligned to gap, in which case F$(i, j) =$ F$(i-1, j)$ – P; or γ_j is aligned to gap, in which case F$(i, j) =$ F$(i, j-1)$ –P. Here s (χ_i, γ_j) is the local score of the previous step and d is the gap penalty, which can be of the format described before. The best score up to (i, j) will be largest of the three points.

There fore, we have

$$F(i,j) = \max \{ F(i-1,j-1) + s(\chi_i, \gamma_j),\quad F(i-1,j) - P,\quad F(i, j-1) - P \}$$

The above equation is applied repeatedly to fill the matrix F(i, j) values, calculating the value in the bottom right-hand corner of each sequence to the top-left. As one fill in the F(i, j) values, the pointer is kept in each cell back to the cell from which its F(i, j) is derived. The boundary conditions are calculated as follows. Along the top row, where $j = 0$, the values F$(i, j-1)$ and F$(i-1, j-1)$ are not defined, so the values F$(i, 0)$ must be handled specially. The value F$(i, 0)$ represent alignments of a prefix of χ to all the gaps in γ, so we can define F$(i, 0) = -i$P. Likewise down the left column F $(0, j) = -j$P. The value in the final cell of matrix, F (n, m), is by definition the best score for a alignment of $\chi_{1\dots n}$ to $\gamma_{1\dots m}$, which is the score of the best global alignment of χ and γ. To find the alignment itself, one should find the path of choices that lead to the final highest score using the pointers. The procedure for doing this is known as *traceback*. It works by building alignment in reverse.

3.1.5.2. Local Alignment: Smith-Waterman Algorithm:

Local alignment arises when say for example one is looking for the best alignment between subsequences of χ, γ. The highest scoring alignment of subsequences of χ and γ is called the best local alignment. The algorithm of local alignment is closely related to that described for global alignment. There are two differences. First, in each cell in the previous set of equation extra possibility is added, allowing F (i, j) to take the value 0 if all other options have value less than 0:

$$F(i,j) = \max \begin{cases} 0, \\ F(i-1, j-1) + s(\chi_i, \gamma_j), \\ F(i-1, j) - P, \\ F(i, j-1) - P \end{cases}$$

Taking option 0 corresponds to starting a new alignment. As a result of it the boundary values of top row and left column will be 0 and not $-i$P and $-j$P respectively.

The second change is that the alignment can start anywhere in the matrix, so instead of taking the value in the bottom right corner, F (n, m), for the best score, one have to look for the highest value of F(i, j) over the whole matrix, and start *traceback* from there. The *traceback* ends when the cell with 0 value is encountered.

3.1.6 The Blast Algorithm For Searching Databases

Database Searching Programs

➢ BLAST	http://www.ncbi.nlm.nih.gov/BLAST/
➢ PSI-BLAST	http://www.ncbi.nlm.nih.gov/BLAST/
➢ FASTA3	http://www.ebi.ac.uk/fasta3/
➢ HMMER	http://hmmer.wustal.edu/
➢ SAM	http://www.cse.ucsc.edu/research/compbio/sam.html
➢ PFSEARCH	http://www.isrec.isb-sib.ch/ftp-server/pftools/pft2.2/

- IMPALA http://bioinformatics.weizmann.ac.il/blocks/impala.html

Sequence searches algorithms like FASTA and BLAST use the word or K-tuple methods. They align two sequences very quickly, by first searching for identical short stretches sequences (called word or k-tuples) and then by joining words in to alignment by the dynamic programming method. These methods are fast and suitable for searching an entire database for the sequences that align best with the query sequence. The FASTA and BLAST methods are heuristic and use feedback to improve performance.

3.1.6.1 The Statistics of Sequence Similarity Scores

To assess whether a given alignment constitutes evidence for homology, it helps to know how strong an alignment can be expected from chance alone. In this context, "chance" can mean the comparison of (i) real but non-homologous sequences; (ii) real sequences that are shuffled to preserve compositional properties (Fitch, 1983; Lipman *et al*., 1984; Altschul, 1985) or (iii) sequences that are generated randomly based upon a DNA or protein sequence model. Analytic statistical results invariably use the last of these definitions of chance, while empirical results based on simulation and curve fitting may use any of the definitions.

3.1.6.2 The statistics of local sequence comparison (BLAST)

Statistics for the scores of local alignments, unlike those of global alignments, are well understood. This is particularly true for local alignments lacking gaps, which we will consider first. Such alignments were precisely those sought by the original Basic Local Alignment Search Tool (BLAST) database search programs (Altschul *et al*., 1990). A local alignment without gaps consists simply of a pair of equal length segments, one from each of the two sequences being compared. A modification of the Smith-Waterman (Smith and Waterman, 1981) or Sellers (Sellers, 1984) algorithms will find all segment pairs whose scores can not be improved by extension or trimming. These are called high-scoring segment pairs or HSPs. To analyze how high a score is likely to arise by chance, a model of random sequences is needed. For proteins, the simplest model chooses the amino acid residues in a sequence independently, with specific background probabilities for the various residues. Additionally, the expected score for aligning a random pair of amino acid is required to be negative. Where this not the case, long alignments would tend to have high score independently of whether the segments aligned were related, and the statistical theory would break down.

Just as the sum of a large number of independent identically distributed (i.i.d) random variables tends to a normal distribution, the maximum of a large number of i.i.d. random variables tends to an extreme value distribution (Gumble, 1958). (We will elide the many technical points required to make this statement rigorous.) In studying optimal local sequence alignments, we are essentially dealing with the latter case (Karlin and Altschul, 1990; Dembo *et al.*, 1994). In the limit of sufficiently large sequence lengths m and n, the statistics of HSP scores are characterized by two parameters, K and λ. Most simply, the expected number of HSPs with score at least S is given by the formula

$$E = K\ mn\ e^{-\lambda S} \qquad (1)$$

We call this the **E-value** for the score S. This formula makes eminently intuitive sense. Doubling the length of either sequence should double the number of HSPs attaining a given score. Also, for an HSP to attain the score 2x it must attain the score x twice in a row, so one expects E to decrease exponentially with score. The parameters K and λ can be thought of simply as natural scales for the search space size and the scoring system respectively.

3.1.6.3 Bit scores

Raw scores have little meaning without detailed knowledge of the scoring system used, or more simply its statistical parameters K and λ.

$$S' = \lambda S - \ln K / \ln 2 \qquad (2)$$

Using above equation one attains a "bit score" S', which has a standard set of units. The E-value corresponding to a given bit score is simply

$$E = mn\ 2^{-S'} \qquad (3)$$

Bit scores subsume the statistical essence of the scoring system employed, so that to calculate significance one needs to know in addition only the size of the search space.

3.1.6.4 P-values

The number of random HSPs with score ≥ S is described by a Poisson distribution (Karlin and Altschul, 1990; Dembo *et al*., 1994). This means that the probability of finding exactly a HSPs with score ≥S is given by

$$e^{-\mathbf{E}} * \mathrm{E}^{\mathbf{a}} / \mathrm{a}! \qquad (4)$$

where E is the E-value of S given by equation (1) above. Specifically the chance of finding zero HSPs with score ≥S is e^{-E}, so the probability of finding at least one such HSP is

$$\mathrm{P} = 1 - e^{-\mathbf{E}} \qquad (5)$$

This is the P-value associated with the score S. For example, if one expects to find three HSPs with score ≥ S, the probability of finding at least one is 0.95. The BLAST programs report E-value rather than P-values because it is easier to understand the difference between, for example, E-value of 5 and 10 than P-values of 0.993 and 0.99995. However, when $E < 0.01$, P-values and E-value are nearly identical.

3.1.6.5 Database searches

The E-value of equation (1) applies to the comparison of two proteins of lengths m and n. How does one assess the significance of an alignment that arises from the comparison of a protein of length m to a database containing many different proteins, of varying lengths? One view is that all proteins in the database are a priori equally likely to be related to the query. This implies that a low E-value for an alignment involving a short database sequence should carry the same weight as a low E-value for an alignment involving a long database sequence. To calculate a "database search" E-value, one simply multiplies the pairwise-comparison E-value by the number of sequences in the database.

An alternative view is that a query is *a priori* more likely to be related to a long than to a short sequence, because long sequences are often composed of multiple distinct domains. If we assume the *a priori* chance of relatedness is proportional to sequence length, then the pairwise E-value involving a database sequence of length n should be multiplied by N/n, where N is the total length of the database in residues. Examining equation (1), this can be accomplished by treating the database as a single long sequence of length N.

The BLAST programs (Smith *et al*., 1985; Collins *et al*., 1988; Altschul *et al*., 1990; Mott, 1992; Waterman and Vingron, 1994; Altschul and Gish, 1996; Altschul *et al*., 1997; Pearson, 1998) take this approach to calculating database E-value.

3.1.6.6 The Statistics of Gapped Alignment:

The statistics developed above have a solid theoretical foundation only for local alignments that are not permitted to have gaps. However, many computational experiments (Altschul and Gish, 1996; Altschul *et al*., 1997; and some analytic results (Arratia and Waterman, 1994) strongly suggest that the same theory applies as well to gapped alignments. For ungapped alignments, the statistical parameters can be calculated, using analytic formulas, from the substitution scores and the background residue frequencies of the sequences being compared. For gapped alignments, these parameters must be estimated from a large-scale comparison of "random" sequences. The BLAST programs achieve much of their speed by avoiding the calculation of optimal alignment scores for all but a handful of unrelated sequences. The must therefore rely upon a pre-estimation of the parameters lambda and K, for a selected set of substitution matrices and gap costs. This estimation could be done using real sequences, but has instead relied upon a random sequence model (Altschul and Gish, 1996), which appears to yield fairly accurate result (Pearson, 1998). The BLAST programs also correct for Edge effects (Altschul and Gish, 1996).

3.1.6.7 The choice of substitution scores

The results a local alignment program produces depend strongly upon the scores it uses. No single scoring scheme is best for all purposes, and an understanding of the basic theory of local alignment scores can improve the sensitivity of one's sequence analyses. A large number of different amino acid substitution scores, based upon a variety of rationales, have been described (Dayhoff *et al*., 1978; Altschul, 1991; Gonnet *et al*., 1992; Henikoff and Henikoff, 1992). However the scores of any substitution matrix with negative expected score can be written uniquely in the form

$$S_{ij} = (\ln q_{ij} / p_i\, p_j) \setminus \lambda \qquad (6)$$

Where, the q_{ij}, called target frequencies, are positive numbers that sum to 1, the p_i are background frequencies for the various residues, and λ is a positive constant (Karlin and Altschul, 1990; Altschul, 1991). The λ here is identical to the λ of equation (1). Multiplying all the scores in a substitution matrix by a positive constant does not change their essence: an alignment that was optimal using the original scores remains optimal. Such multiplication alters the parameter lambda but not the target frequencies $q_{ij.}$ Thus, up to a constant scaling factor, every substitution matrix is uniquely determined by its target frequencies. These frequencies have a special significance (Karlin and Altschul, 1990; Altschul, 1991): A given class of alignments is best distinguished from chance by the substitution matrix whose target frequencies characterize the class. The most direct way to construct appropriate substitution matrices for local sequence comparison is to estimate target and background frequencies, and calculate the corresponding log-odds scores of formula (6). These frequencies in general can not be derived from first principles, and their estimation requires empirical input.

3.1.6.8 The PAM and BLOSUM amino acid substitution matrices

While all substitution matrices are implicitly of log-odds form, the first explicit construction using formula (6) was by Dayhoff and coworkers (Dayhoff *et al*., 1978; Schwartz *et al*., 1978). From a study of observed residue replacements in closely related proteins, they constructed the PAM (point accepted mutation) model of molecular evolution. An alternative approach to estimating target frequencies, and the corresponding log-odds matrices, has been advanced by Henikoff and Henikoff (Henikoff and Henikoff, 1992). They examine multiple alignments of distantly related protein regions directly, rather than extrapolate from closely related sequences. An advantage of this approach is that it cleaves closer to observation; a disadvantage is that it yields no evolutionary model. A number of tests (Pearson, 1995; Henikoff and Henikoff, 1993) suggest that the BLOSUM matrices (Blocks Substitution Matrix derived using BLOCKS database) produced by this method generally are superior to the PAM matrices for detecting biological relationships. BLOSUM62 is default matrix for blast searches.

3.1.6.9 Gap scores and Low Complexity Regions

The theoretical development concerning the optimality of matrices constructed using equation (6) unfortunately is invalid as soon as gaps and associated gap scores are introduced, and no more general theory is available to take its place. However, if the gap scores employed are sufficiently large, one can expect that the optimal substitution scores for a given application will not change substantially. In practice, the same substitution scores have been applied fruitfully to local alignments both with and without gaps. Appropriate gap scores have been selected over the years by trial and error (Pearson, 1995), and most alignment programs will have a default set of gap scores to go with a default set of substitution scores. No clear theoretical guidance can be given, but "affine gap scores" (Gotoh, 1982; Fitch and Smith, 1983; Altschul and Erickson, 1986) with a large penalty for opening a gap and a much smaller one for extending it, have generally proved among the most effective. The BLAST programs employ the SEG algorithm (Wootton and Federhen, 1993) to filter low complexity regions from proteins before executing a database search.

3.1.7 Database Searching with PSI-BLAST

Many functionally and evolutionarily important protein similarities are recognizable only through comparison of three-dimensional structures (Holm and Sander, 1997; Brenner *et al*., 1998). When such structures are not available, patterns of conservation identified from the alignment of related sequences can aid the recognition of distant similarities.

There is a large literature on the definition and construction of these patterns, which have been variously called motifs, profiles, position-specific score matrices, and Hidden Markov Models (Gribskov, 1987; Staden, 1988; Tatusov *et al.*, 1994; Altschul and Gish, 1996; Altschul *et al.*, 1997; Durbin *et al.*, 1998). In essence, for each position in the derived pattern, every amino acid is assigned a score. If a residue is highly conserved at a particular position, that residue is assigned a high positive score, and others are assigned high negative scores. At weakly conserved positions, all residues receive scores near zero. Position-specific scores can also be assigned to potential insertions and deletions (Gribskov *et al.*, 1987; Altschul *et al.*, 1997; Durbin *et al.*, 1998). The power of profile methods can be further enhanced through iteration of the search procedure (Gribskov, 1992; Tatusov, 1994; Yi and Lander, 1994; Altschul *et al.*, 1997). After a profile is run against a database, new similar sequences can be detected. A new multiple alignment, which includes these sequences, can be constructed, a new profile abstracted, and a new database search performed. The procedure can be iterated as often as desired or until the search converges, when no new statistically significant sequences are detected.

3.1.7.1 The design of PSI-BLAST

Iterated profile search methods have led to biologically important observations but, for many years, were quite slow and generally did not provide precise means for evaluating the significance of their results. This limited their utility for systematic mining of the protein databases. The principal design goals in developing the Position-Specific Iterated BLAST (PSI-BLAST) program (Altschul *et al.*, 1997) were speed, simplicity and automatic operation. The procedure PSI-BLAST uses can be summarized in five steps: (1) PSI-BLAST takes as an input a single protein sequence and compares it to a protein database, using the gapped BLAST program (Altschul *et al.*, 1997). (2) The program constructs a multiple alignment, and then a profile, from any significant local alignments found. The original query sequence serves as a template for the multiple alignment and profile, whose lengths are identical to that of the query. Different numbers of sequences can be aligned in different template positions. (3) The profile is compared to the protein database, again seeking local alignments. After a few minor modifications, the BLAST algorithm (Altschul *et al*., 1997; Altschul *et al*., 1990) can be used for this directly. (4) PSI-BLAST estimates the statistical significance of the local alignments found. Because profile substitution scores are constructed to a fixed scale (Karlin and Altschul, 1990), and gap scores remain independent of position, the statistical theory and parameters for gapped BLAST alignments (Altschu and Gish, 1994) remain applicable to profile alignments (Altschul *et al*., 1997). (5) Finally, PSI-BLAST iterates, by returning to step (2), an arbitrary number of times or until convergence. Profile-alignment statistics allow PSI-BLAST to proceed as a natural extension of BLAST; the results produced in iterative search steps are comparable to those produced from the first pass. Unlike most profile-based search methods, PSI-BLAST runs as one program, starting with a single protein sequence, and the intermediate steps of multiple alignment and profile construction are invisible to the user.

3.1.7.2 Estimation of statistical parameters for local alignment scores

As discussed previously, computation experiments strongly suggest that the optimal gapped local alignment scores produced by the Smith-Waterman algorithm (Smith and Waterman, 1981) and approximated by FASTA (Pearson and Lipman, 1988) or Gapped BLAST (Waterman and Vingron, 1994; Altschul and Gish, 1996) follow an extreme value distribution (Gumble, 1958). Specifically, the probability that the optimal score S from the comparison of unrelated proteins is at least x is given by the equation,

$$P\ (S \geq X) = 1 - \exp\ (-K\ mn\ e^{-\lambda x}) \qquad (1)$$

Where, K and λ are statistical parameters dependent upon the scoring system and the background amino acid frequencies of the sequences being compared. BLAST estimates parameters beforehand for specific scoring schemes by comparing many random sequences generated using a standard protein amino acid composition (Robinson and Robinson, 1991). For example, using BLOSUM-62 amino acid substitution scores (Henikoff and Henikoff, 1992), and affine gap costs (Fitch and Smith, 1983; Altschul and Erickson, 1986; Myers and Miller, 1988) in which a gap of length k is assigned a score of -(10 + k), 10,000 pairs of length-1000 random protein sequences were generated, and Smith-Waterman algorithm was used to calculate 10,000 optimal local alignment scores.

From these scores, λ was estimated at 0.252 and K at 0.035 by the method of maximum-likelihood (Lawless, 1982). In general, given M samples from an extreme value distribution, the ratio of the maximum-likelihood estimate of lambda to its actual value is approximately normally distributed, with mean 1.0 and standard deviation 0.78/sqrt(M) (Lawless, 1982). Thus the standard error for our estimate of λ is about 0.002, or less than 1%. The chi-squared goodness-of-fit test for these data, with 34 degrees of freedom, is 25.6, which is lower than would be expected to occur by chance 87% of the time even were the theory precisely valid.

3.1.7.3 Generalization to PSI-BLAST alignment scores

In order for PSI-BLAST to iterate automatically, it needs to be able to generate accurate estimates of the statistical significance of the alignments it produces. Unfortunately, there is no analytic theory with which to estimate the statistical significance of a gapped local alignment of a profile and a simple sequence. One hope is that if amino acid scores within each column of a PSI-BLAST profile can be constructed to the same scale (Karlin and Altschul, 1990; Altschul, 1991) i.e. with the same ungapped λ, as those for a standard amino acid substitution matrix, and then use the same position-independent gap costs, the same gapped λ may result. To review, for ungapped local alignments, any substitution matrix takes the form

$$S_{\mathbf{ij}} = (\ln q_{\mathbf{ij}} / p_{\mathbf{i}}\, p_{\mathbf{j}})\, \lambda_{\mathbf{u}} \qquad (2)$$

Where, the q_{ij} are the target frequencies for aligned pairs of amino acids, the p_i are background frequencies, and the subscript for λ indicates it is the statistical parameter for ungapped local alignments scale (Karlin and Altschul, 1990; Altschul, 1991). For a PSI-BLAST profile (Altschul *et al.*, 1997), each column has its own unique set of amino acid target frequencies q_i. Following (2), the amino acid scores for this column may be constructed to the same scale by using the formula

$$S_{\mathbf{i}} = (\ln q_{\mathbf{i}} / p_{\mathbf{i}}) / \lambda_{\mathbf{u}} \qquad (3)$$

The hope is that, given a specific set of gap costs, the gapped λ for the PSI-BLAST profile will be the same as the gapped λ for the standard substitution matrix, which may be calculated in advance.

3.1.8 Multiple Alignment using CLUSTAL

CLUSTAL has been written and subsequently improved during the span of last ten years (Higgins and Sharp, 1988; Thompson *et al*., 1994a; Higgins *et al*., 1996). CLUSTAL performs a global multiple alignment using following steps: (1) Perform pairwise alignment of all the sequences; (2) use the alignment scores to produce the phylogenetic tree (see later); and (3) align the sequences sequentially, guided by the phylogenetic relationships indicated by the tree. Thus, the most closely related sequences are aligned first, and then additional sequences and groups of sequences are added, guided by the initial alignments to produce a multiple sequence alignment. The quality of the alignments produced in such way is excellent, as judged by the ability to correctly align corresponding domains from sequences of known secondary or tertiary structure. The initial alignments used to produce the guide tree may be obtained by a fast k-tuple or pattern finding approach similar to BLAST that is useful for many sequences, or a slower, dynamic programming method may be used. An enhanced dynamic programming alignment algorithm (Myers and Miller, 1988) is used to obtain optimal alignment scores. For producing a phylogenetic tree, genetic distances between the sequences are required. The genetic distance is the number of mismatched positions in an alignment divided by the total number of matched positions (positions opposite to gaps are not scored).

The recent version is CLUSTALW (Thompson *et al*., 1994) with the W standing for "weighing" represent the ability of the program to provide weights to sequence and program parameters. The sensitivity of the commonly used progressive multiple sequence alignment (CLUSTALV) method has been greatly improved for the alignment of divergent protein sequences using following steps. Firstly, individual weights are assigned to each sequence in a partial alignment in order to downweight near-duplicate sequences and upweight the most divergent ones. Secondly, amino acid substitution matrices are varied at different alignment stages according to the divergence of the sequences to be aligned. Thirdly, residue specific gap penalties and locally reduced gap penalties in hydrophilic regions encourage new gaps in potential loop regions rather than regular secondary structure. Fourthly, positions in early alignments where gaps have been opened receive locally reduced gap penalties to encourage the opening up of new gaps at these positions.

The CLUSTALX (Thompson *et al*., 1997) is graphic interface to CLUSTALW. CLUSTALX is new windows interface for the widely used progressive multiple sequence alignment program CLUSTALW. It is easy to use, providing an integrated system for performing multiple sequence and profile alignments and analyzing the results. CLUSTALX displays the sequence alignment in a window on the screen. A versatile sequence coloring scheme allows the user to highlight conserved features in the alignment. Pull-down menus provide all the options required for traditional multiple sequence and profile alignment. New features include: the ability to cut-and-paste sequences to change the order of the alignment, selection of a subset of the sequences to be realigned, and selection of a sub-range of the alignment to be realigned and inserted back into the original alignment. Alignment quality analysis can be performed and low-scoring segments or exceptional residues can be highlighted. Quality analysis and realignment of selected residue ranges provide the user with a powerful tool to improve and refine difficult alignments and to trap errors in input sequences.

3.1.9 Literature

Searching for literature can be of prime importance for a computational biologist. It is equally important for biologists working in all areas of research to stay acquinted with the latest development in the field. The Literature can be searched over the web in PubMed. PubMed is a project developed by the National Center for Biotechnology Information (NCBI). It has been developed in conjunction with publishers of biomedical literature as a search tool for accessing literature citations and linking to full-text journals at Web sites of participating publishers. The PubMed is available at NCBI web site at http://www.ncbi.nlm.nih.gov/.

3.1.10 Uses of Patterns

Patterns, searched using family alignment databases or multiple sequence alignments, are used to describe the residues that are conserved in a set of sequences. Discovering patterns conserved in a protein family can help in the understanding between sequence, structure and function of the protein under study. When a conserved pattern is discovered, one should analyze how likely it is that pattern has been discovered by chance. The less likely this is, the more likely the pattern is to describe functionally or structurally conserved residues.

If one finds a pattern that not only is conserved in the family, but also is unique to the family, i.e., no (or few) sequences outside the family matches the pattern, then pattern can be used to identify new members of the family. The PROSITE database (Hoffman *et al*., 1999) of protein sites and families illustrates this. The patterns in PROSITE can be used not only for finding out structurally and functionally important residues but also for classification purposes for removing false family members.

3.2 Identification of related structures

PSI-BLAST (Altschul *et al*., 1997) or its relatives has been the best sequence (or homology) searching. Probabilistic or Bayesian models also have been applied (e.g., hidden markov models; Durbin *et al*, 1998) for detection of remote homologues. If structure level similarity in terms of PDB hit(s) is suggested by sequence searching methods, one can straight forward transfer information by homology or can set stage for homology modeling (step 3) for more refined function prediction. But in case that the sequence searches doesn't arrive at any useful hits once can resolve for secondary structure prediction or fold recognition methods for identifying the related structures in fold library.

3.2.1 Secondary Structure Prediction

3.2.1.1 History and General Comments

In one of the earliest studies involved in the analysis of helix content in proteins by optical rotatory dispersion, Szent -Gyorgyi and Cohen (1957) showed that proteins with high proline content also exhibit less helicity. Thus, this established the idea of proline as, in some sense, a helix breaker. Cook in 1967 has given some early rules for helix formation, using then available structures and chemical properties of residues. Some of them are (1) Ala, Val and Leu are the helix formers and they tend to occur in the middle of helix. (2) The size of the side chain of a helix-forming residue is important. (3) Residues Asp, Asn and phe are helix breaking. (4) Asp, Glu, and Thr favor N-termini of α-helical region. (5) Lys, His and Arg prefers the C-termini of α-helical regions.

As observations the above rules were good and that started the search for more sophisticated rules. The x-ray determined structures of 15 proteins were examined by Chou and Fasman (1974a) and the number of occurrence of a given amino acid in the α helix, β sheet and coil was tabulated. From this, the conformational parameters (propensities) for each amino acid within a protein, its occurrence in a given type of secondary structure, and the fraction of residues occurring in that type of structure. The residue preferences found by Chou and Fasman has been quite accurate and has been discussed before while discussing about properties of amino acid side chains. Having computed the propensities Chou and Fasman derived the rules for secondary structure prediction. This rules, when applied then resulted in 70-80% predictive accuracy. However now that accuracy is predicted to be around 50% only. This was the first attempt to apply statistical methods for secondary structure prediction. With this Chou and Fasman has unknowingly set a trend to do a three-state prediction for a given sequence. The GORIII method (Garnier *et al.*, 1978; Gibrat *et al.*, 1987) is a representative of the methods based not only on single residue propensities but also on statistically significant pairwise residue interactions. The preference (information content) I of a residue with sequence number j and residue type Rj for a secondary structure type Z $\in$ {helix, sheet, coil} is approximated as

$\mathrm{I}\ (\mathrm{S}_j = \mathrm{Z};\ \mathrm{R}_{j-8},\ldots,\ \mathrm{R}_{j+8}) = \sum \mathrm{I}\ (\mathrm{S}_j\ \mathrm{Z};\ \mathrm{R}_{j+m} / \mathrm{R}_j)$, where Σ runs from -8 to 8.

in a sequence environment of eight residues on either side of a central one. The information I carried by the amino acid pair ($\mathrm{R}_{j+m} / \mathrm{R}_j$) on the occurrence of the event Z (adoption of a specific secondary structure state) is defined as

$\mathrm{I}\ (\mathrm{S}_j = \mathrm{Z};\ \mathrm{R}_{j+m} / \mathrm{R}_j) = \log \left[\mathrm{P}\ \left(\mathrm{Z} / (\mathrm{R}_{j+m} / \mathrm{R}_j)\right) / \mathrm{P}\ (\mathrm{Z})\right]$

Where, P denotes the conditional probability. The enormous amount of parameters (3 structural states × 20 amino acid types × 20 amino acid types × 17 sequence positions) is estimated from a set of 68 non-redundant protein crystallographic structures. The prediction accuracy achieved was about 63% then (Gibrat *et al.*, 1987; Garnier and Levin, 1991). A further improvement of 2.5 to 6.5% (Biou *et al.*, 1988) was obtained by combining GORIII method with two other prediction schemes. First based on hydrophobicity patterns that are observed in regular secondary structures (bit pattern method, and second using structural similarity between short, sequentially homologous peptides (Levin and Garnier, 1988). It is important to note that the predictive power of methods relying on only sequentially local structure information is limited by about 65% (Gibrat *et al*., 1991). A further increase requires the consideration of tertiary interactions.

3.2.1.2 Importance of Evolutionary information

One of the most successful applications of the multiple sequence alignment has been to improve the accuracy of secondary structure prediction. This has been first used by Zvelebil *et al*., (1987) and subsequently used by Levin *et al*., (1993); Rost and Sander (1993); Salamov *et al*., (1995); Cuff *et al*., (1999) and others for reaching an overall three state prediction accuracy more than 70%. It is around 9% more than single sequence based methods. Some of the methods use multi neuron neural networks and jury of neural network to give three-state prediction.

It is well known that the structure is more conserved than sequences (Chothia and Lesk, 1986; Pastore and Lesk, 1990). What we see in alignment of native proteins is a record of the evolution. If proteins share more than 30% identity most likely they share same fold (Chothia and Lesk, 1986). Of course, not any two residues can be exchanged. On the contrary, the pattern of residue substitutions within one structure family contains specific information about the structure. Gaps in multiple alignments occur more often in loop regions than in regular secondary structure elements such as helix and strand (Pascarella and Argos, 1992). This implies that the number of gaps at a particular position carries information about secondary structure: the more gaps found in a region, the more likely it is a loop region (provided the alignment is correct).

Although secondary structure alone can is generally of limited use, it is nonetheless helpful to be able to refer to a reliable secondary-structure prediction to predict the tertiary structure by fold recognition or motif searches and secondary structure based threading. The following structural clues can sometimes be obtained through inspection of predicted secondary structural elements:

- The structural class of target proteins may be ascertained (all α, all β, or α-β)
- Structural repeats can be detected. By identifying a repeating sequence of secondary structures, it is sometimes possible to identify repeated domains in the target proteins.
- The sequence of secondary structural elements can be compared to the folds matched by fold recognition. For the fold-recognition methods, which do not use predicted secondary structure, this "second opinion" is of great value in determining the degree of confidence to assign to the prediction.

Online servers available for Secondary Structure Prediction

➢ GOR IV	http://npsa-pbil.ibcp.fr/cgi-bin/npsa_automat.pl?page=npsa_gor4.html
➢ PREDATOR	http://www.embl-heidelberg.de/cgi/predator_serv.pl
➢ PHDsec	http://cubic.bioc.columbia.edu/predictprotein/
➢ JPRED	http://jura.ebi.ac.uk/
➢ NNPREDICT	http://www.cmpharm.ucsf.edu/~nomi/nnpredict.html
➢ PSIPRED	http://insulin.brunel.ac.uk/psiform.html

3.2.1.3 Secondary Structure Prediction Using Tertiary Interactions

PREDATOR (Frishman and Argos, 1997) is a secondary structure prediction program. It takes as input a single protein sequence to be predicted and can optimally use a set of unaligned sequences as additional information to predict the query sequence. The principal step in the procedure involves generation of seven secondary structural propensities for input sequences and the related sequences. Three propensities are based on long-range interactions involving potential hydrogen bonding resides in antiparallel (P_1) and parallel (P_2) β strands as well as α-helices (P_3). Three further propensities for helix (P_4), strand (P_5) and coil (P_6) rely on the similarity of the sequence segments to be predicted with those of known conformation (nearest neighbor approach; Zhang *et al*., 1992). Finally a statistically based turn propensity (P_7) is used over a four-residue window (Hutchinson and Thornton, 1994). The mean prediction accuracy of

PREDATOR is 68% for a single sequence and 75% for a set of related sequences. PREDATOR does not use multiple sequence alignment. Instead, it relies on careful pairwise local alignments of the sequences in the set with the query sequence to be predicted.

3.2.1.4 Prediction with Neural Networks

A neural network mimics the architecture of brain neurons. Since 1958, when psychologist Frank Rosenblatt proposed the "Perceptron," a pattern recognition device with learning capabilities, the hierarchical neural network has been the most widely studied form of network structure. A hierarchical neural network is one that links multiple neurons together hierarchically. The special characteristic of this type of network is its simple dynamics. That is, when a signal is input into the input layer, it is propagated to the next layer by the interconnections between the neurons. Simple processing is performed on this signal by the neurons of the receiving layer prior to its being propagated on to the next layer. This process is repeated until the signal reaches the output layer completing the processing process for that signal. The manner in which the various neurons in the intermediary (hidden) layers process the input signal will determine the kind of output signal it becomes (how it is transformed). As you can see, then, hierarchical network dynamics are determined by the weight and threshold parameters of each of their units. If input signals can be transformed to the proper output signals by adjusting these values (parameters), then hierarchical networks can be used effectively to perform information processing.

Since it is difficult to accurately determine multiple parameter values, a learning method is employed. This involves creating a network that randomly determines parameter values. This network is then used to carry out input-to-output transformations for actual problems. The correct final parameters are obtained by properly modifying the parameters in accordance with the errors that the network makes in the process. Error back-propagation learning method has played a major role in the recent neural network computing boom. The back-propagation paradigm has been tested in numerous applications including bond rating, mortgage application evaluation, protein structure determination, backgammon playing, and handwritten digit recognition.

Qian and Sejnowski (1988) presented a neural network method for prediction of secondary structures for single protein sequences using supervised learning method and back-propagation. They trained a standard network with 13 input groups, with 21 units/group using 106 protein structures and different window lengths of 1-21 residues. They achieved a success rate of 64.3% for three-state prediction. This is substantially better than the prediction from statistical methods described before. This however, opened a way for next generation secondary structure prediction methods, as described below.

3.2.1.5 Prediction with Neural Networks and Multiple Alignments:

3.2.1.5.1 PHD Secondary Structure Prediction Method:

PHD is made of three individual prediction methods that use evolutionary information as input to predict secondary structure (PHDsec; Rost and Sander, 1993a,b; 1994a), relative solvent accessibility (PHDacc; 1994b) and transmembrane helices (PHDhtm; Rost *et al*., 1995). Presently it is available on predict protein server. The method consists of following steps.

Generating Multiple Alignment

First step in a PHD prediction is to search for remote homologues from PRODOM domain database using SAM-T98 (Karplus *et al*., 1997). The pairwise profile-based alignment is generated using the program MaxHom (Sander and Schneider, 1991).

Multiple level of Computation

The PHD methods process the input information on multiple levels. The first level is a feed-forward neural network with three layers of units (input, hidden, and output). Input to this first level sequence-to-structure network consists of two contributions: one from the local sequence that is, taken from a window of 13 adjacent residues and another from global sequence. The global information contents for example can be percentage of each amino acid in protein or length of protein etc. Output of the first level network is the 1D structural state for the residue at the center of the output window. For PHDsec and PHDhtm the second level is a structure-to-structure network. The second level structure-to-structure network introduces a correlation between adjacent residues. It is important that the neural network get trained by balanced data for improved prediction of less populated states (e.g., strand) but this is associated with less accurate prediction of more populated states (e.g., loops). Consequently, the overall accuracy is lower for balanced training than for the unbalanced training. To find a compromise between this, a third and final jury decision is performed (effectively a compromise between over- and under prediction). This jury is a simple arithmetic average over, typically, four differently trained networks: all combination of first and second level networks with balanced and unbalanced training, and with balanced and unbalanced training of second level network. The final prediction is assigned to the unit with maximum output value.

Final Filtering

For secondary structure prediction (PHDsec) the filter affects only drastic and unrealistic predictions. Only filter used for predicting transmembrane helices (PHDhtm) is crucial for performance. Predicted transmembrane helices, which are too long, are either split or shortened. Predicted transmembrane helices, which are too short are either elongated or deleted. All decisions are based on the strength of the prediction and length of the transmembrane helix predicted. PHD predicts secondary structure at more than 72% accuracy and transmembrane helices are predicted with accuracy of more than 95%.

3.2.1.5.2 Secondary Structure Prediction using JPRED:

JPRED is a consensus prediction method (Cuff *et al*., 1998) It applies combination of various methods and returns consensus prediction which improves the average three state accuracy of prediction by 1% that to PHD. The server simplifies the use of current prediction algorithms and allows conservation patterns important to structure and function to be identified. The server accepts two input types, a family of aligned protein sequences or a single protein sequence. If a single sequence is submitted, an automatic process creates a multiple sequence alignment, prior to prediction (Cuff & Barton, 1998). Six different prediction methods: DSC (King & Sternberg, 1996), PHD (Rost & Sander, 1993), NNSSP (Salamov & Solovyev, 1995), PREDATOR (Frishman & Argos, 1997), ZPRED (Zvelebil *et al*., 1987) and MULPRED (Barton, 1988, unpublished) are then run, and the results from each method are combined into a simple file format.

The NNSSP, DSC, PREDATOR, MULPRED, ZPRED and PHD methods were chosen as representatives of current state of the art secondary structure prediction methods, that exploit the evolutionary information from multiple sequences. Each derives its prediction using a different heuristic, based upon nearest neighbors (NNSSP), jury decision neural networks (PHD), linear discrimination (DSC), consensus single sequence method combination (MULPRED), hydrogen bonding propensities (PREDATOR), or conservation number weighted prediction (ZPRED).

The predictions and corresponding sequence alignment are rendered in colored HTML, Java (Clamp *et al*., 1998) and Postscript. The predictions are colored and aligned with their corresponding family of sequences. Physico-chemical properties, solvent accessibility, prediction reliability and conservation number values (Zvelebil *et al*., 1987) for each amino acid are included in the output. The original ASCII text data from each of the prediction methods can also be downloaded. For example, BLAST results, MSF and HSSP format alignments, pair comparison files and so on.

3.2.1.6 Transmembrane Region Prediction

Onlione Servers for Transmembrane Region Detection

➢ DAS	http://www.sbc.su.se/~miklos/DAS/
➢ HMMTOP	http://www.enzim.hu/hmmtop/submit.html

PHDhtm	http://dodo.cpmc.columbia.edu/predictprotein/
SOSUI	http://sosui.proteome.bio.tuat.ac.jp/sosuiframe0.html
TMAP	http://www.mbb.ki.se/tmap/
TMHMM	http://www.cbs.dtu.dk/services/TMHMM-1.0/
TMpred	http://www.ch.embnet.org/software/TMPRED_form.html
TopPred2	http://www.sbc.su.se/~erikw/toppred2/

3.2.1.6 .1 Using TOPRED

TOPPRED (von Heijne, 1992) is strategy for predicting the topology of bacterial inner membrane proteins and it is proposed on the basis of hydrophobicity analysis, automatic generation of a set of possible topologies and ranking of these according to the positive-inside rule. It is shown that positively charged residues in short loop guide the orientation of helices by preventing translocation across membranes (von Heijne, 1994). It applies two empirical hydrophobicity cutoffs to the output of a sliding trapezoid window in order to compile certain and putative transmembrane helices. The combination of the putative helices that produces strongest enrichment of positively charged residues on the cytoplasmic side is selected as best prediction.

3.2.1.6 .2 Using TMPRED

The TMpred (Hofmann and Stoffel, 1993) program makes a prediction of membrane-spanning regions and their orientation. The algorithm is based on the statistical analysis of TMbase, a database of naturally occuring transmembrane proteins. The prediction is made using a combination of several weight-matrices for scoring. TMbase is mainly based on SwissProt, but contains informations from other sources as well. All data is stored in different tables, suited for use with any relational database management system. These tables are distributed as ASCII files.

3.2.1.6 .3 Using HMMTOP

HMMTOP (Tusnády and Simon, 1998) is based on the hypothesis that the localization of the transmembrane segments and the topology are determined by the difference in the amino acid distributions in various structural parts of these proteins rather than by specific amino acid compositions of these parts. Five structural parts were defined in membrane proteins: membrane helix (H), inside and outside helix tail (i and o), inside and outside loop (I and O). Topology is determined by partitioning amino acid sequence in a way that product of the relative frequencies of amino acids in these structural parts along the sequence should be maximal. This task can be solved by the hidden Markov model (HMM), in which biological constraints can be taken into account by the architecture of HMM using the Baum-Welch algorithm. The structural parts, which are described above, correspond to the five states used by the model. With use of this HMM architecture a state sequence (i.e. a prediction) can be generated as follows: first a state is chosen according to the initial state probabilities. Every following state is chosen according to the transition probabilities of the present state. The aim is to maximize the product of these probabilities and the emission symbol probabilities along the given sequence. The method has been a successful demonstration of HMM in secondary structure prediction.

3.2.1.6 .4 Using TMHMM

TMHMM (Sonnhammer *et al*., 1998) is based on a hidden Markov model (HMM) with an architecture that corresponds closely to the biological system. The model is cyclic with 7 types of states for helix core, helix caps on either side, loop on the cytoplasmic side, two loops for the non-cytoplasmic side, and a globular domain state in the middle of each loop. The two loop paths on the non-cytoplasmic side are used to model short and long loops separately, which corresponds biologically to the two known different membrane insertion mechanisms. The close mapping between the biological and computational states allows us to infer which parts of the model architecture are important to capture the information that encodes the membrane topology, and to gain a better understanding of the mechanisms and constraints involved. Models were estimated both by maximum likelihood and a discriminative method, and a method for reassignment of the membrane helix boundaries was developed. In a cross-validated test on single sequences, our TMHMM correctly predicts the entire topology for 77% of the sequences in a standard dataset of 83 proteins with known topology. The same accuracy was achieved on a larger dataset of 160 proteins. These results compare favorably with existing methods. The TMHMM method is very similar to HMMTOP and uses the same algorithm for training the internal parameter of markov model.

3.2.1.6 .5 Using SOSUI

SOSUI (Hirokawa *et al*., 1998) is a system for discrimination of membrane proteins together with soluble ones and the prediction of transmembrane helices. One important assumption SOSUI system makes is that, a primary transmembrane helix is stabilized by a combination of amphiphilic side chains at helix ends as well as high hydrophobicity in the central region. The system uses four paramamters in form of four indices. A hydropathy index (Kyte and Doolittle, 1982), an amphphphilicity index, an index of amino acid charges and length of each sequence. The SOSUI output contains (i) the type of protein; (ii) the region of transmembrane helices; (iii) a graph of the hydropathy plot; and (iv) helix wheel diagram for all transmembrane helices.

3.2.1.7 Perscan: a method for predicting 3D models of transmembrane helices

The structure prediction of integral membrane proteins is a difficult task. However since the membranes are essentially 2 dimensional, they provide a powerful constraint upon arrangement of the elements that cross them. Therefore structure prediction of $\alpha-$ helical membrane proteins can often be viewd as a two dimensional problem for which four pieces of information are required: (1) The region of the sequences that form transmembrane helices (2) the basic topology of transmembrane domain; (3) The side of each helix that faces the helix bundle. (4) The relative depth that each helix is inserted into membrane.

Perscan is a collection of programs that attempts to address some of the requirements in order to get information about system under study. Perscan (V7.0) is a collection of 13 FORTRAN programs that detect and display periodicity in protein sequences or structures. These are 2 'PROF' programs, 5 'PER' and 5 'SCAN' programs and one utility program called SELHEL. The PROF programs are more traditional method for searching transmembrane helices. Perscan use Fourier transform methods in order to identify periodicity of hydrophobic and hydrophilic residues in sequence and sequence alignments to identify amphipathic helices (Eisenberg *et al*., 1984; Cornette *et al*., 1987). The periodicity of conserved/variable residues can be used to predict the presence of helix (Komiya *et al*., 1988). The third method uses different between substitution patterns described for soluble (Overington *et al*., 1990; Overington *et al*., 1992) and membrane proteins (Donnelly *et al*., 1993). These environment-specific substitution tables can also be used to assign a value that quantifies the extent to which each position in a sequence alignment is buried. The periodicity in such values can be used to assign values to predict the presence of α-helix and also allows the buried face of each helix to be identified.

The SCAN programs (SCANHYD, SCANVAR, SCANCON, SCANMUT and SCANACC) are designed to look for sequences in complete sequence alignments or structures, whereas the PER programs (PERHYD, PERVAR, PERCON, PERMUT and PERACC) carry out a more detailed analysis of a single putative helical region. The five identifiers (HYD, VAR, CON, MUT and ACC) indicate the different properties for which helical periodicity is searched (i.e., hydrophobicity, variability, conservation, substitution-patterns and solvent accessibility). This information is then used to predict the point at which the helix makes contact with the aqueous environment at the borders of bilayer (Donnelly *et al*., 1993;Donnelly and Codgell, 1993).

PERSCAN is also useful as secondary structure prediction method. The results of PERSCAN including the number of helices, variable and constant faces of a helix, buried faces, hydrophobic moments combined with helix-wheel diagram provided by it, can be used to build a model of the system under study.

3.2.2 Tertiary Structure Prediction (or Fold Recognition)

Lists of threading servers

➢ 123D	http://www-lmmb.ncifcrf.gov/~nicka/123D.html
➢ 3D-PSSM	http://www.bmm.icnet.uk/~3dpssm
➢ Honig lab	http://honiglab.cpmc.columbia.edu/
➢ Libra I	http://www.ddbj.nig.ac.jp/htmls/E-mail/libra/libra/libra.html
➢ NCBI	http://www.ncbi.nlm.nih.gov/Structure/RESEARCH/threading.html
➢ Profit	http://lore.came.sbg.ac.at/home.html
➢ Threader2	http://insulin.brunel.ac.uk/threader/threader.html
➢ TOPITS	http://www.embl-heidelberg.de/predictprotein/help05.html
➢ UCLA-DOE	http://www.doe-mbi.ucla.edu/people/frsvr/srsvr.html
➢ GenThreader	http://insulin.brunel.ac.uk/psiform.html

The term "threading was first coined in 1992 by Jones *et al*. (Jones *et al*., 1992), but the field has grown considerably with many different methods being proposed. The idea behind threading comes from the fact that a large percentage of proteins adopt a limited number of folds (Orengo *et al*., 1994).

Description of the methods is out of scope of this thesis. How ever, the most important methods so far had been the 1-D-3-D profiles (Bowie *et al*., 1991), threading (Jones *et al*., 1992), using secondary structure predictions (Rost, 1997), combining sequence similarity with threading as implimented in Gen-THREADER (Jones, 1999), and using structural profiles (3D-PSSM; Kelly *et al*., 2000).

3.3 Derivation of Model from Template(s)

3.3.1 History and General Comments

Comparative modeling uses experimentally determined protein structures to predict conformation of other proteins with similar amino acid sequences. This is possible because a small change in the sequence usually results in a small change in structure (Lesk and Chothia, 1986; Hubbard and Blundell, 1987). The accuracy of protein models obtained by comparative modeling compares favorably with the model calculated by other theoretical models. The comparative method produces models with an r.m.s error as low as 1Å for the sequences that have sufficiently similar homologues with known 3D structures (Topham *et al.*, 1991); in contrast, physical prediction methods and combinatorial modeling calculates structures with r.m.s. error of approximately 3.5Å for small proteins (Cohen and Kuntz, 1989; Wilson and Doniach, 1989). On the other hand, comparative modeling is not as accurate as X-ray crystallography and NMR, which can determine protein structures with an r.m.s. error of approximately 0.3 and 0.5Å, respectively (Clore and Gronenborn, 1991). It is also restricted to sequences with closely related proteins with known structures. It has been estimated that approximately one third of all knows sequences are related to at least one protein of known structure (Rost and Sander, 1996). With approximately 0.7 million sequences known, comparative modeling had been applied to 2,43,410 domains in known sequences (Sanchez *et al.*, 2000). This is an order of magnitude more proteins than experimentally determined protein structures (~15,600). Furthermore, the usefulness of comparative modeling is steadily increasing because the number of different structural folds that protein can adopt is limited (Chothia, 1992), and because the number of experimentally determined new structures is increasong exponentially (Holm and Sander, 1996). Due to Structural Genomics Initiative in less than 10years, atleast one example of most structural folds will be known, making comparative modeling applicable to most globular domains in most protein sequences (Sali *et al*., 1998).

Early modeling studies frequently relied on the construction of wire or plastic models and only later incorporated interactive computer graphics. The first models produced from homologous proteins were constructed by taking the existing coordinates of a single known structure and then altering those side chains that were not identical in the protein to be modeled. Browne and co-workers (1969) published the first model, they modeled bovine α-lactalbumin on the three dimensional structure of hen egg-white lysozyme. For reviews on the history and development of homology modeling please see Johnson *et al*., 1994; Sanchez and Sali, 1997 and Sanchez and Sali, 2000 etc.

3.3.2 Modeling Procedure

Modeling procedures can be envisaged as two steps. The first step is to solve the inverse folding problem: to define all those sequences that can adopt a particular fold (step1 and step 2 of this thesis; figure 1.1). It involves projecting restraints from a three-dimensional structure onto a one-dimensional sequence. The second step is to use the sequence with the knowledge that the protein belongs to a family of known fold to construct a model.

The modeling techniques used for comparative modeling generally falls into two classes: (a) assembly of rigid fragments and (b) use of distance geometry to construct the models that are in best agreement with the distance constraints. Both the approaches have been used while working towards this thesis. The packages used are COMPOSER (Sutcliffe *et al*., 1987a,b; a part of SYBYL suite) and MODELER (Sali and Blundell, 1993) respectively. The flow chart of the methodology used by COMPOSER and MODELER is given in figure 3.1 and figure 3.2 respectively. The step obviously important to both the methods is defining the topologically equivalent parts using the superposition of homologous structures and other structural properties. COMPOSER uses it to derive the structural framework (Structurally Conserved Regions or SCRs; Sutcliffe *et al*., 1987a) for the model, while MODELER uses it to derive the spatial restraints for the model (Sali *et al*., 1993. The rules for comparative modeling are also derived from the database of homologous structures (Sali and Overington, 1994). Several methods are available for defining topological equivalence of residues. Most of them use superposition of the structures. However proteins can be compared at residue, secondary structure, supersecondary structure, motif or domain levels also. The features that can be used for the comparison at residue and segment levels of two structures is summarized table (derived from Sali and Blundell, 1990).

Comparison at Residue level

Properties:

Identity, Residue type properties, Local conformation, Distance from gravity centre, Side- chain orientation, Main-chain orientation, Solvent accessibility, Position in space

Relations:

Hydrogen bond, Distance to one or more nearest neighbors, Disulfide bond, Ionic bond, Hydrophobic cluster

Comparison at Segment level

Properties:

Secondary structure type, Amphipathicity, Improper-dihedral angle, Distance form gravity centre, Orientation relative to gravity centre, Solvent accessibility, Position in space, Orientation in space

Relations:

Distance to one or more nearest neighbors, Relative orientation of two or more segments

Table 3.1 Showing different levels at which two protein structures can be compared to derive topological equivalence

The methods in this thesis for superposition and generating structure-based alignments are MNYFIT (Sutcliffe *et al.*, 1987), COMPARER. (Sali and Blundell, 1990) and STAMP (Russell and Barton, 1992). COMPOSER uses MNYFIT for generating the structural framework while the structure-based alignment for MODELER input can be prepared using either method.

3.3.2.1 MNYFIT

MNYFIT (Sutcliffe *et al.*, 1987) works by method of unweighed least square fitting (Hermans and Ferro, 1971; McLachlan 1979, 1982; Sutcliffe *et al.*, 1987a) choosing one of the structures at random to the framework and fit all the others to it pairwise. The process is iterative and, it does the fitting till an r.m.s of 10^{-5}Å is reached. In the second step, atomic positions are weighed while doing least square fit as to reflect how representative it is of the set of topologically equivalent positions. The third step generates a framework that is close to the specific structure to be modeled.

3.3.2.2 STAMP

STAMP (Russell and Barton, 1992) is designed with specific purpose of generating multiple sequence alignment from tertiary structure comparison. It provides not only multiple alignments and the corresponding 'best-fit' superpositions, but also a systematic and reproducible method for assessing the quality of such alignments. It also provides a method for protein 3D-structure database scanning.

STAMP uses Rossman and Argos equation (Rossman *et al.*, 1975) for expressing the probability of equivalence of residue structural equivalence. STAMP then uses Smith Waterman dynamic programming algorithm (Smith and Waterman, 1981; Sankoff and Kruskal, 1983; Barton, 1994) for fast determination of best path through a matrix containing a numerical measure of the pairwise similarity of each position in one sequence to each position in another sequence. Within STAMP, these similarity values correspond to the modified values of Rossman and Argos equation. From this a set of equivalent C_α positions are obtained. These are used to obtain a best fit transformation and r.m.s. deviation by a least square method (Kabsch, 1978; McLachlan, 1979). This transformation is applied to yield two new sets of coordinates for which the entire procedure is repeated in iterative fashion until the two sets of coordinates, and the corresponding alignment, converge on a single solution.

3.3.2.3 COMPARER

COMPARER (Sali and Blundell, 1990) attempts to define topological equivalences in protein structures by comparing properties of protein structures at various levels. Residue and segment properties that COMPARER takes in to are: residue local fold, residue type properties, residue distance from molecular gravity centre, side-chain orientation relative to molecular gravity centre, side-chain orientation relative to main-chain, main-chain orientation relative to molecular gravity centre, side-chain solvent accessibility, main-chain solvent accessibility, hydrogen bonding relationship, residue identity, residue position in space, φ, ψ dihedral angle and main chain directions. A normalized difference of a certain feature between residues from the pair of proteins is computed. A scaling factor is defined that determines the relative importance of a feature used for comparison. From this a weighted sum is calculated. Relationships were weighed using simulated annealing methods. Once the dissimilarity matrices are computed. Best pairwise or multiple alignment is searched using dynamic programming approach described before (Needleman and Wunsch, 1970; Sankoff and Kruskal).

COMPARER alignments are more useful in terms of modeling by spatial restraints since it gives the topologically equivalent residues using hierarchical definition of structure and used in MODELER (Sali and Blundell, 1993).

Description of Modeling Programs

3.3.2.4 COMPOSER

As mentioned before COMPOSER (Sutcliffe *et al*., 1987a,b) is an automated approach of comparative modeling based on assembly of rigid fragments. It is available as a part of SYBYL module of TRIPOS Inc. The flow chart of the COMPOSER methodology is as shown in figure 3.1. As described before for homologous structures are used to derive the structural framework or SCRs using MNYFIT. Modeling of gaps or Structurally Variable Regions (SVRs) involves search for fragments of suitable length and end-to-end

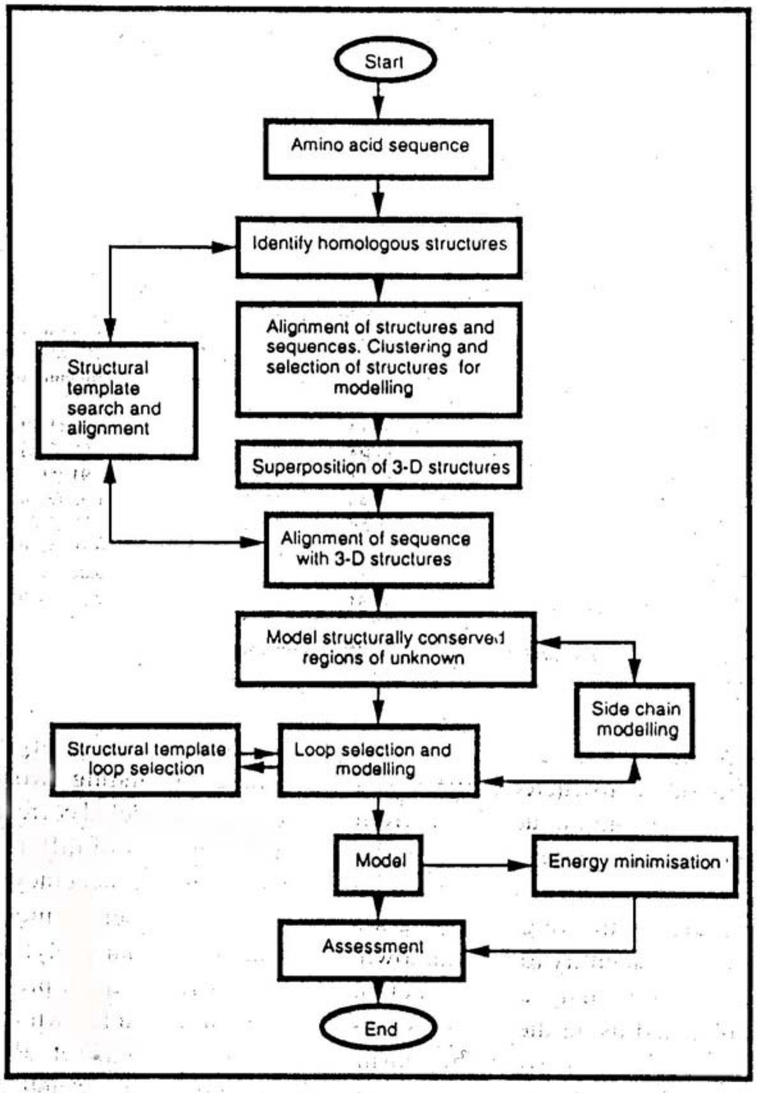
Start
Amino acid sequence
Identify homologous structures
Alignment of structures and sequences. Clustering and selection of structures for modelling
Structural template search and alignment
Superposition of 3-D structures
Alignment of sequence with 3-D structures
Model structurally conserved regions of unknown
Side chain modelling
Structural template loop selection
Loop selection and modelling
Model
Energy minimisation
Assessment
End

Figure 3.1 Showing the flowchart of methodology implimented in COMPOSER homology modeling program.

distances with a check that the modeled loop does not clash with the rest of the proteins. The identified region is usually fitted to anchor regions (the ends of the intervening regions in the model that are mainly the helices and strands). The selection of the correct conformation can be improved by considering the r.m.s. difference in the anchor regions and sequence similarity between the identified segment and one to be modeled.

Candidate loops can also be ranked by using structural templates (Topham *et al*., 1993). The templates reflect amino acid substitutions that are compatible with the local structural environment for each amino acid defined in terms of main chain conformation, solvent accessibility, hydrogen bonding, disulfide bonding, and *cis*-peptide conformation (Overington *et al*., 1990; 1992). The side chains are modeled depending on the orientation of the side chains in the equivalent positions in the known homologues or based on a large number of rules derived for their preferred conformations in various secondary structures (Sutcliffe *et al*., 1987b). Other techniques, including energy minimization and localized molecular dynamics can then be applied to the model.

3.3.2.5 MODELLER

MODELLER is an implementation of an automated approach to comparative protein structure modeling by satisfaction of spatial restraints extrapolated from homologous 3D-structures to the sequences to be modeled (Sali and Blundell, 1993, Sali *et al*., 1995). The modeling procedure begins with an alignment of the sequence to be modeled (target) with related known structures (templates). This alignment is usually the input to the program. The output is a 3D model for the target sequence containing all main chain and side chain non-hydrogen atoms.

First, many distance and dihedral angle restraints on the sequence are calculated from its alignment with template 3D structures. The form of these restraints was obtained from a statistical analysis of the relationship between many pairs of homologous structures. This analysis relied on the database of 105 family alignments that included 146 known structures (Sali and Overington, 1994). By scanning the database, tables quantifying various correlations were obtained, such as correlations between two equivalent C_α–C_α distances, or between equivalent main chain dihedral angles from two related proteins. These relationships were expressed as conditional probability density functions (pdf's) and can be used directly as spatial restraints. For example, probabilities for different values of the main chain dihedral angles are calculated from the type of a residue

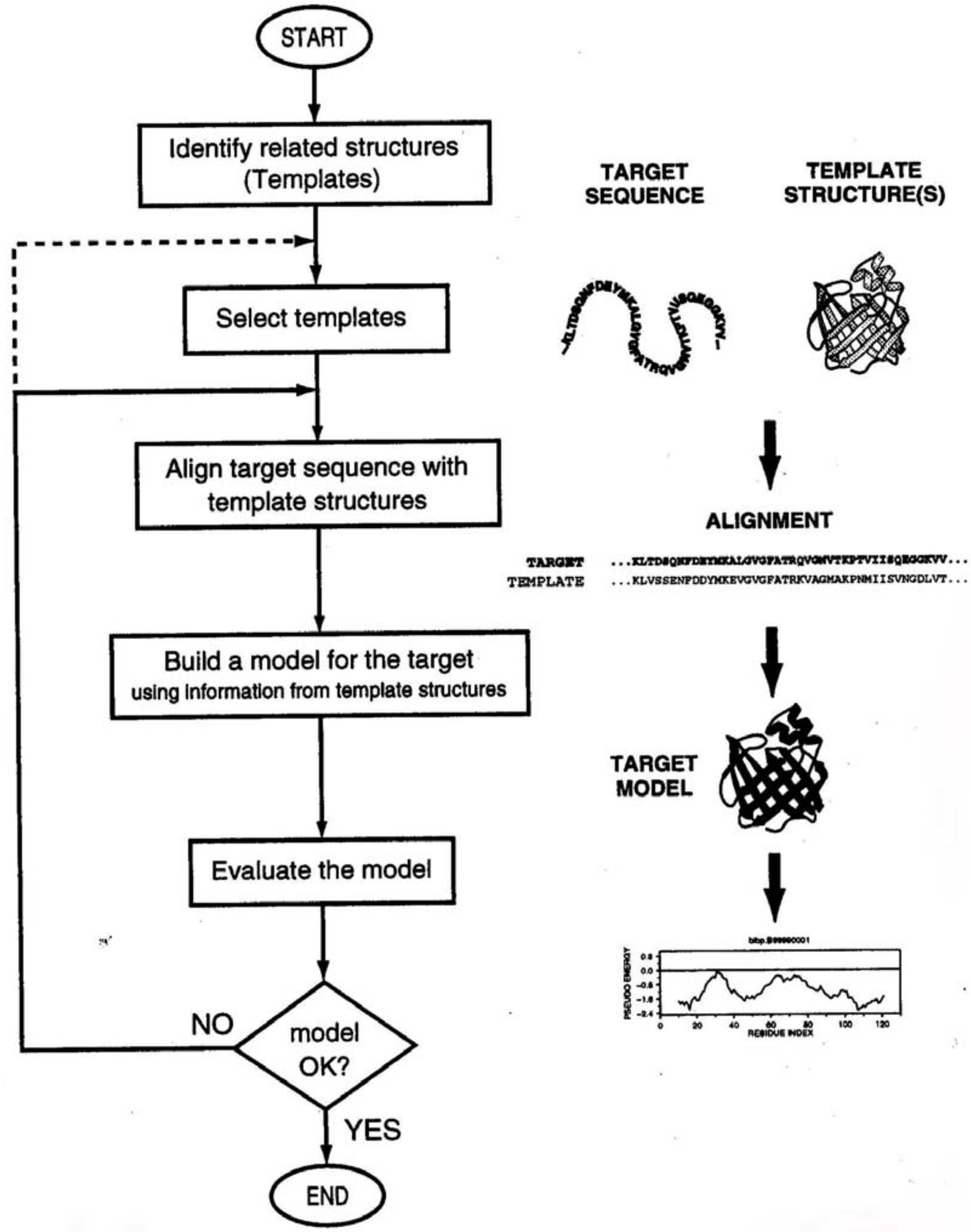
START
Identify related structures (Templates)
Select templates
Align target sequence with template structures
Build a model for the target using information from template structures
Evaluate the model
model OK?
NO
YES
END
TARGET SEQUENCE
TEMPLATE STRUCTURE(S)
ALIGNMENT
TARGET ...KLTDSQNFDEYMKALGVGFATRQVGNVTKPTVIISQEGGKVV...
TEMPLATE ...KLVSSENFDDYMKEVGVGFATRKVAGMAKPNMIISVNGDLVT...
TARGET MODEL
bibp.B99990001
PSEUDO ENERGY
0.8
0.0
-0.8
-1.6
-2.4
0
20
40
60
80
100
120
RESIDUE INDEX

Figure 3.2 Flowchart showing methodology implimented in homology program MODELLER.

considered, form the main chain conformation of an equivalent residue, and form the sequence similarity between the two proteins. Another example is the pdf for a certain C_α–C_α distance given equivalent distances in two related protein structures. An important feature of the method is that the spatial restraints are obtained empirically, from the database of protein structure alignments. Next, the spatial restraints and CHARMM energy terms enforcing proper stereochemistry are combined in to an objective function. Finally, the model is obtained by optimizing the objective function in Cartesian space. The optimization is carried out by the use of the variable target function method (Braun and Go, 1985) employing methods of conjugate gradients and molecular dynamics with simulated annealing.

Several slightly different models can be calculated by varying the initial structure. The variability among these models can be used to estimate the errors in the corresponding regions of the fold. MODELLER evaluates the model internally. The internal self-consistency check is that the model has to satisfy most restraints used to calculate it, especially the stereochemical restraints. If some restraints are grossly violated in all models it is likely that the alignment in the corresponding region is incorrect. The restraint violations are reported at the end of the log file.

3.4 Model Evaluation

Evaluation of the 3D model is an essential step that can be performed at different levels of structural organization, namely, to identify (1) the correctness of the overall fold, (2) detect errors over more localized regions, and (3) check stereochemical parameters like bond lengths, bond angles, and hydrogen bond geometry.

Model Evaluation programs and sites

➢ PROCHECK	www.biochem.ucl.ac.uk/~roman/procheck/procheck.html
➢ WHATCHECK	www.sander.embl-heidelberg.de/whatcheck
➢ PROSAII	www.came.sbg.ac.at
➢ PROCYON	www.horus.com/sippl/
➢ BIOTECH	biotech.embl-ebi.ac.uk:8400/
➢ VERIFY3D	www.doe-mbi.ucla.edu/verify3d.html
➢ ERRAT	www.doe-mbi.ucla.edu/errat_server.html

It is recommended to evaluate the model obtained by homology modeling for errors. Various programs that are developed for checking the quality of protein structures are also used for checking quality of models derived from homology modeling.

3.4.1 PROCHECK

PROCHECK (Laskowski *et al.*, 1993) makes use of properties originally derived from a set of 119 non-homologous protein crystal structures at a resolution of 2.0 Å or higher and having an R-factor no greater than 20% (Morris *et al.*, 1992). It checks the stereochemistry using C_α chirality, Percentage of residues found (more than 90%) in the core region of Ramachandran plot, torsion angles for secondary structures and χ_1, χ_2, χ_3 torsional angles etc. It also calculates the main chain hydrogen bond energy. The output is a series of postscript files. The most important file is the one that gives the Ramachandran plot, which has been discussed extensively before.