Comparative Protein Structure Modeling Using MODELLER

Benjamin Webb[1] and Andrej Sali[1]

[1]University of California at San Francisco, San Francisco, California

ABSTRACT

Functional characterization of a protein sequence is one of the most frequent problems in biology. This task is usually facilitated by accurate three-dimensional (3-D) structure of the studied protein. In the absence of an experimentally determined structure, comparative or homology modeling can sometimes provide a useful 3-D model for a protein that is related to at least one known protein structure. Comparative modeling predicts the 3-D structure of a given protein sequence (target) based primarily on its alignment to one or more proteins of known structure (templates). The prediction process consists of fold assignment, target-template alignment, model building, and model evaluation. This unit describes how to calculate comparative models using the program MODELLER and discusses all four steps of comparative modeling, frequently observed errors, and some applications. Modeling lactate dehydrogenase from *Trichomonas vaginalis* (TvLDH) is described as an example. The download and installation of the MODELLER software is also described. *Curr. Protoc. Bioinform.* 47:5.6.1-5.6.32. © 2014 by John Wiley & Sons, Inc.

Keywords: Modeller • protein structure • comparative modeling • structure prediction • protein fold

INTRODUCTION

Functional characterization of a protein sequence is one of the most frequent problems in biology. This task is usually facilitated by an accurate three-dimensional (3-D) structure of the studied protein. In the absence of an experimentally determined structure, comparative or homology modeling often provides a useful 3-D model for a protein that is related to at least one known protein structure (Marti-Renom et al., 2000; Fiser, 2004; Misura and Baker, 2005; Petrey and Honig, 2005; Misura et al., 2006). Comparative modeling predicts the 3-D structure of a given protein sequence (target) based primarily on its alignment to one or more proteins of known structure (templates).

Comparative modeling consists of four main steps (Marti-Renom et al., 2000; Figure 5.6.1): (i) fold assignment, which identifies similarity between the target and at least one known template structure; (ii) alignment of the target sequence and the template(s); (iii) building a model based on the alignment with the chosen template(s); and (iv) predicting model errors.

There are several computer programs and Web servers that automate the comparative modeling process (Table 5.6.1). The accuracy of the models calculated by many of these servers is evaluated by CAMEO (Haas et al., 2013) and the biannual CASP (Critical Assessment of Techniques for Protein Structure Prediction; Moult, 2005; Moult et al., 2009) experiment.

While automation makes comparative modeling accessible to both experts and nonspecialists, manual intervention is generally still needed to maximize the accuracy of the models in the difficult cases. A number of resources useful in comparative modeling are listed in Table 5.6.1.

Current Protocols in Bioinformatics 5.6.1-5.6.32, September 2014
Published online September 2014 in Wiley Online Library (wileyonlinelibrary.com).
DOI: 10.1002/0471250953.bi0506s47

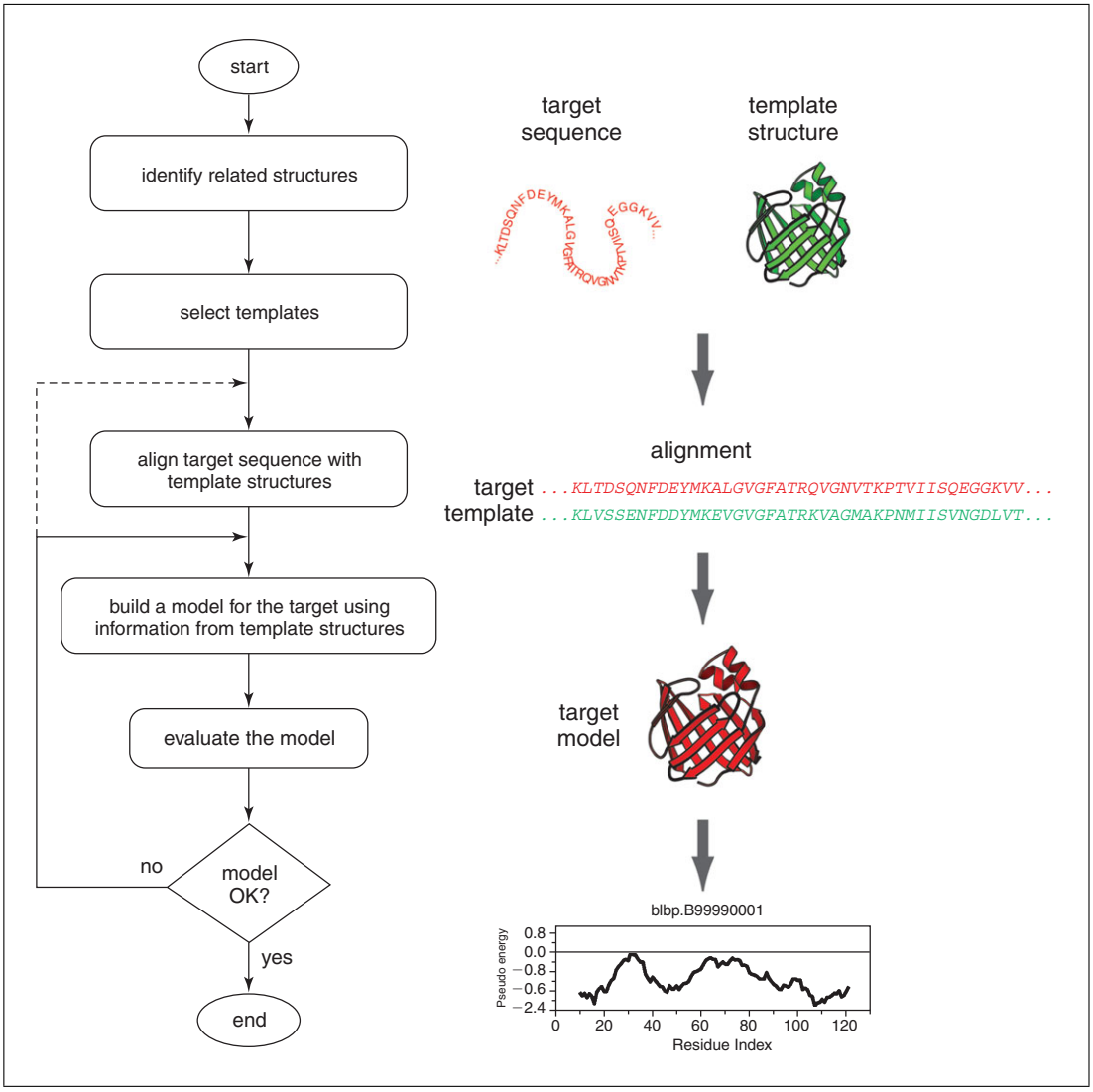

Figure 5.6.1 Steps in comparative protein structure modeling. See text for details.

This unit describes how to calculate comparative models using the program MODELLER (Basic Protocol). The Basic Protocol goes on to discuss all four steps of comparative modeling (Figure 5.6.1), frequently observed errors, and some applications. The Support Protocol describes how to download and install MODELLER.

BASIC PROTOCOL

MODELING LACTATE DEHYDROGENASE FROM *TRICHOMONAS VAGINALIS* (TvLDH) BASED ON A SINGLE TEMPLATE USING MODELLER

MODELLER is a computer program for comparative protein structure modeling (Sali and Blundell, 1993; Fiser et al., 2000). In the simplest case, the input is an alignment of a sequence to be modeled with the template structures, the atomic coordinates of the templates, and a simple script file. MODELLER then automatically calculates a model containing all non-hydrogen atoms, within minutes on a modern PC and with no user intervention. Apart from model building, MODELLER can perform additional auxiliary tasks, including fold assignment, alignment of two protein sequences or their profiles (Marti-Renom et al., 2004), multiple alignment of protein sequences and/or structures (Madhusudhan et al., 2006; Madhusudhan et al., 2009), calculation of phylogenetic trees, and de novo modeling of loops in protein structures (Fiser et al., 2000).

NOTE: Further help for all the described commands and parameters may be obtained from the MODELLER Web site (see Internet Resources).

Table 5.6.1 Programs and Web Servers Useful in Comparative Protein Structure Modeling

Name	URL
Databases	
Protein sequence databases	
Ensembl (Flicek et al., 2013)	*http://www.ensembl.org*
GENBANK (Benson et al., 2013)	*http://www.ncbi.nlm.nih.gov/Genbank/*
Protein Information Resource (Huang et al., 2007)	*http://pir.georgetown.edu/*
UniProtKB (Bairoch et al., 2005)	*http://www.uniprot.org*
Domains and superfamilies	
CATH/Gene3D (Pearl et al., 2005)	*http://www.cathdb.info*
InterPro (Hunter et al., 2012)	*http://www.ebi.ac.uk/interpro/*
MEME (Bailey and Elkan, 1994)	*http://meme.nbcr.net/meme/*
Pfam (Bateman et al., 2004)	*http://pfam.sanger.ac.uk/*
PRINTS (Attwood et al., 2012)	*http://www.bioinf.manchester.ac.uk/dbbrowser/PRINTS/index.php*
ProDom (Bru et al., 2005)	*http://prodom.prabi.fr*
ProSite (Hulo et al., 2006)	*http://prosite.expasy.org/*
SCOP (Andreeva et al., 2004)	*http://scop.mrc-lmb.cam.ac.uk/scop/*
SFLD (Brown and Babbitt, 2012)	*http://sfld.rbvi.ucsf.edu/*
SMART (Letunic et al., 2012)	*http://smart.embl-heidelberg.de/*
SUPERFAMILY (Gough et al., 2001)	*http://supfam.cs.bris.ac.uk/SUPERFAMILY/*
Protein structures and models	
ModBase (Pieper et al., 2011)	*http://www.salilab.org/modbase/*
PDB (Berman et al., 2000)	*http://www.pdb.org/*
Protein Model Portal (Arnold et al., 2009; Haas et al., 2013)	*http://www.proteinmodelportal.org/*
SwissModel Repository (Kiefer et al., 2009)	*http://swissmodel.expasy.org/repository/*
Miscellaneous	
DBALI (Marti-Renom et al., 2001)	*http://www.salilab.org/dbali*
GENECENSUS (Lin et al., 2002)	*http://bioinfo.mbb.yale.edu/genome/*
Alignment	
Sequence- and structure-based sequence alignment	
AlignMe (Khafizov et al., 2010)	*http://www.bioinfo.mpg.de/AlignMe/*
CLUSTALW (Thompson et al., 1994)	*http://www2.ebi.ac.uk/clustalw/*
COMPASS (Sadreyev and Grishin, 2003)	*ftp://iole.swmed.edu/pub/compass/*
EXPRESSO (Armougom et al., 2006)	*http://igs-server.cnrs-mrs.fr/Tcoffee/tcoffee_cgi/index.cgi*
FastA (Pearson, 2000)	*http://www.ebi.ac.uk/Tools/sss/fasta/*
FFAS03 (Jaroszewski et al., 2005)	*http://ffas.burnham.org/*
FUGUE (Shi et al., 2001)	*http://www-cryst.bioc.cam.ac.uk/fugue*
GENTHREADER (Jones, 1999; McGuffin and Jones, 2003)	*http://bioinf.cs.ucl.ac.uk/psipred/*

continued

Table 5.6.1 Programs and Web Servers Useful in Comparative Protein Structure Modeling, *continued*

Name	URL
HHBlits/HHsearch (Remmert et al., 2012)	*http://toolkit.lmb.uni-muenchen.de/hhsuite*
MAFFT (Katoh and Standley, 2013)	*http://mafft.cbrc.jp/alignment/software/*
MUSCLE (Edgar, 2004)	*http://www.drive5.com/muscle*
MUSTER (Wu and Zhang, 2008)	*http://zhanglab.ccmb.med.umich.edu/MUSTER*
PROMALS3D (Pei et al., 2008)	*http://prodata.swmed.edu/promals3d/promals3d.php*
PSI-BLAST (Altschul et al., 1997)	*http://blast.ncbi.nlm.nih.gov/Blast.cgi*
PSIPRED (McGuffin et al., 2000)	*http://bioinf.cs.ucl.ac.uk/psipred/*
SALIGN (Eswar et al., 2003)	*http://www.salilab.org/salign/*
SAM-T08 (Karplus et al., 2003; Karplus, 2009)	*http://compbio.soe.ucsc.edu/HMM-apps/*
Staccato (Shatsky et al., 2006)	*http://bioinfo3d.cs.tau.ac.il/staccato/*
T-Coffee (Notredame, 2010; also see *UNIT 3.8*)	*http://www.tcoffee.org/*
Structure	
CE (Prlic et al., 2010)	*http://source.rcsb.org/jfatcatserver/ceHome.jsp*
GANGSTA+ (Guerler and Knapp, 2008)	*http://agknapp.chemie.fu-berlin.de/gplus/index.php*
HHsearch (Soding, 2005)	*ftp://toolkit.lmb.uni-muenchen.de/hhsearch/*
Mammoth (Ortiz et al., 2002)	*http://ub.cbm.uam.es/software/mammoth.php*
Mammoth-mult (Lupyan et al., 2005)	*http://ub.cbm.uam.es/software/mammothm.php*
MASS (Dror et al., 2003)	*http://bioinfo3d.cs.tau.ac.il/MASS/*
MultiProt (Shatsky et al., 2004)	*http://bioinfo3d.cs.tau.ac.il/MultiProt*
MUSTANG (Konagurthu et al., 2006)	*http://www.csse.monash.edu.au/~karun/Site/mustang.html*
PDBeFold (Dietmann et al., 2001)	*http://www.ebi.ac.uk/msd-srv/ssm/*
SALIGN (Eswar et al., 2003)	*http://www.salilab.org/salign/*
TM-align (Zhang and Skolnick, 2005)	*http://zhanglab.ccmb.med.umich.edu/TM-align/*
Alignment modules in molecular graphics programs	
Discovery Studio	*http://www.accelrys.com*
PyMol	*http://www.pymol.org/*
Swiss-PDB Viewer (Kaplan and Littlejohn, 2001)	*http://spdbv.vital-it.ch/*
UCSF Chimera (Huang et al., 2000)	*http://www.cgl.ucsf.edu/chimera*
Comparative modeling, threading, and refinement	
Web servers	
3d-jigsaw (Bates et al., 2001)	*http://www.bmm.icnet.uk/servers/3djigsaw/*
HHPred (Soding et al., 2005)	*http://toolkit.genzentrum.lmu.de/hhpred*

continued

Table 5.6.1 Programs and Web Servers Useful in Comparative Protein Structure Modeling, *continued*

Name	URL
IntFold (Roche et al., 2011)	*http://www.reading.ac.uk/bioinf/IntFOLD/*
i-TASSER (Roy et al., 2010)	*http://zhanglab.ccmb.med.umich.edu/I-TASSER/*
M4T (Fernandez-Fuentes et al., 2007)	*http://manaslu.aecom.yu.edu/M4T/*
ModWeb (Eswar et al., 2003)	*http://salilab.org/modweb/*
Phyre2 (Kelley and Sternberg, 2009)	*http://www.sbg.bio.ic.ac.uk/phyre2*
RaptorX (Kallberg et al., 2012)	*http://raptorx.uchicago.edu/*
Robetta (Song et al., 2013)	*http://robetta.bakerlab.org/*
SWISS-MODEL (Schwede et al., 2003)	*http://www.expasy.org/swissmod*
Programs	
HHsuite (Soding, 2005)	*ftp://toolkit.genzentrum.lmu.de/pub/HH-suite/*
Modeller (Sali and Blundell, 1993)	*http://www.salilab.org/modeller/*
MolIDE (Wang et al., 2008)	*http://dunbrack.fccc.edu/molide/*
Rosetta@home	*http://boinc.bakerlab.org/rosetta/*
RosettaCM (Song et al., 2013)	*https://www.rosettacommons.org/home*
SCWRL (Krivov et al., 2009)	*http://dunbrack.fccc.edu/scwrl4/SCWRL4.php*
Quality estimation	
ANOLEA (Melo and Feytmans, 1998)	*http://melolab.org/anolea/index.html*
ERRAT (Colovos and Yeates, 1993)	*http://nihserver.mbi.ucla.edu/ERRAT/*
ModEval	*http://salilab.org/modeval/*
ProQ2 (Ray et al., 2012)	*http://proq2.theophys.kth.se/*
PROCHECK (Laskowski et al., 1993)	*http://www.ebi.ac.uk/thornton-srv/software/PROCHECK/*
Prosa2003 (Sippl, 1993; Wiederstein and Sippl, 2007)	*http://www.came.sbg.ac.at*
QMEAN local (Benkert et al., 2011)	*http://www.openstructure.org/download/*
SwissModel Workspace (Arnold et al., 2006)	*http://swissmodel.expasy.org/workspace/index.php?func=tools_structureassessment1*
VERIFY3D (Luthy et al., 1992)	*http://www.doe-mbi.ucla.edu/Services/Verify_3D/*
WHATCHECK (Hooft et al., 1996)	*http://www.cmbi.kun.nl/gv/whatcheck/*
Methods evaluation	
CAMEO (Haas et al., 2013)	*http://cameo3d.org/*
CASP (Moult et al., 2003)	*http://predictioncenter.llnl.gov*

Necessary Resources

Hardware

A computer running RedHat Linux (PC, Opteron or EM64T/Xeon64 systems) or other version of Linux/Unix (x86/x86_64 Linux, AIX), Apple Mac OSX (10.6 or later), or Microsoft Windows (XP or later)

Software

The MODELLER 9.13 program, downloaded and installed from *http://salilab.org/modeller/download_installation.html* (see Support Protocol)

```
>P1;TvLDH
sequence:TvLDH:::::::0.00: 0.00
MSEAAHVLITGAAGQIGYILSHWIASGELYGDRQVYLHLLDIPPAMNRLTALTMELEDCAFPHLAGFVATTDPKA
AFKDIDCAFLVASMPLKPGQVRADLISSNSVIFKNTGEYLSKWAKPSVKVLVIGNPDNTNCEIAMLHAKNLKPEN
FSSLSMLDQNRAYYEVASKLGVDVKDVHDIIVWGNHGESMVADLTQATFTKEGKTQKVVDVLDHDYVFDTFFKKI
GHRAWDILEHRGFTSAASPTKAAIQHMKAWLFGTAPGEVLSMGIPVPEGNPYGIKPGVVFSFPCNVDKEGKIHVV
EGFKVNDWLREKLDFTEKDLFHEKEIALNHLAQGG*
```

Figure 5.6.2 File `TvLDH.ali`. Sequence file in PIR format.

Files

All files required to complete this protocol can be downloaded from *http://salilab.org/modeller/tutorial/basic-example.tar.gz* (Unix/Linux) or *http://salilab.org/modeller/tutorial/basic-example.zip* (Windows)

Background to TvLDH

A novel gene for lactate dehydrogenase (LDH) was identified from the genomic sequence of *Trichomonas vaginalis* (TvLDH). The corresponding protein had higher sequence similarity to the malate dehydrogenase of the same species (TvMDH) than to any other LDH. The authors hypothesized that TvLDH arose from TvMDH by convergent evolution relatively recently (Wu et al., 1999). Comparative models were constructed for TvLDH and TvMDH to study the sequences in a structural context and to suggest site-directed mutagenesis experiments to elucidate changes in enzymatic specificity in this apparent case of convergent evolution. The native and mutated enzymes were subsequently expressed and their activities compared (Wu et al., 1999).

Searching structures related to TvLDH

Conversion of sequence to PIR file format

It is first necessary to convert the target TvLDH sequence into a format that is readable by MODELLER (file `TvLDH.ali`; Fig. 5.6.2). MODELLER uses the PIR format to read and write sequences and alignments. The first line of the PIR-formatted sequence consists of `P1;` followed by the identifier of the sequence. In this example, the sequence is identified by the code `TvLDH`. The second line, consisting of ten fields separated by colons, usually contains details about the structure, if any. In the case of sequences with no structural information, only two of these fields are used: the first field should be `sequence` (indicating that the file contains a sequence without a known structure) and the second should contain the model file name (`TvLDH` in this case). The rest of the file contains the sequence of TvLDH, with an asterisk (`*`) marking its end. The standard uppercase single-letter amino acid codes are used to represent the sequence.

Searching for suitable template structures

A search for potentially related sequences of known structure can be performed using the `profile.build()` command of MODELLER (file `build_profile.py`). The command uses the local dynamic programming algorithm to identify related sequences (Smith and Waterman, 1981). In the simplest case, the command takes as input the target sequence and a database of sequences of known structure (file `pdb_95.pir`) and returns a set of statistically significant alignments. The input script file for the command is shown in Figure 5.6.3.

The script, `build_profile.py`, does the following:

1. Initializes the "environment" for this modeling run by creating a new `environ` object (called `env` here). Almost all MODELLER scripts require this step, as the new object is needed to build most other useful objects.

```
from modeller import *

log.verbose()
env = environ()

#-- Prepare the input files

#-- Read in the sequence database
sdb = sequence_db(env)
sdb.read(seq_database_file='pdb_95.pir', seq_database_format='PIR',
         chains_list='ALL', minmax_db_seq_len=(30, 4000), clean_sequences=True)

#-- Write the sequence database in binary form
sdb.write(seq_database_file='pdb_95.bin', seq_database_format='BINARY',
          chains_list='ALL')

#-- Now, read in the binary database
sdb.read(seq_database_file='pdb_95.bin', seq_database_format='BINARY',
         chains_list='ALL')

#-- Read in the target sequence/alignment
aln = alignment(env)
aln.append(file='TvLDH.ali', alignment_format='PIR', align_codes='ALL')

#-- Convert the input sequence/alignment into
#   profile format
prf = aln.to_profile()

#-- Scan sequence database to pick up homologous sequences
prf.build(sdb, matrix_offset=-450, rr_file='${LIB}/blosum62.sim.mat',
          gap_penalties_1d=(-500, -50), n_prof_iterations=1,
          check_profile=False, max_aln_evalue=0.01)

#-- Write out the profile in text format
prf.write(file='build_profile.prf', profile_format='TEXT')

#-- Convert the profile back to alignment format
aln = prf.to_alignment()

#-- Write out the alignment file
aln.write(file='build_profile.ali', alignment_format='PIR')
```

Figure 5.6.3 File `build_profile.py`. Input script file that searches for templates against a database of nonredundant PDB sequences.

2. Creates a new `sequence_db` object, calling it `sdb`, which is used to contain large databases of protein sequences.
3. Reads a file, in text format, containing nonredundant PDB sequences, into the `sdb` database. The sequences can be found in the file `pdb_95.pir`. This file is also in the PIR format. Each sequence in this file is representative of a group of PDB sequences that share 95% or more sequence identity to each other and have less than 30 residues or 30% sequence length difference.
4. Writes a binary machine-independent file containing all sequences read in the previous step.
5. Reads the binary format file back in for faster execution.
6. Creates a new "alignment" object (`aln`), reads the target sequence `TvLDH` from the file `TvLDH.ali`, and converts it to a profile object (`prf`). Profiles contain similar information to alignments, but are more compact and better for sequence database searching.
7. `prf.build()` searches the sequence database (`sdb`) with the target profile (`prf`). Matches from the sequence database are added to the profile.
8. `prf.write()` writes a new profile containing the target sequence and its homologs into the specified output file (file `build_profile.prf`; Fig. 5.6.4). The equivalent information is also written out in standard alignment format.

The `profile.build()` command has many options (see Internet Resources for the MODELLER Web site). In this example, `rr_file` is set to use the BLOSUM62 similarity matrix (file `blosum62.sim.mat` provided in the MODELLER distribution). Accordingly, the parameters `matrix_offset` and `gap_penalties_1d` are set to the appropriate values for the BLOSUM62 matrix. For this example, only one search

```
# Number of sequences:      30
# Length of profile  :     335
# N_PROF_ITERATIONS  :       1
# GAP_PENALTIES_1D   :    -900.0  -50.0
# MATRIX_OFFSET      :    0.0
# RR_FILE            : ${MODINSTALL8v0}/modlib//asl.sim.mat

    1  TvLDH                                S     0   335    1   335    0      0     0    0.  0.0
    2  1a5z                                 X     1   312   75   242   63    229   164   28.  0.83E-08
    3  1b8pA                                X     1   327    7   331    6    325   316   42.  0.0
    4  1bdmA                                X     1   318    1   325    1    310   309   45.  0.0
    5  1t2dA                                X     1   315    5   256    4    250   238   25.  0.66E-04
    6  1civA                                X     1   374    6   334   33    358   325   35.  0.0
    7  2cmd                                 X     1   312    7   320    3    303   289   27.  0.16E-05
    8  1o6zA                                X     1   303    7   320    3    287   278   26.  0.27E-05
    9  1ur5A                                X     1   299   13   191    9    171   158   31.  0.25E-02
   10  1guzA                                X     1   305   13   301    8    280   265   25.  0.28E-08
   11  lgvØA                                X     1   301   13   323    8    289   274   26.  0.28E-04
   12  1hyeA                                X     1   307    7   191    3    183   173   29.  0.14E-07
   13  liOzA                                X     1   332   85   300   94    304   207   25.  0.66E-05
   14  lilOA                                X     1   331   85   295   93    298   196   26.  0.86E-05
   15  lidnA                                X     1   316   78   298   73    301   214   26.  0.19E-03
   16  61dh                                 X     1   329   47   301   56    302   244   23.  0.17E-02
   17  21dx                                 X     1   331   66   306   67    306   227   26.  0.25E-04
   18  51dh                                 X     1   333   85   300   94    304   207   26.  0.30E-05
   19  91dtA                                X     1   331   85   301   93    304   207   26.  0.10E-05
   20  111c                                 X     1   321   64   239   53    234   164   26.  0.20E-03
   21  111dA                                X     1   313   13   242    9    233   216   31.  0.31E-07
   22  5mdhA                                X     1   333    2   332    1    331   328   44.  0.0
   23  7mdhA                                X     1   351    6   334   14    339   325   34.  0.0
   24  lml.dA                               X     1   313    5   198    1    189   183   26.  0.13E-05
   25  1oc4A                                X     1   315    5   191    4    186   174   28.  0.18E-04
   26  1ojuA                                X     1   294   78   320   68    285   218   28.  0.43E-05
   27  1pzgA                                X     1   327   74   191   71    190   114   30.  0.16E-06
   28  1smkA                                X     1   313    7   202    4    198   188   34.  0.0
   29  1sovA                                X     1   316   81   256   76    248   160   27.  0.93E-03
   30  1y63A                                X     1   289   77   191   58    167   109   33.  0.32E-05
```

Figure 5.6.4 An excerpt from the file `build_profile.prf`. The aligned sequences have been removed for convenience.

iteration is run, by setting the parameter `n_prof_iterations` equal to 1. Thus, there is no need to check the profile for deviation (`check_profile` set to False). Finally, the parameter `max_aln_evalue` is set to `0.01`, indicating that only sequences with *E*-values smaller than or equal to 0.01 will be included in the output.

Execute the script using the command:

```
python build_profile.py > build_profile.log
```

(or, if Python is not installed on your machine, with `mod9.13 build_profile.py`). At the end of the execution, a log file is created (`build_profile.log`). MODELLER always produces a log file. Errors and warnings in log files can be found by searching for the `_E>` and `_W>` strings, respectively.

Selecting a template

An extract (omitting the aligned sequences) from the file `build_profile.prf` is shown in Figure 5.6.4. The first six commented lines indicate the input parameters used in MODELLER to create the alignments. Subsequent lines correspond to the detected similarities by `profile.build()`. The most important columns in the output are the second, tenth, eleventh, and twelfth columns. The second column reports the code of the PDB sequence that was aligned to the target sequence. The eleventh column reports the percentage sequence identities between TvLDH and the PDB sequence normalized by the length of the alignment (indicated in the tenth column). In general, a sequence identity value above ~25% indicates a potential template, unless the alignment is too short (i.e., <100 residues). A better measure of the significance of the alignment is given in the twelfth column by the *E*-value of the alignment (lower the *E*-value the better).

```
from modeller import *

env = environ()
aln = alignment(env)
for (pdb, chain) in (('1b8p', 'A'), ('1bdm', 'A'), ('1civ', 'A'),
                     ('5mdh', 'A'), ('7mdh', 'A'), ('1smk', 'A')):
    m = model(env, file=pdb, model_segment=('FIRST:'+chain, 'LAST:'+chain))
    aln.append_model(m, atom_files=pdb, align_codes=pdb+chain)
aln.malign()
aln.malign3d()
aln.compare_structures()
aln.id_table(matrix_file='family.mat')
env.dendrogram(matrix_file='family.mat', cluster_cut=-1.0)
```

Figure 5.6.5 Script file `compare.py`.

In this example, six PDB sequences show very significant similarities to the query sequence, with *E*-values equal to 0. As expected, all the hits correspond to malate dehydrogenases (1bdm:A, 5mdh:A, 1b8p:A, 1civ:A, 7mdh:A, and 1smk:A). To select the appropriate template for the target sequence, the `alignment.compare_structures()` command will first be used to assess the sequence and structure similarity between the six possible templates (file `compare.py`; Fig. 5.6.5).

In `compare.py`, the alignment object `aln` is created and MODELLER is instructed to read into it the protein sequences and information about their PDB files. The command `malign()` calculates their multiple sequence alignment, which is subsequently used as a starting point for creating a multiple structure alignment by `malign3d()`. Based on this structural alignment, the `compare_structures()` command calculates the RMS and DRMS deviations between atomic positions and distances, differences between the main-chain and side-chain dihedral angles, percentage sequence identities, and several other measures. Finally, the `id_table()` command writes a file (`family.mat`) with pairwise sequence distances that can be used as input to the `dendrogram()` command (or the clustering programs in the PHYLIP package; Felsenstein, 1989). `dendrogram()` calculates a clustering tree from the input matrix of pairwise distances, which helps visualizing differences among the template candidates. Excerpts from the log file (`compare.log`) are shown in Figure 5.6.6.

The objective of this step is to select the most appropriate single template structure from all the possible templates. The dendrogram in Figure 5.6.6 shows that 1civ:A and 7mdh:A are almost identical, both in terms of sequence and structure. However, 7mdh:A has a better crystallographic resolution than 1civ:A (2.4 Å versus 2.8 Å). From the second group of similar structures (5mdh:A, 1bdm:A, and 1b8p:A), 1bdm:A has the best resolution (1.8 Å). 1smk:A is most structurally divergent among the possible templates. However, it is also the one with the lowest sequence identity (34%) to the target sequence (`build_profile.prf`). 1bdm:A is finally picked over 7mdh:A as the final template because of its higher overall sequence identity to the target sequence (45%).

Aligning TvLDH with the template

One way to align the sequence of TvLDH with the structure of 1bdm:A is to use the `align2d()` command in MODELLER (Madhusudhan et al., 2006). Although `align2d()` is based on a dynamic programming algorithm (Needleman and Wunsch, 1970), it is different from standard sequence-sequence alignment methods because it takes into account structural information from the template when constructing an alignment. This task is achieved through a variable gap penalty function that tends to place gaps in solvent-exposed and curved regions, outside secondary structure segments, and between two positions that are close in space. In the current example, the target-template similarity

```
Sequence identity comparison (ID_TABLE):

   Diagonal        ... number of residues;
   Upper triangle  ... number of identical residues;
   Lower triangle  ... % sequence identity, id/min(length).

         lb8pA @1lbdmA @1lcivA @25mdhA @27mdhA @2lsmkA @2
1b8pA @1      327     194     147     151     153      49
1bdmA @1       61     318     152     167     155      56
1civA @2       45      48     374     139     304      53
5mdhA @2       46      53      42     333     139      57
7mdhA @2       47      49      87      42     351      48
1smkA @2       16      18      17      18      15     313


Weighted pair-group average clustering based on a distance matrix:


                                        ---------------------- 1b8pA @1.9    39.0000
                                  ------                       1bdmA @1.8    50.5000
                              ----------------------------------
                                                               5mdhA @2.4    55.3750
                                                          ---- 1civA @2.8    13.0000
                                                               7mdhA @2.4    83.2500
                                                               lsmkA @2.5

    +----+----+----+----+----+----+----+----+----+----+----+----+----+
  86.0600   73.4150   60.7700   48.1250   35.4800   22.8350   10.1900
       79.7375   67.0925   54.4475   41.8025   29.1575   16.5125
```

Figure 5.6.6 Excerpts from the log file `compare.log`.

```
from modeller import *

env = environ()
aln = alignment(env)
mdl = model(env, file='1bdm', model_segment=('FIRST:A','LAST:A'))
aln.append_model(mdl, align_codes='1bdmA', atom_files='1bdm.pdb')
aln.append(file='TvLDH.ali', align_codes='TvLDH')
aln.align2d()
aln.write(file='TvLDH-1bdmA.ali', alignment_format='PIR')
aln.write(file='TvLDH-1bdmA.pap', alignment_format='PAP')
```

Figure 5.6.7 The script file `align2d.py`, used to align the target sequence against the template structure.

is so high that almost any alignment method with reasonable parameters will result in the same alignment.

The MODELLER script shown in Figure 5.6.7 aligns the TvLDH sequence in file `TvLDH.ali` with the 1bdm:A structure in the PDB file `1bdm.pdb` (file `align2d.py`). In the first line of the script, an empty alignment object `aln`, and a new model object `mdl`, into which the chain A of the 1bdm structure is read, are created. `append_model()` transfers the PDB sequence of this model to `aln` and assigns it the name of `1bdmA` (`align_codes`). The `TvLDH` sequence, from file `TvLDH.ali`, is then added to `aln` using `append()`. The `align2d()` command aligns the two sequences and the alignment is written out in two formats, PIR (`TvLDH-1bdmA.ali`) and PAP (`TvLDH-1bdmA.pap`). The PIR format is used by MODELLER in the subsequent model-building stage, while the PAP alignment format is easier to inspect visually. In

```
 _aln.pos        10        20        30        40        50        60
ThdmA    MKAPVRVAVTGAAGQIGYSLLFRIAAGEMLGKDQPVILQLLEIPQAMKALEGVVMELEDCAFPLLAGL
TvLDH    MSEAAHVLITGAAGQIGYILSHWIASGELYG-DRQVYLHLLDIPPAMNRLTALTMELEDCAFPHLAGF
_consrvd *     *  ********* *   ** **  * *  * * ** ** **  *    ********* ***

_aln.p   70        80        90       100       110       120       130
ThdmA    EATDDPDVAFKDADYALLVGAAPRL --------QVNGKIFTEQGRALAEVAKKDVKVLVVGNPANTN
TvLDH    VATTDPKAAFKDIDCAFLVASMPLKPGQVRADLISSNSVIFKNTGEYLSKWAKPSVKVLVIGNPDNTN
_consrvd  ** **  **** * * **   *             *  **   *  *   **  ***** *** ***

_aln.pos  140       150       160       170       180       190       200
1bdmA    ALIAYKNAPGLNPRNFTAMTRLDHNRAKAQLAKKTGTGVDRIRRMTVWGNHSSIMFPDLFHAEVD---
TvLDH    CEIAMLHAKNLKPENFSSLSMLDQNRAYYEVASKLGVDVKDVHDIIVWGNHGESMVADLTQATFTKEG
_consrvd   **   *  * * **     ** ***    * * *  *       *****   *  **  *

_aln.pos    210       220       230       240       250       260       270
ThdmA    -GRPALELVDMEWYEKVFIPTVAQRGAAIIQARGASSAASAANAAIEHIRDWALGTPEGDWVSMAVPS
TvLDH    KTQKVVDVLDHDYVFDTFFKKIGHRAWDILEHRGFTSAASPTKAAIQHMKAWLFGTAPGEVLSMGIPV
_consrvd          *       *      *   *   **  ****   *** *    * **  *   **  *

_aln.pos      280       290       300       310       320       330
ThdmA    Q--GEYGIPEGIVYSFPVTAK-DGAYRVVEGLEINEFARKRMEITAQELLDEMEQVKAL--GLI
TvLDH    PEGNPYGIKPGVVFSFPCNVDKEGKIHVVEGFKVNDWLREKLDFTEKDLFHEKEIALNHLAQGG
_consrvd      ***  * * ***      *   ****   *   *     *   *  * *
```

Figure 5.6.8 The alignment between sequences `TvLDH` and `1bdmA`, in the MODELLER PAP format. File `TvLDH-1bdmA.pap`.

```
from modeller import *
from modeller.automodel imort *
#from modeller import soap_protein_od

env = environ()
a = automodel(env, alnfile= TvLDH-1bdmA.ali ,
              knowns= 1bdmA , sequence=  TvLDH ,
              assess_methods=(assess.DOPE,
                              #soap_protein_od.Scorer(),
                              assess.GA341))

a.starting_model = 1
a.ending_model = 5
a.make()
```

Figure 5.6.9 Script file, `model-single.py`, that generates five models.

the PAP format, all identical positions are marked with a `*` (file `TvLDH-1bdmA.pap`; Fig. 5.6.8). Due to the high target-template similarity, there are only a few gaps in the alignment.

Model building

Once a target-template alignment is constructed, MODELLER calculates a 3-D model of the target completely automatically, using its `automodel` class. The script in Figure 5.6.9 will generate five different models of TvLDH based on the 1bdm:A template structure and the alignment in file `TvLDH-1bdmA.ali` (file `model-single.py`).

The first line (Fig. 5.6.9) loads the `automodel` class and prepares it for use. An `automodel` object is then created and called "a" and parameters are set to guide the model-building procedure. `alnfile` names the file that contains the target-template alignment in the PIR format. `knowns` defines the known template structure(s) in `alnfile` (`TvLDH-1bdmA.ali`) and `sequence` defines the code of the target sequence. `starting_model` and `ending_model` define the number of models that are calculated (their indices will run from 1 to 5). The last line in the file calls the `make` method that actually calculates the models. The most important output files are

`model-single.log`, which reports warnings, errors and other useful information including the input restraints used for modeling that remain violated in the final model, and `TvLDH.B9999000[1-5].pdb`, which contain the coordinates of the five produced models, in the PDB format. The models can be viewed by any program that reads the PDB format, such as Chimera (*http://www.cgl.ucsf.edu/chimera/*) or RasMol (*http://www.rasmol.org*).

Evaluating a model

If several models are calculated for the same target, the best model can be selected by picking the model with the lowest value of the MODELLER objective function or the DOPE (Shen and Sali, 2006) or SOAP (Dong et al., 2013) assessment scores, which are reported at the end of the log file. (To calculate the SOAP score, download the SOAP-Protein library file from *http://salilab.org/SOAP/* and uncomment the two SOAP-related lines in `model-single.py` by removing the '#' characters.) In this example, the second model (`TvLDH.B99990002.pdb`) has the lowest objective function and is selected. None of these scores are absolute measures, meaning that they can only be used to rank models calculated from the same alignment.

Once a final model is selected, there are many ways to further assess it. In this example, the DOPE potential in MODELLER is used to evaluate the fold of the selected model. Links to other programs for model assessment can be found in Table 5.6.1. However, before any external evaluation of the model, one should check the log file from the modeling run for runtime errors (`model-single.log`) and restraint violations (see the MODELLER manual for details).

The script, `evaluate_model.py` (Fig. 5.6.10) evaluates the model with the DOPE potential. In this script, the atomic coordinates of the PDB file are read in (using `complete_pdb()`) to a model object, `mdl`. This is necessary for MODELLER to correctly calculate the energy, and additionally allows for the possibility of the PDB file having atoms in a nonstandard order, or having different subsets of atoms (e.g., all atoms including hydrogens, while MODELLER uses only heavy atoms, or vice versa). The DOPE energy is then calculated using `assess_dope()`. An energy profile is additionally requested, smoothed over a 15-residue window, and normalized by the number of restraints acting on each residue. This profile is written to a file `TvLDH.profile`, which can be used as input to a graphing program such as GNUPLOT.

Similarly, the profile can be calculated for the template structure (see the scripts `evaluate_template.py` and `plot_profiles.py` in the zipfile). A comparison of the two profiles is shown in Figure 5.6.11. It can be seen that the DOPE score profile

```
from modeller import *
from modeller.scripts import complete_pdb

log.verbose()    # request verbose output
env = environ()
env.libs.topology.read(file='$(LIB)/top_heav.lib') # read topology
env.libs.parameters.read(file='$(LIB)/par.lib') # read parameters

# read model file
mdl = complete_pdb(env, 'TvLDH.B99990002.pdb')

# Assess with DOPE:
s = selection(mdl)   # all atom selection
s.assess_dope(output='ENERGY_PROFILE NO_REPORT', file='TvLDH.profile',
              normalize_profile=True, smoothing_window=15)
```

Figure 5.6.10 File `evaluate_model.py`, used to generate a pseudo-energy profile for the model.

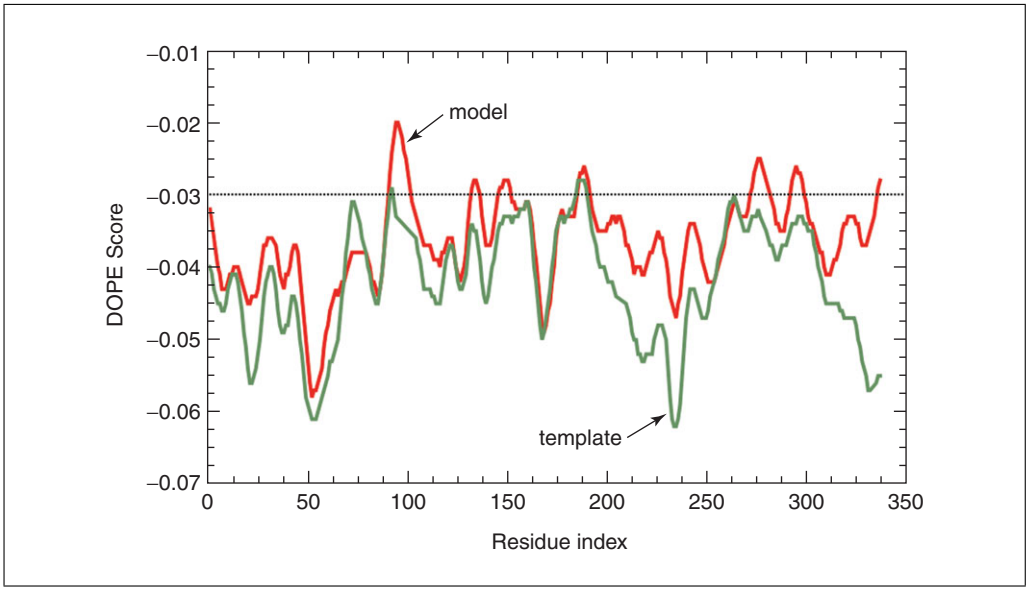

Figure 5.6.11 A comparison of the pseudo-energy profiles of the model (red) and the template (green) structures.

shows clear differences between the two profiles for the long active-site loop between residues 90 and 100 and the long helices at the C-terminal end of the target sequence. This long loop interacts with region 220 to 250, which forms the other half of the active site. This latter region is well resolved in both the template and the target structure. However, probably due to the unfavorable nonbonded interactions with the 90 to 100 region, it is reported to be of high energy by DOPE. It is to be noted that a region of high energy indicated by DOPE may not always necessarily indicate actual error, especially when it highlights an active site or a protein-protein interface. However, in this case, the same active-site loops have a better profile in the template structure, which strengthens the argument that the model is probably incorrect in the active-site region. Resolution of such problems is beyond the scope of this unit, but is described in a more advanced modeling tutorial available at *http://salilab.org/modeller/tutorial/advanced.html*.

OBTAINING AND INSTALLING MODELLER

SUPPORT PROTOCOL

MODELLER is written in Fortran 90 and uses Python for its control language. All input scripts to MODELLER are, hence, Python scripts. While knowledge of Python is not necessary to run MODELLER, it can be useful in performing more advanced tasks. Precompiled binaries for MODELLER can be downloaded from *http://salilab.org/modeller*.

Necessary Resources

Hardware

A computer running RedHat Linux (PC, Opteron or EM64T/Xeon64 systems) or other version of Linux/Unix (x86/x86_64 Linux, AIX), Apple Mac OS X (10.6 or later), or Microsoft Windows (XP or later)

Software

An up-to-date Internet browser, such as Internet Explorer (*http://www.microsoft.com/ie*); Chrome (*http://chrome.google.com*); Firefox (*http://www.mozilla.org/firefox*); or Safari (*http://www.apple.com/safari*)

Installation

The steps involved in installing MODELLER on a computer depend on its operating system. The following procedure describes the steps for installing MODELLER on a generic x86 PC running any Unix/Linux operating system. The procedures for other operating systems differ slightly. Detailed instructions for installing MODELLER on machines running other operating systems can be found at *http://salilab.org/modeller/release.html*. In particular, installer packages are available for Windows, Mac, RedHat Linux, and Debian/Ubuntu Linux.

1. Point browser to *http://salilab.org/modeller/download_installation.html*.
2. On the page that appears, download the distribution by clicking on the link entitled "Generic Unix tarball" under "Available downloads . . . ".
3. A valid license key, distributed free of cost to academic users, is required to use MODELLER. To obtain a key, go to the URL *http://salilab.org/modeller/registration.html*, fill in the simple form at the bottom of the page, and read and accept the license agreement. The key will be e-mailed to the address provided.
4. Open a terminal or console and change to the directory containing the downloaded distribution. The distributed file is a compressed archive file called `modeller-9.13.tar.gz`.
5. Unpack the downloaded file with the following commands:

```
gunzip modeller-9.13.tar.gz
tar -xvf modeller-9.13.tar
```

6. The files needed for the installation can be found in a newly created directory called `modeller-9.13`. Move into that directory and start the installation with the following commands:

```
cd modeller-9.13
./Install
```

7. The installation script will prompt the user with several questions and suggest default answers. To accept the default answers, press the `Enter` key. The various prompts are briefly discussed below:

   a. For the prompt below, choose the appropriate combination of the machine architecture and operating system. For this example, choose the default answer by pressing the `Enter` key.

```
The currently supported architectures are as follows:

     1) Linux x86 PC (e.g., RedHat, SuSe).
     2) IBM AIX OS.
     3) x86_64 (Opteron/EM64T) box (Linux).
     4) Alternative Linux x86 PC binary (e.g., for
       FreeBSD).
     Select the type of your computer from the list
       above [1]:
```

   b. For the prompt below, tell the installer where to install the MODELLER executables. The default choice will place it in the directory indicated, but any directory to which the user has write permissions may be specified.

```
Full directory name for the installed MODELLER9.13
[<YOUR-HOME-DIRECTORY>/bin/modeller9.13]:
```

   c. For the prompt below, enter the MODELLER license key obtained in step 3.

KEY_MODELLER9v13, obtained from our academic license server at *http://salilab.org/modeller/registration.html*:

8. The installer will now confirm the answers to the above prompts. Press `Enter` to begin the installation. The `mod9.13` script installed in the chosen directory can now be used to invoke MODELLER. The installer will also provide information on how to set up MODELLER to work with your operating system's copy of Python.

Other resources

9. The MODELLER Web site provides links to several additional resources that can supplement the tutorial provided in this unit, as follows.

   a. News about the latest MODELLER releases can be found at *http://salilab.org/modeller/news.html*.
   b. There is a discussion forum, operated through a mailing list, devoted to providing tips, tricks, and practical help in using MODELLER. Users can subscribe to the mailing list at *http://salilab.org/modeller/discussion_forum.html*. Users can also browse through or search the archived messages of the mailing list.
   c. The documentation section of the web page contains links to Frequently Asked Questions (FAQ; *http://salilab.org/modeller/FAQ.html*), tutorial examples (*http://salilab.org/modeller/tutorial*), an online version of the manual (*http://salilab.org/modeller/manual*), and user-editable Wiki pages (*http://salilab.org/modeller/wiki/*) to exchange tips, scripts, and examples.

COMMENTARY

Background Information

As stated earlier, comparative modeling consists of four main steps: fold assignment, target-template alignment, model building and model evaluation (Marti-Renom et al., 2000; Fig. 5.6.1).

Fold assignment and target-template alignment

Although fold assignment and sequence-structure alignment are logically two distinct steps in the process of comparative modeling, in practice, almost all fold-assignment methods also provide sequence-structure alignments. In the past, fold-assignment methods were optimized for better sensitivity in detecting remotely related homologs, often at the cost of alignment accuracy. However, recent methods simultaneously optimize both the sensitivity and alignment accuracy. Therefore, in the following discussion, fold assignment and sequence-structure alignment will be treated as a single procedure, explaining the differences as needed.

Fold assignment

The primary requirement for comparative modeling is the identification of one or more known template structures with detectable similarity to the target sequence. The identification of suitable templates is achieved by scanning structure databases, such as PDB (Berman et al., 2000), SCOP (Andreeva et al., 2004), DALI, *UNIT 5.5* (Dietmann et al., 2001), and CATH (Pearl et al., 2005), with the target sequence as the query. The detected similarity is usually quantified in terms of sequence identity or statistical measures such as *E*-value or *z*-score, depending on the method used.

Three regimes of the sequence-structure relationship

The sequence-structure relationship can be subdivided into three different regimes in the sequence similarity spectrum: (i) the easily detected relationships, characterized by >30% sequence identity; (ii) the "twilight zone" (Rost, 1999), corresponding to relationships with statistically significant sequence similarity, with identities in the 10% to 30% range; and (iii) the "midnight zone" (Rost, 1999), corresponding to statistically insignificant sequence similarity.

Pairwise sequence alignment methods

For closely related protein sequences with identities higher than 30% to 40%, the alignments produced by all methods are almost always largely correct. The quickest way to search for suitable templates in this regime is to use simple pairwise sequence alignment methods such as SSEARCH (Pearson, 1994), BLAST (Altschul et al., 1997), and FASTA (Pearson, 1994). Brenner et al. (1998) showed that these methods detect only ~18% of the

homologous pairs at less than 40% sequence identity, while they identify more than 90% of the relationships when sequence identity is between 30% and 40% (Brenner et al., 1998). Another benchmark, based on 200 reference structural alignments with 0% to 40% sequence identity, indicated that BLAST is able to correctly align only 26% of the residue positions (Sauder et al., 2000).

Profile-sequence alignment methods

The sensitivity of the search and accuracy of the alignment become progressively difficult as the relationships move into the twilight zone (Saqi et al., 1998; Rost, 1999). A significant improvement in this area was the introduction of profile methods by Gribskov et al. (1987). The profile of a sequence is derived from a multiple sequence alignment and specifies residue-type occurrences for each alignment position. The information in a multiple sequence alignment is most often encoded as either a position-specific scoring matrix (PSSM; Henikoff and Henikoff, 1994; Altschul et al., 1997) or as a Hidden Markov Model (HMM; Krogh et al., 1994; Eddy, 1998). In order to identify suitable templates for comparative modeling, the profile of the target sequence is used to search against a database of template sequences. The profile-sequence methods are more sensitive in detecting related structures in the twilight zone than the pairwise sequence-based methods; they detect approximately twice the number of homologs under 40% sequence identity (Park et al., 1998; Lindahl and Elofsson, 2000; Sauder et al., 2000). The resulting profile-sequence alignments correctly align approximately 43% to 48% of residues in the 0% to 40% sequence identity range (Sauder et al., 2000; Marti-Renom et al., 2004); this number is almost twice as large as that of the pairwise sequence methods. Frequently used programs for profile-sequence alignment are PSI-BLAST (Altschul et al., 1997), SAM (Karplus et al., 1998), HMMER (Eddy, 1998), HHsearch (Soding, 2005), HHBlits (Remmert et al., 2012), and BUILD_PROFILE (part of MODELLER; Sali and Blundell, 1993).

Profile-profile alignment methods

As a natural extension, the profile-sequence alignment methods have led to profile-profile alignment methods that search for suitable template structures by scanning the profile of the target sequence against a database of template profiles as opposed to a database of template sequences. These methods have proven to include the most sensitive and accurate fold assignment and alignment protocols to date (Edgar and Sjolander, 2004; Marti-Renom et al., 2004; Ohlson et al., 2004; Wang and Dunbrack, 2004). Profile-profile methods detect ~28% more relationships at the superfamily level and improve the alignment accuracy by 15% to 20%, compared to profile-sequence methods (Marti-Renom et al., 2004; Zhou and Zhou, 2005). There are a number of variants of profile-profile alignment methods that differ in the scoring functions they use (Pietrokovski, 1996; Rychlewski et al., 1998; Yona and Levitt, 2002; Panchenko, 2003; Sadreyev and Grishin, 2003; von Ohsen et al., 2003; Edgar and Sjolander, 2004; Marti-Renom et al., 2004; Zhou and Zhou, 2005). However, several analyses have shown that the overall performances of these methods are comparable (Edgar and Sjolander, 2004; Marti-Renom et al., 2004; Ohlson et al., 2004; Wang and Dunbrack, 2004). Some of the programs that can be used to detect suitable templates are FFAS (Jaroszewski et al., 2005), SP3 (Zhou and Zhou, 2005), SALIGN (Marti-Renom et al., 2004), HHBlits (Remmert et al., 2012), HHsearch (Soding, 2005), and PPSCAN (part of MODELLER; Sali and Blundell, 1993).

Sequence-structure threading methods

As the sequence identity drops below the threshold of the twilight zone, there is usually insufficient signal in the sequences or their profiles for the sequence-based methods discussed above to detect true relationships (Lindahl and Elofsson, 2000). Sequence-structure threading methods are most useful in this regime, as they can sometimes recognize common folds even in the absence of any statistically significant sequence similarity (Godzik, 2003). These methods achieve higher sensitivity by using structural information derived from the templates. The accuracy of a sequence-structure match is assessed by the score of a corresponding coarse model and not by sequence similarity, as in sequence-comparison methods (Godzik, 2003). The scoring scheme used to evaluate the accuracy is either based on residue substitution tables dependent on structural features such as solvent exposure, secondary structure type, and hydrogen-bonding properties (Shi et al., 2001; Karchin et al., 2003; McGuffin and Jones, 2003; Zhou and Zhou, 2005), or on statistical potentials for residue interactions implied by the alignment (Sippl, 1990, 1995; Bowie et al., 1991; Skolnick and Kihara, 2001; Xu

et al., 2003). The use of structural data does not have to be restricted to the structure side of the aligned sequence-structure pair. For example, SAM-T08 makes use of the predicted local structure for the target sequence to enhance homolog detection and alignment accuracy (Karplus et al., 2003). Commonly used threading programs are GenTHREADER (Jones, 1999; McGuffin and Jones, 2003), 3D-PSSM (Kelley et al., 2000), FUGUE (Shi et al., 2001), SP3 (Zhou and Zhou, 2005), SAM-T08 multi-track HMM (Karchin et al., 2003; Karplus et al., 2003), and MUSTER (Wu and Zhang, 2008).

Iterative sequence-structure alignment and model building

Yet another strategy is to optimize the alignment by iterating over the process of calculating alignments, building models, and evaluating models. Such a protocol can sample alignments that are not statistically significant and identify the alignment that yields the best model. Although this procedure can be time consuming, it can significantly improve the accuracy of the resulting comparative models in difficult cases (John and Sali, 2003).

Importance of an accurate alignment

Regardless of the method used, searching in the twilight and midnight zones of the sequence-structure relationship often results in false negatives, false positives, or alignments that contain an increasingly large number of gaps and alignment errors. Improving the performance and accuracy of methods in this regime remains one of the main tasks of comparative modeling today (Moult, 2005). It is imperative to calculate an accurate alignment between the target-template pair, as comparative modeling can almost never recover from an alignment error (Sanchez and Sali, 1997a).

Template selection

After a list of all related protein structures and their alignments with the target sequence have been obtained, template structures are prioritized depending on the purpose of the comparative model. Template structures may be chosen based purely on the target-template sequence identity, or on a combination of several other criteria, such as experimental accuracy of the structures (resolution of X-ray structures, number of restraints per residue for NMR structures), conservation of active-site residues, holo-structures that have bound ligands of interest, and prior biological information that pertains to the solvent, pH, and quaternary contacts. It is not necessary to select only one template. In fact, the use of several templates approximately equidistant from the target sequence generally increases the model accuracy (Srinivasan and Blundell, 1993; Sanchez and Sali, 1997b).

Model building

Modeling by assembly of rigid bodies

The first and still widely used approach in comparative modeling is to assemble a model from a small number of rigid bodies obtained from the aligned protein structures (Browne et al., 1969; Greer, 1981; Blundell et al., 1987). The approach is based on the natural dissection of the protein structures into conserved core regions, variable loops that connect them, and side chains that decorate the backbone. For example, the following semiautomated procedure is implemented in the computer program COMPOSER (Sutcliffe et al., 1987a). First, the template structures are selected and superposed. Second, the "framework" is calculated by averaging the coordinates of the C^{α} atoms of structurally conserved regions in the template structures. Third, the main-chain atoms of each core region in the target model are obtained by superposing the core segment, from the template whose sequence is closest to the target, on the framework. Fourth, the loops are generated by scanning a database of all known protein structures to identify the structurally variable regions that fit the anchor core regions and have a compatible sequence (Topham et al., 1993). Fifth, the side chains are modeled based on their intrinsic conformational preferences and on the conformation of the equivalent side chains in the template structures (Sutcliffe et al., 1987b). Finally, the stereochemistry of the model is improved either by a restrained energy minimization or a molecular dynamics refinement. The accuracy of a model can be somewhat increased when more than one template structure is used to construct the framework and when the templates are averaged into the framework using weights corresponding to their sequence similarities to the target sequence (Srinivasan and Blundell, 1993). Possible future improvements of modeling by rigid-body assembly include incorporation of rigid body shifts, such as the relative shifts in the packing of a helices and β-sheets (Nagarajaram et al., 1999). Three other programs that implement this method are 3D-JIGSAW (Bates et al., 2001), RosettaCM (Song et al., 2013), and SWISS-MODEL (Schwede et al., 2003).

Modeling by segment matching or coordinate reconstruction

The basis of modeling by coordinate reconstruction is the finding that most hexapeptide segments of protein structure can be clustered into only 100 structurally different classes (Jones and Thirup, 1986; Claessens et al., 1989; Unger et al., 1989; Levitt, 1992; Bystroff and Baker, 1998). Thus, comparative models can be constructed by using a subset of atomic positions from template structures as guiding positions to identify and assemble short, all-atom segments that fit these guiding positions. The guiding positions usually correspond to the C^{α} atoms of the segments that are conserved in the alignment between the template structure and the target sequence. The all-atom segments that fit the guiding positions can be obtained either by scanning all known protein structures, including those that are not related to the sequence being modeled (Claessens et al., 1989; Holm and Sander, 1991), or by a conformational search restrained by an energy function (Bruccoleri and Karplus, 1987; van Gelder et al., 1994). This method can construct both main-chain and side-chain atoms, and can also model unaligned regions (gaps). It is implemented in the program SegMod (Levitt, 1992). Even some side-chain modeling methods (Chinea et al., 1995) and the class of loop-construction methods based on finding suitable fragments in the database of known structures (Jones and Thirup, 1986) can be seen as segment-matching or coordinate-reconstruction methods.

Modeling by satisfaction of spatial restraints

The methods in this class begin by generating many constraints or restraints on the structure of the target sequence, using its alignment to related protein structures as a guide. The procedure is conceptually similar to that used in determination of protein structures from NMR-derived restraints. The restraints are generally obtained by assuming that the corresponding distances between aligned residues in the template and the target structures are similar. These homology-derived restraints are usually supplemented by stereochemical restraints on bond lengths, bond angles, dihedral angles, and nonbonded atom-atom contacts that are obtained from a molecular mechanics force field. The model is then derived by minimizing the violations of all the restraints. This optimization can be achieved either by distance geometry or real-space optimization. For example, an elegant distance geometry approach constructs all-atom models from lower and upper bounds on distances and dihedral angles (Havel and Snow, 1991).

Comparative protein structure modeling by MODELLER. MODELLER, the authors' own program for comparative modeling, belongs to this group of methods (Sali and Blundell, 1993; Sali and Overington, 1994; Fiser et al., 2000; Fiser et al., 2002). MODELLER implements comparative protein structure modeling by satisfaction of spatial restraints. The program was designed to use as many different types of information about the target sequence as possible.

Homology-derived restraints. In the first step of model building, distance and dihedral angle restraints on the target sequence are derived from its alignment with template 3-D structures. The form of these restraints was obtained from a statistical analysis of the relationships between similar protein structures. The analysis relied on a database of 105 family alignments that included 416 proteins of known 3-D structure (Sali and Overington, 1994). By scanning the database of alignments, tables quantifying various correlations were obtained, such as the correlations between two equivalent C^{α}-C^{α} distances, or between equivalent main-chain dihedral angles from two related proteins (Sali and Blundell, 1993). These relationships are expressed as conditional probability density functions (pdf's), and can be used directly as spatial restraints. For example, probabilities for different values of the main-chain dihedral angles are calculated from the type of residue considered, from main-chain conformation of an equivalent residue, and from sequence similarity between the two proteins. Another example is the pdf for a certain C^{α}-C^{α} distance given equivalent distances in two related protein structures. An important feature of the method is that the form of spatial restraints was obtained empirically, from a database of protein structure alignments.

Stereochemical restraints. In the second step, the spatial restraints and the CHARMM22 force-field terms enforcing proper stereochemistry (MacKerell et al., 1998) are combined into an objective function. The general form of the objective function is similar to that in molecular dynamics programs, such as CHARMM22 (MacKerell et al., 1998). The objective function depends on the Cartesian coordinates of ~10,000 atoms (3-D points) that form the modeled molecules. For a 10,000-atom system, there can be on the order of 200,000 restraints. The functional

form of each term is simple; it includes a quadratic function, harmonic lower and upper bounds, cosine, a weighted sum of a few Gaussian functions, Coulomb's law, Lennard-Jones potential, and cubic splines. The geometric features presently include a distance, an angle, a dihedral angle, a pair of dihedral angles between two, three, four, and eight atoms, respectively, the shortest distance in the set of distances, solvent accessibility, and atom density that is expressed as the number of atoms around the central atom. Some restraints can be used to restrain pseudo-atoms, e.g., the gravity center of several atoms.

Optimization of the objective function. Finally, the model is obtained by optimizing the objective function in Cartesian space. The optimization is carried out by the use of the variable target function method (Braun and Go, 1985), employing methods of conjugate gradients and molecular dynamics with simulated annealing (Clore et al., 1986). Several slightly different models can be calculated by varying the initial structure, and the variability among these models can be used to estimate the lower bound on the errors in the corresponding regions of the fold.

Restraints derived from experimental data. Because the modeling by satisfaction of spatial restraints can use many different types of information about the target sequence, it is perhaps the most promising of all comparative modeling techniques. One of the strengths of modeling by satisfaction of spatial restraints is that restraints derived from a number of different sources can easily be added to the homology-derived restraints. For example, restraints could be provided by rules for secondary-structure packing (Cohen et al., 1989), analyses of hydrophobicity (Aszodi and Taylor, 1994) and correlated mutations (Taylor et al., 1994), empirical potentials of mean force (Sippl, 1990), nuclear magnetic resonance (NMR) experiments (Sutcliffe et al., 1992), cross-linking experiments, fluorescence spectroscopy, image reconstruction in electron microscopy, site-directed mutagenesis (Boissel et al., 1993), and intuition, among other sources. Especially in difficult cases, a comparative model could be improved by making it consistent with available experimental data and/or with more general knowledge about protein structure.

Relative accuracy, flexibility, and automation. Accuracies of the various model-building methods are relatively similar when used optimally (Marti-Renom et al., 2002). Other factors such as template selection and alignment accuracy usually have a larger impact on the model accuracy, especially for models based on low sequence identity to the templates. However, it is important that a modeling method allow a degree of flexibility and automation to obtain better models more easily and rapidly. For example, a method should allow for an easy recalculation of a model when a change is made in the alignment. It should also be straightforward enough to calculate models based on several templates, and should provide tools for incorporation of prior knowledge about the target (e.g., cross-linking restraints, predicted secondary structure) and allow ab initio modeling of insertions (e.g., loops), which can be crucial for annotation of function.

Loop modeling

Loop modeling is an especially important aspect of comparative modeling in the range of 30% to 50% sequence identity. In this range of overall similarity, loops among the homologs vary while the core regions are still relatively conserved and aligned accurately. Loops often play an important role in defining the functional specificity of a given protein, forming the active and binding sites. Loop modeling can be seen as a mini protein folding problem, because the correct conformation of a given segment of a polypeptide chain has to be calculated mainly from the sequence of the segment itself. However, loops are generally too short to provide sufficient information about their local fold. Even identical decapeptides in different proteins do not always have the same conformation (Kabsch and Sander, 1984; Mezei, 1998). Some additional restraints are provided by the core anchor regions that span the loop and by the structure of the rest of the protein that cradles the loop. Although many loop-modeling methods have been described, it is still challenging to correctly and confidently model loops longer than ~10 to 12 residues (Fiser et al., 2000; Jacobson et al., 2004; Zhu et al., 2006).

There are two main classes of loop-modeling methods: (i) database search approaches that scan a database of all known protein structures to find segments fitting the anchor core regions (Jones and Thirup, 1986; Chothia and Lesk, 1987); (ii) conformational search approaches that rely on optimizing a scoring function (Moult and James, 1986; Bruccoleri and Karplus, 1987; Shenkin et al., 1987). There are also methods that combine these two approaches (van Vlijmen and Karplus, 1997; Deane and Blundell, 2001).

Loop modeling by database search. The database search approach to loop modeling is accurate and efficient when a database of specific loops is created to address the modeling of the same class of loops, such as β-hairpins (Sibanda et al., 1989), or loops on a specific fold, such as the hypervariable regions in the immunoglobulin fold (Chothia and Lesk, 1987; Chothia et al., 1989). There are attempts to classify loop conformations into more general categories, thus extending the applicability of the database search approach (Ring et al., 1992; Oliva et al., 1997; Rufino et al., 1997; Fernandez-Fuentes and Fiser, 2006). However, the database methods are limited because the number of possible conformations increases exponentially with the length of a loop, and until the late 1990s only loops up to 7 residues long could be modeled using the database of known protein structures (Fidelis et al., 1994; Lessel and Schomburg, 1994). However, the growth of the PDB in recent years has largely eliminated this problem (Fernandez-Fuentes and Fiser, 2006).

Loop modeling by conformational search. There are many such methods, exploiting different protein representations, objective functions, and optimization or enumeration algorithms. The search algorithms include the minimum perturbation method (Fine et al., 1986), dihedral angle search through a rotamer library (Zhu et al., 2006; Sellers et al., 2008), molecular dynamics simulations (Bruccoleri and Karplus, 1990; van Vlijmen and Karplus, 1997), genetic algorithms (Ring et al., 1993), Monte Carlo and simulated annealing (Higo et al., 1992; Collura et al., 1993; Abagyan and Totrov, 1994), multiple copy simultaneous search (Zheng et al., 1993), self-consistent field optimization (Koehl and Delarue, 1995), robotics-inspired kinematic closure (Mandell et al., 2009), and enumeration based on graph theory (Samudrala and Moult, 1998). The accuracy of loop predictions can be further improved by clustering the sampled loop conformations and partially accounting for the entropic contribution to the free energy (Xiang et al., 2002). Another way to improve the accuracy of loop predictions is to consider the solvent effects. Improvements in implicit solvation models, such as the Generalized Born solvation model, motivated their use in loop modeling. The solvent contribution to the free energy can be added to the scoring function for optimization, or it can be used to rank the sampled loop conformations after they are generated with a scoring function that does not include the solvent terms (Fiser et al., 2000; Felts et al., 2002; de Bakker et al., 2003; DePristo et al., 2003).

Loop modeling in MODELLER. The loop-modeling module in MODELLER implements the optimization-based approach (Fiser et al., 2000; Fiser and Sali, 2003b). The main reasons for choosing this implementation are the generality and conceptual simplicity of scoring function minimization. Loop prediction by optimization is applicable to simultaneous modeling of several loops and loops interacting with ligands, which is not straightforward with the database-search approaches. Loop optimization in MODELLER relies on conjugate gradients and molecular dynamics with simulated annealing. The pseudo energy function is a sum of many terms, including some terms from the CHARMM22 molecular mechanics force field (MacKerell et al., 1998) and spatial restraints based on distributions of distances (Sippl, 1990; Melo et al., 2002) and dihedral angles in known protein structures. The method was tested on a large number of loops of known structure, both in the native and near-native environments (Fiser et al., 2000).

Comparative model building by iterative alignment, model building, and model assessment

Comparative or homology protein structure modeling is severely limited by errors in the alignment of a modeled sequence with related proteins of known three-dimensional structure. To ameliorate this problem, one can use an iterative method that optimizes both the alignment and the model implied by it (Sanchez and Sali, 1997a; Miwa et al., 1999). This task can be achieved by a genetic algorithm protocol that starts with a set of initial alignments and then iterates through realignment, model building, and model assessment to optimize a model assessment score (John and Sali, 2003). During this iterative process: (1) new alignments are constructed by the application of a number of genetic algorithm operators, such as alignment mutations and crossovers; (2) comparative models corresponding to these alignments are built by satisfaction of spatial restraints, as implemented in the program MODELLER; and (3) the models are assessed by a composite score, partly depending on an atomic statistical potential (Melo et al., 2002). When testing the procedure on a very difficult set of 19 modeling targets sharing only 4% to 27% sequence identity with their template structures, the average final alignment accuracy increased from 37% to 45% relative to the initial alignment (the

alignment accuracy was measured as the percentage of positions in the tested alignment that were identical to the reference structure-based alignment). Correspondingly, the average model accuracy increased from 43% to 54% (the model accuracy was measured as the percentage of the C^{α} atoms of the model that were within 5 Å of the corresponding C^{α} atoms in the superimposed native structure).

Errors in comparative models

As the similarity between the target and the templates decreases, the errors in the model increase. Errors in comparative models can be divided into five categories (Sanchez and Sali, 1997a,b; Fig. 5.6.12), as follows:

Errors in side-chain packing (Fig. 5.6.12A). As the sequences diverge, the packing of side chains in the protein core changes. Sometimes even the conformation of identical side chains is not conserved, a pitfall for many comparative modeling methods. Side-chain errors are critical if they occur in regions that are involved in protein function, such as active sites and ligand-binding sites.

Distortions and shifts in correctly aligned regions (Fig. 5.6.12B). As a consequence of sequence divergence, the main-chain conformation changes, even if the overall fold remains the same. Therefore, it is possible that in some correctly aligned segments of a model the template is locally different (>3 Å) from the target, resulting in errors in that region. The structural differences are sometimes not due to differences in sequence, but are a consequence of artifacts in structure determination or structure determination in different environments (e.g., packing of subunits in a crystal). The simultaneous use of several templates can minimize this kind of error (Srinivasan and Blundell, 1993; Sanchez and Sali, 1997a,b).

Errors in regions without a template (Fig. 5.6.12C). Segments of the target sequence that have no equivalent region in the template structure (i.e., insertions or loops) are the most difficult regions to model. If the insertion is relatively short, <9 residues long, some methods can correctly predict the conformation of the backbone (van Vlijmen and Karplus, 1997; Fiser et al., 2000; Jacobson et al., 2004). Conditions for successful prediction are the correct alignment and an accurately modeled environment surrounding the insertion.

Errors due to misalignments (Fig. 5.6.12D). The largest single source of errors in comparative modeling is misalignments, especially when the target-template sequence identity decreases below 30%. However, alignment errors can be minimized in two ways. First, it is usually possible to use a large number of sequences to construct a multiple alignment, even if most of these sequences do not have known structures. Multiple alignments are generally more reliable than pairwise alignments (Barton and Sternberg, 1987; Taylor et al., 1994). The second way of improving the alignment is to iteratively modify those regions in the alignment that correspond to predicted errors in the model (Sanchez and Sali, 1997a,b; John and Sali, 2003).

Incorrect templates (Fig. 5.6.12E). This is a potential problem when distantly related proteins are used as templates (i.e., <25% sequence identity). Distinguishing between a model based on an incorrect template and a model based on an incorrect alignment with a correct template is difficult. In both cases, the evaluation methods will predict an unreliable model. The conservation of the key functional or structural residues in the target sequence increases the confidence in a given fold assignment.

Predicting the model accuracy

The accuracy of the predicted model determines the information that can be extracted from it. Thus, estimating the accuracy of a model in the absence of the known structure is essential for interpreting it.

Initial assessment of the fold. As discussed earlier, a model calculated using a template structure that shares more than 30% sequence identity is indicative of an overall accurate structure. However, when the sequence identity is lower, the first aspect of model evaluation is to confirm whether or not a correct template was used for modeling. It is often the case, when operating in this regime, that the fold-assignment step produces only false positives. A further complication is that at such low similarities the alignment generally contains many errors, making it difficult to distinguish between an incorrect template on one hand and an incorrect alignment with a correct template on the other hand. There are several methods that use 3-D profiles and statistical potentials (Sippl, 1990; Luthy et al., 1992; Melo et al., 2002) to assess the compatibility between the sequence and modeled structure by evaluating the environment of each residue in a model with respect to the expected environment as found in native high-resolution experimental structures. These methods can be used to assess whether or not the correct

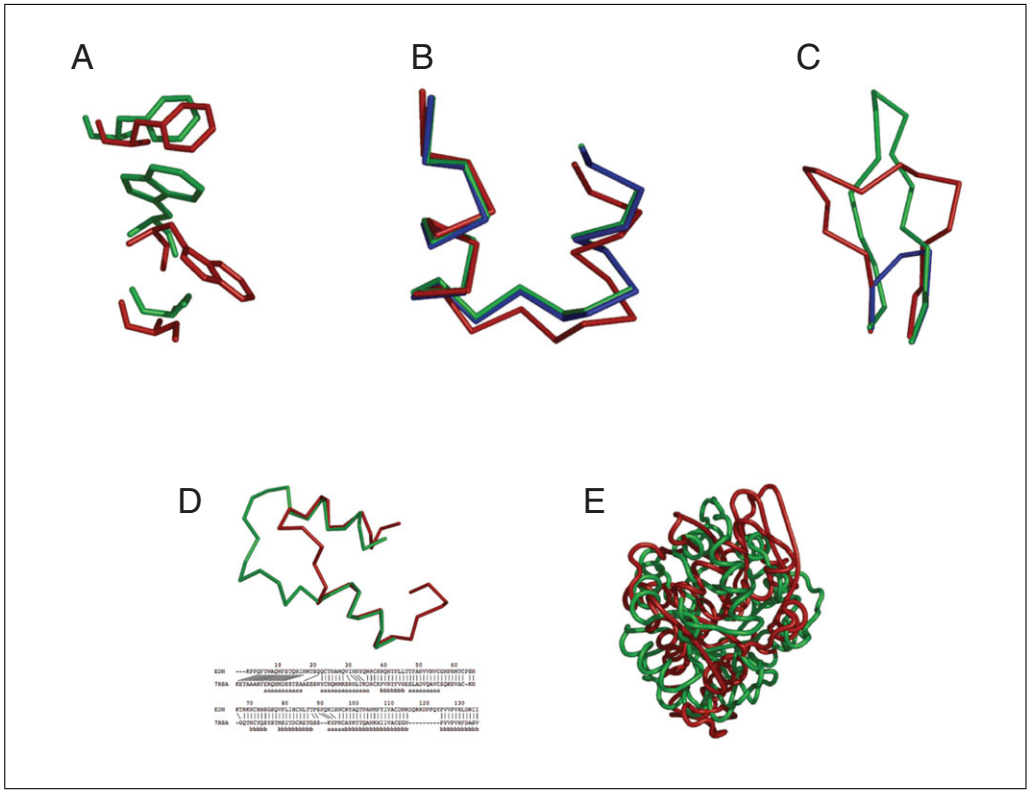

Figure 5.6.12 Typical errors in comparative modeling. (**A**) Errors in side-chain packing. The Trp 109 residue in the crystal structure of mouse cellular retinoic acid binding protein I (red) is compared with its model (green). (**B**) Distortions and shifts in correctly aligned regions. A region in the crystal structure of mouse cellular retinoic acid binding protein I (red) is compared with its model (green) and with the template fatty acid binding protein (blue). (**C**) Errors in regions without a template. The C^{α} trace of the 112–117 loop is shown for the X-ray structure of human eosinophil neurotoxin (red), its model (green), and the template ribonuclease A structure (residues 111–117; blue). (**D**) Errors due to misalignments. The N-terminal region in the crystal structure of human eosinophil neurotoxin (red) is compared with its model (green). The corresponding region of the alignment with the template ribonuclease A is shown. The red lines show correct equivalences, that is, residues whose C^{α} atoms are within 5 Å of each other in the optimal least-squares superposition of the two X-ray structures. The "a" characters in the bottom line indicate helical residues and "b" characters, the residues in sheets. (**E**) Errors due to an incorrect template. The X-ray structure of α-trichosanthin (red) is compared with its model (green) that was calculated using indole-3-glycerophosphate synthase as the template.

template was used for the modeling. They include VERIFY3D (Luthy et al., 1992), Prosa2003 (Sippl, 1993; Wiederstein and Sippl, 2007), HARMONY (Topham et al., 1994), ANOLEA (Melo and Feytmans, 1998), DFIRE (Zhou and Zhou, 2002), DOPE (Shen and Sali, 2006), SOAP (Dong et al., 2013), QMEAN local (Benkert et al., 2011), and TSV-Mod (Eramian et al., 2008).

Even when the model is based on alignments that have >30% sequence identity, other factors, including the environment, can strongly influence the accuracy of a model. For instance, some calcium-binding proteins undergo large conformational changes when bound to calcium. If a calcium-free template is used to model the calcium-bound state of the target, it is likely that the model will be incorrect irrespective of the target-template similarity or accuracy of the template structure (Pawlowski et al., 1996).

Evaluations of self-consistency. The model should also be subjected to evaluations of self-consistency to ensure that it satisfies the restraints used to calculate it. Additionally, the stereochemistry of the model (e.g., bond-lengths, bond-angles, backbone torsion angles, and nonbonded contacts) may be evaluated using programs such as PROCHECK (Laskowski et al., 1993) and WHATCHECK (Hooft et al., 1996). Although errors in stereochemistry are rare and less informative than errors detected by statistical potentials, a cluster of stereochemical errors may indicate that

there are larger errors (e.g., alignment errors) in that region.

Applications

Comparative modeling is often an efficient way to obtain useful information about the protein of interest. For example, comparative models can be helpful in designing mutants to test hypotheses about the protein's function (Wu et al., 1999; Vernal et al., 2002); in identifying active and binding sites (Sheng et al., 1996); in searching for, designing, and improving ligand binding strength for a given binding site (Ring et al., 1993; Li et al., 1996; Selzer et al., 1997; Enyedy et al., 2001; Que et al., 2002); in modeling substrate specificity (Xu et al., 1996); in predicting antigenic epitopes (Sali and Blundell, 1993); in simulating protein-protein docking (Vakser, 1995); in inferring function from calculated electrostatic potential around the protein (Matsumoto et al., 1995); in facilitating molecular replacement in X-ray structure determination (Howell et al., 1992); in refining models based on NMR constraints (Modi et al., 1996); in testing and improving a sequence-structure alignment (Wolf et al., 1998); in annotating single nucleotide polymorphisms (Mirkovic et al., 2004; Karchin et al., 2005); in structural characterization of large complexes by docking to low-resolution cryo-electron density maps (Spahn et al., 2001; Gao et al., 2003); and in rationalizing known experimental observations.

Fortunately, a 3-D model does not have to be absolutely perfect to be helpful in biology, as demonstrated by the applications listed above. The type of a question that can be addressed with a particular model does depend on its accuracy (Fig. 5.6.13).

At the low end of the accuracy spectrum, there are models that are based on less than 25% sequence identity, and that sometimes have less than 50% of their C^{α} atoms within 3.5 Å of their correct positions. However, such models still have the correct fold, and even knowing only the fold of a protein may sometimes be sufficient to predict its approximate biochemical function. Models in this low range of accuracy, combined with model evaluation, can be used for confirming or rejecting a match between remotely related proteins (Sanchez and Sali, 1997a; 1998).

In the middle of the accuracy spectrum are the models based on approximately 35% sequence identity, corresponding to 85% of the C^{α} atoms modeled within 3.5 Å of their correct positions. Fortunately, the active and binding sites are frequently more conserved than the rest of the fold, and are thus modeled more accurately (Sanchez and Sali, 1998). In general, medium-resolution models frequently allow a refinement of the functional prediction based on sequence alone, because ligand binding is most directly determined by the structure of the binding site rather than its sequence. It is frequently possible to correctly predict important features of the target protein that do not occur in the template structure. For example, the location of a binding site can be predicted from clusters of charged residues (Matsumoto et al., 1995), and the size of a ligand may be predicted from the volume of the binding-site cleft (Xu et al., 1996). Medium-resolution models can also be used to construct site-directed mutants with altered or destroyed binding capacity, which in turn could test hypotheses about the sequence-structure-function relationships. Other problems that can be addressed with medium-resolution comparative models include designing proteins that have compact structures, without long tails, loops, and exposed hydrophobic residues, for better crystallization, or designing proteins with added disulfide bonds for extra stability.

The high end of the accuracy spectrum corresponds to models based on 50% sequence identity or more. The average accuracy of these models approaches that of low-resolution X-ray structures (3 Å resolution) or medium-resolution NMR structures (10 distance restraints per residue; Sanchez and Sali, 1997b). The alignments on which these models are based generally contain almost no errors. Models with such high accuracy have been shown to be useful even for refining crystallographic structures by the method of molecular replacement (Howell et al., 1992; Baker and Sali, 2001; Jones, 2001; Claude et al., 2004; Schwarzenbacher et al., 2004).

Conclusion

Over the past few years, there has been a gradual increase in both the accuracy of comparative models and the fraction of protein sequences that can be modeled with useful accuracy (Marti-Renom et al., 2000; Baker and Sali, 2001; Pieper et al., 2011). The magnitude of errors in fold assignment, alignment, and modeling of side-chains and loops have decreased considerably. These improvements are a consequence both of better techniques and a larger number of known protein sequences and structures. Nevertheless, all the errors remain

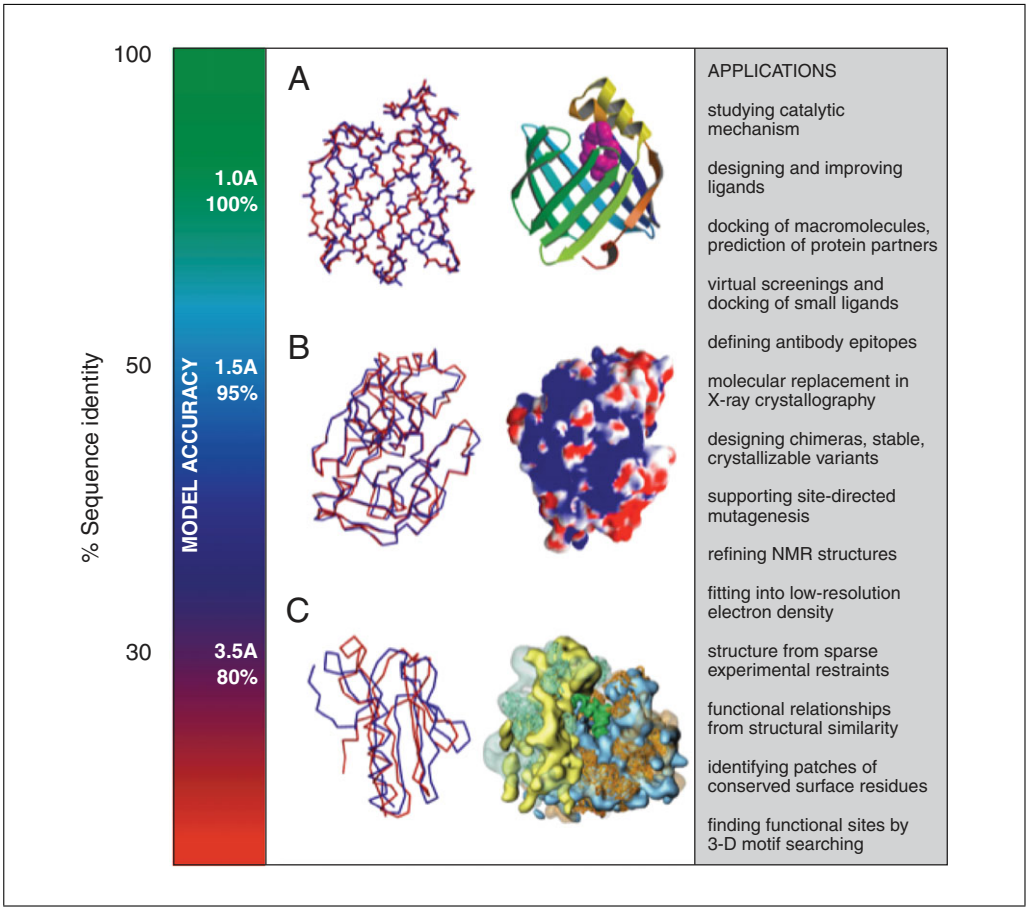

Figure 5.6.13 Accuracy and application of protein structure models. The vertical axis indicates the different ranges of applicability of comparative protein structure modeling, the corresponding accuracy of protein structure models, and their sample applications. (**A**) The docosahexaenoic fatty acid ligand (violet) was docked into a high-accuracy comparative model of brain lipid-binding protein (right), modeled based on its 62% sequence identity to the crystallographic structure of adipocyte lipid-binding protein (PDB code *1adl*). A number of fatty acids were ranked for their affinity to brain lipid-binding protein consistently with site-directed mutagenesis and affinity chromatography experiments (Xu et al., 1996), even though the ligand specificity profile of this protein is different from that of the template structure. Typical overall accuracy of a comparative model in this range of sequence similarity is indicated by a comparison of a model for adipocyte fatty acid binding protein with its actual structure (left). (**B**) A putative proteoglycan binding patch was identified on a medium-accuracy comparative model of mouse mast cell protease 7 (right), modeled based on its 39% sequence identity to the crystallographic structure of bovine pancreatic trypsin (*2ptn*) that does not bind proteoglycans. The prediction was confirmed by site-directed mutagenesis and heparin-affinity chromatography experiments (Matsumoto et al., 1995). Typical accuracy of a comparative model in this range of sequence similarity is indicated by a comparison of a trypsin model with the actual structure. (**C**) A molecular model of the whole yeast ribosome (right) was calculated by fitting atomic rRNA and protein models into the electron density of the 80S ribosomal particle, obtained by electron microscopy at 15 Å resolution (Spahn et al., 2001). Most of the models for 40 out of the 75 ribosomal proteins were based on template structures that were approximately 30% sequence-identical. Typical accuracy of a comparative model in this range of sequence similarity is indicated by a comparison of a model for a domain in L2 protein from *B. stearothermophilus* with the actual structure (*1rl2*).

significant and demand future methodological improvements. In addition, there is a great need for more accurate modeling of distortions and rigid-body shifts, as well as detection of errors in a given protein structure model. Error detection is useful both for refinement and interpretation of the models.

Acknowledgments

The authors wish to express gratitude to all members of their research group. This review is partially based on the authors' previous reviews (Marti-Renom et al., 2000; Eswar et al., 2003; Fiser and Sali, 2003a; also see previous version of this unit

at *http://onlinelibrary.wiley.com/doi/10.1002/0471250953.bi0506s15/full*). The authors wish to acknowledge funding from Sandler Family Supporting Foundation, NIH R01 GM54762, P01 GM71790, P01 A135707, and U54 GM62529, as well as hardware gifts from IBM and Intel.

Literature Cited

Abagyan, R. and Totrov, M. 1994. Biased probability Monte Carlo conformational searches and electrostatic calculations for peptides and proteins. *J. Mol. Biol.* 235:983-1002.

Altschul, S.F., Madden, T.L., Schaffer, A.A., Zhang, J., Zhang, Z., Miller, W., and Lipman, D.J. 1997. Gapped BLAST and PSI-BLAST: A new generation of protein database search programs. *Nucleic Acids Res.* 25:3389-3402.

Andreeva, A., Howorth, D., Brenner, S.E., Hubbard, T.J., Chothia, C., and Murzin, A.G. 2004. SCOP database in 2004: Refinements integrate structure and sequence family data. *Nucleic Acids Res.* 32:D226-D229.

Armougom, F., Moretti, S., Poirot, O., Audic, S., Dumas, P., Schaeli, B., Keduas, V., and Notredame, C. 2006. Expresso: Automatic incorporation of structural information in multiple sequence alignments using 3D-Coffee. *Nucleic Acids Res.* 34:W604-W608.

Arnold, K., Bordoli, L., Kopp, J., and Schwede, T. 2006. The SWISS-MODEL workspace: A web-based environment for protein structure homology modelling. *Bioinformatics* 22:195-201.

Arnold, K., Kiefer, F., Kopp, J., Battey, J.N., Podvinec, M., Westbrook, J.D., Berman, H.M., Bordoli, L., and Schwede, T. 2009. The Protein Model Portal. *J. Struct. Funct. Genomics* 10:1-8.

Aszodi, A. and Taylor, W.R. 1994. Secondary structure formation in model polypeptide chains. *Protein Eng.* 7:633-644.

Attwood, T.K., Coletta, A., Muirhead, G., Pavlopoulou, A., Philippou, P.B., Popov, I., Roma-Mateo, C., Theodosiou, A., and Mitchell, A.L. 2012. The PRINTS database: A fine-grained protein sequence annotation and analysis resource—its status in 2012. *Database (Oxford)* 2012:bas019.

Bailey, T.L. and Elkan, C. 1994. Fitting a mixture model by expectation maximization to discover motifs in biopolymers. *Proc. Int. Conf. Intell. Syst. Mol. Biol.* 2:28-36.

Bairoch, A., Apweiler, R., Wu, C.H., Barker, W.C., Boeckmann, B., Ferro, S., Gasteiger, E., Huang, H., Lopez, R., Magrane, M., Martin, M.J., Natale, D.A., O'Donovan, C., Redaschi, N., and Yeh, L.S. 2005. The Universal Protein Resource (UniProt). *Nucleic Acids Res.* 33:D154-D159.

Baker, D. and Sali, A. 2001. Protein structure prediction and structural genomics. *Science* 294:93-96.

Barton, G.J. and Sternberg, M.J. 1987. A strategy for the rapid multiple alignment of protein sequences. Confidence levels from tertiary structure comparisons. *J. Mol. Biol.* 198:327-337.

Bateman, A., Coin, L., Durbin, R., Finn, R.D., Hollich, V., Griffiths-Jones, S., Khanna, A., Marshall, M., Moxon, S., Sonnhammer, E.L., Studholme, D.J., Yeats, C., and Eddy, S.R. 2004. The Pfam protein families database. *Nucleic Acids Res.* 32:D138-D141.

Bates, P.A., Kelley, L.A., MacCallum, R.M., and Sternberg, M.J. 2001. Enhancement of protein modeling by human intervention in applying the automatic programs 3D-JIGSAW and 3D-PSSM. *Proteins* 5:39-46.

Benkert, P., Biasini, M., and Schwede, T. 2011. Toward the estimation of the absolute quality of individual protein structure models. *Bioinformatics* 27:343-350.

Benson, D. A., Cavanaugh, M., Clark, K., Karsch-Mizrachi, I., Lipman, D.J., Ostell, J., and Sayers, E.W. 2013. GenBank. *Nucleic Acids Res.* 41:D36-D42.

Berman, H.M., Westbrook, J., Feng, Z., Gilliland, G., Bhat, T.N., Weissig, H., Shindyalov, I.N., and Bourne, P.E. 2000. The Protein Data Bank. *Nucleic Acids Res.* 28:235-242.

Blundell, T.L., Sibanda, B.L., Sternberg, M.J., and Thornton, J.M. 1987. Knowledge-based prediction of protein structures and the design of novel molecules. *Nature* 326:347-352.

Boissel, J.P., Lee, W.R., Presnell, S.R., Cohen, F.E., and Bunn, H.F. 1993. Erythropoietin structure-function relationships. Mutant proteins that test a model of tertiary structure. *J. Biol. Chem.* 268:15983-15993.

Bowie, J.U., Luthy, R., and Eisenberg, D. 1991. A method to identify protein sequences that fold into a known three-dimensional structure. *Science* 253:164-170.

Braun, W. and Go, N. 1985. Calculation of protein conformations by proton-proton distance constraints. A new efficient algorithm. *J. Mol. Biol.* 186:611-626.

Brenner, S.E., Chothia, C., and Hubbard, T.J. 1998. Assessing sequence comparison methods with reliable structurally identified distant evolutionary relationships. *Proc. Natl. Acad. Sci. U.S.A.* 95:6073-6078.

Brown, S.D., and Babbitt, P.C. 2012. Inference of functional properties from large-scale analysis of enzyme superfamilies. *J. Biol. Chem.* 287:35-42.

Browne, W.J., North, A.C., Phillips, D.C., Brew, K., Vanaman, T.C., and Hill, R.L. 1969. A possible three-dimensional structure of bovine alpha-lactalbumin based on that of hen's egg-white lysozyme. *J. Mol. Biol.* 42:65-86.

Bru, C., Courcelle, E., Carrere, S., Beausse, Y., Dalmar, S., and Kahn, D. 2005. The ProDom database of protein domain families: More emphasis on 3D. *Nucleic Acids Res.* 33:D212-D215.

Bruccoleri, R.E. and Karplus, M. 1987. Prediction of the folding of short polypeptide segments by

uniform conformational sampling. *Biopolymers* 26:137-168.

Bruccoleri, R.E. and Karplus, M. 1990. Conformational sampling using high-temperature molecular dynamics. *Biopolymers* 29:1847-1862.

Bystroff, C. and Baker, D. 1998. Prediction of local structure in proteins using a library of sequence-structure motifs. *J. Mol. Biol.* 281:565-577.

Chinea, G., Padron, G., Hooft, R.W., Sander, C., and Vriend, G. 1995. The use of position-specific rotamers in model building by homology. *Proteins* 23:415-421.

Chothia, C. and Lesk, A.M. 1987. Canonical structures for the hypervariable regions of immunoglobulins. *J. Mol. Biol.* 196:901-917.

Chothia, C., Lesk, A.M., Tramontano, A., Levitt, M., Smith-Gill, S.J., Air, G., Sheriff, S., Padlan, E.A., Davies, D., Tulip, W.R., et al. 1989. Conformations of immunoglobulin hypervariable regions. *Nature* 342:877-883.

Claessens, M., Van Cutsem, E., Lasters, I., and Wodak, S. 1989. Modelling the polypeptide backbone with 'spare parts' from known protein structures. *Protein Eng.* 2:335-345.

Claude, J.B., Suhre, K., Notredame, C., Claverie, J.M., and Abergel, C. 2004. CaspR: A web server for automated molecular replacement using homology modelling. *Nucleic Acids Res.* 32:W606-W609.

Clore, G.M., Brunger, A.T., Karplus, M., and Gronenborn, A.M. 1986. Application of molecular dynamics with interproton distance restraints to three-dimensional protein structure determination. A model study of crambin. *J. Mol. Biol.* 191:523-551.

Cohen, F.E., Gregoret, L., Presnell, S.R., and Kuntz, I.D. 1989. Protein structure predictions: New theoretical approaches. *Prog. Clin. Biol. Res.* 289:75-85.

Collura, V., Higo, J., and Garnier, J. 1993. Modeling of protein loops by simulated annealing. *Protein Sci.* 2:1502-1510.

Colovos, C. and Yeates, T.O. 1993. Verification of protein structures: patterns of nonbonded atomic interactions. *Protein Sci.* 2:1511-1519.

Deane, C.M. and Blundell, T.L. 2001. CODA: A combined algorithm for predicting the structurally variable regions of protein models. *Protein Sci.* 10:599-612.

de Bakker, P.I., DePristo, M.A., Burke, D.F., and Blundell, T.L. 2003. Ab initio construction of polypeptide fragments: Accuracy of loop decoy discrimination by an all-atom statistical potential and the AMBER force field with the Generalized Born solvation model. *Proteins* 51:21-40.

DePristo, M.A., de Bakker, P.I., Lovell, S.C., and Blundell, T.L. 2003. Ab initio construction of polypeptide fragments: Efficient generation of accurate, representative ensembles. *Proteins* 51:41-55.

Dietmann, S., Park, J., Notredame, C., Heger, A., Lappe, M., and Holm, L. 2001. A fully automatic evolutionary classification of protein folds: Dali Domain Dictionary version 3. *Nucleic Acids Res.* 29:55-57.

Dong, G.Q., Fan, H., Schneidman-Duhovny, D., Webb, B., and Sali, A. 2013. Optimized atomic statistical potentials: Assessment of protein interfaces and loops. *Bioinformatics* 29:3158-3166

Dror, O., Benyamini, H., Nussinov, R., and Wolfson, H. 2003. MASS: Multiple structural alignment by secondary structures. *Bioinformatics* 19:i95-i104.

Eddy, S.R. 1998. Profile hidden Markov models. *Bioinformatics* 14:755-763.

Edgar, R.C. 2004. MUSCLE: Multiple sequence alignment with high accuracy and high throughput. *Nucleic Acids Res.* 32:1792-1797.

Edgar, R.C. and Sjolander, K. 2004. A comparison of scoring functions for protein sequence profile alignment. *Bioinformatics* 20:1301-1308.

Enyedy, I.J., Lee, S.L., Kuo, A.H., Dickson, R.B., Lin, C.Y., and Wang, S. 2001. Structure-based approach for the discovery of bis-benzamidines as novel inhibitors of matriptase. *J. Med. Chem.* 44:1349-1355.

Eramian, D., Eswar, N., Shen, M., and Sali, A. 2008. How well can the accuracy of comparative protein structure models be predicted? *Protein Sci.* 17:1881-1893.

Eswar, N., John, B., Mirkovic, N., Fiser, A., Ilyin, V.A., Pieper, U., Stuart, A.C., Marti-Renom, M.A., Madhusudhan, M.S., Yerkovich, B., and Sali, A. 2003. Tools for comparative protein structure modeling and analysis. *Nucleic Acids Res.* 31:3375-3380.

Felsenstein, J. 1989. PHYLIP—Phylogeny Inference Package (Version 3.2). *Cladistics* 5:164-166.

Felts, A.K., Gallicchio, E., Wallqvist, A., and Levy, R.M. 2002. Distinguishing native conformations of proteins from decoys with an effective free energy estimator based on the OPLS all-atom force field and the Surface Generalized Born solvent model. *Proteins* 48:404-422.

Fernandez-Fuentes, N. and Fiser, A. 2006. Saturating representation of loop conformational fragments in structure databanks. *BMC Struct. Biol.* 6:15.

Fernandez-Fuentes, N., Rai, B.K., Madrid-Aliste, C.J., Fajardo, J.E., and Fiser, A. 2007. Comparative protein structure modeling by combining multiple templates and optimizing sequence-to-structure alignments. *Bioinformatics* 23:2558-2565.

Fidelis, K., Stern, P.S., Bacon, D., and Moult, J. 1994. Comparison of systematic search and database methods for constructing segments of protein structure. *Protein Eng.* 7:953-960.

Fine, R.M., Wang, H., Shenkin, P.S., Yarmush, D.L., and Levinthal, C. 1986. Predicting antibody hypervariable loop conformations. II: Minimization and molecular dynamics studies of MCPC603 from many randomly generated loop conformations. *Proteins* 1:342-362.

Fiser, A. 2004. Protein structure modeling in the proteomics era. *Expert Rev. Proteomics* 1:97-110.

Fiser, A. and Sali, A. 2003a. Modeller: Generation and refinement of homology-based protein structure models. *Methods Enzymol.* 374:461-491.

Fiser, A. and Sali, A. 2003b. ModLoop: Automated modeling of loops in protein structures. *Bioinformatics* 19:2500-2501.

Fiser, A., Do, R.K.G., and Sali, A. 2000. Modeling of loops in protein structures. *Protein Sci.* 9:1753-1773.

Fiser, A., Feig, M., Brooks, C.L., and Sali, A. 2002. Evolution and physics in comparative protein structure modeling. *Acc. Chem. Res.* 35:413-421.

Flicek, P., Ahmed, I., Amode, M.R., Barrell, D., Beal, K., Brent, S., Carvalho-Silva, D., Clapham, P., Coates, G., Fairley, S., Fitzgerald, S., Gil, L., Garcia-Giron, C., Gordon, L., Hourlier, T., Hunt, S., Juettemann, T., Kahari, A. K., Keenan, S., Komorowska, M., Kulesha, E., Longden, I., Maurel, T., McLaren, W.M., Muffato, M., Nag, R., Overduin, B., Pignatelli, M., Pritchard, B., Pritchard, E., Riat, H.S., Ritchie, G.R., Ruffier, M., Schuster, M., Sheppard, D., Sobral, D., Taylor, K., Thormann, A., Trevanion, S., White, S., Wilder, S.P., Aken, B.L., Birney, E., Cunningham, F., Dunham, I., Harrow, J., Herrero, J., Hubbard, T.J., Johnson, N., Kinsella, R., Parker, A., Spudich, G., Yates, A., Zadissa, A., and Searle, S.M. 2013. Ensembl 2013. *Nucleic Acids Res.* 41:D48-D55.

Gao, H.X., Sengupta, J., Valle, M., Korostelev, A., Eswar, N., Stagg, S.M., Van Roey, P., Agrawal, R.K., Harvey, S.C., Sali, A., Chapman, M.S., and Frank, J. 2003. Study of the structural dynamics of the E-coli 70S ribosome using real-space refinement. *Cell* 113:789-801.

Godzik, A. 2003. Fold recognition methods. *Methods Biochem. Anal.* 44:525-546.

Gough, J., Karplus, K., Hughey, R., and Chothia, C. 2001. 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.

Greer, J. 1981. Comparative model-building of the mammalian serine proteases. *J. Mol. Biol.* 153:1027-1042.

Gribskov, M., McLachlan, A.D., and Eisenberg, D. 1987. Profile analysis: Detection of distantly related proteins. *Proc. Natl. Acad. Sci. U.S.A.* 84:4355-4358.

Guerler, A. and Knapp, E.W. 2008. Novel protein folds and their nonsequential structural analogs. *Protein Sci.* 17:1374-1382.

Haas, J., Roth, S., Arnold, K., Kiefer, F., Schmidt, T., Bordoli, L., and Schwede, T. 2013. The Protein Model Portal—a comprehensive resource for protein structure and model information. *Database (Oxford)* 2013:bat031.

Havel, T.F. and Snow, M.E. 1991. A new method for building protein conformations from sequence alignments with homologues of known structure. *J. Mol. Biol.* 217:1-7.

Henikoff, S. and Henikoff, J.G. 1994. Position-based sequence weights. *J. Mol. Biol.* 243:574-578.

Higo, J., Collura, V., and Garnier, J. 1992. Development of an extended simulated annealing method: Application to the modeling of complementary determining regions of immunoglobulins. *Biopolymers* 32:33-43.

Holm, L. and Sander, C. 1991. Database algorithm for generating protein backbone and side-chain co-ordinates from a C alpha trace application to model building and detection of co-ordinate errors. *J. Mol. Biol.* 218:183-194.

Hooft, R.W., Vriend, G., Sander, C., and Abola, E.E. 1996. Errors in protein structures. *Nature* 381:272.

Howell, P.L., Almo, S.C., Parsons, M.R., Hajdu, J., and Petsko, G.A. 1992. Structure determination of turkey egg-white lysozyme using Laue diffraction data. *Acta Crystallogr. B* 48:200-207.

Huang, C.C., Novak, W.R., Babbitt, P.C., Jewett, A.I., Ferrin, T.E., and Klein, T.E. 2000. Integrated tools for structural and sequence alignment and analysis. *Pac. Symp. Biocomput.* 2000:230-241.

Huang, H., Hu, Z.Z., Arighi, C.N., and Wu, C.H. 2007. Integration of bioinformatics resources for functional analysis of gene expression and proteomic data. *Front. Biosci.* 12:5071-5088.

Hulo, N., Bairoch, A., Bulliard, V., Cerutti, L., De Castro, E., Langendijk-Genevaux, P.S., Pagni, M., and Sigrist, C.J. 2006. The PROSITE database. *Nucleic Acids Res.* 34:D227-D230.

Hunter, S., Jones, P., Mitchell, A., Apweiler, R., Attwood, T.K., Bateman, A., Bernard, T., Binns, D., Bork, P., Burge, S., de Castro, E., Coggill, P., Corbett, M., Das, U., Daugherty, L., Duquenne, L., Finn, R.D., Fraser, M., Gough, J., Haft, D., Hulo, N., Kahn, D., Kelly, E., Letunic, I., Lonsdale, D., Lopez, R., Madera, M., Maslen, J., McAnulla, C., McDowall, J., McMenamin, C., Mi, H., Mutowo-Muellenet, P., Mulder, N., Natale, D., Orengo, C., Pesseat, S., Punta, M., Quinn, A.F., Rivoire, C., Sangrador-Vegas, A., Selengut, J.D., Sigrist, C.J., Scheremetjew, M., Tate, J., Thimmajanarthanan, M., Thomas, P.D., Wu, C.H., Yeats, C., and Yong, S.Y. 2012. InterPro in 2011: New developments in the family and domain prediction database. *Nucleic Acids Res.* 40:D306-D312.

Jacobson, M.P., Pincus, D.L., Rapp, C.S., Day, T.J., Honig, B., Shaw, D.E., and Friesner, R.A. 2004. A hierarchical approach to all-atom protein loop prediction. *Proteins* 55:351-367.

Jaroszewski, L., Rychlewski, L., Li, Z., Li, W., and Godzik, A. 2005. FFAS03: A server for profile–profile sequence alignments. *Nucleic Acids Res.* 33:W284-W288.

John, B. and Sali, A. 2003. Comparative protein structure modeling by iterative alignment,

model building and model assessment. *Nucleic Acids Res.* 31:3982-3992.

Jones, D.T. 1999. GenTHREADER: An efficient and reliable protein fold recognition method for genomic sequences. *J. Mol. Biol.* 287:797-815.

Jones, D.T. 2001. Evaluating the potential of using fold-recognition models for molecular replacement. *Acta Crystallogr. D Biol. Crystallogr.* 57:1428-1434.

Jones, T.A. and Thirup, S. 1986. Using known substructures in protein model building and crystallography. *EMBO J.* 5:819-822.

Kabsch, W. and Sander, C. 1984. On the use of sequence homologies to predict protein structure: Identical pentapeptides can have completely different conformations. *Proc. Natl. Acad. Sci. U.S.A.* 81:1075-1078.

Kallberg, M., Wang, H., Wang, S., Peng, J., Wang, Z., Lu, H., and Xu, J. 2012. Template-based protein structure modeling using the RaptorX web server. *Nat. Protoc.* 7:1511-1522.

Kaplan, W. and Littlejohn, T.G. 2001. Swiss-PDB Viewer (Deep View). *Brief. Bioinform.* 2:195-197.

Karchin, R., Cline, M., Mandel-Gutfreund, Y., and Karplus, K. 2003. Hidden Markov models that use predicted local structure for fold recognition: Alphabets of backbone geometry. *Proteins* 51:504-514.

Karchin, R., Diekhans, M., Kelly, L., Thomas, D.J., Pieper, U., Eswar, N., Haussler, D., and Sali, A. 2005. LS-SNP: Large-scale annotation of coding non-synonymous SNPs based on multiple information sources. *Bioinformatics* 21:2814-2820.

Karplus, K. 2009. SAM-T08, HMM-based protein structure prediction. *Nucleic Acids Res.* 37:W492-W497.

Karplus, K., Barrett, C., and Hughey, R. 1998. Hidden Markov models for detecting remote protein homologies. *Bioinformatics* 14:846-856.

Karplus, K., Karchin, R., Draper, J., Casper, J., Mandel-Gutfreund, Y., Diekhans, M., and Hughey, R. 2003. Combining local-structure, fold-recognition, and new fold methods for protein structure prediction. *Proteins* 6:491-496.

Katoh, K. and Standley, D.M. 2013. MAFFT multiple sequence alignment software version 7: Improvements in performance and usability. *Mol. Biol. Evol.* 30:772-780.

Kelley, L.A. and Sternberg, M.J. 2009. Protein structure prediction on the Web: A case study using the Phyre server. *Nat. Protoc.* 4:363-371.

Kelley, L.A., MacCallum, R.M., and Sternberg, M.J. 2000. Enhanced genome annotation using structural profiles in the program 3D-PSSM. *J. Mol. Biol.* 299:499-520.

Khafizov, K., Staritzbichler, R., Stamm, M., and Forrest, L.R. 2010. A study of the evolution of inverted-topology repeats from LeuT-fold transporters using AlignMe. *Biochemistry (Mosc).* 49:10702-10713.

Kiefer, F., Arnold, K., Kunzli, M., Bordoli, L., and Schwede, T. 2009. The SWISS-MODEL Repository and associated resources. *Nucleic Acids Res.* 37:D387-D392.

Koehl, P. and Delarue, M. 1995. A self consistent mean field approach to simultaneous gap closure and side-chain positioning in homology modelling. *Nat. Struct. Biol.* 2:163-170.

Konagurthu, A.S., Whisstock, J.C., Stuckey, P.J., and Lesk, A.M. 2006. MUSTANG: A multiple structural alignment algorithm. *Proteins* 64:559-574.

Krivov, G.G., Shapovalov, M.V., and Dunbrack, R.L. Jr. 2009. Improved prediction of protein side-chain conformations with SCWRL4. *Proteins* 77:778-795.

Krogh, A., Brown, M., Mian, I.S., Sjolander, K., and Haussler, D. 1994. Hidden Markov models in computational biology. Applications to protein modeling. *J. Mol. Biol.* 235:1501-1531.

Laskowski, R., MacArthur, M., Moss, D., and Thornton, J. 1993. PROCHECK-a program to check the stereochemical quality of protein structures. *J. Appl. Cryst.* 26:283-291.

Lessel, U. and Schomburg, D. 1994. Similarities between protein 3-D structures. *Protein Eng.* 7:1175-1187.

Letunic, I., Doerks, T., and Bork, P. 2012. SMART 7: Recent updates to the protein domain annotation resource. *Nucleic Acids Res.* 40:D302-D305.

Levitt, M. 1992. Accurate modeling of protein conformation by automatic segment matching. *J. Mol. Biol.* 226:507-533.

Li, R., Chen, X., Gong, B., Selzer, P.M., Li, Z., Davidson, E., Kurzban, G., Miller, R.E., Nuzum, E.O., McKerrow, J.H., Fletterick, R.J., Gillmor, S.A., Craik, C.S., Kuntz, I.D., Cohen, F.E., and Kenyon, G.L. 1996. Structure-based design of parasitic protease inhibitors. *Biorg. Med. Chem.* 4:1421-1427.

Lin, J., Qian, J., Greenbaum, D., Bertone, P., Das, R., Echols, N., Senes, A., Stenger, B., and Gerstein, M. 2002. GeneCensus: Genome comparisons in terms of metabolic pathway activity and protein family sharing. *Nucleic Acids Res.* 30:4574-4582.

Lindahl, E. and Elofsson, A. 2000. Identification of related proteins on family, superfamily and fold level. *J. Mol. Biol.* 295:613-625.

Lupyan, D., Leo-Macias, A., and Ortiz, A.R. 2005. A new progressive-iterative algorithm for multiple structure alignment. *Bioinformatics* 21:3255-3263.

Luthy, R., Bowie, J.U., and Eisenberg, D. 1992. Assessment of protein models with three-dimensional profiles. *Nature* 356:83-85.

MacKerell, J.A.D., Bashford, D., Bellott, M., Dunbrack, R.L. Jr., Evanseck, J.D., Field, M.J., Fischer, S., Gao, J., Guo, H., Ha, S., Joseph-McCarthy, D., Kuchnir, L., Kuczera, K., Lau, F.T.K., Mattos, C., Michnick, S., Ngo, T., Nguyen, D.T., Prodhom, B., Reiher, I.W.E.,

Roux, B., Schlenkrich, M., Smith, J.C., Stote, R., Straub, J., Watanabe, M., Wiorkiewicz-Kuczera, J., Yin, D., and Karplus, M. 1998. All-atom empirical potential for molecular modeling and dynamics Studies of proteins. *J. Phys. Chem. B* 102:3586-3616.

Madhusudhan, M.S., Marti-Renom, M.A., Sanchez, R., and Sali, A. 2006. Variable gap penalty for protein sequence-structure alignment. *Protein Eng. Des. Sel.* 19:129-133.

Madhusudhan, M.S., Webb, B.M., Marti-Renom, M.A., Eswar, N., and Sali, A. 2009. Alignment of multiple protein structures based on sequence and structure features. *Protein Eng. Des. Sel.* 22:569-574.

Mandell, D.J., Coutsias, E.A., and Kortemme, T. 2009. Sub-angstrom accuracy in protein loop reconstruction by robotics-inspired conformational sampling. *Nat. Methods* 6:551-552.

Marti-Renom, M.A., Stuart, A.C., Fiser, A., Sanchez, R., Melo, F., and Sali, A. 2000. Comparative protein structure modeling of genes and genomes. *Annu. Rev. Biophys. Biomol. Struct.* 29:291-325.

Marti-Renom, M.A., Ilyin, V.A., and Sali, A. 2001. DBAli: A database of protein structure alignments. *Bioinformatics* 17:746-747.

Marti-Renom, M.A., Madhusudhan, M.S., Fiser, A., Rost, B., and Sali, A. 2002. Reliability of assessment of protein structure prediction methods. *Structure* 10:435-440.

Marti-Renom, M.A., Madhusudhan, M.S., and Sali, A. 2004. Alignment of protein sequences by their profiles. *Protein Sci.* 13:1071-1087.

Matsumoto, R., Sali, A., Ghildyal, N., Karplus, M., and Stevens, R.L. 1995. Packaging of proteases and proteoglycans in the granules of mast cells and other hematopoietic cells. A cluster of histidines on mouse mast cell protease 7 regulates its binding to heparin serglycin proteoglycans. *J. Biol. Chem.* 270:19524-19531.

McGuffin, L.J. and Jones, D.T. 2003. Improvement of the GenTHREADER method for genomic fold recognition. *Bioinformatics* 19:874-881.

McGuffin, L.J., Bryson, K., and Jones, D.T. 2000. The PSIPRED protein structure prediction server. *Bioinformatics* 16:404-405.

Melo, F. and Feytmans, E. 1998. Assessing protein structures with a non-local atomic interaction energy. *J. Mol. Biol.* 277:1141-1152.

Melo, F., Sanchez, R., and Sali, A. 2002. Statistical potentials for fold assessment. *Protein Sci.* 11:430-448.

Mezei, M. 1998. Chameleon sequences in the PDB. *Protein Eng.* 11:411-414.

Mirkovic, N., Marti-Renom, M.A., Weber, B.L., Sali, A., and Monteiro, A.N. 2004. Structure-based assessment of missense mutations in human BRCA1: implications for breast and ovarian cancer predisposition. *Cancer Res.* 64:3790-3797.

Misura, K.M. and Baker, D. 2005. Progress and challenges in high-resolution refinement of protein structure models. *Proteins* 59:15-29.

Misura, K.M., Chivian, D., Rohl, C.A., Kim, D.E., and Baker, D. 2006. Physically realistic homology models built with ROSETTA can be more accurate than their templates. *Proc. Natl. Acad. Sci. U.S.A.* 103:5361-5366.

Miwa, J.M., Ibanez-Tallon, I., Crabtree, G.W., Sanchez, R., Sali, A., Role, L.W., and Heintz, N. 1999. lynx1, an endogenous toxin-like modulator of nicotinic acetylcholine receptors in the mammalian CNS. *Neuron* 23:105-114.

Modi, S., Paine, M.J., Sutcliffe, M.J., Lian, L.Y., Primrose, W.U., Wolf, C.R., and Roberts, G.C. 1996. A model for human cytochrome P450 2D6 based on homology modeling and NMR studies of substrate binding. *Biochemistry (Mosc).* 35:4540-4550.

Moult, J. 2005. A decade of CASP: Progress, bottlenecks and prognosis in protein structure prediction. *Curr. Opin. Struct. Biol.* 15:285-289.

Moult, J. and James, M.N. 1986. An algorithm for determining the conformation of polypeptide segments in proteins by systematic search. *Proteins* 1:146-163.

Moult, J., Fidelis, K., Zemla, A., and Hubbard, T. 2003. Critical assessment of methods of protein structure prediction (CASP)-round V. *Proteins* 6:334-339.

Moult, J., Fidelis, K., Kryshtafovych, A., Rost, B., and Tramontano, A. 2009. Critical assessment of methods of protein structure prediction - Round VIII. *Proteins* 77:1-4.

Nagarajaram, H.A., Reddy, B.V., and Blundell, T.L. 1999. Analysis and prediction of interstrand packing distances between beta-sheets of globular proteins. *Protein Eng.* 12:1055-1062.

Needleman, S.B. and Wunsch, C.D. 1970. A general method applicable to the search for similarities in the amino acid sequence of two proteins. *J. Mol. Biol.* 48:443-453.

Notredame, C., Higgins, D.G., and Heringa, J. 2000. T-Coffee: A novel method for fast and accurate multiple sequence alignment. *J. Mol. Biol.* 302:205-217.

Ohlson, T., Wallner, B., and Elofsson, A. 2004. Profile-profile methods provide improved fold-recognition: A study of different profile-profile alignment methods. *Proteins* 57:188-197.

Oliva, B., Bates, P.A., Querol, E., Aviles, F.X., and Sternberg, M.J. 1997. An automated classification of the structure of protein loops. *J. Mol. Biol.* 266:814-830.

Ortiz, A.R., Strauss, C.E., and Olmea, O. 2002. MAMMOTH (matching molecular models obtained from theory): An automated method for model comparison. *Protein Sci.* 11:2606-2621.

Panchenko, A.R. 2003. Finding weak similarities between proteins by sequence profile comparison. *Nucleic Acids Res.* 31:683-689.

Park, J., Karplus, K., Barrett, C., Hughey, R., Haussler, D., Hubbard, T., and Chothia, C. 1998. Sequence comparisons using multiple sequences detect three times as many remote homologues as pairwise methods. *J. Mol. Biol.* 284:1201-1210.

Pawlowski, K., Bierzynski, A., and Godzik, A. 1996. Structural diversity in a family of homologous proteins. *J. Mol. Biol.* 258:349-366.

Pearl, F., Todd, A., Sillitoe, I., Dibley, M., Redfern, O., Lewis, T., Bennett, C., Marsden, R., Grant, A., Lee, D., Akpor, A., Maibaum, M., Harrison, A., Dallman, T., Reeves, G., Diboun, I., Addou, S., Lise, S., Johnston, C., Sillero, A., Thornton, J., and Orengo, C. 2005. The CATH domain structure database and related resources Gene3D and DHS provide comprehensive domain family information for genome analysis. *Nucleic Acids Res.* 33:D247-D251.

Pearson, W.R. 1994. Using the FASTA program to search protein and DNA sequence databases. *Methods Mol. Biol.* 24:307-331.

Pearson, W.R. 2000. Flexible sequence similarity searching with the FASTA3 program package. *Methods Mol. Biol.* 132:185-219.

Pei, J., Kim, B.H., and Grishin, N.V. 2008. PROMALS3D: A tool for multiple protein sequence and structure alignments. *Nucleic Acids Res.* 36:2295-2300.

Petrey, D. and Honig, B. 2005. Protein structure prediction: Inroads to biology. *Mol. Cell* 20:811-819.

Pieper, U., Webb, B.M., Barkan, D.T., Schneidman-Duhovny, D., Schlessinger, A., Braberg, H., Yang, Z., Meng, E.C., Pettersen, E.F., Huang, C.C., Datta, R.S., Sampathkumar, P., Madhusudhan, M.S., Sjolander, K., Ferrin, T.E., Burley, S.K., and Sali, A. 2011. ModBase, a database of annotated comparative protein structure models, and associated resources. *Nucleic Acids Res.* 39:465-474.

Pietrokovski, S. 1996. Searching databases of conserved sequence regions by aligning protein multiple-alignments. *Nucleic Acids Res.* 24:3836-3845.

Prlic, A., Bliven, S., Rose, P.W., Bluhm, W.F., Bizon, C., Godzik, A., and Bourne, P.E. 2010. Precalculated protein structure alignments at the RCSB PDB website. *Bioinformatics* 26:2983-2985.

Que, X., Brinen, L.S., Perkins, P., Herdman, S., Hirata, K., Torian, B.E., Rubin, H., McKerrow, J.H., and Reed, S.L. 2002. Cysteine proteinases from distinct cellular compartments are recruited to phagocytic vesicles by *Entamoeba histolytica*. *Mol. Biochem. Parasitol.* 119:23-32.

Ray, A., Lindahl, E., and Wallner, B. 2012. Improved model quality assessment using ProQ2. *BMC Bioinformatics* 13:224.

Remmert, M., Biegert, A., Hauser, A., and Soding, J. 2012. HHblits: Lightning-fast iterative protein sequence searching by HMM-HMM alignment. *Nat. Methods* 9:173-175.

Ring, C.S., Kneller, D.G., Langridge, R., and Cohen, F.E. 1992. Taxonomy and conformational analysis of loops in proteins. *J. Mol. Biol.* 224:685-699.

Ring, C.S., Sun, E., McKerrow, J.H., Lee, G.K., Rosenthal, P.J., Kuntz, I.D., and Cohen, F.E. 1993. Structure-based inhibitor design by using protein models for the development of antiparasitic agents. *Proc. Natl. Acad. Sci. U.S.A.* 90:3583-3587.

Roche, D.B., Buenavista, M.T., Tetchner, S.J., and McGuffin, L.J. 2011. The IntFOLD server: An integrated web resource for protein fold recognition, 3D model quality assessment, intrinsic disorder prediction, domain prediction and ligand binding site prediction. *Nucleic Acids Res.* 39:W171-W176.

Rost, B. 1999. Twilight zone of protein sequence alignments. *Protein Eng.* 12:85-94.

Roy, A., Kucukural, A., and Zhang, Y. 2010. I-TASSER: A unified platform for automated protein structure and function prediction. *Nat. Protoc.* 5:725-738.

Rufino, S.D., Donate, L.E., Canard, L.H., and Blundell, T.L. 1997. Predicting the conformational class of short and medium size loops connecting regular secondary structures: Application to comparative modelling. *J. Mol. Biol.* 267:352-367.

Rychlewski, L., Zhang, B., and Godzik, A. 1998. Fold and function predictions for *Mycoplasma genitalium* proteins. *Fold. Des.* 3:229-238.

Sadreyev, R. and Grishin, N. 2003. COMPASS: A tool for comparison of multiple protein alignments with assessment of statistical significance. *J. Mol. Biol.* 326:317-336.

Sali, A. and Blundell, T.L. 1993. Comparative protein modelling by satisfaction of spatial restraints. *J. Mol. Biol.* 234:779-815.

Sali, A. and Overington, J.P. 1994. Derivation of rules for comparative protein modeling from a database of protein structure alignments. *Protein Sci.* 3:1582-1596.

Samudrala, R. and Moult, J. 1998. A graph-theoretic algorithm for comparative modeling of protein structure. *J. Mol. Biol.* 279:287-302.

Sanchez, R. and Sali, A. 1997a. Advances in comparative protein-structure modelling. *Curr. Opin. Struct. Biol.* 7:206-214.

Sanchez, R. and Sali, A. 1997b. Evaluation of comparative protein structure modeling by MODELLER-3. *Proteins* 1:50-58.

Sanchez, R. and Sali, A. 1998. Large-scale protein structure modeling of the *Saccharomyces cerevisiae* genome. *Proc. Natl. Acad. Sci. U.S.A.* 95:13597-13602.

Saqi, M.A., Russell, R.B., and Sternberg, M.J. 1998. Misleading local sequence alignments: Implications for comparative protein modelling. *Protein Eng.* 11:627-630.

Sauder, J.M., Arthur, J.W., and Dunbrack, R.L. Jr. 2000. Large-scale comparison of protein

sequence alignment algorithms with structure alignments. *Proteins* 40:6-22.

Schwarzenbacher, R., Godzik, A., Grzechnik, S.K., and Jaroszewski, L. 2004. The importance of alignment accuracy for molecular replacement. *Acta Crystallogr. D Biol. Crystallogr.* 60:1229-1236.

Schwede, T., Kopp, J., Guex, N., and Peitsch, M.C. 2003. SWISS-MODEL: An automated protein homology-modeling server. *Nucleic Acids Res.* 31:3381-3385.

Sellers, B.D., Zhu, K., Zhao, S., Friesner, R.A., and Jacobson, M.P. 2008. Toward better refinement of comparative models: predicting loops in inexact environments. *Proteins* 72:959-971.

Selzer, P.M., Chen, X., Chan, V.J., Cheng, M., Kenyon, G.L., Kuntz, I.D., Sakanari, J.A., Cohen, F.E., and McKerrow, J.H. 1997. Leishmania major: Molecular modeling of cysteine proteases and prediction of new nonpeptide inhibitors. *Exp. Parasitol.* 87:212-221.

Shatsky, M., Nussinov, R., and Wolfson, H.J. 2004. A method for simultaneous alignment of multiple protein structures. *Proteins* 56:143-156.

Shatsky, M., Nussinov, R., and Wolfson, H.J. 2006. Optimization of multiple-sequence alignment based on multiple-structure alignment. *Proteins* 62:209-217.

Shen, M.Y. and Sali, A. 2006. Statistical potential for assessment and prediction of protein structures. *Protein Sci.* 15:2507-2524.

Sheng, Y., Sali, A., Herzog, H., Lahnstein, J., and Krilis, S.A. 1996. Site-directed mutagenesis of recombinant human beta 2-glycoprotein I identifies a cluster of lysine residues that are critical for phospholipid binding and anti-cardiolipin antibody activity. *J. Immunol.* 157:3744-3751.

Shenkin, P.S., Yarmush, D.L., Fine, R.M., Wang, H.J., and Levinthal, C. 1987. Predicting antibody hypervariable loop conformation. I. Ensembles of random conformations for ringlike structures. *Biopolymers* 26:2053-2085.

Shi, J., Blundell, T.L., and Mizuguchi, K. 2001. FUGUE: Sequence-structure homology recognition using environment-specific substitution tables and structure-dependent gap penalties. *J. Mol. Biol.* 310:243-257.

Sibanda, B.L., Blundell, T.L., and Thornton, J. M. 1989. Conformation of beta-hairpins in protein structures. A systematic classification with applications to modelling by homology, electron density fitting and protein engineering. *J. Mol. Biol.* 206:759-777.

Sippl, M.J. 1993. Recognition of errors in three-dimensional structures of proteins. *Proteins* 17:355-362.

Sippl, M.J. 1990. Calculation of conformational ensembles from potentials of mean force. An approach to the knowledge-based prediction of local structures in globular proteins. *J. Mol. Biol.* 213:859-883.

Sippl, M.J. 1995. Knowledge-based potentials for proteins. *Curr. Opin. Struct. Biol.* 5:229-235.

Skolnick, J. and Kihara, D. 2001. Defrosting the frozen approximation: PROSPECTOR—a new approach to threading. *Proteins* 42:319-331.

Smith, T.F. and Waterman, M.S. 1981. Identification of common molecular subsequences. *J. Mol. Biol.* 147:195-197.

Soding, J. 2005. Protein homology detection by HMM-HMM comparison. *Bioinformatics* 21:951-960.

Soding, J., Biegert, A., and Lupas, A.N. 2005. The HHpred interactive server for protein homology detection and structure prediction. *Nucleic Acids Res.* 33:W244-W248.

Song, Y., Dimaio, F., Wang, R.Y., Kim, D., Miles, C., Brunette, T., Thompson, J., and Baker, D. 2013. High-resolution comparative modeling with RosettaCM. *Structure* 21:1735-1742.

Spahn, C.M., Beckmann, R., Eswar, N., Penczek, P.A., Sali, A., Blobel, G., and Frank, J. 2001. Structure of the 80S ribosome from *Saccharomyces cerevisiae*—tRNA-ribosome and subunit-subunit interactions. *Cell* 107:373-386.

Srinivasan, N. and Blundell, T.L. 1993. An evaluation of the performance of an automated procedure for comparative modelling of protein tertiary structure. *Protein Eng.* 6:501-512.

Sutcliffe, M.J., Haneef, I., Carney, D., and Blundell, T.L. 1987a. Knowledge based modelling of homologous proteins, Part I: Three-dimensional frameworks derived from the simultaneous superposition of multiple structures. *Protein Eng.* 1:377-384.

Sutcliffe, M.J., Hayes, F.R., and Blundell, T.L. 1987b. Knowledge based modelling of homologous proteins, Part II: Rules for the conformations of substituted sidechains. *Protein Eng.* 1:385-392.

Sutcliffe, M.J., Dobson, C.M., and Oswald, R.E. 1992. Solution structure of neuronal bungarotoxin determined by two-dimensional NMR spectroscopy: Calculation of tertiary structure using systematic homologous model building, dynamical simulated annealing, and restrained molecular dynamics. *Biochemistry (Mosc).* 31:2962-2970.

Taylor, W.R., Flores, T.P., and Orengo, C.A. 1994. Multiple protein structure alignment. *Protein Sci.* 3:1858-1870.

Thompson, J.D., Higgins, D.G., and Gibson, T.J. 1994. CLUSTAL W: Improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice. *Nucleic Acids Res.* 22:4673-4680.

Topham, C.M., McLeod, A., Eisenmenger, F., Overington, J.P., Johnson, M.S., and Blundell, T.L. 1993. Fragment ranking in modelling of protein structure. Conformationally constrained environmental amino acid substitution tables. *J. Mol. Biol.* 229:194-220.

Topham, C.M., Srinivasan, N., Thorpe, C.J., Overington, J.P., and Kalsheker, N.A. 1994. Comparative modelling of major house dust mite allergen Der p I: Structure validation using an

extended environmental amino acid propensity table. *Protein Eng.* 7:869-894.

Unger, R., Harel, D., Wherland, S., and Sussman, J.L. 1989. A 3D building blocks approach to analyzing and predicting structure of proteins. *Proteins* 5:355-373.

Vakser, I.A. 1995. Protein docking for low-resolution structures. *Protein Eng.* 8:371-377.

van Gelder, C.W., Leusen, F.J., Leunissen, J.A., and Noordik, J.H. 1994. A molecular dynamics approach for the generation of complete protein structures from limited coordinate data. *Proteins* 18:174-185.

van Vlijmen, H.W. and Karplus, M. 1997. PDB-based protein loop prediction: Parameters for selection and methods for optimization. *J. Mol. Biol.* 267:975-1001.

Vernal, J., Fiser, A., Sali, A., Muller, M., Cazzulo, J.J., and Nowicki, C. 2002. Probing the specificity of a trypanosomal aromatic alpha-hydroxy acid dehydrogenase by site-directed mutagenesis. *Biochem. Biophys. Res. Commun.* 293:633-639.

von Ohsen, N., Sommer, I., and Zimmer, R. 2003. Profile-profile alignment: A powerful tool for protein structure prediction. *Pac. Symp. Biocomput.* 252-263.

Wang, G. and Dunbrack, R.L. Jr. 2004. Scoring profile-to-profile sequence alignments. *Protein Sci.* 13:1612-1626.

Wang, Q., Canutescu, A.A., and Dunbrack, R.L. Jr. 2008. SCWRL and MolIDE: Computer programs for side-chain conformation prediction and homology modeling. *Nat. Protoc.* 3:1832-1847.

Wiederstein, M. and Sippl, M.J. 2007. ProSA-web: Interactive web service for the recognition of errors in three-dimensional structures of proteins. *Nucleic Acids Res.* 35:W407-W410.

Wolf, E., Vassilev, A., Makino, Y., Sali, A., Nakatani, Y., and Burley, S.K. 1998. Crystal structure of a GCN5-related N-acetyltransferase: *Serratia marcescens* aminoglycoside 3-N-acetyltransferase. *Cell* 94:439-449.

Wu, G., Fiser, A., ter Kuile, B., Sali, A., and Muller, M. 1999. Convergent evolution of *Trichomonas vaginalis* lactate dehydrogenase from malate dehydrogenase. *Proc. Natl. Acad. Sci. U.S.A.* 96:6285-6290.

Wu, S. and Zhang, Y. 2008. MUSTER: Improving protein sequence profile-profile alignments by using multiple sources of structure information. *Proteins* 72:547-556.

Xiang, Z., Soto, C.S., and Honig, B. 2002. Evaluating conformational free energies: The colony energy and its application to the problem of loop prediction. *Proc. Natl. Acad. Sci. U.S.A.* 99:7432-7437.

Xu, J., Li, M., Kim, D., and Xu, Y. 2003. RAPTOR: Optimal protein threading by linear programming. *J. Bioinf. Comput. Biol.* 1:95-117.

Xu, L.Z., Sanchez, R., Sali, A., and Heintz, N. 1996. Ligand specificity of brain lipid-binding protein. *J. Biol. Chem.* 271:24711-24719.

Yona, G. and Levitt, M. 2002. Within the twilight zone: A sensitive profile-profile comparison tool based on information theory. *J. Mol. Biol.* 315:1257-1275.

Zhang, Y. and Skolnick, J. 2005. TM-align: A protein structure alignment algorithm based on the TM-score. *Nucleic Acids Res.* 33:2302-2309.

Zheng, Q., Rosenfeld, R., Vajda, S., and DeLisi, C. 1993. Determining protein loop conformation using scaling-relaxation techniques. *Protein Sci.* 2:1242-1248.

Zhou, H. and Zhou, Y. 2002. Distance-scaled, finite ideal-gas reference state improves structure-derived potentials of mean force for structure selection and stability prediction. *Protein Sci.* 11:2714-2726.

Zhou, H. and Zhou, Y. 2005. Fold recognition by combining sequence profiles derived from evolution and from depth-dependent structural alignment of fragments. *Proteins* 58:321-328.

Zhu, K., Pincus, D.L., Zhao, S., and Friesner, R.A. 2006. Long loop prediction using the protein local optimization program. *Proteins* 65:438-452.

Internet Resources

http://salilab.org/modeller/

B. Webb, M.S. Madhusudhan, M-Y. Shen, G.Q. Dong, M.A. Marti-Renom, N. Eswar, F. Alber, M. Topf, B. Oliva, A. Fiser, R. Sanchez, B. Yerkovich, A. Badretdinov, F. Melo, J.P. Overington, E. Feyfant and A. Sali. 2014. MODELLER, A Protein Structure Modeling Program, Release 9.13.