1

The Basics of Protein Sequence Analysis

Katarzyna H. Kaminska, Kaja Milanowska and Janusz M. Bujnicki

1.1 Introduction

Genes and proteins are products of evolution. Over the course of evolution, the nucleotide sequences of genes undergo numerous changes. First, duplications (or deletions) may lead to creation of additional copies (or removal) of genes or gene fragments. Second, local mutations: substitutions, insertions and deletions within genes may result in changes to the amino acid sequence of proteins they encode. Thus, the initially identical copies of duplicated genes over time accumulate divergent mutations that make their sequences progressively dissimilar. Not all positions of protein-encoding genes are equally susceptible to mutation, as some amino acid residues may be very important for protein function, stability, or folding and may thus be more constrained in the residue types allowed. Therefore, although mutations are random, in nature we observe only such protein variants, in which sequence changes have been 'accepted' by the evolutionary pressure. Proteins with mutations that cause detrimental changes in structure and/or function are usually eliminated. If the protein is important to the integrity of the organism, the organism that bears the mutant gene dies, and the structurally/functionally compromised variant ceases to exist; or if it is not important, then the inactivated gene may be eventually 'purged' from the genome by random deletions. On the other hand, if the mutant variant brings an additional and beneficial new function to the organism, it is likely to be retained and further 'optimized' towards the activity favored by the selective pressure.

The above-mentioned evolutionary mechanisms have given rise to families of evolutionarily related proteins (homologs), which share a common ancestor. Duplicated proteins are

Prediction of Protein Structures, Functions, and Interactions Edited by Janusz Bujnicki

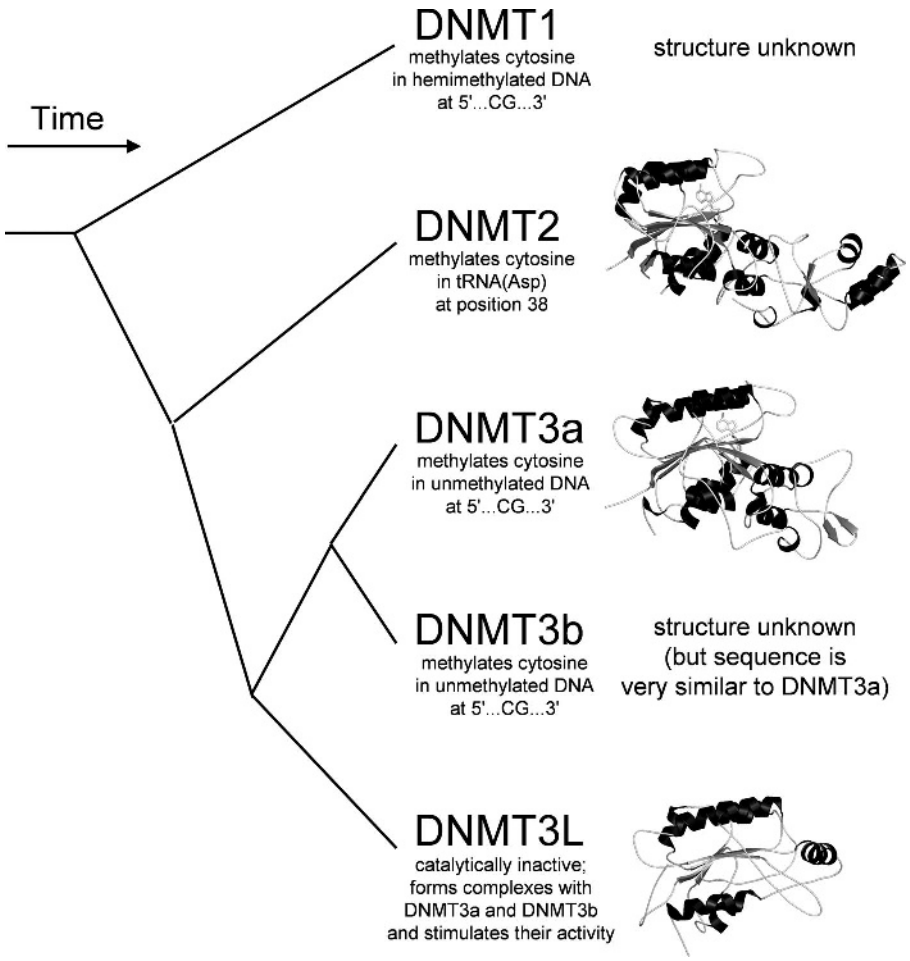

Figure 1.1 *In the course of the evolution, protein-encoding genes undergo duplications, and the resulting copies accumulate differentiating mutations (substitutions, insertions, deletions). As long as a small subset of residues important for internal stability and interactions with key partner molecules is preserved, the overall structure and mode of action of diverging homologous proteins is likely to remain similar. As a result, we observe that extant homologous proteins retain similar tertiary structure, while sequence similarity becomes less and less evident. Mutations may cause the protein or one of its paralogous copies to lose its function (and be eliminated), or to develop a new function, usually by somehow modifying the previous function. Example: a family of cytosine-C5 methyltransferases. Most members methylate cytosine in DNA; however, DNMT2 has apparently changed its specificity and acts on tRNA, and DNMT3L has lost the original catalytic activity, but instead gained a new regulatory activity*

described as paralogs, and in these relatives the sequences and functions can diverge considerably from the original variant (see Figure 1.1). A general function of paralogs (such as the ability to bind a certain type of molecule or to catalyze a certain type of chemical reaction) often remains conserved, but they tend to specialize in different specific roles

(e.g. catalysis of a similar reaction on different substrates, different mode of regulation, or being directed to different cellular compartments etc.). It has been found that new activities and entire biochemical pathways evolve by recruitment and 'tinkering' of enzymes that are already capable of performing the desired chemistry, rather than by developing new functions from scratch.[1] Thus, paralogous enzymes are often found to carry out similar reactions in different pathways. Examples of large groups of paralogous proteins include: different kinases or different helix-turn-helix transcription factors encoded in the human genome. On the other hand, proteins from different organisms that have diverged from the ancestral gene present in the last common ancestor of these organisms (i.e. copies of 'the same protein in different organisms') are called orthologs. They tend to retain very similar functions and their sequences usually show higher conservation than between paralogs (for a detailed review on orthology and paralogy and discussion of several caveats, see ref. [2]). Thus, members of a protein family exhibit divergence, but usually share a specific biological function despite high sequence diversity.

As the number of known protein structures solved by X-ray crystallography and NMR techniques increased, it became clear that protein structure is much more highly conserved throughout evolution than protein's sequence.[3] While in many families sequence identity between members can drop below 5% identical residues, they tend to retain most of their common structural scaffold, mainly in the core of the protein. Structure is also more conserved than function; remote paralogs that retain common fold but replace functionally important residues may fulfill completely different roles in the cell (examples include a non-enzymatic heme-binding protein nitrophorin of a bedbug, which is related to an enzyme inositol polyphosphate 5-phosphatase[4]). Counterexamples may be found: proteins that exhibit high sequence similarity but different functions and/or structures (for a review see ref. [5]), however they are relatively rare. This suggests that structure comparison is the best method to detect remote evolutionary relationship.[6] Unfortunately, protein structure determination is considerably more costly and time consuming than gene sequencing, therefore the sequence databases have always been several orders of magnitude larger than the structure databases. There has been an exponential increase in the sizes of both types of data since the early 1970s but the largest sequence database GenBank[7] doubles in size roughly every 18 months, while the number of protein structures deposited in the Protein Data Bank[8] doubles roughly every three years, hence the gap keeps growing and is unlikely to be closed in the near future.

Not only have the structures lagged behind sequences, but also functional characterization. With the current pace of data generation by high-throughput sequencing projects, it is an impossible task to study all proteins by experiment. Thus, it is imperative to develop methods that use sequence information to identify evolutionary relationships and/or predict common structures and functions (or at least some aspect thereof). In this chapter, we discuss bioinformatic approaches for analyzing protein sequences, in particular aiming at identification and basic characterization of evolutionary relationships. The following chapters in this volume focus on direct prediction of functional properties from sequence (Chitale *et al.*), prediction of local conformation (Majorek *et al.*), and construction of three-dimensional structural models based on sequence analyses (Kosinski *et al.*). Here, we first define the primary functional units in protein sequence (domains and motifs) and describe how domains are duplicated and combined in various ways to give different protein families. We then briefly describe the major classifications and databases of protein

families, domains, and motifs. In the main part of the chapter we review algorithms for protein sequence analyses, with the particular focus on their implementations that have been made freely available as web servers or downloadable computer software. We concentrate on methods for database searches and identification of sequence similarities, clustering of sequences into homologous families, multiple sequence alignment, and inference of evolutionary relationships. Finally, we consider an iterative procedure utilizing these methods for identification of domains and motifs in the protein sequence and their functional characterization.

1.2 Domains: Primary Functional Units in Protein Sequence

Proteins are modular, containing discrete regions that perform different roles. The primary modular unit is called a domain. Regrettably, there is no standard definition of what a domain really is. Structural biologists put emphasis on structural autonomy, biochemists and geneticists refer to regions with autonomous function detectable in their experimental assays, while evolutionary biologists focus on regions that are conserved throughout the evolution. Here, we adapt a definition based mostly on structural and evolutionary criteria.

The structural domain has been first defined in the 1960s with the advent of the first structures of water-soluble globular proteins determined by X-ray crystallography (for review see ref. [9]). Globular domains are characterized by ellipsoidal or spherical shape, and a relatively stable internal structure, which is defined by the amino acid sequence. In structural domains the backbone of a polypeptide chain exhibits elements of regular secondary structure (α-helices and/or β-strands) that forms a unique three-dimensional arrangement called a 'fold', which serves as a scaffold for functionally important side chains of amino acid residues. Some amino acid residues form a hydrophobic core, from which water molecules are excluded, while others are exposed at the hydrophilic surface, where they form sites of interactions with other molecules. In order to satisfy these requirements, protein domains are typically formed by amino acid sequences of high informational complexity. Globular domains typically range from 50 to 300 residues with a few larger and smaller exceptions (review: [6]). Domains located within biological membranes exhibit similar structures, with a few exceptions: they are usually barrel-shaped, with a hydrophobic 'belt' on the outside that ensures a seamless fit to the hydrocarbon tails of the lipid bilayer. One type of transmembrane (TM) proteins is composed exclusively of α-helices, while the other contains only β-strands; the latter type of structures form pores, and contain an internal hydrophilic channel instead of the hydrophobic core (review: [10]).

Since 1970 it emerged that structural domains may recur in different structural contexts or in multiple copies in the same polypeptide chain. More recent comparative analyses of large numbers of protein sequences and structures confirmed that a structural domain is also a fundamental unit in evolution (reviews: [6,11]). The same domains can be found in different proteins in all three forms of life, Archaea, Bacteria and Eukaryota, as well as in viruses that infect them. Examples of frequently recurring domains include: a helix-turn-helix domain often found in DNA-binding proteins (20∼100 residues), a TIM-barrel domain present in many enzymes (∼200 residues), or a transmembrane domain found in G-protein coupled receptors (∼250 residues).

Gene fragments encoding domains may undergo duplication (e.g. leading to proteins with tandem copies of the same domain), fusion with other genes or gene fragments (leading to multi-domain proteins). Protein families are usually defined based on a presence of one common domain, which does not exclude the possible presence of additional domains. For example in enzyme families, the common homologous domain is usually responsible for performing catalysis, while the auxiliary domains may be responsible for recognition of various substrates. (e.g. in enzymes acting on DNA, they may recognize different specific DNA sequences). These auxiliary domains may formally belong to different families or even exhibit different folds. Thus, it is important to remember that proteins may comprise multiple domains, of which some may be homologous, while other may be non-homologous. Certain combinations of domains that are found recurring in diverse proteins are often referred to as modules or supradomains. They duplicate and are selected as one evolutionary unit either because it is functionally beneficial to have both activities present in one polypeptide or because the functional site is created between the domains.[12] Examples of such modules can be found e.g. in nucleic acid polymerases, which often have the polymerization domain fused to an exonuclease proof-reading domain or in proteins involved in signal transduction, which have a nuclear receptor ligand-binding domain fused to a DNA-binding domain.

It is important to remember that a conserved 3D structure in the different context does not guarantee the same amino-acid sequence or function; in fact these features may differ substantially for remotely related proteins and domains. An insertion of one domain into another may cause the latter domain to become discontinuous in sequence, even though its original three-dimensional fold is preserved, with distant sequence elements brought together to form a stable structure. Another example of a complex rearrangement is circular permutation (review: [13]), when a sequence fragment from one terminus is transferred to the other terminus, thereby changing the order of sequence motifs within the domain. A circularly permuted sequence may still form the same three-dimensional fold, albeit with a different connectivity of the polypeptide chain (N- and C-termini of a protein appear in a different position in the structure). Sequence rearrangements that do not preserve the order of primary sequence make detection of structurally conserved domains a very difficult task (see the final section of this chapter).

In addition to stably folded domains, many proteins possess segments that are non-globular in the sense that they lack a tightly packed hydrophobic core. They are often formed by compositionally biased sequences that are poor in hydrophobic residues and enriched in charged residues, and exhibit different types of 'low complexity' regions, e.g. short-period repeats, near-homopolymeric residue clusters, or aperiodic mosaics of only a few residue types.[14,15] Such segments may form fibrous or filamentous structures (e.g. in collagen or keratins) or exhibit conformational heterogeneity, so called 'intrinsic disorder' (see refs. [16,17]). Some of these regions form linkers that permit the correct spacing between globular domains, but others play more specific roles, in particular harbor sites for interactions with other molecules, including proteins and nucleic acids. The review of the variety of structures assumed by non-globular regions is beyond the scope of this chapter; here, we will discuss only those of their features that are directly related to sequence–function relationships. For recent reviews on structure–function relationships of fibrous and intrinsically disordered proteins (IDPs) proteins the reader should consult ref. [18] and refs [19–21], respectively. Bioinformatics methodology for prediction of

regions of disorder is reviewed in detail in the chapter by Majorek *et al*. in this volume.

1.3 Sequence Motifs

While essentially all protein sequences can be subdivided into globular domains and non-globular segments, the most basic functional unit in protein sequence is called a motif. Motifs usually correspond to short sequence fragments (a few, typically up to 10 amino acids) that reflect some vital biological role in terms of structure or function (e.g. are responsible for stabilizing interactions or promote a particular conformation within a protein molecule or take part in binding of another molecule). Motifs occur frequently both in globular and non-globular sequence segments, but depending on the structural context, they fulfill different roles. Structured motifs (SMs) are fingerprints of globular domains. They are conserved in the evolution because of critical involvement in activity, for which the entire domain is selected, e.g. binding of the ligand that serves as a cofactor in the enzymatic reaction catalyzed by the enzyme. They are usually conformationally rigid (or at least their fragments are, while some parts may show mobility required for function). Examples of SMs include Walker A GXXGXGK(T/S) and Walker B '(R/K)X(6-7)Lh(4)D' motifs involved in ATP-binding in a large group of ATP-utilizing enzymes.[22] Other SMs may be required for structural stability, e.g. contain Zn-binding Cys and His residues in e.g. C_2H_2–type Zn-finger domains: 'CX(2-4)C. . .HX(2-4)H'.[23] The presence of a common SM in a particular set of domains suggests the presence of a similar well-defined structure required for binding of a ligand, but may or may not indicate homology. In particular, motifs involved in binding of widespread ligands are found in several protein families that are unrelated to each other. Thus, caution must be exerted when identification of a single common SM is used to infer evolutionary relationship, and it should be accompanied by analysis of global sequence similarity (see below) and preferably, also global structural similarity.

Linear motifs (LMs) are a different group of functionally heterogeneous sites. They mediate interactions of proteins with other molecules, are responsible for cell compartment targeting, or represent the sites of post-translational modification, such as phosphorylation, glycosylation, fucosylation, methylation etc. (review: [24]). Motifs of this kind are typically embedded in locally unstructured regions, but possess a few specificity-determining residues favoring disorder-order transition upon binding. LMs have a unique amino acid composition, dissimilar to either globular domains or non-globular segments; they are enriched in Pro, hydrophobic residues Trp, Leu, Phe, and Tyr, as well as charged residues Arg and Asp.[25] Examples of LMs include the PXXP motif for binding to SH3 domains, the NPXY motif for the interaction with PTB domains, the WXXW C-mannosylation site, and the WXXX(Y/F) peroxisomal targeting signal. LMs rarely occur in 'conventional' globular domains, but if they do, these domains almost invariably undergo posttranslational modifications. LMs also show completely different conservation patterns than SMs. SMs are evolutionarily constrained by many interactions within the globular domains and/or stable binding to high-affinity ligands, therefore they are often conserved in entire protein families or superfamilies. LMs are typically involved in transient interactions, rely on a very few specific interactions and their structure is loosely constrained, therefore they may be easily created as well as removed due to few accidental mutations. If they appear in

locations that confer selective advantage, e.g. due to introducing a regulatory switch, they may be preserved in the course of the evolution. However, due to the relative redundancy of LMs, removal of a single site (e.g. one of many phosphorylation sites within a regulatory region of a particular protein) rarely has as drastic effects as removal of an individual SM (e.g. a catalytic motif in the enzyme). As a result LMs tend to be conserved only among very close homologs, and are frequently in non-homologous proteins that nonetheless share the same functionality (e.g. the ability to be phosphorylated by the same protein kinase). Thus, Nature appears to use LMs as evolutionary interaction switches.[24]

1.4 Databases of Protein Families, Domains, and Motifs

The importance of domains as structural building blocks, basic elements of biochemical function, and elements of evolution, has brought about many automated methods for their identification and classification in proteins of known structure. However, as mentioned before there is no standard definition of what a domain really is, therefore assigning domain boundaries even for proteins with known structures is not a trivial task. While human experts disagree for approximately 10% of structures, automatic methods for domain assignment show much larger discrepancy even for structures that the human experts agree on.[26] Expectedly, assignment of domains for proteins in the absence of structural information varies enormously; hence prediction of domains from sequence remains a challenging problem. However, before we describe bioinformatic methods that approach this problem, we will describe databases of protein families and domains, and tools for database searches and multiple sequence alignments.

A number of databases have been created to facilitate classification and identification of domains and motifs, and using them for protein function prediction. They usually classify proteins based on the presence of conserved domains (defined according to many different criteria) and/or motifs and group them according to sequence or structural similarity or based on predicted evolutionary relationships, such as orthology. Table 1.1 lists some of the most comprehensive and well-established databases of families, domains, and motifs, whose entries have been created and are curated at least partially by protein experts.

The most popular databases that classify protein domains based on structural comparisons are SCOP [27] and CATH.[28] Domain definitions used by these databases are based on very similar geometric criteria and therefore usually coincide with each other. Both databases are organized hierarchically, with the top level corresponding to structural class of a domain, i.e. the proportion of residues adopting α-helical or β-strand conformation (see the chapter by Majorek *et al*. for the discussion on secondary structure assignment). Within each class, domains are classified into folds, which group together proteins exhibiting significant structural similarity, both in terms of the arrangement of structures in three dimensions, and connectivity between them (as a result, circularly permuted variants that differ in connectivity should fall into different folds, hence this criterion is sometimes relaxed). Further, proteins with the same fold and evidence for evolutionary relationships are classified into homologous superfamilies. Within superfamilies proteins with clear sequence similarity are grouped into families. SCOP is maintained by mostly manual analysis for recognizing relationships to generate superfamilies, while CATH uses a combination of automatic and manual analysis.

Table 1.1 *Databases of domains and protein families*

Method	URL (http://)	Data*	Description
			Structural
SCOP[27]	scop.mrc-lmb.cam.ac.uk/scop	D	Hierarchical classification of domain structures, with four levels: Class, Fold, Superfamily, Family. Linked to the SUPERFAMILY database[48], which maps protein sequences from fully sequenced genomes onto SCOP superfamilies and represents the resulting sequence superfamilies as HMMs.
CATH[28]	www.biochem.ucl.ac.uk/bsm/cath	D	Hierarchical classification of domain structures, with four levels: Class, Architecture, Topology, and Homologous superfamily. Linked to the Gene3D sequence database[49], which assigns proteins into HMMs based on CATH domain families.
			Sequence
InterPro[29]	www.ebi.ac.uk/interpro/	F, D, SM, LM	An integrative 'meta-database' that collects annotations at the level of families, domains, and motifs.
Pfam[31]	pfam.sanger.ac.uk/	F, D	Collects MSAs and HMMs covering protein families and domains. Provides information about protein domain architectures, species distributions, and known protein structures.
PANTHER[50]	www.pantherdb.org/	F	Classifies genes and proteins into families and subfamilies by their functions, using published experimental evidence and predictions based on evolutionary relationships. Families and subfamilies are then categorized by molecular function and biological process ontology terms. For some entries pathway information is also provided.
TIGRFAMs[51]	www.tigr.org/TIGRFAMs/	F	Collection of protein families encoded as HMMs.
iProClass[52]	http://pir.georgetown.edu/iproclass/	F	Classifies proteins according to PIR superfamilies and annotates them with PROSITE signatures.
ProDom[53]	prodom.prabi.fr/prodom/current/html/home.php	D	Contains domain families automatically generated from SWISS-PROT and TrEMBL sequence databases, and information about protein domain architectures
SMART[54]	smart.embl-heidelberg.de/	D	Stores information about protein domain architectures and protein-protein interactions.

Table 1.1 *(continued)*

PRINTS[55]	http://www.bioinf.manchester.ac.uk/dbbrowser/PRINTS/	SM	Database of protein family/domain fingerprints
PROSITE[33]	www.expasy.ch/prosite/	SM, LM, F, D,	Consists of documentation entries describing protein domains, families and functional sites, associated patterns and profiles to identify them. Complemented by ProRule, a collection of rules based on profiles and patterns, which increases the discriminatory power of profiles and patterns by providing additional information about functionally and/or structurally critical amino acids.
CDD[35]	www.ncbi.nlm.nih.gov/Structure/cdd/cdd.shtml	F, D	Collection of MSAs for domains and full-length proteins, contains its own curated domains and un-curated entries imported from other databases. Allows searching for proteins with similar sequences (CD-search) and similar domain architectures (CDART)
COG[37]	www.ncbi.nih.gov/COG/	F, D	Collection of MSAs and phyletic profiles for groups of orthologs or close paralogs from at least 3 fully sequenced genomes.

*D, F, SM, and LM indicate domains, families, structured motifs and linear motifs.

Among protein family/domain databases that classify protein sequences, there are two comprehensive meta-databases developed at the EBI in the UK and at the NCBI in the USA EBI's INTERPRO[29,30] is a major resource for protein families, domains and functional sites, which integrates the protein sequence database UniProt (which itself is a meta-database of Swiss-Prot, TrEMBL, and PIR) with databases of protein structure (MSD, SCOP, and CATH) and databases of families, domains, and patterns: Pfam, PROSITE, PRINTS, ProDom, SMART, TIGRFAMs, PIRSF, SUPERFAMILY, Gene3D, and PANTHER (see Table 1.1). Among the latter class of databases, particularly important are Pfam,[31,32] currently the most comprehensive primary protein family/domain resource based on sequence data, and PROSITE,[33,34] which focuses on motifs. Pfam-A entries are high quality, manually curated families, which are further grouped into higher order clans (based on sequence or structure similarity). Pfam-B entries are additional entries generated automatically by referring to the ProDom database.

Conserved Domain Database (CDD) is a sequence database meta-resource within NCBI's Entrez database system.[35,36] The CDD collection contains MSAs of protein families and domains imported from Pfam, SMART and COG databases, as well as additional domains curated at NCBI. CDD-specific domains are organized into evolutionary hierarchies. The Clusters of Orthologous Groups (COG/KOG) database[37] groups together families of entire proteins and evolutionarily conserved modules from completely sequenced genomes, which are predicted to form orthologous clusters. The database is split into a two components: COGs group together proteins encoded by numerous bacterial and archaeal genomes and two yeast genomes, while KOGs group together a relatively smaller number of eukaryotic genomes (including yeasts). COG and Pfam definitions of families are the most commonly referred to in the scientific literature to describe yet uncharacterized proteins or domains.

In addition to databases of protein families curated by experts, a number of studies have reported databases resulting from fully automatic clustering of protein sequences, where 'families' indicate groups of proteins classified according to certain numerical value of sequence similarity. Examples among recently created or updated databases include: CluSTr (http://www.ebi.ac.uk/clustr/),[38] ProtoNet (http://www.protonet.cs.huji.ac.il/),[39] SYSTERS (http://systers.molgen.mpg.de/),[40] eggNOG (http://eggnog.embl.de),[41] InParanoid (http://InParanoid.sbc.su.se/),[42] OrthoDB (http://cegg.unige.ch/orthodb),[43] SIMAP (http://mips.gsf.de/simap/).[44]

While protein databases contain thousands of domain families and associated SMs, known LMs are limited in number. There are also only a few general LM databases such as ELM[45] or Scansite.[46] A number of programs specialize in cataloging and predicting motifs with narrowly defined function and distribution, e.g. sites of different posttranslational modification, often restricted to particular taxonomic groups (see ref. [47] for review). Table 1.2 lists some of the databases and predictive servers; however a comprehensive review of such databases is beyond the scope of this chapter.

1.5 Database Searches and Pairwise Alignments

The key step in analyzing our sequence of interest (hereafter referred to as 'query' or 'target') is to determine whether it shows any similarity to other protein sequences. The

Table 1.2 *Databases of motifs and software for motif finding*

SCANSITE[46]	scansite.mit.edu	A database of LMs that are recognized by globular domains (phosphorylation and protein-binding sites: contains LM-domain pairs) and a search utility. Allows searches for combinations of motifs
ELM[45]	elm.eu.org	Catalogues LMs in eukaryotic proteins, searches for defined motifs in a query sequence. Employs a set of filters
Phospho.ELM	phospho.elm.eu.org	ELM version specialized in phosphorylation sites
Minimotif Miner[56]	mnm.engr.uconn.edu	A database of known motifs (partially compiled from other databases) and a search utility. Scores motifs with several methods
CBS prediction servers[47,57]	www.cbs.dtu.dk/services/	A number of independent methods for identification of known posttranslational modification sites, targeting sites, and peptide cleavage sites
MEME/MAST[58]	meme.sdsc.edu/meme/	Searches for user-defined motifs (MAST) or performs de novo motif identification.(MEME)
Gibbs Sampler[59]	bayesweb.wadsworth.org/gibbs/gibbs.html	Searches for user-defined motifs or performs de novo motif identification for a set of up to 1000 sequences. Allows sampling by different strategies: Site, Motif, Recursive, or Centroid
EasyGibbs	www.cbs.dtu.dk/biotools/EasyGibbs/	Requires submission of submission of training examples and evaluation examples to train a motif prediction method
IBM's Bioinformatics and Pattern Discovery[60]	cbcsrv.watson.ibm.com/Tspd.html	A number of tools for sequence pattern discovery
DILIMOT server[61]	dilimot.embl.de	De novo LM finder for a set of unaligned sequences, employs a set of filters
SLiMDisc[62]	bioware.ucd.ie/~slimdisc/	De novo motif finder. Uses TEIRESIAS algorithm to find patterns in a set of unaligned sequences, down-weights motifs found in groups of proteins found to be mutually related
NestedMICA[63]	www.sanger.ac.uk/Software/analysis/nmica	De novo motif finder for a set of sequences
DEME[64]	bioinformatics.org.au/deme/	De novo discriminative motif finder, searches only for patterns that can differentiate the two sets of sequences. Uses an informative Bayesian prior on protein motif columns, allowing it to incorporate prior knowledge of residue characteristics
IBM[65]	www.research.ibm.com/bioinformatics	De novo identification of over-represented motifs

determination of sequence similarity, from which functional similarity and/or homology is inferred, may be carried out in two independent (and complementary) ways, namely searches for patterns of characters or applying statistical models such as profiles of Hidden Markov Models (HMMs). Typically, searches against a database of motifs and full sequences are employed in parallel to see whether the query protein exhibits known LMs, SMs and domains.

Motifs can be represented as strings of characters from a specific alphabet, which discriminates between invariant residues, alternative conserved residues, unspecified residues, excluded residues, repetitions, and other features. A motif can be written as a regular expression such as 'Y.A(4){C}[DE]$', which can be interpreted as Y followed by any residue, followed by four As, followed by a non-C residue, followed by D or E, followed by C-terminus. With this representation, identification of exact matches between the sequence and a database of motifs is fairly simple, as the regular expression either is present in the sequence or not. However, this way of searching is likely to miss relevant motif variants that exhibit slight variations. Allowing for approximate matches allows for detection of more variants, but inevitably causes appearance of false positives. The major limitation of regular expressions is that they do not take into account the information about the relative frequency of residues at different positions. Statistical models such as profiles (also called positional weight matrices, PWMs) give the probability of observing each amino acid in each position. They allow for partial matches and in general have stronger predictive power, i.e. enable detection of diverged but genuine motifs. Some popular software tools for detection of known motifs and '*de novo*' discovery of previously unknown motifs in functionally related sequences are summarized in Table 1.2. Once sequences sharing a common motif are identified and the motif variants are aligned with each other (see below for explanation of alignment techniques), they can be represented as Sequence Logos[66] for visual inspection (e.g. using the WebLogo server[67] at http://weblogo.berkeley.edu/logo.cgi).

Recognition of very short motifs (e.g. most of LMs) remains problematic, as they are often presented in many sequences solely to the sequence composition of the proteome. Thus database searches with most method yield many false positives that have to be filtered out by considering additional information, e.g. presence of globular domains, which usually contain SMs but are depleted in functionally relevant LMs. On the one hand, presence of non-globular, e.g. disordered regions, can be exploited to detect certain LMs, such as phosphorylation sites; this rule has been implemented in the DisPhos server[68] (http://www.ist.temple.edu/DISPHOS/). On the other hand, homologous globular domains often contain conserved sets of SMs, e.g. in spatially adjacent regions involved in formation of binding sites. Typically, the order of SMs is preserved and a pattern of motifs may be exploited to build a diagnostic tool for detection of new members of a protein family. Nonetheless, because of problems with assessment of statistical significance of short motifs, it is recommended that homology predicted via motif searches is confirmed by one of the tools that provides a more global estimate of sequence similarity, e.g. sequence alignment.

Sequence alignments usually assume (or search for) evolutionary conservation, as opposed to similarity of short motifs that may result from convergent evolution. The statistical significance of alignment can be established by estimating the likelihood that the similarity between two sequences is due to their divergence from a common ancestor, rather than pure accident. First, the query sequence and the potentially homologous sequence are searched

for a series of similar amino acid residues or residue patterns that are in the same order. Then, gaps are inserted between the residues and sequence fragments are shifted so that residues with identical or similar characters in both sequences are aligned in successive columns. If two sequences are indeed homologous (i.e. they diverged from a common ancestor), matches in the alignment represent residues that have been conserved in the evolution, while mismatches can be interpreted as point mutations and gaps as indels (that is, insertion or deletion mutations) introduced in one or both lineages in the time since they diverged from one another. The biological relevance of sequence alignment is usually assessed by comparison with a structure-based alignment, in which residues are considered homologous if they are spatially superimposable. Structural alignments are considered a 'gold standard' in bioinformatics (review: [69]). Since only a small fraction of protein sequences have known structures, the accuracy of sequence alignment measured on the references is merely an estimation of how well a given algorithm reproduces a structurally correct alignment for a collection of standard datasets.

There are two types of algorithms for sequence alignment based on dynamic programming: global Needleman-Wunsch[70] and local Smith-Waterman.[71] In global alignment, an attempt is made to align the entire sequence, using as many matching amino acid residues as possible, up to both ends of each sequence. Thus, best candidates for global alignment are sequences that are approximately the same length. 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 and/or sequences that differ in length. Local alignment is particularly useful for identification of regions of homology between proteins composed of different domains, i.e. sequences that are only partially homologous. Such multidomain proteins are very common in Eukaryota, in contrast to Prokaryota (Bacteria and Archaea), which are more frequently composed of single domains and exhibit 'global' homology.

The above methods of establishing sequence relationships have been utilized in database similarity searches. In the initial step the query sequence is compared to every sequence in the selected database, and similar sequences are identified. Pairwise alignments between the target sequence and the best-matching database entries are constructed, typically using dynamic programming algorithms, and scored. Although percent identity of amino acid residues between two sequences is intuitive and easy to calculate, it is a poor measure of protein similarity, especially for more diverged sequences. Protein alignments are typically aligned and scored using substitution matrices that reflect statistical probabilities of one residue being substituted by another. PAM[72] (and its newer versions Gonnet[73] or Jones-Taylor-Thornton/JTT[74]) and BLOSUM[75] are the two most commonly used types of matrices, with PAM being based on an evolutionary model and extrapolation of probabilities calculates for closely related sequences and BLOSUM based on alignments of more remotely related sequences. Different matrices allow for detecting sequences with varying levels of divergence. A scoring function includes also penalties for the introduction of gaps corresponding to insertion or deletion (indel) mutations. Finally, statistical methods are used to determine the likelihood of a particular alignment between sequences or sequence regions arising by chance, given the size and composition of the database being searched. Alignments that have a low probability of occurrence by chance are interpreted as likely to indicate homology. However, the likelihood of finding a given alignment by chance can

vary significantly depending on the size and composition of the database. For the search for homologs to be effective and the score to be accurately estimated, the database must contain many unrelated sequences. It is important to remember that pairwise similarities (especially if confined to very short regions) can also reflect convergent evolution or simply coincidental resemblance. Thus, repetitive sequences in the database or query can distort both the search results and the assessment of statistical significance.

The most popular methods for sequence database searches (Table 1.3) are FASTA[76] and BLAST.[77] They identify a series of short non-overlapping subsequences in the query sequence that are then matched to candidate database sequences. Query-database matches are subsequently extended and combined into a local pairwise alignment using a variation of the Smith-Waterman algorithm. Both FASTA and BLAST employ extreme value distributions to estimate the distribution of the scores between the query and the database entries and a probability of a random match.[78,79] The result of a database search is a list of pairwise alignments ranked according to the expectation value (E) that represents a number of sequences that are not related to the query sequence and are predicted to produce as good an alignment score as the query sequence. As a rule of thumb, alignments that exhibit small E value (<0.001 for large databases), presence of long stretches of aligned regions without gaps, and absence of low-complexity regions are likely to indicate homology. Nonetheless, homologous sequences can be so diverged that their pairwise similarity scores are in the range of random noise.

Detection of more remote relationships requires taking into account not only individual sequence pairs, but also analyzing similarities in the context of entire families of homologous proteins. For instance, PSI-BLAST (Position-Specific Iterated BLAST) allows for finding very distant relatives of a protein by first invoking regular BLAST and retrieving statistically significant alignments, calculating a 'sequence profile', or a position-specific score matrix (PSSM) that describes the frequency of amino acids found at each position in aligned sequences, and then searching the database using this matrix.[80] Alternatively to PSSMs, the set of query-database alignments can be used to create a Hidden Markov Model (HMM), which also can be iteratively compared with the database to identify new statistically significant matches (as implemented in methods such as HMMER[81]). The list of detected statistically similar (and presumably homologous) sequences aligned to the query can be then updated with new sequences and searches can be carried out in an iterative fashion until no new sequences are reported with the similarity score above the threshold of statistical significance. It must be emphasized that in rounds >1 the similarity scores are calculated with respect to the whole group of aligned sequences (represented by PSSM or a HMM) rather than to the single query sequence, therefore erroneous addition of unrelated sequences at an early stage of the search can lead to further degeneration of the result and inclusion of many false positives. Thus, e.g. for PSI-BLAST it is recommended to initialize searches with a stringent E-value threshold for inclusion of database sequences in the query PSSM (e.g. 10^{-20}-10^{-3} for typical protein families), and progressive relaxation of the threshold (to e.g. 10^{-3}) in subsequent iterations, depending on the number of reported sequences and their similarity to the query.

The 'intermediate sequence search' (ISS) strategy[82,83] is an alternative to profile-based methods. It employs a series of database searches initiated with the query and then continued in a pairwise manner with its homologs. Saturated BLAST is a freely available software package that performs ISS with BLAST in an automated manner.[84] Since all

Table 1.3 *Elected representative methods for sequence database searches*

Method	Search strategy	URL (http://)	Description
FASTA[76]	query sequence vs. database	www.ebi.ac.uk/fasta/index.html	Searches for matching sequence patterns or words, rescans matched regions using scoring matrices, and trims the ends of the region to include only sequence contributing to the highest score. It uses a Smith-Waterman algorithm to calculate an optimal score for a local alignment
BLAST[77]	query sequence vs. database	www.ncbi.nlm.nih.gov/blast/Blast.cgi	Uses a heuristic approach to search for exact matches of a small fixed length between the query and sequences in the database, tries to extend the match in both directions, and performs a gapped alignment between the query sequence and the database sequence using a variation of the Smith-Waterman algorithm. Faster than FASTA
PSI-BLAST[80]	iteratated profile search	www.ncbi.nlm.nih.gov/blast/Blast.cgi	A BLAST search is performed and an alignment from the best local hits is built. This alignment is then used as a query for the next round of search. After each round the search alignment is updated
RPS-BLAST[88]	iteratated profile search	www.ncbi.nlm.nih.gov/blast/Blast.cgi	RPS-BLAST (Reverse PSI-BLAST) searches a query sequence against a database of profiles
SENSER[85]	profile-sequence	available from the authors	Performs a PSI-BLAST search, in addition to significant matches extracts candidates for remote homologs from alignments reported with scores below the level of statistical significance. Candidates are then validated by reciprocal PSI-BLAST searches. Aligned homologs are used to build a HMM that is used as a query in subsequent database searches. The procedure may be iterated
PROF_SIM[89]	profile-profile	available from the authors	Compares two input profiles (like those that are generated by PSI-BLAST) and assigns a similarity score to assess their statistical similarity
COMPASS[90]	profile-profile	prodata.swmed.edu/compass/compass.php	Derives numerical profiles from given multiple sequence alignments, constructs local profile-profile alignments and analytically estimates E-values for the detected similarities
HHsearch[91]	profile-profile	toolkit.tuebingen.mpg.de/hhpred	Builds a profile-HMM from a query sequence and compares it with a database of HMMs representing annotated protein families (e.g. PFAM, COGs,) or domains with known structure (PDB, SCOP)
HHsenser[92]	profile-profile	toolkit.tuebingen.mpg.de/hhsenser	Similar to SENSER, but involves 'profile-HMM to profile-HMM' instead of 'profile-HMM to sequence' comparisons to search for remote similarities between whole (super)families

homologs are used as search targets, this strategy is computationally demanding, but it can identify links to remotely related outliers, which may be missed by MSA-based profile or HMM searches that preferentially detect typical sequences. A variant of ISS strategy that includes profile-sequence searches with PSI-BLAST and attempts to extract remote homologs from alignments reported with scores below the level of statistical significance, has been implemented in the method SENSER.[85]

The introduction of profile-based methods, in particular PSI-BLAST, has truly revolutionarized the field of evolutionary bioinformatics, resulting in characterization of numerous conserved domains and detection of remote homologies between many sequences and sequence families that were undetectable in pairwise searches.[86,87] It has also prompted development of several databases of protein families or protein domains (see below), accompanied by the appearance of special bioinformatics tools for searching of these databases. One example is RPS-BLAST (Reverse Position-Specific BLAST) implemented in the IMPALA package,[88] which, as its name implies, reverses the PSI-BLAST approach by comparing a single query sequence against a collection of PSSMs pre-calculated for a number of previously characterized protein families, to determine whether the query sequence is likely to belong to one of these families. Currently the most widely used algorithms for sequence database searches (apart from still extremely popular PSI-BLAST) belong to the newer generation of methods that carry out profile-profile comparisons and allow for detection of even more remote relationships than profile-sequence comparisons. These tools are typically available as web servers; they parse the query sequence provided by the user, automatically run PSI-BLAST to retrieve a profile corresponding to reliably identified candidate homologs (i.e. the query family), and compare it with profiles pre-calculated for a large number of protein families. Examples include PROF_SIM,[89] COMPASS,[90] and HHsearch.[91] Profile-profile search methods have been also adapted to assist in template-based protein structure prediction (described in more detail in chapter by Kosinski *et al.*). The last generation of methods for automated database searches is represented by HHsenser, which combines SENSER-like exhaustive intermediate profile-sequence searches with HHsearch-like pairwise comparison of HMMs.[92]

Once an initial search for homologs of the query sequence is performed, the detected sequences are extracted from the database. Database searches are usually carried out with local alignment programs and extraction of sequences results in retrieval of full length entries from database. Outside the homologous region that has been detected by a local search these sequences may contain regions that are non-homologous to the query, or regions that are homologous to the query but local alignment methods failed to detect them. As mentioned earlier, database searches may result in retrieval of false positives, i.e. sequences that exhibit similarity score above the threshold (e.g. due to biased sequence composition), but nonetheless are not true homologs of the query. Besides, all major databases contain redundant multiple copies of the same protein that differ by only a few residues (e.g. variants with alternative translation codons or results of different sequencing experiments) or exhibit various errors (e.g. terminal truncations or indels caused by incorrect prediction of gene boundaries or exon/intron structure). Such incorrect or redundant sequence variants have to be removed from the preliminary sequence dataset (or corrected, if need be) prior to any advanced analyses. Identification of erroneous sequences is best done at the level of global multiple sequence alignment, which facilitates visualization of missing or redundant regions corresponding to erroneous deletions and insertions. Although there exist a number

of fully automated methods for multiple sequence alignment (MSA, see below), thus far no method allows for automated 'purging' of the alignment of all incorrect sequences and this stage has to be done manually, with the aid of methods for graphical representation and editing of alignments. Such analysis becomes very difficult when the number of sequences to be analyzed is significantly larger than 100, and the workstation's screen becomes too small to display them all. On the other hand, identification of redundant sequences is best done by clustering analysis which may or may not require prior calculation of the MSA, and is capable of processing large number of sequences. In our experience, the most useful procedure is to carry out general clustering first to identify major subgroups (potential families) that are possible to handle by alignment editors, followed by calculation and editing of MSA for each subgroup, followed by merging of all edited sequence groups and repeating MSA and carrying out final quality checks. Below, we describe in more detail methods for MSA-independent sequence clustering, MSA construction, and for MSA-based calculation of phylogenetic trees.

1.6 Sequence Clustering

It is well known that protein families can be classified into subfamilies using phylogenetic analysis to calculate a hierarchy of relationships. The traditional representation of this hierarchy is a treelike dendrogram, with individual elements ('leaves') at one end and a single cluster containing every element ('root') at the other. Phylogenetic analysis requires, however, the availability of MSA and intensive calculations to obtain evolutionary distances and generate an accurate treelike representation of mutual relationships within the protein family. There have been many attempts to circumvent this problem, in particular by using various 'surrogate' measures of pairwise sequence similarity, rather than evolutionary distances, and by applying various hierarchical clustering techniques to build treelike representations.

An important step in clustering is to select a distance measure, which will determine how the similarity of two elements is calculated. Sequence clustering algorithms typically employ the value of pairwise sequence similarity, e.g. calculated by BLAST or the Smith-Waterman algorithm (see above) and aim at identifying groups of sequences that are more similar to each other than to other members of the input set. Typically, the aim of protein sequence clustering is to identify groups of homologs exhibiting statistically significant similarity, thus the threshold value for cutting the tree should correspond to the desired evolutionary distance (e.g. to split a superfamily into families and then into subfamilies). An appropriate cutoff should also separate true homologs from non-homologs, which can be used to purge the initial dataset from potential false positives. Clustering can also be used to split a group of functionally similar but not necessarily evolutionarily related proteins into subgroups of homologs that are further analyzed independently from each other. The presence of well-characterized proteins within a family can then allow one to reliably assign functions to other family members whose functions are not known or not well understood. Finding proteins with different functions within the same family may suggest caution in extrapolating functional information. On the other hand, finding families with only uncharacterized members may prompt them as sources of interesting candidates for experimental analyses.

Single linkage (SL) clustering is a simple and intuitive algorithm, in which the distance between two clusters is computed as the distance between the two closest elements in these clusters. It has been implemented e.g. in the BLASTCLUST method from the popular BLAST package[80] (ftp://ftp.ncbi.nih.gov/blast/, also available via a third-party web server http://toolkit.tuebingen.mpg.de/blastclust/). The SL algorithm is known to produce accurate clustering when different subgroups show similar level of internal similarity and an appropriate threshold is given to separate families from each other. A drawback of this method is that clusters may be forced together due to single elements being similar to each other, even though other elements in each cluster may be dissimilar to each other. Thus, the SL analysis is not appropriate for analyzing sets of largely non-homologous multidomain proteins, which may be falsely chained to each other (e.g. a cluster of many proteins comprising domain A and one protein with domains A and B may be chained to a cluster composed of proteins with domain B and one protein with domains B and C, then chained to a cluster of domains C and so on). In particular, many proteins possess small, widespread protein domains (e.g. SH2, WD40, and DnaJ) that are known to have very different functions. The presence of such a common domain within a group of proteins does not necessarily imply that these proteins perform the same function. Ideally, these types of proteins should be classified into a single cluster only if they exhibit highly similar domain architectures. Another drawback is that in many protein superfamilies the degree of similarity within different families varies greatly, and e.g. subfamilies within one family may be more diverged from each other than two other families. Therefore, application of only one average threshold may produce many too small clusters and a few too large clusters.

Due to the fact that SL method has difficulty in detecting an appropriate threshold for identification of clusters, modern protein clustering applications employ other algorithms. In particular graph theory allows the classification of objects into groups based on a global treatment of all relationships in similarity space simultaneously. Thus, proteins and their similarities may be represented as vertices and edges of a graph, respectively, and the initial partition produced, e.g. by SL clustering, may be post-processed by a graph partitioning algorithm (see Chapter 10 by Nabieva and Singh in this volume for a detailed discussion of different clustering algorithms, in the context of graphs representing networks of protein–protein interactions). CLANS (CLuster ANalysis of Sequences)[93] (ftp://ftp.tuebingen.mpg.de/pub/protevo/CLANS) is a freely available Java application, which runs all-against-all BLAST searches for all sequences in the input set, and then applies the Fruchterman–Reingold graph layout algorithm to visualize pairwise sequence similarities based on BLAST P-values in either two-dimensional or three-dimensional space. CLANS allows the user to select different thresholds and parameters for calculation of distances and to carry out clustering using several different algorithms, including single and multiple linkage, network-based, and convex clustering. LGL[94] is a similar clustering algorithm with a Java front end for visualization, however it requires pre-computed similarity values as an input. ProClust (http://pig-pbil.ibcp.fr/magos)[95] is another graph-based clustering algorithm, which scales similarity values based on the length of the protein sequences compared, and takes into account the significance of alignment scores to filter for spurious links. Post-processing to merge clusters is based on comparison of clusters with each other using profile-HMMs (see further sections in this chapter for review of methodology for profile-profile comparisons). MCL[96] relies on the Markov cluster (MCL)

algorithm, which finds clusters by calculating the probabilities associated with a transition from one protein to another within the graph and passing the matrix of probabilities through iterative rounds of 'multiplication' and 'inflation' until convergence. The 'inflation' value parameter is used to control the 'tightness' of final clusters. The MCL algorithm is relatively insensitive to the presence of multi-domain proteins, promiscuous domains or fragmented sequences. Super Paramagnetic Clustering (SPC)[97] (http://www.vcclab.org/lab/spc/) is a different approach that clusters input data based on analogy to the physics of an inhomogeneous ferromagnet; a stepwise implementation of this algorithm, called global SPC (gSPC) was shown to be even more robust than TRIBE-MCL. FlowerPower[98] (http://phylogenomics.berkeley.edu/cgi-bin/flowerpower/input_flowerpower.py) has been designed specifically for the identification of subfamilies with global homology (e.g. from a set of sequences with different domain compositions) using the SCI-PHY algorithm based on HMMs.[99] Finally, unlike other methods that calculate their similarity matrices based on alignments, CLUSS[100] (http://prospectus.usherbrooke.ca/CLUSS/) performs clustering based on a matching amino acid subsequences, which makes it applicable both to alignable and unalignable sequences, e.g. products of circular permutation etc. A number of other clustering approaches have been used to cluster various sequence data sets and construct databases of clusters (see the section on protein family databases); however, the underlying clustering programs have not been made available as standalone applications.

1.7 Multiple Sequence Alignment

As soon as sets of homologous sequences with similar domain composition are identified, or the domain subsequences isolated from non-homologous fragments, they can be aligned together to study sequence conservation across the entire family. Multiple sequence alignment (MSA) is an extension of pairwise alignment, in which multiple related sequences are optimally matched, by bringing the greatest number of similar characters into register in the same column. In this manner, protein sequences are arranged into a rectangular array with the goal that residues in a given column are homologous (derived from a single position in an ancestral sequence), superimposable (in a structural alignment) or play a common functional role. The advantage of the MSA is that it reveals more biological information than a set of pairwise alignments, e.g. conserved patterns and motifs that are common to the whole sequence family and may indicate functionally or structurally important elements. However, finding an optimal alignment of more than two sequences that includes matches, mismatches, and gaps, and that takes into account the degree of variation in all of the sequences at the same time, is very difficult. Usually, an arrangement of amino acid residues that maximizes the sum of similarities for all pairs of sequences (the sum-of-pairs, or SP, score) is sought. Unlike in pairwise alignments, the SP score has no rigorous theoretical foundation for the MSA and, in particular, fails to incorporate an evolutionary model. Moreover, the dynamic programming algorithm used for optimal alignment of pairs of sequences can be extended to multiple sequences, but the computational time and memory required to maximize the SP score has been shown to scale exponentially with the number of sequences and becomes prohibitively expensive for data sets larger than a few proteins.[101] Thus, approximate alternatives are used. The majority of programs (Table 1.4) are based on the 'progressive algorithm' approach, where the MSA

Table 1.4 *Methods for calculation of MSAs*

Method	URL (http://)	Description
CLUSTALW[110]	www.ebi.ac.uk/clustalw/	Performs pairwise alignments of input sequences, produces a tree based on similarity scores, and realigns sequences sequentially, guided by the tree. Old and inferior to newer methods, but still very popular
DbClustal[111]	bips.u-strasbg.fr/PipeAlign/	Carries out BLAST searches and incorporates local alignment information into a CLUSTAL global alignment in the form of a list of anchor points between pairs of sequences. Allows for incorporation of very long insertions and terminal extensions
SAM[112]	www.soe.ucsc.edu/compbio/sam.html	Employs HMM for MSA. Evolved into a fold-recognition tool SAM-T, current version: SAM-T06 (www.soe.ucsc.edu/research/compbio/SAM_T06/T06-query.html)
HMMer[113]	hmmer.janelia.org/	Employs HMM. Implemented in PFAM database for grouping of sequences into families
MUSCLE[114]	www.drive5.com/muscle/	Rapidly generates a very crude guide tree, generates MSA using a profile function (log-expectation score) and refines it using tree-dependent restricted partitioning. Very fast
PRRN[103]	prrn.hgc.jp align.genome.jp/prrn/	Adopts a doubly nested randomized iterative refinement strategy to make alignment, phylogenetic tree and pair weights mutually consistent. Performs a large number of pairwise group-to-group alignments to gradually improve overall weighted sum-of-pairs score
T-Coffee[104]	www.tcoffee.org/	Employs a consistency measure by considering information from all of the sequences during pairwise sequence alignments, not just those being aligned at that stage. Combines a collection of multiple/pairwise, global/local alignments into a single MSA. Version 2.00 and higher can combine sequences and structures
MAFFT[115]	align.bmr.kyushu-u.ac.jp/mafft/online/server	Uses a fast Fourier transform to generate a guide tree. Refines the alignment by optimizing the weighted sum of pairs (WSP) objective function
PRRN[116]	prrn.hgc.jp/	Optimizes a weighted sum-of-pairs score, in which the weights given to individual sequence pairs are adjusted to compensate for the biased contributions. MSA is refined through partitioning and realignment restricted to the edges of the tree
PRALINE[117]	zeus.cs.vu.nl/programs/pralinewww/	Runs PSI-BLAST for each sequence in the input set to generate a PSSM pre-profile. Pre-profiles are then aligned hierarchically by a profile-profile alignment method
ProbCons[118]	probcons.stanford.edu/	During pairwise alignments employs 3-state HMMs, uses maximum expected accuracy as an objective function, and applies probabilistic consistency transformation to incorporate multiple sequence conservation information
SPEM[108]	sparks.informatics.iupui.edu/Softwares-Services_files/spem.htm	Runs PSI-BLAST and makes secondary structure prediction for each sequence in the input set. Uses SP2 with combined scoring of sequence and structure, then applies probabilistic consistency-based scoring for refinement of pairwise alignments.

MUMMALS[119]	prodata.swmed.edu/mummals/	Employs complex HMMs with multiple match states that capture local structural information. Applies a probabilistic consistency-based scoring function
PROMALS[120]	prodata.swmed.edu/promals/	Runs PSI-BLAST and makes secondary structure prediction for each sequence in the input set. Uses a HMM with combined scoring of sequence and structure, applies probabilistic consistency-based scoring. Slow, but accurate. Works poorly with multidomain proteins
SATCHMO[121]	phylogenomics.berkeley.edu/cgi-bin/satchmo/input_satchmo.py	Simultaneously constructs a tree and a set of MSAs, one for each internal node of the tree (for all sequences within its sub-tree). Generates profile-HMMs at each node; these are used to determine branching order, to align sequences and to predict structurally alignable regions
PRIME[122]	prime.cbrc.jp	Employs doubly nested randomized iterative refinement strategy, based on a group-to-group sequence alignment algorithm with piecewise linear gap cost, instead of traditional affine gap cost
Kalign[123]	msa.cgb.ki.se/	A progressive method, which relies on the Wu-Manber approximate string-matching algorithm in the distance calculation and optionally in the dynamic programming to align the profiles
DIALIGN[124]	bibiserv.techfak.uni-bielefeld.de/dialign/	Constructs pairwise and multiple alignments by comparing segments instead of full-length sequences. Employs a fragment-chaining algorithm
MANGO[125]	www.bioinfo.org.cn/mango/	Identifies motifs shared by two or more sequences, constructs skeletal alignment, extends it to a full MSA, which is iteratively refined
ALIGN-M[126]	bioinformatics.vub.ac.be/software/software.html	Uses a non-progressive local approach to guide a global alignment. Designed to deal with particularly diverged sequences
POA[127]	bioinfo.mbi.ucla.edu/poa2/	Replaces the row–column representation of a MSA with a graph in which each node corresponds to a set of aligned residues. Enables alignment of protein sequences with multiple domains
AliWABA[128]	aba.nbcr.net/	A-Bruijn Alignment represents an alignment as a directed graph, possibly containing cycles. Enables alignment of protein sequences with shuffled and/or repeated domain structure
ProDA[129]	proda.stanford.edu/	Does not assume global alignability; allows repeated, shuffled and absent domains. Clusters alignable regions and returns a collection of local MSAs
ComAlign[130]	www.daimi.au.dk/~ocaprani/ComAlign/programs/	Meta-method. Combines qualitatively good sub-alignments from a set of input MSAs. Software for download. Server: mobyle.pasteur.fr/cgi-bin/MobylePortal/portal.py?form=comalign
M-Coffee[131]	www.tcoffee.org	Meta-method. Runs several methods from the COFFEE family to compute alternative alignments and calculates a consensus MSA

is constructed by a series of pairwise alignments, starting with the most related sequences, followed by progressively adding less related sequences (to construct partial alignments of three or more sequences) or aligning partial alignments with each other.[102]

The knowledge of evolutionary relationships among sequences is a very useful criterion for selecting the order of pairwise alignments. Although the calculation of a phylogenetic tree requires the availability of the MSA (see below), an initial tree for construction of the MSA may be calculated based on preliminary evolutionary distances calculated from pairwise comparisons of sequences. The major problem with progressive alignment programs is the dependence of the ultimate MSA on the initial pairwise sequence alignments. The more distantly related these sequences, the more errors will be made, and these errors will be propagated to the MSA. Two main techniques are utilized to correct or minimize mistakes made in the progressive alignment process. One is iterative refinement of the MSA, e.g. by repeatedly dividing the aligned sequences into subgroups and realigning the subgroups, as implemented in PRRN.[103] The other technique makes a consistency measure among a set of pairwise sequence alignments before the progressive alignment steps.[104] Many methods combine iterative optimization with either progressive algorithm and/or consistency-based scoring (review: [105]). An alternative approach for MSA, which does not require calculation of trees, relies on identification of locally conserved patterns found in the same order in the sequences (e.g. as implemented in the DIALIGN method[106]).

Another possibility is to employ a HMM, a statistical model in which an MSA is represented as a form of directed acyclic graph (also called a partial-order graph), which consists of a series of nodes representing possible entries in the columns of an MSA. In this representation a column that contains the same residue in all sequences is coded as a single node with as many outgoing connections as there are possible characters in the next column of the alignment. Sequences are aligned using the Viterbi algorithm, a variant of a dynamic programming algorithm. Several software programs are available in which variants of HMM-based methods have been implemented, including SAM and HMMER (see Table 1.4). Some of these methods allow for the presence of non-alignable (non-homologous) regions of sequence to be present in the input set. In the approach implemented in AliWABA the graph may contain cycles, which enables alignment of protein sequences with shuffled and/or repeated domain structure.[107]

Currently the best methods for MSA such as SPEM[108] or PROMALS[109] employ PSI-BLAST database searches and secondary structure prediction to construct meta-profiles for all input sequences, then carry out profile-profile alignments (with HMMs or with regular profile methods), often refine these alignments based on consistency scoring, and only then combine the input sequences into an MSA. These methods are therefore much slower than simple (but still very popular) methods like CLUSTAL, but are much more accurate, at least for individual domains. However, they might be more prone to errors in case of data sets comprising proteins of uneven length, e.g. some with single domains, and others with the same domain fused to others. Therefore, comparison of MSAs generated with different methods may provide hints as to reliability of the results. As with most bioinformatics methods, algorithms for MSA rarely generate solutions that ideally reflect the biological reality, especially for large datasets of strongly diverged sequences. However, expert knowledge concerning relationships within a given protein family can be used to improve suboptimal MSAs obtained from automatic software packages. A number of methods exist that allow for graphical visualization and manual editing of MSAs to make

them agree with observations that cannot be easily incorporated into the scoring function of most algorithms (e.g. knowledge that particular residues in different sequences that must correspond to each other or agreement of structural patterns obtained from experiment or from predictions). Example tools for displaying and editing protein (often also nucleic acid) sequences and alignments have been listed in Table 1.4.

1.8 Relationship of Multiple Sequence Alignments to Phylogenetic Analysis

A biologically meaningful MSA contains sequences that are all homologous, i.e. derived from a common ancestor sequence. Further, in an ideal MSA, all columns contain amino acid residues that were derived from an ancestral residue in the ancestral sequence (if these conditions are not fulfilled, MSA is 'biologically wrong' and cannot be used for phylogenetic analyses). Within the column are original characters that were present early, as well as other derived characters that appeared later in evolutionary time. In some cases, the position is so important for function that mutational changes are not observed. It is these conserved positions that usually serve as 'anchor points' for producing an alignment. In other cases, the position is less important, and substitutions are observed. Deletions and insertions are also typically more frequent in the variable regions of the alignment. If the sequences in the MSA show evident similarities (e.g. >30% identity and relatively few insertions and deletions), they are likely to be recently derived from a common ancestor sequence. Conversely, sequences with multiple differences are likely to be remotely related. Thus, the number and types of changes in the MSA may be used to infer the mutations that occurred during the evolution of the sequence family. It is also possible to dissect the order of appearance of the sequences during evolution and to relate the relationships between sequences to the relationships between their hosts (organisms). A number of packages for phylogenetic calculations based on user-defined MSAs have been made available, including PHYLIP (http://evolution.genetics.washington.edu/phylip.html),[146] MEGA (http://www.megasoftware.net/),[147] PHYML (http://atgc.lirmm.fr/phyml/),[148] PAML (http://abacus.gene.ucl.ac.uk/software/paml.html),[149] TREE-PUZZLE[150] (http://www.tree-puzzle.de), or MrBayes[151] (http://mrbayes.csit.fsu.edu). Among web resources, MultiPhyl (http://www.cs.nuim.ie/distributed/multiphyl.php)[152] is a particularly useful site, which allows the users to carry out computationally very expensive inference of Maximum Likelihood trees using distributed computing. A review of methods for phylogenetic calculations is outside the scope of this chapter, interested readers should consult reviews: e.g. ref. [153–155]

The result of phylogenetic analysis can be used as a feedback for revising particularly challenging MSAs that are suspect of errors (e.g. sequences may be split it into subgroups and realigned separately or the tree may be used to guide the progressive alignment algorithm). As an example, the SCI-PHY server (http://phylogenomics.berkeley.edu/SCI-PHY/) allows users to upload a MSA for subfamily identification and subfamily HMM construction.[99] Further, analysis of the phylogenetic tree in connection with the known (or assumed) tree of hosts (organisms) can be used to deduce major evolutionary events in the protein family, e.g. gene duplications, gene losses, which provide the basis for discrimination between orthologs and paralogs and may guide functional predictions (review: [156]). Another application of MSA and phylogenetic analysis is the inference of ancestral sequences,

Table 1.4 *Multiple alignment editors*

Jalview[132]	www.jalview.org/	Java tool (OSindependent). Standalone version allows for calculation and manipulation of protein MSAs, (does not work with nucleotide sequences) calculates trees and PCA, displays structures. Web applet version allows visualization of pre-calculated alignments. Coupled with structure prediction server JNet. Used as a default viewer in many web servers and databases
Panta rhei (QAlign2)[133]	gi.cebitec.uni-bielefeld.de/qalign	Standalone tool (Windows and Mac OS X). Allows for manipulations of huge protein and nucleotide datasets in multiple parallel sessions. Calculates MSAs and phylogenetic trees
STRAP[134]	www.charite.de/bioinf/strap/	Standalone Java tool (OS independent) for huge MSAs of protein sequences and structures. Supports annotation of mRNA, intron/exon gene structure. Allows exporting data to Jalview
BioEdit	www.mbio.ncsu.edu/BioEdit/bioedit.html	Standalone tool for MS Windows. Multiple options for manual editing, graphical display and basic analyses of sequence conservation for proteins and nucleic acids, links to external servers. Last update: July 2007. As of February 2008: no longer being reliably maintained, and the documentation is out of date
GeneDoc	www.nrbsc.org/gfx/genedoc/index.html	Standalone tool for MS Windows. Multiple options for manual editing, graphical display and basic analyses of sequence conservation for proteins and nucleic acids. Supports phylogenetic trees and integrates sequence and structure information. Last update: July 2001
JEMBOSS[135]	emboss.sourceforge.net/Jemboss/	Java tool – standalone and web version. A graphical user interface to EMBOSS. Very simple
INTERALIGN[136]	see the right panel for an unusually long link to this program	Java tool (for Linux and Windows) to interactively manipulate and refine multiple sequence alignments using 3D structures. www-dsv.cea.fr/instituts/institut-de-biologie-environnementale-et-biotechnologie-ibeb/unites-de-recherche/service-de-biochimie-et-toxicologie-nucleaire-sbtn/interalign-download-page
CINEMA[137]	utopia.cs.manchester.ac.uk/cinema	Standalone tool for MS Windows, Linux, and Mac OS X. Interactive editor for proteins and nucleic acids. Java-based applets. Serious security issue: data are saved on a remote server and are publicly available. Built into UTOPIA, comprising also protein structure viewer Ambrosia and search and management tool Find-O-Matic allowing for access of remote databases

Base-By-Base[138]	athena.bioc.uvic.ca/workbench.php?tool=basebybase	Java web tool. Developed for comparative analysis of viral genomes, but handles also proteins. Relatively slow
SQUINT[139]	www.cebl.auckland.ac.nz/index.php?target=software&item=6	Standalone Java tool. Allows for calculation and editing of MSAs both for DNA sequences and the corresponding protein sequences
MACAW[140]	genamics.com/software/downloads/	A standalone (Windows and Mac OS) interactive program for locating, and combining 'blocks' of similar sequence segments. Employs Gibbs sampling and pattern searches
SeaView[141]	pbil.univ-lyon1.fr/software/seaview.html	A standalone application for a variety of systems (including MS Windows, Linux, Mac OS X, and Solaris) as well as a helper application via web browser. Allows for manual editing of the MSAs and basic comparative analyses
ViTO[142]	bioserv.cbs.cnrs.fr/VITO/DOC/	An interactive program coupling a MSA editor with a 3D viewer, especially useful for preparing input files for comparative modeling. Supports macros. Connected to SCWRL and MODELLER for 3D structure modeling (see the chapter by Kosinski *et al*. in this volume)
POAViz[143]	www.bioinformatics.ucla.edu/poa	A visualization tool for POA alignments (see Table 1.4)
AltAVisT[144]	Bibiserv.techfak.uni-bielefeld.de/altavist/	A web server able to compare two alternative MSAs of a given sequence set to each other. Color-coded regions where MSAs coincide and can be considered to be most reliable
BOXSHADE	www.ch.embnet.org/software/BOX_form.html	Standalone Linux tool and a web server to generate a rendered PostScript, rtf or pict output from an MSA
ESPript[145]	www.lg.ndirect.co.uk/chroma	Standalone Linux tool and a web server to generate a rendered PostScript output from an MSA

with methods such MrBayes or ANCESCON (ftp://iole.swmed.edu/pub/ANCESCON/)[157] (review: ref. [158]).

1.9 Prediction of Domains

It has been reported that around 65% or eukaryotic and around 40% of prokaryotic proteins are composed of two or more globular domains.[159] In addition, 30–60% of eukaryotic proteins are predicted to contain long stretches of disordered residues.[160] Unfortunately, many experimental as well as computational techniques work effectively only on single domains. For instance, experimental structure determination using NMR and in many cases also X-ray crystallography is more successful for isolated globular domains, devoid of disordered regions, rather than for complete multi-domain proteins, unless their constituent parts form a tight complex. Also, many computational methods for protein sequence alignment, phylogenetic analyses (see above), or three-dimensional structure prediction (fold recognition and *de novo* folding – see chapters by Kosinski *et al*. and by Gront *et al*. in this volume) have been designed to work with single domains and may produce erroneous results when presented with multidomain proteins. Thus, identification of domain boundaries from amino acid sequence (hereafter referred to as 1D domain prediction) is an essential step in many protein analyses. However, as mentioned earlier, there is no precise definition of what constitutes a domain even if the structure is known; therefore 1D domain prediction from sequence without structural information presents a great challenge and interpretation of results must consider a certain degree of fuzziness.

Jones and coworkers[161] have classified 1D domain prediction methods into three broad and partially overlapping classes, analogous to 3D structure prediction methods: domain homology prediction, domain recognition (these two classes can be considered 'template-based'), and new domain ('template-free') prediction methods. The most effective way of domain prediction is by detecting its homology to known domain structures (e.g. those classified in SCOP or CATH databases) or to domains from manually curated sequence databases, such as Pfam or CDD (Table 1.1). Main problems in predicting homology occur when the domain is discontinuous (e.g. in the case of insertion of another domain), exhibits circular permutation or forms an evolutionarily conserved module with another associated domain. In this context it must be remembered that some of the entries in domain databases correspond in fact to evolutionarily conserved modules that comprise several structural domains. For sequence regions that cannot be assigned to known domain 'by homology', domain recognition methods can be used. One approach is to apply 3D fold-recognition methods that allow for prediction of structural similarity to known domain structures due to extremely distant homology and sometimes also due to analogy (see Chapter 4 by Kosinski *et al*.) Another approach is to predict secondary structure for the query sequence (see Chapter 2 by Majorek *et al*.) and search for known domains with similar patterns. Finally, new domain prediction rely either on machine learning methods for recognition of sequence features that generally characterize domains or on methods for *de novo* folding (see Chapter 5 by Gront *et al*.) that generate a set of possible tertiary structures, in which compact units are identified. This last class of method is extremely computationally expensive. A list of currently available web servers is shown in Table 1.5; besides, some of domain databases mentioned in Table 1.1 have their own search utilities.

Table 1.5 *Domain prediction methods*

Method	URL (http://)	Description
		Template-based domain prediction
SSEP-Domain[165]	www.bio.ifi.lmu.de/SSEP	Server input is restricted to 50-600 amino acids. Applies secondary structure element alignment (SSEA) and profile-profile alignment (PPA) in combination with InterPro pattern searches
SBASE[166]	hydra.icgeb.trieste.it/sbase/	Searches a database of known domains and applies SVM to post-process results using a 'similarity network' of inter-sequence similarity scores for known domains
Ginzu[164]	www.robetta.org	Searches for homologous domains in PDB using first PSI-BLAST, then fold-recognition method 3D-Jury, retrieved structures are parsed into domains. In the remaining regions domains are predicted according to the pattern of conservation in PSI-BLAST alignments. Domain boundaries are assigned based on patterns of sequence edges and low-occupied positions in the PSI-BLAST output and secondary structure predicted by PSI-PRED
DOMAINATION[167]	mathbio.nimr.mrc.ac.uk	Infers putative domains and their boundaries in a query sequence from local gapped alignments generated using PSI-BLAST, then submits delineated domains as successive database queries in further iterative steps
Biozon[168]	biozon.org/tools/domains/ (in February 2008 down until further notice)	Analyzes the results of a database search by an ANN, the output is further smoothed and post-processed using a probabilistic model to predict the most likely transition positions between domains
PPRODO[169]	gene.kias.re.kr/∼jlee/pprodo (standalone tool available for download)	Analyzes the results of a PSI-BLAST database search by an ANN
DOMpro[170]	www.ics.uci.edu/∼baldig/dompro.html	Predicts protein domain boundaries based on bidirectional recurrent ANNs and statistical methods from PSI-BLAST PSSMs, predicted secondary structure and solvent accessibility
		New domain prediction
Globplot[171]	globplot.embl.de/	Identifies putative domains by identifying the globular and non-globular regions within protein sequence based on the amino acid propensities for random coil (disordered) or secondary structure. See also Chapter 2 by Majorek *et al.* in this volume

(*continued overleaf*)

Table 1.6 *(continued)*

Method	URL (http://)	Description
DomCut[172]	www.bork.embl-heidelberg.de/%7Esuyama/domcut/	Identifies domain boundaries by discriminating between regions with amino acid composition characteristic for globular domains and interdomain linkers in multidomain proteins
Scooby-domain[173]	ibivu.cs.vu.nl/programs/scoobywww	Identifies putative globular domains in protein sequence based on the observed lengths and hydrophobicities of domains from proteins with known tertiary structure
CHOPnet[174]	www.rostlab.org/services/CHOP/submit.html	Uses ANN to predicts domain boundaries from sequence conservation, predicted secondary structure, solvent accessibility, amino acid flexibility, and amino acid composition
		Meta-servers
Meta-DP[162]	meta-dp.cse.buffalo.edu/	Meta-server for prediction of globular domains by calculating simple consensus of 10 different primary methods: Adda, Biozon, DomPred-DomSSEA, InterProScan, Mateo, Globplot, ROBETTA-Ginzu, Dopro, Ssep-domain, Dompro
DomPred[161]	bioinf.cs.ucl.ac.uk/dompred/DomPredform.html	Meta-server, consists of: (1) domain homology searches against the Pfam database, (2) DPS method, which predicts domain boundaries from the distribution of termini of sequence matches reported by PSI-BLAST, and (3) DomSSEA, which compares a pattern of secondary structures predicted for the target protein with secondary structure patterns of domains with known 3D structures
DOMAC[163]	www.bioinfotool.org/domac.html	Meta-server, first runs PSI-BLAST to detect similarity to known structures, then builds 3D models using MODELLER, and parses them into domains using PDP[175]. For the remaining regions uses DOMro (see above)

As with most of bioinformatics predictions, the recommended protocol for 1D domain prediction involves application of the consensus rule. A meta-server for domain prediction Meta-DP has been developed[162] that allows for comparison and averaging of results reported by several algorithms. However, the best results are achieved if 1D domain prediction is carried out hierarchically, starting with the template-based methods, followed by the more demanding (and more error-prone) *de novo* methods. This hybrid approach has been already implemented in a few fully automated methods that were shown to outperform individual methods within the framework of the CASP competition. Examples include DOMAC[163] (available as a server, see Table 1.5) and DP_Hybrid (comprising Ginzu and RosettaDOM,[164] components of the Rosetta suite, not available as a standalone server).

1.10 Summary

In this chapter we discussed methods for primary structure analysis of proteins, including identification of short motifs, database searches to detect significantly similar sequences (candidate homologs), sequence clustering to identify protein families regions of homology to sequences, multiple sequence alignment, and identification of globular domains. We have not covered the issue of predicting non-globular or disordered regions and secondary structure prediction, as these analyses are reviewed in depth in another chapter in this volume (Majorek *et al.*). In addition to reviewing theory, we provided tables summarizing different programs dedicated to carry out various types of sequence analyses. These are mostly web servers, and some standalone packages for local installation. We must mention, however, that many databases and methods that have been described in the literature and used to be available as web servers, have now disappeared from the Internet or at least have not been available during preparation of this chapter, therefore were omitted from the tables. It is also expected that with time some of the methods mentioned here will also completely disappear or will move to different websites; on the other hand, new interesting methods will be made available. The readers / potential users are therefore encouraged to consult the periodically updated collections of web servers e.g. the annual special issue of Nucleic Acid Research (http://nar.oxfordjournals.org/) and the Bioinformatics Links Directory, (http://bioinformatics.ca/links_directory/).

There are several considerations in choosing a set of programs to analyze a sequence of interest, including biological accuracy, complexity of the analysis and time required to complete it (without asking a sequence analysis expert for help), and software/hardware usage. In Figure 1.2 we present a flowchart illustrating the recommended protocol of protein sequence analysis, from basic searches to domain prediction, which can be used to generate input data for more subsequent computational or experimental analyses. If the aim is simple, e.g. to obtain an approximate sequence alignment of a few homologs and illustrate the most obvious motifs (both SMs and LMs), then a simple sequence search (e.g. with BLAST) of one of protein family/domain databases is often sufficient to check, whether an annotated data set is already available for download, without the need to carry out new analyses. However, we suggest that web servers for identification of motifs should be queried, as they often provide information that is more up to date than pre-calculated data sets in family databases. In case of novel sequences that are not yet present in major databases, a PSI-BLAST search of one of sequence databases

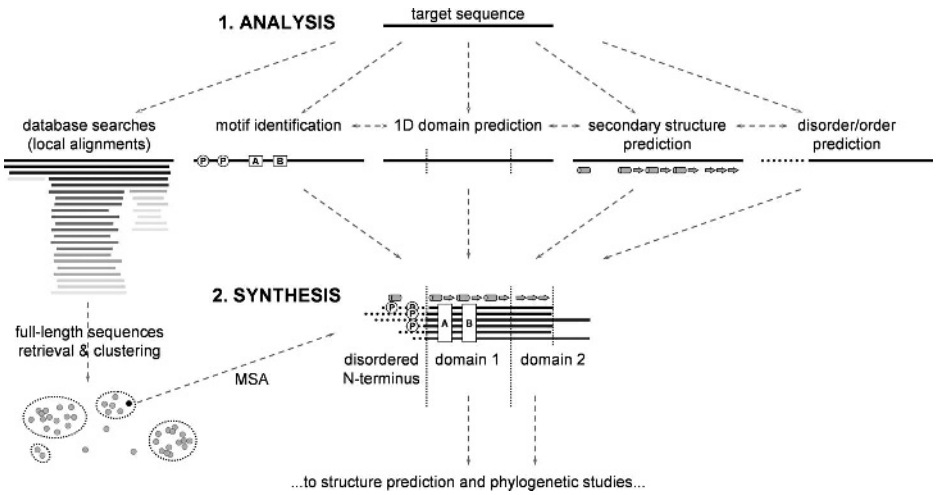

Figure 1.2 *Suggested workflow for protein sequence analysis. Basic sequence analyses involve usually some or all of five tasks: (1) identification of locally similar sequences in databases (here, sequences with decreasing level of similarity are indicated by fading shades of gray) followed by retrieval of full-length sequences and their clustering to identify families, and finally MSA of the family; (2) identification of motifs (LMs and SMs); (3) prediction of putative domains; (4) prediction of secondary structure; (5) prediction of disordered/ordered regions. Tasks (4) and (5) are reviewed in Chapter 2 by Majorek* et al. *in this volume. Subsequently, results from these analyses (as well as useful data and predictions from other sources, if available) are combined and individual domain families may be subjected to detailed structural and phylogenetic analyses. Alternatively, predicted domain structure may be used to carry out another round of basic analyses, with adjusted parameters (e.g. new database searches, with correction for compositionally biased sequence, and e.g. removed N-terminal region or protein sequence split into individual domains)*

(e.g. nr at the NCBI) is recommended, to be followed by clustering of the extracted homologs and identification of the putative orthologous family, which may be aligned using one of the recent methods for MSA calculation. In parallel, domain databases should be searched by sensitive profile methods to detect potential presence of known domains. If no evident similarity to known protein families or domains is observed, domain prediction methods should be used, preferably in connection with prediction of disordered regions and secondary structure. If the aim of the analysis is an experimental characterization of protein function, such combination of methods is usually sufficient to delineate major domains and conserved regions. However, if an advanced comparative analysis is desired, e.g. calculation of a phylogenetic tree or prediction of protein structure, the MSA must be carefully refined to remove or 'mask' unalignable (e.g. non-homologous) regions. For multidomain proteins domain boundaries must be judiciously localized, and domains should be submitted independently for phylogenetic and modeling calculations, unless there are specific reasons to believe that a set of domains should be analyzed together (e.g. if it forms an evolutionarily conserved module). At all stages of analysis (perhaps with the exception of database searches), we recommend using several alternative methods and comparing their results. As a rule of thumb, consistency between different algorithms

indicates higher likelihood that a given result is close to optimal. On the other hand, automatically generated results are seldom ideal and they can be often improved by human experts. Finally, it must be remembered that uncorrected errors tend to accumulate, and 'higher level' methods usually assume that their input is error-free, thus it is very important to carefully check results returned by all automated methods before submitting them to next, usually more time-consuming stages.

Acknowledgements

We thank present and former members of the Bujnicki lab in IIMCB and at the UAM for stimulating discussions and contribution of ideas and information to this article. The authors acknowledge the support from past and current grants for the development of bioinformatics methods from Polish Ministry of Science, NIH, Framework Programme of the EU, EMBO, and HHMI. KHK has worked on this article while being supported by a fellowship from EMBO and a travel grant from Polish Academy of Sciences and JSPS. JMB has worked on this article while being supported by the Institute of Medical Science at the University of Tokyo.

References

1. R.A. Jensen, Enzyme recruitment in evolution of new function, *Annu Rev Microbiol*, **30**, 409–425 (1976).
2. E.V. Koonin, Orthologs, paralogs, and evolutionary genomics, *Annu Rev Genet*, **39**, 309–338 (2005).
3. C. Chothia, and A.M. Lesk, The relation between the divergence of sequence and structure in proteins, *Embo J*, **5**, 823–826 (1986).
4. A. Weichsel, E.M. Maes, J.F. Andersen, *et al*., Heme-assisted s-nitrosation of a proximal thiolate in a nitric oxide transport protein, *Proc Natl Acad Sci U S A*, **102**, 594–599 (2005).
5. L.N. Kinch, and N.V. Grishin, Evolution of protein structures and functions, *Curr Opin Struct Biol*, **12**, 400–408 (2002).
6. C.A. Orengo, and J.M. Thornton, Protein families and their evolution – a structural perspective, *Annu Rev Biochem*, **74**, 867–900 (2005).
7. D.L. Wheeler, T. Barrett, D.A. Benson, *et al*., Database resources of the National Center for Biotechnology Information, *Nucleic Acids Res*, **36**, D13-21 (2008).
8. K. Henrick, Z. Feng, W.F. Bluhm, *et al*., Remediation of the Protein Data Bank Archive, *Nucleic Acids Res*, **36**, D426-433 (2008).
9. M. Perutz, Early days of protein crystallography, *Methods Enzymol*, **114**, 3–18 (1985).
10. A. Elofsson, and G. von Heijne, Membrane protein structure: prediction versus reality, *Annu Rev Biochem*, **76**, 125–140 (2007).
11. C.P. Ponting, and R.R. Russell, The natural history of protein domains, *Annu Rev Biophys Biomol Struct*, **31**, 45–71 (2002).
12. C. Vogel, C. Berzuini, M. Bashton, J. Gough, and S.A. Teichmann, Supra-domains: Evolutionary units larger than single protein domains, *J Mol Biol*, **336**, 809–823 (2004).
13. Y. Lindqvist, and G. Schneider, Circular permutations of natural protein sequences: Structural evidence, *Curr Opin Struct Biol*, **7**, 422–427 (1997).
14. J.C. Wootton, and S. Federhen, Analysis of compositionally biased regions in sequence databases, *Methods Enzymol*., **266**, 554–571 (1996).
15. P. Romero, Z. Obradovic, X. Li, E.C. Garner, C.J. Brown, and A.K. Dunker, Sequence complexity of disordered protein, *Proteins*, **42**, 38–48 (2001).

16. P. Radivojac, L.M. Iakoucheva, C.J. Oldfield, Z. Obradovic, V.N. Uversky, and A.K. Dunker, Intrinsic disorder and functional proteomics, *Biophys J*, **92**, 1439–1456 (2007).
17. V. Csizmok, Z. Dosztanyi, I. Simon, and P. Tompa, Towards proteomic approaches for the identification of structural disorder, *Curr Protein Pept Sci*, **8**, 173–179 (2007).
18. D.A. Parry, Structural and functional implications of sequence repeats in fibrous proteins, *Adv Protein Chem*, **70**, 11–35 (2005).
19. H. Xie, S. Vucetic, L.M. Iakoucheva, *et al*., Functional anthology of intrinsic disorder. 3. Ligands, post-translational modifications, and diseases associated with intrinsically disordered proteins, *J Proteome Res*, **6**, 1917–1932 (2007).
20. S. Vucetic, H. Xie, L.M. Iakoucheva, *et al*., Functional anthology of intrinsic disorder. 2. Cellular components, domains, technical terms, developmental processes, and coding sequence diversities correlated with long disordered regions, *J Proteome Res*, **6**, 1899–1916 (2007).
21. H. Xie, S. Vucetic, L.M. Iakoucheva, *et al*., Functional anthology of intrinsic disorder. 1. Biological processes and functions of proteins with long disordered regions, *J Proteome Res*, **6**, 1882–1898 (2007).
22. J.E. Walker, M. Saraste, M.J. Runswick, and N.J. Gay, Distantly related sequences in the alpha- and beta-subunits of Atp synthase, myosin, kinases and other Atp-requiring enzymes and a common nucleotide binding fold, *Embo J*, **1**, 945–951 (1982).
23. K. Struhl, Helix-turn-helix, zinc-finger, and leucine-zipper motifs for eukaryotic transcriptional regulatory proteins, *Trends Biochem Sci*, **14**, 137–140 (1989).
24. V. Neduva, and R.B. Russell, Linear motifs: evolutionary interaction switches, *FEBS Lett*, **579**, 3342–3345 (2005).
25. M. Fuxreiter, P. Tompa, and I. Simon, Local structural disorder imparts plasticity on linear motifs, *Bioinformatics*, **23**, 950–956 (2007).
26. T.A. Holland, S. Veretnik, I.N. Shindyalov, and P.E. Bourne, Partitioning protein structures into domains: why is it so difficult?, *J Mol Biol*, **361**, 562–590 (2006).
27. A.G. Murzin, S.E. Brenner, T. Hubbard, and C. Chothia, Scop: A Structural Classification of Proteins Database for the investigation of sequences and structures, *J Mol Biol*, **247**, 536–540 (1995).
28. C.A. Orengo, A.D. Michie, S. Jones, D.T. Jones, M.B. Swindells, and J.M. Thornton, Cath: a hierarchic classification of protein domain structures, *Structure*, **5**, 1093–1108 (1997).
29. R. Apweiler, T.K. Attwood, A. Bairoch, *et al*., The Interpro Database, an integrated documentation resource for protein families, domains and functional sites, *Nucleic Acids Res*, **29**, 37–40 (2001).
30. N.J. Mulder, R. Apweiler, T.K. Attwood, *et al*., New developments in the Interpro Database, *Nucleic Acids Res*, **35**, D224–228 (2007).
31. E.L. Sonnhammer, S.R. Eddy, and R. Durbin, Pfam: A comprehensive database of protein domain families based on seed alignments, *Proteins*, **28**, 405–420 (1997).
32. R.D. Finn, J. Tate, J. Mistry, *et al*., The Pfam Protein Families Database, *Nucleic Acids Res*, **36**, D281–288 (2008).
33. A. Bairoch, Prosite: A dictionary of sites and patterns in proteins, *Nucleic Acids Res*, **19 Suppl**, 2241–2245 (1991).
34. N. Hulo, A. Bairoch, V. Bulliard, *et al*., The 20 years of prosite, *Nucleic Acids Res*, **36**, D245–249 (2008).
35. A. Marchler-Bauer, A.R. Panchenko, B.A. Shoemaker, P.A. Thiessen, L.Y. Geer, and S.H. Bryant, Cdd: A database of conserved domain alignments with links to domain three-dimensional structure, *Nucleic Acids Res*, **30**, 281–283 (2002).
36. A. Marchler-Bauer, J.B. Anderson, M.K. Derbyshire, *et al*., Cdd: A conserved domain database for interactive domain family analysis, *Nucleic Acids Res*, **35**, D237–240 (2007).
37. R.L. Tatusov, M.Y. Galperin, D.A. Natale, and E.V. Koonin, The Cog Database: A tool for genome-scale analysis of protein functions and evolution, *Nucleic Acids Res*, **28**, 33–36 (2000).
38. E.V. Kriventseva, W. Fleischmann, E.M. Zdobnov, and R. Apweiler, Clustr: A database of clusters of Swiss-Prot+Trembl proteins, *Nucleic Acids Res*, **29**, 33–36 (2001).
39. N. Kaplan, O. Sasson, U. Inbar, *et al*., Protonet 4.0: A hierarchical classification of one million protein sequences, *Nucleic Acids Res*, **33**, D216–218 (2005).

40. T. Meinel, A. Krause, H. Luz, M. Vingron, and E. Staub, The Systers Protein Family Database in 2005, *Nucleic Acids Res*, **33**, D226–229 (2005).
41. L.J. Jensen, P. Julien, M. Kuhn, *et al*., Eggnog: Automated construction and annotation of orthologous groups of genes, *Nucleic Acids Res*, **36**, D250–254 (2008).
42. A.C. Berglund, E. Sjolund, G. Ostlund, and E.L. Sonnhammer, Inparanoid 6: Eukaryotic ortholog clusters with inparalogs, *Nucleic Acids Res*, **36**, D263–266 (2008).
43. E.V. Kriventseva, N. Rahman, O. Espinosa, and E.M. Zdobnov, Orthodb: The hierarchical catalog of eukaryotic orthologs, *Nucleic Acids Res*, **36**, D271–275 (2008).
44. T. Rattei, P. Tischler, R. Arnold, *et al*., Simap–Structuring the network of protein similarities, *Nucleic Acids Res*, **36**, D289–292 (2008).
45. P. Puntervoll, R. Linding, C. Gemund, *et al*., Elm server: A new resource for investigating short functional sites in modular eukaryotic proteins, *Nucleic Acids Res*, **31**, 3625–3630 (2003).
46. J.C. Obenauer, L.C. Cantley, and M.B. Yaffe, Scansite 2.0: Proteome-wide prediction of cell signaling interactions using short sequence motifs, *Nucleic Acids Res*, **31**, 3635–3641 (2003).
47. N. Blom, T. Sicheritz-Ponten, R. Gupta, S. Gammeltoft, and S. Brunak, Prediction of post-translational glycosylation and phosphorylation of proteins from the amino acid sequence, *Proteomics*, **4**, 1633–1649 (2004).
48. J. Gough, K. Karplus, R. Hughey, and C. Chothia, Assignment of homology to genome sequences using a library of hidden Markov models that represent all proteins of known structure, *J Mol Biol*, **313**, 903–919 (2001).
49. C. Yeats, M. Maibaum, R. Marsden, *et al*., Gene3d: Modelling protein structure, function and evolution, *Nucleic Acids Res*, **34**, D281–284 (2006).
50. P.D. Thomas, A. Kejariwal, M.J. Campbell, *et al*., Panther: A browsable database of gene products organized by biological function, using curated protein family and subfamily classification, *Nucleic Acids Res*, **31**, 334–341 (2003).
51. D.H. Haft, B.J. Loftus, D.L. Richardson, *et al*., Tigrfams: A protein family resource for the functional identification of proteins, *Nucleic Acids Res*, **29**, 41–43 (2001).
52. C.H. Wu, S. Zhao, and H.L. Chen, A protein class database organized with prosite protein groups and PIR superfamilies, *J Comput Biol*, **3**, 547–561 (1996).
53. E.L. Sonnhammer, and D. Kahn, Modular arrangement of proteins as inferred from analysis of homology, *Protein Sci*, **3**, 482–492 (1994).
54. J. Schultz, F. Milpetz, P. Bork, and C.P. Ponting, Smart, a Simple Modular Architecture Research Tool: Identification of signaling domains, *Proc Natl Acad Sci U S A*, **95**, 5857–5864 (1998).
55. T.K. Attwood, M.E. Beck, A.J. Bleasby, and D.J. Parry-Smith, Prints: a Database of protein motif fingerprints, *Nucleic Acids Res*, **22**, 3590–3596 (1994).
56. S. Balla, V. Thapar, S. Verma, *et al*., Minimotif miner: A tool for investigating protein function, *Nat Methods*, **3**, 175–177 (2006).
57. O. Emanuelsson, S. Brunak, G. von Heijne, and H. Nielsen, Locating proteins in the cell using Targetp, Signalp and related tools, *Nat Protoc*, **2**, 953–971 (2007).
58. T.L. Bailey, N. Williams, C. Misleh, and W.W. Li, Meme: Discovering and analyzing DNA and protein sequence motifs, *Nucleic Acids Res*, **34**, W369–373 (2006).
59. C.E. Lawrence, S.F. Altschul, M.S. Boguski, J.S. Liu, A.F. Neuwald, and J.C. Wootton, Detecting subtle sequence signals: A Gibbs sampling strategy for multiple alignment, *Science*, **262**, 208–214 (1993).
60. T. Huynh, I. Rigoutsos, L. Parida, D. Platt, and T. Shibuya, The web server of IBM's Bioinformatics and Pattern Discovery Group, *Nucleic Acids Res*, **31**, 3645–3650 (2003).
61. V. Neduva, and R.B. Russell, Dilimot: Discovery of linear motifs in proteins, *Nucleic Acids Res*, **34**, W350–355 (2006).
62. N.E. Davey, D.C. Shields, and R.J. Edwards, Slimdisc: Short, linear motif discovery, correcting for common evolutionary descent, *Nucleic Acids Res*, **34**, 3546–3554 (2006).
63. M. Dogruel, T.A. Down, and T.J. Hubbard, Nestedmica as an ab initio protein motif discovery tool, *BMC Bioinformatics*, **9**, 19 (2008).
64. E. Redhead, and T.L. Bailey, Discriminative motif discovery in DNA and protein sequences using the Deme algorithm, *BMC Bioinformatics*, **8**, 385 (2007).

65. A. Apostolico, M. Comin, and L. Parida, Conservative extraction of over-represented extensible motifs, *Bioinformatics*, **21 Suppl 1**, i9–18 (2005).
66. T.D. Schneider, and R.M. Stephens, Sequence logos: A new way to display consensus sequences, *Nucleic Acids Res*, **18**, 6097–6100 (1990).
67. G.E. Crooks, G. Hon, J.M. Chandonia, and S.E. Brenner, Weblogo: A sequence logo generator, *Genome Res*, **14**, 1188–1190 (2004).
68. L.M. Iakoucheva, P. Radivojac, C.J. Brown, T.R. O'Connor, J.G. Sikes, Z. Obradovic, and A.K. Dunker, The importance of intrinsic disorder for protein phosphorylation, *Nucleic Acids Res*, **32**, 1037–1049 (2004).
69. G. Blackshields, I.M. Wallace, M. Larkin, and D.G. Higgins, Analysis and comparison of benchmarks for multiple sequence alignment, *In Silico Biol*, **6**, 321–339 (2006).
70. S.B. Needleman, and C.D. Wunsch, A general method applicable to the search for similarities in the amino acid sequence of two proteins, *J.Mol.Biol.*, **48**, 443–453 (1970).
71. T.F. Smith, and M.S. Waterman, Identification of common molecular subsequences, *J.Mol.Biol.*, **147**, 195–197 (1981).
72. M.O. Dayhoff, R.M. Schwartz, and B.C. Orcutt, A model of evolutionary change in proteins, in *Atlas of Protein Sequence and Structure*, M.O. Dayhoff (ed.), Natl. Biomed. Res. Found., Washington, DC., 1978.
73. S.A. Benner, M.A. Cohen, and G.H. Gonnet, Amino acid substitution during functionally constrained divergent evolution of protein sequences, *Protein Eng*, **7**, 1323–1332 (1994).
74. D.T. Jones, W.R. Taylor, and J.M. Thornton, The rapid generation of mutation data matrices from protein sequences, *Comput Appl Biosci*, **8**, 275–282 (1992).
75. S. Henikoff, and J.G. Henikoff, Amino acid substitution matrices from protein blocks, *Proc Natl Acad Sci U S A*, **89**, 10915–10919 (1992).
76. W.R. Pearson, and D.J. Lipman, Improved tools for biological sequence comparison, *Proc.Natl.Acad.Sci.U.S.A.*, **85**, 2444–2448 (1988).
77. S.F. Altschul, W. Gish, W. Miller, E.W. Myers, and D.J. Lipman, Basic local alignment search tool, *J Mol Biol*, **215**, 403–410 (1990).
78. W.R. Pearson, Empirical statistical estimates for sequence similarity searches, *J Mol Biol*, **276**, 71–84 (1998).
79. M. Pagni, and C.V. Jongeneel, Making sense of score statistics for sequence alignments, *Brief Bioinform*, **2**, 51–67 (2001).
80. S.F. Altschul, T.L. Madden, A.A. Schaffer, *et al.*, Gapped blast and Psi-blast: A new generation of protein database search programs, *Nucleic Acids Res*, **25**, 3389–3402 (1997).
81. S.R. Eddy, G. Mitchison, and R. Durbin, Maximum discrimination hidden Markov models of sequence consensus, *J Comput Biol*, **2**, 9–23 (1995).
82. J. Park, S.A. Teichmann, T. Hubbard, and C. Chothia, Intermediate sequences increase the detection of homology between sequences, *J Mol Biol*, **273**, 349–354 (1997).
83. J. Park, K. Karplus, C. Barrett, *et al.*, Sequence comparisons using multiple sequences detect three times as many remote homologues as pairwise methods, *J Mol Biol*, **284**, 1201–1210 (1998).
84. W. Li, F. Pio, K. Pawlowski, and A. Godzik, Saturated blast: An automated multiple intermediate sequence search used to detect distant homology, *Bioinformatics*, **16**, 1105–1110 (2000).
85. K.K. Koretke, R.B. Russell, and A.N. Lupas, Fold recognition without folds, *Protein Sci*, **11**, 1575–1579 (2002).
86. S.F. Altschul, and E.V. Koonin, Iterated profile searches with Psi-blast–a tool for discovery in protein databases, *Trends Biochem Sci*, **23**, 444–447 (1998).
87. L. Aravind, and E.V. Koonin, Gleaning non-trivial structural, functional and evolutionary information about proteins by iterative database searches, *J Mol Biol*, **287**, 1023–1040 (1999).
88. A.A. Schaffer, Y.I. Wolf, C.P. Ponting, E.V. Koonin, L. Aravind, and S.F. Altschul, Impala: Matching a protein sequence against a collection of Psi-blast-constructed position-specific score matrices, *Bioinformatics*, **15**, 1000–1011 (1999).
89. G. Yona, and M. Levitt, Within the twilight zone: A sensitive profile-profile comparison tool based on information theory, *J Mol Biol*, **315**, 1257–1275 (2002).

90. R. Sadreyev, and N. Grishin, Compass: A tool for comparison of multiple protein alignments with assessment of statistical significance, *J Mol Biol*, **326**, 317–336 (2003).
91. J. Soding, Protein homology detection by Hmm-Hmm comparison, *Bioinformatics*, **21**, 951–960 (2005).
92. J. Soding, M. Remmert, A. Biegert, and A.N. Lupas, Hhsenser: Exhaustive transitive profile search using Hmm-Hmm comparison, *Nucleic Acids Res*, **34**, W374–378 (2006).
93. T. Frickey, and A. Lupas, Clans: A Java application for visualizing protein families based on pairwise similarity, *Bioinformatics*, **20**, 3702–3704 (2004).
94. A.T. Adai, S.V. Date, S. Wieland, and E.M. Marcotte, Lgl: Creating a map of protein function with an algorithm for visualizing very large biological networks, *J Mol Biol*, **340**, 179–190 (2004).
95. P. Pipenbacher, A. Schliep, S. Schneckener, A. Schonhuth, D. Schomburg, and R. Schrader, Proclust: Improved clustering of protein sequences with an extended graph-based approach, *Bioinformatics*, **18 Suppl 2**, S182–191 (2002).
96. A.J. Enright, S. Van Dongen, and C.A. Ouzounis, An efficient algorithm for large-scale detection of protein families, *Nucleic Acids Res*, **30**, 1575–1584 (2002).
97. I.V. Tetko, A. Facius, A. Ruepp, and H.W. Mewes, Super paramagnetic clustering of protein sequences, *BMC Bioinformatics*, **6**, 82 (2005).
98. N. Krishnamurthy, D. Brown, and K. Sjolander, Flowerpower: Clustering proteins into domain architecture classes for phylogenomic inference of protein function, *BMC Evol Biol*, **7 Suppl 1**, S12 (2007).
99. D.P. Brown, N. Krishnamurthy, and K. Sjolander, Automated protein subfamily identification and classification, *PLoS Comput Biol*, **3**, e160 (2007).
100. A. Kelil, S. Wang, R. Brzezinski, and A. Fleury, Cluss: Clustering of protein sequences based on a new similarity measure, *BMC Bioinformatics*, **8**, 286 (2007).
101. L. Wang, and T. Jiang, On the complexity of multiple sequence alignment, *J Comput Biol*, **1**, 337–348 (1994).
102. P. Hogeweg, and B. Hesper, The alignment of sets of sequences and the construction of phyletic trees: an integrated method, *J Mol Evol*, **20**, 175–186 (1984).
103. O. Gotoh, Significant improvement in accuracy of multiple protein sequence alignments by iterative refinement as assessed by reference to structural alignments, *J Mol Biol*, **264**, 823–838 (1996).
104. C. Notredame, D.G. Higgins, and J. Heringa, T-Coffee: A novel method for fast and accurate multiple sequence alignment, *J Mol Biol*, **302**, 205–217 (2000).
105. I.M. Wallace, O. O'Sullivan, and D.G. Higgins, Evaluation of iterative alignment algorithms for multiple alignment, *Bioinformatics*, **21**, 1408–1414 (2005).
106. B. Morgenstern, K. Frech, A. Dress, and T. Werner, Dialign: Finding local similarities by multiple sequence alignment, *Bioinformatics*, **14**, 290–294 (1998).
107. B. Raphael, D. Zhi, H. Tang, and P. Pevzner, A novel method for multiple alignment of sequences with repeated and shuffled elements, *Genome Res*, **14**, 2336–2346 (2004).
108. H. Zhou, and Y. Zhou, Spem: Improving multiple sequence alignment with sequence profiles and predicted secondary structures, *Bioinformatics*, **21**, 3615–3621 (2005).
109. J. Pei, B.H. Kim, M. Tang, and N.V. Grishin, Promals web server for accurate multiple protein sequence alignments, *Nucleic Acids Res.* (2007).
110. J.D. Thompson, T.J. Gibson, F. Plewniak, F. Jeanmougin, and D.G. Higgins, The Clustal_X Windows Interface: Flexible strategies for multiple sequence alignment aided by quality analysis tools, *Nucleic.Acids Res.*, **25**, 4876–4882 (1997).
111. J.D. Thompson, F. Plewniak, J. Thierry, and O. Poch, Dbclustal: Rapid and reliable global multiple alignments of protein sequences detected by database searches, *Nucleic Acids Res*, **28**, 2919–2926 (2000).
112. R. Hughey, and A. Krogh, Hidden Markov models for sequence analysis: Extension and analysis of the basic method, *Comput Appl Biosci*, **12**, 95–107 (1996).
113. S.R. Eddy, Multiple alignment using hidden Markov models, *Proc Int Conf Intell Syst Mol Biol*, **3**, 114–120 (1995).

114. R.C. Edgar, Muscle: Multiple sequence alignment with high accuracy and high throughput, *Nucleic Acids Res*, **32**, 1792–1797 (2004).
115. K. Katoh, K. Misawa, K. Kuma, and T. Miyata, Mafft: A novel method for rapid multiple sequence alignment based on fast Fourier transform, *Nucleic Acids Res*, **30**, 3059–3066 (2002).
116. O. Gotoh, A weighting system and algorithm for aligning many phylogenetically related sequences, *Comput Appl Biosci*, **11**, 543–551 (1995).
117. V.A. Simossis, J. Kleinjung, and J. Heringa, Homology-extended sequence alignment, *Nucleic Acids Res*, **33**, 816–824 (2005).
118. C.B. Do, M.S. Mahabhashyam, M. Brudno, and S. Batzoglou, Probcons: Probabilistic consistency-based multiple sequence alignment, *Genome Res*, **15**, 330–340 (2005).
119. J. Pei, and N.V. Grishin, Mummals: Multiple sequence alignment improved by using hidden Markov models with local structural information, *Nucleic Acids Res*, **34**, 4364–4374 (2006).
120. J. Pei, and N.V. Grishin, Promals: Towards accurate multiple sequence alignments of distantly related proteins, *Bioinformatics*, **23**, 802–808 (2007).
121. R.C. Edgar, and K. Sjolander, Satchmo: Sequence alignment and tree construction using hidden Markov models, *Bioinformatics*, **19**, 1404–1411 (2003).
122. S. Yamada, O. Gotoh, and H. Yamana, Improvement in accuracy of multiple sequence alignment using novel group-to-group sequence alignment algorithm with piecewise linear gap cost, *BMC Bioinformatics*, **7**, 524 (2006).
123. T. Lassmann, and E.L. Sonnhammer, Kalign: An accurate and fast multiple sequence alignment algorithm, *BMC Bioinformatics*, **6**, 298 (2005).
124. A.R. Subramanian, J. Weyer-Menkhoff, M. Kaufmann, and B. Morgenstern, Dialign-T: An improved algorithm for segment-based multiple sequence alignment, *BMC Bioinformatics*, **6**, 66 (2005).
125. Z. Zhang, H. Lin, and M. Li, Mango: A new approach to multiple sequence alignment, *Comput Syst Bioinformatics Conf*, **6**, 237–247 (2007).
126. I. Van Walle, I. Lasters, and L. Wyns, Align-M: A new algorithm for multiple alignment of highly divergent sequences, *Bioinformatics*, **20**, 1428–1435 (2004).
127. C. Lee, C. Grasso, and M.F. Sharlow, Multiple sequence alignment using partial order graphs, *Bioinformatics*, **18**, 452–464 (2002).
128. N.C. Jones, D. Zhi, and B.J. Raphael, Aliwaba: Alignment on the Web through an a-Bruijn approach, *Nucleic Acids Res*, **34**, W613–616 (2006).
129. T.M. Phuong, C.B. Do, R.C. Edgar, and S. Batzoglou, Multiple alignment of protein sequences with repeats and rearrangements, *Nucleic Acids Res*, **34**, 5932-5942 (2006).
130. K. Bucka-Lassen, O. Caprani, and J. Hein, Combining many multiple alignments in one improved alignment, *Bioinformatics*, **15**, 122–130 (1999).
131. I.M. Wallace, O. O'Sullivan, D.G. Higgins, and C. Notredame, M-Coffee: Combining multiple sequence alignment methods with T-Coffee, *Nucleic Acids Res*, **34**, 1692–1699 (2006).
132. M. Clamp, J. Cuff, S.M. Searle, and G.J. Barton, The Jalview Java Alignment Editor, *Bioinformatics*, **20**, 426–427 (2004).
133. M. Sammeth, T. Griebel, F. Tille, and J. Stoye, Panta Rhei (Qalign2): An open graphical environment for sequence analysis, *Bioinformatics*, **22**, 889–890 (2006).
134. C. Gille, and C. Frommel, Strap: Editor for structural alignments of proteins, *Bioinformatics*, **17**, 377–378 (2001).
135. T.J. Carver, and L.J. Mullan, Jae: Jemboss alignment editor, *Appl Bioinformatics*, **4**, 151–154 (2005).
136. O. Pible, G. Imbert, and J.L. Pellequer, Interalign: Interactive alignment editor for distantly related protein sequences, *Bioinformatics*, **21**, 3166–3167 (2005).
137. D.J. Parry-Smith, A.W. Payne, A.D. Michie, and T.K. Attwood, Cinema: A novel colour interactive editor for multiple alignments, *Gene*, **221**, GC57–63 (1998).
138. R. Brodie, A.J. Smith, R.L. Roper, V. Tcherepanov, and C. Upton, Base-by-Base: Single nucleotide-level analysis of whole viral genome alignments, *BMC Bioinformatics*, **5**, 96 (2004).
139. M.G. Goode, and A.G. Rodrigo, Squint: A multiple alignment program and editor, *Bioinformatics*, **23**, 1553–1555 (2007).

140. G.D. Schuler, S.F. Altschul, and D.J. Lipman, A workbench for multiple alignment construction and analysis, *Proteins*, **9**, 180–190 (1991).
141. N. Galtier, M. Gouy, and C. Gautier, Seaview and Phylo_Win: Two graphic tools for sequence alignment and molecular phylogeny, *Comput Appl Biosci*, **12**, 543–548 (1996).
142. V. Catherinot, and G. Labesse, Vito: Tool for refinement of protein sequence-structure alignments, *Bioinformatics*, **20**, 3694–3696 (2004).
143. C. Grasso, M. Quist, K. Ke, and C. Lee, Poaviz: A partial order multiple sequence alignment visualizer, *Bioinformatics*, **19**, 1446–1448 (2003).
144. B. Morgenstern, S. Goel, A. Sczyrba, and A. Dress, Altavist: Comparing alternative multiple sequence alignments, *Bioinformatics*, **19**, 425–426 (2003).
145. P. Gouet, X. Robert, and E. Courcelle, Espript/Endscript: Extracting and rendering sequence and 3D information from atomic structures of proteins, *Nucleic Acids Res*, **31**, 3320–3323 (2003).
146. J. Felsenstein, Phylip – Phylogeny Inference Package (Version 3.2), *Cladistics*, **5**, 164–166 (1989).
147. K. Tamura, J. Dudley, M. Nei, and S. Kumar, Mega4: Molecular Evolutionary Genetics Analysis (Mega) Software Version 4.0, *Mol Biol Evol*, **24**, 1596–1599 (2007).
148. S. Guindon, and O. Gascuel, A simple, fast, and accurate algorithm to estimate large phylogenies by maximum likelihood, *Syst Biol*, **52**, 696–704 (2003).
149. Z. Yang, Paml 4: Phylogenetic analysis by maximum likelihood, *Mol Biol Evol*, **24**, 1586–1591 (2007).
150. H.A. Schmidt, K. Strimmer, M. Vingron, and A. von Haeseler, Tree-Puzzle: Maximum likelihood phylogenetic analysis using quartets and parallel computing, *Bioinformatics*, **18**, 502–504 (2002).
151. F. Ronquist, and J.P. Huelsenbeck, Mrbayes 3: Bayesian phylogenetic inference under mixed models, *Bioinformatics*, **19**, 1572–1574 (2003).
152. T.M. Keane, T.J. Naughton, and J.O. McInerney, Multiphyl: A high-throughput phylogenomics webserver using distributed computing, *Nucleic Acids Res*, **35**, W33–37 (2007).
153. S. Whelan, P. Lio, and N. Goldman, Molecular phylogenetics: State-of-the-art methods for looking into the past, *Trends Genet*, **17**, 262–272 (2001).
154. J.P. Huelsenbeck, F. Ronquist, R. Nielsen, and J.P. Bollback, Bayesian inference of phylogeny and its impact on evolutionary biology, *Science*, **294**, 2310–2314 (2001).
155. C. Kosiol, L. Bofkin, and S. Whelan, Phylogenetics by likelihood: Evolutionary modeling as a tool for understanding the genome, *J Biomed Inform*, **39**, 51–61 (2006).
156. K. Sjolander, Phylogenomic inference of protein molecular function: Advances and challenges, *Bioinformatics*, **20**, 170–179 (2004).
157. W. Cai, J. Pei, and N.V. Grishin, Reconstruction of ancestral protein sequences and its applications, *BMC Evol Biol*, **4**, 33 (2004).
158. P.D. Williams, D.D. Pollock, B.P. Blackburne, and R.A. Goldstein, Assessing the accuracy of ancestral protein reconstruction methods, *PLoS Comput Biol*, **2**, e69 (2006).
159. S.K. Kummerfeld, and S.A. Teichmann, Relative rates of gene fusion and fission in multidomain proteins, *Trends Genet*, **21**, 25–30 (2005).
160. C.J. Oldfield, Y. Cheng, M.S. Cortese, C.J. Brown, V.N. Uversky, and A.K. Dunker, Comparing and combining predictors of mostly disordered proteins, *Biochemistry*, **44**, 1989–2000 (2005).
161. K. Bryson, D. Cozzetto, and D.T. Jones, Computer-assisted protein domain boundary prediction using the Dompred server, *Curr Protein Pept Sci*, **8**, 181–188 (2007).
162. H.K. Saini, and D. Fischer, Meta-Dp: Domain prediction meta server, *Bioinformatics* (2005).
163. J. Cheng, Domac: An accurate, hybrid protein domain prediction server, *Nucleic Acids Res*, **35**, W354–356 (2007).
164. D.E. Kim, D. Chivian, L. Malmstrom, and D. Baker, Automated prediction of domain boundaries in Casp6 targets using Ginzu and Rosettadom, *Proteins* (2005).
165. J.E. Gewehr, and R. Zimmer, Ssep-Domain: Protein domain prediction by alignment of secondary structure elements and profiles, *Bioinformatics*, **22**, 181–187 (2006).

166. K. Vlahovicek, L. Kajan, V. Agoston, and S. Pongor, The Sbase domain sequence resource, release 12: Prediction of protein domain-architecture using support vector machines, *Nucleic Acids Res*, **33 Database Issue**, D223–225 (2005).
167. R.A. George, and J. Heringa, Protein domain identification and improved sequence similarity searching using Psi-blast, *Proteins*, **48**, 672–681 (2002).
168. N. Nagarajan, and G. Yona, Automatic prediction of protein domains from sequence information using a hybrid learning system, *Bioinformatics*, **20**, 1335–1360 (2004).
169. J. Sim, S.Y. Kim, and J. Lee, Pprodo: Prediction of protein domain boundaries using neural networks, *Proteins*, (2005).
170. J. Cheng, M.J. Sweredoski, and P. Baldi, Dompro: Protein domain prediction using profiles, secondary structure, relative solvent accessibility, and recursive neural networks, *Data Mining and Knowledge Discovery*, **13**, 1–10 (2006).
171. R. Linding, R.B. Russell, V. Neduva, and T.J. Gibson, Globplot: Exploring protein sequences for globularity and disorder, *Nucleic Acids Res*, **31**, 3701–3708 (2003).
172. M. Suyama, and O. Ohara, Domcut: Prediction of inter-domain linker regions in amino acid sequences, *Bioinformatics*, **19**, 673–674 (2003).
173. R.A. George, K. Lin, and J. Heringa, Scooby-Domain: Prediction of globular domains in protein sequence, *Nucleic Acids Res*, **33**, W160–163 (2005).
174. J. Liu, and B. Rost, Sequence-based prediction of protein domains, *Nucleic Acids Res*, **32**, 3522–3530 (2004).
175. N. Alexandrov, and I. Shindyalov, Pdp: Protein domain parser, *Bioinformatics*, **19**, 429–430 (2003).