

---

## HOMOLOGY MODELING

---

Elmar Krieger, Sander B. Nabuurs, and Gert Vriend

The ultimate goal of protein modeling is to predict a structure from its sequence with an accuracy that is comparable to the best results achieved experimentally. This would allow users to safely use rapidly generated *in silico* protein models in all the contexts where today only experimental structures provide a solid basis: structure-based drug design, analysis of protein function, interactions, antigenic behavior, and rational design of proteins with increased stability or novel functions. In addition, protein modeling is the only way to obtain structural information if experimental techniques fail. Many proteins are simply too large for NMR analysis and cannot be crystallized for X-ray diffraction.

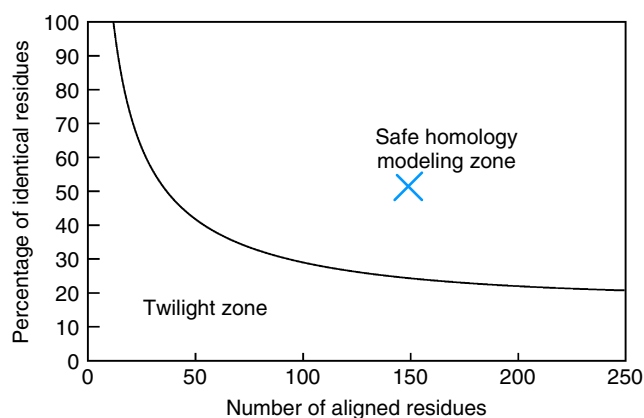
Among the three major approaches to three-dimensional (3D) structure prediction described in this and the following two chapters, homology modeling is the easiest one. It is based on two major observations:

1. The structure of a protein is uniquely determined by its amino acid sequence (Epstein, Goldberger, and Anfinsen, 1963). Knowing the sequence should, at least in theory, suffice to obtain the structure.
2. During evolution, the structure is more stable and changes much slower than the associated sequence, so that similar sequences adopt practically identical structures, and distantly related sequences still fold into similar structures. This relationship was first identified by Chothia and Lesk (1986) and later quantified by Sander and Schneider (1991). Thanks to the exponential growth of the Protein Data Bank (PDB), Rost (1999) could recently derive a precise limit for this rule, shown in Figure 25.1. As long as the length of two sequences and the percentage of identical residues fall in the region marked as “safe,” the two sequences are practically guaranteed to adopt a similar structure.

*Structural Bioinformatics*

Edited by Philip E. Bourne and Helge Weissig

ISBN 0-471-20199-5 Copyright © 2003 by Wiley-Liss, Inc.

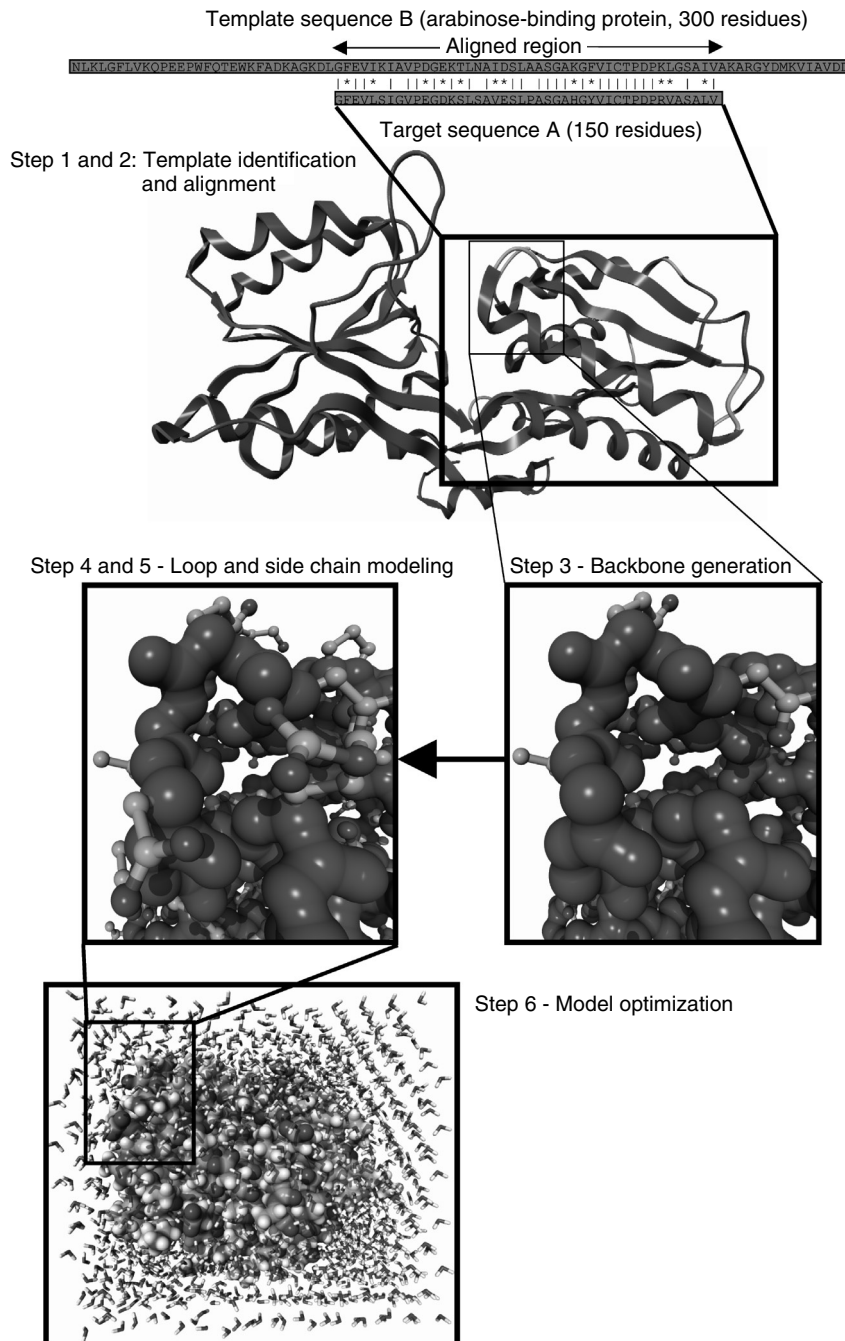


**Figure 25.1.** The two zones of sequence alignments. Two sequences are practically guaranteed to fold into the same structure if their length and percentage sequence identity fall into the region marked as “safe.” An example of two sequences with 150 amino acids, 50% of which are identical, is shown (gray cross).

Imagine that we want to know the structure of sequence A (150 amino acids long, Figure 25.2, steps 1 and 2). We compare sequence A to all the sequences of known structures stored in the PDB (using, for example, BLAST), and luckily find a sequence B (300 amino acids long) containing a region of 150 amino acids that match sequence A with 50% identical residues. As this match (alignment) clearly falls in the safe zone (Fig. 25.1), we can simply take the known structure of sequence B (the template), cut out the fragment corresponding to the aligned region, mutate those amino acids that differ between sequences A and B, and finally arrive at our model for structure A. Structure A is called the *target* and is of course not known at the time of modeling. In practice, homology modeling is a multistep process that can be summarized in seven steps:

1. Template recognition and initial alignment
2. Alignment correction
3. Backbone generation
4. Loop modeling
5. Side-chain modeling
6. Model optimization
7. Model validation

At almost all the steps choices have to be made. The modeler can never be sure to make the best ones, and thus a large part of the modeling process consists of serious thought about how to gamble between multiple seemingly similar choices. A lot of research has been spent on teaching the computer how to make these decisions, so that homology models can be built fully automatically. Currently, this allows modelers to construct models for about 25% of the amino acids in a genome, thereby supplementing the efforts of structural genomics projects (Sanchez and Sali, 1999, Peitsch, Schwede, and Guex, 2000). This average value of 25% differs significantly



**Figure 25.2.** The steps to homology modeling. The fragment of the template (arabinose-binding protein) corresponding to the region aligned with the target sequence forms the basis of the model (including conserved side chains). Loops and missing side chains are predicted, then the model is optimized (in this case together with surrounding water molecules). Images created with Yasara ([www.yasara.com](http://www.yasara.com)).

between individual genomes, ranging from 16% (*Mycoplasma pneumoniae*) to 30% (*Haemophilus influenzae*) and increasing steadily thanks to the continuous growth of the PDB. For the remaining  $\sim 75\%$  of a genome, no template with a known structure is available (or cannot be detected with a simple BLAST run), and one must use fold recognition (Chapter 26), *ab initio* folding techniques (Chapter 27), or simply an experiment to obtain structural data (Chapters 4, 5, and 6). While automated model building provides high throughput, the evaluation of these methods during CASP (Chapter 24) indicated that human expertise is still helpful, especially if the alignment is close to the twilight zone (Fischer et al., 1999).

## THE SEVEN STEPS TO HOMOLOGY MODELING

### Step 1: Template Recognition and Initial Alignment

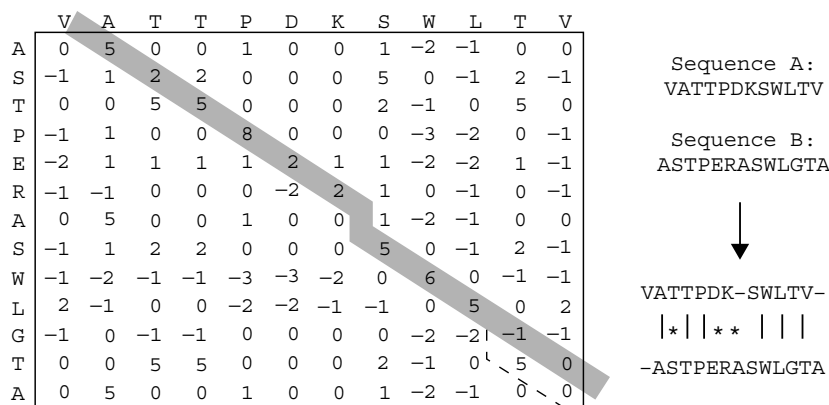
In the safe homology modeling zone (Fig. 25.1), the percentage identity between the sequence of interest and a possible template is high enough to be detected with simple sequence alignment programs such as BLAST (Altschul et al., 1990) or FASTA (Pearson, 1990).

To identify these hits, the program compares the query sequence to all the sequences of known structures in the PDB using mainly two matrices:

1. A residue exchange matrix (Fig. 25.3). The elements of this  $20 \times 20$  matrix define the likelihood that any two of the 20 amino acids ought to be aligned. It is clearly seen that the values along the diagonal (representing conserved residues) are highest, but one can also observe that exchanges between residue types with similar physicochemical properties (for example  $F \rightarrow Y$ ) get a better score than exchanges between residue types that widely differ in their properties.
2. An alignment matrix (Fig. 25.4). The axes of this matrix correspond to the two sequences to align, and the matrix elements are simply the values from the

	A	C	D	E	F	G	H	I	K	L	M	N	P	Q	R	S	T	V	W	Y
A	5	-2	0	1	-2	0	0	-1	0	-1	0	0	1	0	-1	1	0	0	-2	-2
C	-2	8	-2	-3	-3	-2	0	-2	-3	-3	0	-2	-3	-3	-2	-1	-1	-2	-1	-2
D	0	-2	5	2	-2	0	1	-3	0	-2	-1	2	0	1	-2	0	0	-2	-3	-2
E	1	-3	2	5	-3	0	-1	-2	1	-2	-2	1	1	2	0	1	1	-1	-2	-1
F	-2	-3	-2	-3	6	-3	1	0	-3	2	2	-3	-2	-3	-2	-1	-2	0	3	3
G	0	-2	0	0	-3	5	-1	-2	0	-2	-2	0	0	-1	0	0	-1	-1	-2	-3
H	0	0	1	-1	1	-1	5	-1	1	-1	0	1	0	1	2	0	1	-1	0	1
I	-1	-2	-3	-2	0	-2	-1	5	-2	2	2	-2	-2	-3	-2	-1	0	2	0	0
K	0	-3	0	1	-3	0	1	-2	5	-1	-2	1	0	1	2	0	0	-1	-2	-2
L	-1	-3	-2	-2	2	-2	-1	2	-1	5	3	-2	-2	0	-1	-1	0	2	0	0
M	0	0	-1	-2	2	-2	0	2	-2	3	5	-1	-2	0	-2	-1	0	1	-2	-1
N	0	-2	2	1	-3	0	1	-2	1	-2	-1	5	-2	1	0	2	0	-2	-3	-1
P	1	-3	0	1	-2	0	0	-2	0	-2	-2	-2	8	0	0	0	0	-1	-3	-3
Q	0	-3	1	2	-3	-1	1	-3	1	0	0	1	0	5	2	1	0	-1	-1	-2
R	-1	-2	-2	0	-2	0	2	-2	2	-1	-2	0	0	2	5	1	0	-1	0	-1
S	1	-1	0	1	-1	0	0	-1	0	-1	-1	2	0	1	1	5	2	-1	0	0
T	0	-1	0	1	-2	-1	1	0	0	0	0	0	0	0	0	2	5	0	-1	-2
V	0	-2	-2	-1	0	-1	-1	2	-1	2	1	-2	-1	-1	-1	-1	0	5	-1	0
W	-2	-1	-3	-2	3	-2	0	0	-2	0	-2	-3	-3	-1	0	0	-1	-1	6	3
Y	-2	-2	-2	-1	3	-3	1	0	-2	0	-1	-1	-3	-2	-1	0	-2	0	3	6

**Figure 25.3.** A typical residue exchange or scoring matrix used by alignment algorithms. Because the score for aligning residues A and B is normally the same as for B and A, this matrix is symmetric.



**Figure 25.4.** The alignment matrix for the sequences VATTPDKSWLTV and ASTPERASWLGTA, using the scores from Figure 25.3. The optimum path corresponding to the alignment on the right side is shown in gray. Residues with similar properties are marked with a star (\*). The dashed line marks an alternative alignment that scores more points but requires opening a second gap.

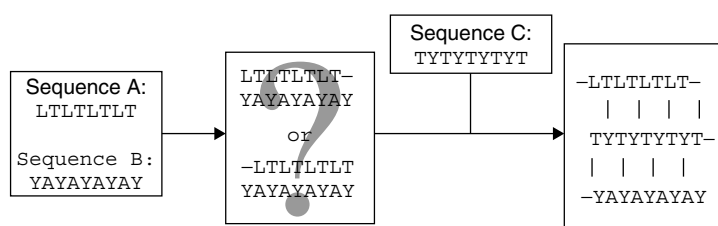
residue exchange matrix (Fig. 25.3) for a given pair of residues. During the alignment process, one tries to find the best path through this matrix, starting from a point near the top left, and going down to the bottom right. To make sure that no residue is used twice, one must always take at least one step to the right and one step down. A typical alignment path is shown in Figure 25.4. At first sight, the dashed path in the bottom right corner would have led to a higher score. However, it requires the opening of an additional gap in sequence A (Gly of sequence B is skipped). By comparing thousands of sequences and sequence families, it became clear that the opening of gaps is about as unlikely as at least a couple of nonidentical residues in a row. The jump roughly in the middle of the matrix, however, is justified, because after the jump we earn lots of points (5,6,5), which would have been (1,0,0) without the jump. The alignment algorithm therefore subtracts an “opening penalty” for every new gap and a much smaller “gap extension penalty” for every residue that is skipped in the alignment. The gap extension penalty is smaller simply because one gap of three residues is much more likely than three gaps of one residue each.

In practice, one just feeds the query sequence to one of the countless BLAST servers on the web, selects a search of the PDB, and obtains a list of hits—the modeling templates and corresponding alignments (Fig. 25.2).

## Step 2: Alignment Correction

Having identified one or more possible modeling templates using the fast methods described above, it is time to consider more sophisticated methods to arrive at a better alignment.

Sometimes it may be difficult to align two sequences in a region where the percentage sequence identity is very low. One can then use other sequences from homologous



**Figure 25.5.** A pathological alignment problem. Sequences A and B are impossible to align, unless one considers a third sequence C from a homologous protein.

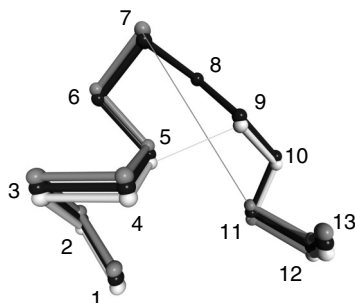
proteins to find a solution. A pathological example is shown in Figure 25.5: Suppose you want to align the sequence LTLTLTLT with YAYAYAYAY. There are two equally poor possibilities, and only a third sequence, TYTYTYTYT, that aligns easily to both of them can solve the issue.

The example above introduced a very powerful concept called “multiple sequence alignment.” Many programs are available to align a number of related sequences, for example CLUSTALW (Thompson, Higgins, and Gibson, 1994), and the resulting alignment contains a lot of additional information. Think about an Ala → Glu mutation. Relying on the matrix in Figure 25.3, this exchange always gets a score of 1. In the 3D structure of the protein, it is however very unlikely to see such an Ala → Glu exchange in the hydrophobic core, but on the surface this mutation is perfectly normal. The multiple sequence alignment implicitly contains information about this structural context. If at a certain position only exchanges between hydrophobic residues are observed, it is highly likely that this residue is buried. To consider this knowledge during the alignment, one uses the multiple sequence alignment to derive position-specific scoring matrices, also called *profiles* (Taylor, 1986, Dodge, Schneider, and Sander, 1998).

When building a homology model, we are in the fortunate situation of having an almost perfect profile—the known structure of the template. We simply know that a certain alanine sits in the protein core and must therefore not be aligned with a glutamate. Multiple sequence alignments are nevertheless useful in homology modeling, for example, to place deletions (missing residues in the model) or insertions (additional residues in the model) only in areas where the sequences are strongly divergent. A typical example for correcting an alignment with the help of the template is shown in Figures 25.6 and 25.7. Although a simple sequence alignment gives the highest score for the wrong answer (alignment 1 in Fig. 25.6), a simple look at the structure of the template reveals that alignment 2 is correct, because it leads to a small gap, compared to a huge hole associated with alignment 1.

		1	2	3	4	5	6	7	8	9	10	11	12	13
<b>Template</b>		PHE	ASP	ILE	CYS	ARG	LEU	PRO	GLY	SER	ALA	GLU	ALA	VAL
Model (bad) 1		PHE	ASN	VAL	CYS	ARG	ALA	PRO	---	---	---	GLU	ALA	ILE
Model (good) 2		PHE	ASN	VAL	CYS	ARG	---	---	---	ALA	PRO	GLU	ALA	ILE

**Figure 25.6.** Example of a sequence alignment where a three-residue deletion must be modeled. While the first alignment appears better when considering just the sequences (a matching proline at position 7), a look at the structure of the template leads to a different conclusion (Figure 25.7).



**Figure 25.7.** Correcting an alignment based on the structure of the modeling template ( $C\alpha$ -trace shown in black). While the alignment with the highest score (dark gray, also in Figure 25.6) leads to a gap of 7.5 Å between residues 7 and 11, the second option (white) creates only a tiny hole of 1.3 Å between residues 5 and 9. This can easily be accommodated by small backbone shifts. (The normal  $C\alpha-C\alpha$  distance of 3.8 Å has been subtracted).

### Step 3: Backbone Generation

When the alignment is ready, the actual model building can start. Creating the backbone is trivial for most of the model: One simply copies the coordinates of those template residues that show up in the alignment with the model sequence (Fig. 25.2). If two aligned residues differ, only the backbone coordinates (N,  $C\alpha$ , C and O) can be copied. If they are the same, one can also include the side chain (at least the more rigid side chains, since rotamers tend to be conserved).

Experimentally determined protein structures are not perfect (but still better than models in most cases). There are countless sources of errors, ranging from poor electron density in the X-ray diffraction map to simple human errors when preparing the PDB file for submission. A lot of work has been spent on writing software to detect these errors (correcting them is even more difficult), and the current count is at more than 10,000,000 problems in the 17,000 structures deposited in the PDB by the end of 2001. It is obvious that a straightforward way to build a good model is to choose the template with the fewest errors (the PDBREPORT database [Hoof et al., 1996] at [www.cmbi.nl/gv/pdbreport](http://www.cmbi.nl/gv/pdbreport) can be very helpful). But what if two templates are available, and each has a poorly determined region, but these regions are not the same? One should clearly combine the good parts of both templates in one model—an approach known as multiple template modeling. (The same applies if the alignments between the model sequence and possible templates show good matches in different regions). Although in principle multiple template modeling is simple (and done by automated modeling servers such as Swiss-Model [Peitsch, Schwede, and Guex, 2000]), it is difficult in practice to achieve results that are really closer to the true structure than all the templates. Nevertheless, it is possible, as has been shown by AndrejŠalis' group in CASP4 (see Chapter 24).

### Step 4: Loop Modeling

In the majority of cases, the alignment between model and template sequence contains gaps. Either gaps in the model sequence (deletions as shown in Figs. 25.6 and 25.7) or in the template sequence (insertions). In the first case, one simply omits residues

from the template, creating a hole in the model that must be closed. In the second case, one takes the continuous backbone from the template, cuts it, and inserts the missing residues. Both cases imply a conformational change of the backbone. The good news is that conformational changes cannot happen within regular secondary structure elements. It is therefore safe to shift all insertions or deletions in the alignment out of helices and strands, placing them in loops and turns. The bad news is that these changes in loop conformation are notoriously difficult to predict (the big unsolved problem in homology modeling). To make things worse, even without insertions or deletions we often find quite different loop conformations in template and target. Three main reasons can be identified (Rodriguez, <http://www.cmbi.kun.nl/gv/articles/text/gambling.html>):

1. Surface loops tend to be involved in crystal contacts, leading to a significant conformational change between template and target.
2. The exchange of small to bulky side chains underneath the loop pushes it aside.
3. The mutation of a loop residue to proline or from glycine to any other residue. In both cases, the new residue must fit into a more restricted area in the Ramachandran plot, which most of the time requires conformational changes of the loop.

There are two main approaches to loop modeling:

1. Knowledge based: one searches the PDB for known loops with endpoints that match the residues between which the loop has to be inserted, and simply copies the loop conformation. All major molecular modeling programs and servers support this approach (e.g., 3D-Jigsaw [Bates and Sternberg, 1999], Insight [Dayringer, Tramontano, and Fletterick, 1986], Modeller [Sali and Blundell, 1993], Swiss-Model [Peitsch, Schwede, and Guex, 2000], or WHAT IF [Vriend, 1990]).
2. Energy based: as in true *ab initio* fold prediction, an energy function is used to judge the quality of a loop. Then this function is minimized, using Monte Carlo (Simons et al., 1999) or molecular dynamics techniques (Fiser, Do, and Sali, 2000) to arrive at the best loop conformation. Often the energy function is modified (e.g., smoothed) to facilitate the search (Tappura, 2001).

At least for short loops (up to 5–8 residues), the various methods have a reasonable chance of predicting a loop conformation that superimposes well on the true structure. As mentioned above, surface loops tend to change their conformation due to crystal contacts. So if the prediction is made for an isolated protein and then found to differ from the crystal structure, it might still be correct.

### Step 5: Side-Chain Modeling

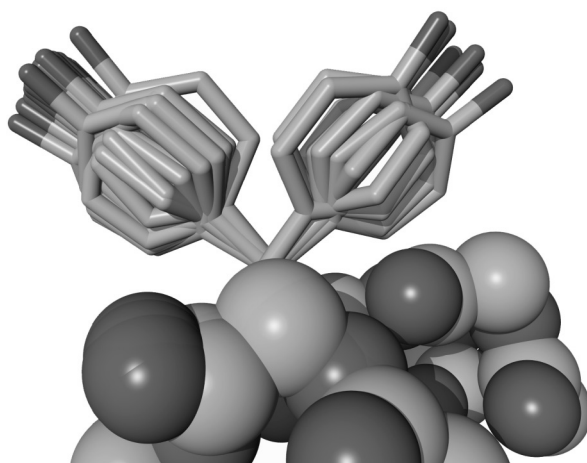
When we compare the side-chain conformations (rotamers) of residues that are conserved in structurally similar proteins, we find that they often have similar  $\chi_1$ -angles (i.e., the torsion angle about the  $C_\alpha-C_\beta$  bond). It is therefore possible to simply copy conserved residues entirely from the template to the model (see also Step 3) and achieve a higher accuracy than by copying just the backbone and repredicting the side chains. In practice, this rule of thumb holds only at high levels of sequence identity, when the conserved residues form networks of contacts. When they get isolated (<35%



sequence identity), the rotamers of conserved residues may differ in up to 45% of the cases (Sanchez and Sali, 1997).

Practically all successful approaches to side-chain placement are at least partly knowledge based. They use libraries of common rotamers extracted from high-resolution X-ray structures. The various rotamers are tried successively and scored with a variety of energy functions. Intuitively, one might expect rotamer prediction to be computationally demanding due to the combinatorial explosion—the choice of a certain rotamer automatically affects the rotamers of all neighboring residues, which in turn affect their neighbors and so on. With 100 residues and on average  $\sim 5$  rotamers per residue, one would already end up at  $5^{100}$  different combinations to score. A lot of research has been spent on the development of methods to make this enormous search space tractable (Desmet et al., 1992). The number of combinations is in fact so large, that even nature could not try all of them during the folding process, which indicates that there must exist mechanisms to shrink down the search space.

Beside the trivial fact that copying conserved rotamers from the template often splits up the protein into distinct regions where rotamers can be predicted independently, the key to handling the combinatorial explosion lies in the protein backbone. Certain backbone conformations strongly favor certain rotamers (allowing, for example, a hydrogen bond between side chain and backbone) and thus greatly reduce the search space. For a given backbone conformation, there may be only one strongly populated rotamer that can be modeled right away, thereby providing an anchor for surrounding, more flexible side chains. An example for a backbone conformation that favors two different tyrosine rotamers is shown in Figure 25.8. These position-specific rotamer libraries are widely used today (de Filippis, Sander, and Vriend, 1994, Stites, Meeker, and Shortle, 1994, Dunbrack and Karplus, 1994). To build such a library, one takes high-resolution structures and collects all stretches of three to seven residues (depending on the method) with a given amino acid at the center. To predict a rotamer, the corresponding backbone stretch in the template is superposed on all the collected examples,



**Figure 25.8.** Example of a backbone-dependent rotamer library. The current backbone conformation (space-filling display) favors two different rotamers for Tyrosine (sticks), which appear about equally often in the database.

and the possible side-chain conformations are selected from the best backbone matches (Chinae et al., 1995).

Further evidence that the combinatorial problem of rotamer prediction is far smaller than originally believed was found recently. Xiang and Honig (2001) first removed one single side chain from known structures and repredicted it. In a second step, they removed all the side chains and added them again using the same simple search strategy. Surprisingly, it turned out that the accuracy was only marginally higher in the much easier first case.

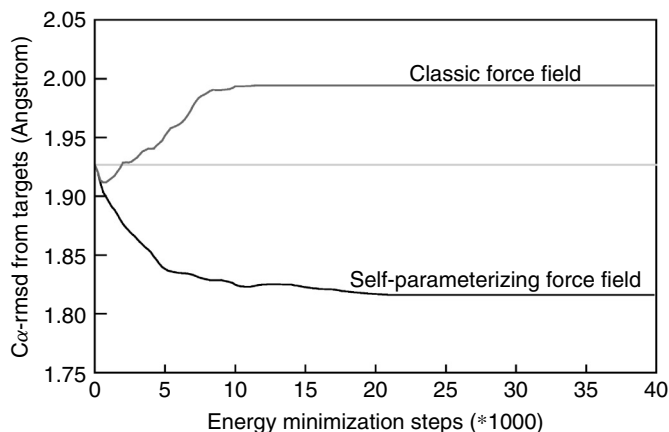
The prediction accuracy is usually quite high for residues in the hydrophobic core where more than 90% of all  $\chi_1$ -angles fall within  $\pm 20^\circ$  from the experimental values, but much lower for residues on the surface where the percentage is often even below 50%. There are two reasons for this:

1. Experimental reasons: flexible side chains on the surface tend to adopt multiple conformations, which are additionally influenced by crystal contacts. So even experiment cannot provide one single correct answer.
2. Theoretical reasons: the energy functions used to score rotamers can easily handle the hydrophobic packing in the core (mainly Van der Waals interactions), but are not precise enough to get the complicated electrostatic interactions on the surface right, including hydrogen bonds with water molecules and associated entropic effects.

It is important to note that the prediction accuracies given in most publications cannot be reached in real-life applications. This situation is simply due to the fact that the methods are evaluated by taking a known structure, removing the side chains and repredicting them. The algorithms thus rely on the correct backbone, which is not available in homology modeling. The backbone of the template often differs significantly from the target. The rotamers must thus be predicted based on an incorrect backbone and prediction accuracies tend to be lower in this case.

### Step 6: Model Optimization

The problem just mentioned above leads to a classical chicken-and-egg situation. To predict the side-chain rotamers with high accuracy, we need the correct backbone, which in turn depends on the rotamers and their packing. The common approach to such a problem is an iterative one: predict the rotamers, then the resulting shifts in the backbone, then the rotamers for the new backbone, and so on, until the procedure converges. This method boils down to a sequence of rotamer prediction and energy minimization steps. The latter use the methods from the loop-modeling step above, but this time they must be applied to the entire protein structure, not just an isolated loop. This requires an enormous precision in the energy function, because there are many more paths leading away from the answer (the target structure) than toward it, which is why energy minimization must be used carefully. At every minimization step, a few big errors (like bumps, i.e., too short atomic distances) are removed while many small errors are introduced. When the big errors are gone, the small ones start accumulating and the model moves away from the target (Fig. 25.9). As a rule of thumb, today's modeling programs therefore either restrain the atom positions and/or apply only a few hundred steps of energy minimization. In short, model optimization does not work until



**Figure 25.9.** The average rmsd between models and targets during an extensive energy minimization of 14 homology models with two different force fields. Both force fields improve the models during the first  $\sim 500$  energy minimization steps but then the small errors sum up in the classic force field and guide the minimization in the wrong direction, away from the target while the self-parameterizing force field goes in the right direction. To reach experimental precision, the minimization would have to proceed all the way down to  $\sim 0.5$  Å, which is the uncertainty in experimentally determined coordinates.

energy functions (force fields) get more precise. Two ways to achieve that precision are currently being pursued:

1. Quantum force fields: protein force fields must be fast to handle these large molecules efficiently, energies are therefore normally expressed as a function of the positions of the atomic nuclei only. The continuous increase of computer power has now finally made it possible to apply methods of quantum chemistry to entire proteins, arriving at more accurate descriptions of the charge distribution (Liu et al., 2001). It is however still difficult to overcome the inherent approximations of today's quantum chemical calculations. Attractive Van der Waals forces are, for example, so difficult to treat, that they must often be completely omitted. While providing more accurate electrostatics, the overall precision achieved is still about the same as in the classical force fields.
2. Self-parameterizing force fields: the precision of a force field depends to a large extent on its parameters (e.g., Van der Waals radii, atomic charges). These parameters are usually obtained from quantum chemical calculations on small molecules and fitting to experimental data, following elaborate rules (Wang, Cieplak, and Kollman, 2000). By applying the force field to proteins, one implicitly assumes that a peptide chain is just the sum of its individual small molecule building blocks—the amino acids. Alternatively, one can just state a goal, for example, improve the models during an energy minimization, and then let the force field parameterize itself while trying to optimally fulfill this goal (Krieger, Koraimann, and Vriend, 2002). This method

leads to a computationally rather expensive procedure. Take initial parameters (for example, from an existing force field), change a parameter randomly, energy minimize models, see if the result improved, keep the new force field if yes, otherwise go back to the previous force field. With this procedure, the force field precision increases enough to go in the right direction during an energy minimization (Fig. 25.9), but experimental precision is still far out of reach.

The most straightforward approach to model optimization is simply to run a molecular dynamics simulation of the model. Such a simulation follows the motions of the protein on a femtosecond ( $10^{-15}$  s) timescale and mimics the true folding process. One thus hopes that the model will complete its folding and “home in” to the true structure during the simulation. The advantage is that a molecular dynamics simulation implicitly contains entropic effects that are otherwise difficult to treat; the disadvantage is that the force fields are again not precise enough to make it work. (One must in fact be happy if the model is not messed up during the simulation). Nevertheless, one of the main tasks of *Blue Gene*, the forthcoming fastest computer in the world, will be to run exactly this type of molecular dynamics simulations (IBM Blue Gene team, 2001). More precise force fields will have to be available when *Blue Gene* goes online in 2005.

### Step 7: Model Validation

Every homology model contains errors. The number of errors (for a given method) mainly depends on two values:

1. The percentage sequence identity between template and target. If it is greater than 90%, the accuracy of the model can be compared to crystallographically determined structures, except for a few individual side chains (Chothia and Lesk, 1986; Sippl, 1993). From 50% to 90% identity, the rms error in the modeled coordinates can be as large as 1.5 Å, with considerably larger local errors. If the sequence identity drops to 25%, the alignment turns out to be the main bottleneck for homology modeling, often leading to very large errors.
2. The number of errors in the template.

Errors in a model become less of a problem if they can be localized. It is, for example, hardly important that a loop far away from an enzyme’s active site is placed incorrectly. An essential step in the homology modeling process is therefore the verification of the model. There are two principally different ways to estimate errors in a structure:

1. Calculating the model’s energy based on a force field: This method checks if the bond lengths and bond angles are within normal ranges, and if there are lots of bumps in the model (corresponding to a high Van der Waals energy). Essential questions such as “Is the model folded correctly?” cannot yet be answered this way, because completely misfolded but well-minimized models often reach the same force field energy as the target structure (Novotny, Rashin, and Bruccoleri, 1988). This result is mainly due to the fact that molecular dynamics force fields do not explicitly contain entropic terms (such as the

hydrophobic effect), but rely on the simulation to generate them. Although this problem can be addressed by extending the force field and adding, for example, solvation, the major drawback is that one always obtains a single number for the entire protein and cannot easily trace problems down to individual residues.

2. Determination of normality indices that describe how well a given characteristic of the model resembles the same characteristic in real structures. Many features of protein structures are well suited for normality analysis. Most of them are directly or indirectly based on the analysis of interatomic distances and contacts. Some published examples are:
  - General checks for the normality of bond lengths, bond and torsion angles (Morris et al., 1992; Czaplowski et al., 2000) are good checks for the quality of experimentally determined structures, but are less suitable for the evaluation of models because the better model-building programs simply do not make this kind of error.
  - Inside/outside distributions of polar and apolar residues can be used to detect completely misfolded models (Baumann, Frommel, and Sander, 1989).
  - The radial distribution function for a given type of atom (i.e., the probability to find certain other atoms at a given distance) can be extracted from the library of known structures and converted into an energylike quantity, called a “potential of mean force” (Sippl, 1990). Such a potential can easily distinguish good contacts (e.g., between a C $\gamma$  of valine and a C $\delta$  of isoleucine) from bad ones (e.g., between the same C $\gamma$  of valine and the positively charged amino group of lysine).
  - If not only the distance, but also the direction of atomic contacts is taken into account, one arrives at 3D distribution functions that can also easily identify misfolded proteins and are good indicators of local model building problems (Vriend and Sander, 1993).

Most methods used for the verification of models can also be applied to experimental structures (and hence to the templates used for model building). A detailed verification is essential when trying to derive new information from the model, either to interpret or predict experimental results or plan new experiments.

In summary, it is safe to say that homology modeling is unfortunately not as easy as stated in the beginning. Ideally, homology modeling uses threading (Chapter 26) to improve the alignment, and *ab initio* folding (Chapter 27) to predict the loops and molecular dynamics simulations with a perfect force field to home in to the true structure. Doing all that correctly will keep researchers busy for a long time, leaving lots of fascinating discoveries to good old experiment.

## ACKNOWLEDGMENTS

We thank Rolando Rodriguez, Chris Spronk, and Rob Hooft for stimulating discussions and practical help. We apologize to the numerous crystallographers who made all this work possible by depositing structures in the PDB for not referring to each of the 16,000 very important articles describing these structures.

## FURTHER READING

- Gregoret LM, Cohen FE (1990): Novel method for the rapid evaluation of packing in protein structures. *J Mol Biol* 211:959–74.
- Holm L, Sander C (1992): Evaluation of protein models by atomic solvation preference. *J Mol Biol* 225:93–105.

## REFERENCES

- Altschul SF, Gish W, Miller W, Myers EW, Lipman DJ (1990): Basic local alignment search tool. *J Mol Biol* 215:403–10.
- Bates PA, Sternberg MJE (1999): Model building by comparison at CASP3: using expert knowledge and computer automation. *Proteins* (Suppl. 3):47–54.
- Baumann G, Frommel C, Sander C (1989): Polarity as a criterion in protein design. *Protein Eng* 2:329–34.
- China G, Padron G, Hooft RWW, Sander C, Vriend G (1995): The use of position specific rotamers in model building by homology. *Proteins* 23:415–21.
- Chothia C, Lesk AM (1986): The relation between the divergence of sequence and structure in proteins. *EMBO J* 5:823–36.
- Czaplewski C, Rodziewicz-Motowidlo S, Liwo A, Ripoll DR, Wawak RJ, Scheraga HA (2000): Molecular simulation study of cooperativity in hydrophobic association. *Protein Sci* 9:1235–45.
- Dayringer HE, Tramontano A, Fletterick RJ (1986): Interactive program for visualization and modelling of proteins, nucleic acids and small molecules. *J Mol Graph* 4:82–7.
- de Filippis V, Sander C, Vriend G (1994): Predicting local structural changes that result from point mutations. *Protein Eng* 7:1203–8.
- Desmet J, De Maeyer M, Hazes B, Lasters I (1992): The dead-end elimination theorem and its use in protein side-chain positioning. *Nature* 356:539–42.
- Dodge C, Schneider R, Sander C (1998): The HSSP database of protein structure–sequence alignments and family profiles. *Nucleic Acids Res* 26:313–5.
- Dunbrack RL Jr, Karplus M (1994): Conformational analysis of the backbone dependent rotamer preferences of protein side chains. *Nat Struct Biol* 5:334–40.
- Epstein CJ, Goldberger RF, Anfinsen CB (1963): *Cold Spring Harb Symp Quant Biol* 28:439.
- Fischer D, Barret C, Bryson K, Elofsson A, Godzik A, Jones D, Karplus KJ, Kelley LA, MacCallum RM, Pawowski K, Rost B, Rychlewski L, Sternberg MJE (1999): CAFASP1: Critical assessment of fully automated structure prediction methods. *Proteins* (Suppl. 3):209–17.
- Fiser A, Do RK, Sali A (2000): Modeling of loops in protein structures. *Protein Sci* 9:1753–73.
- Hooft RWW, Vriend G, Sander C, Abola EE (1996): Errors in protein structures. *Nature* 381:272.
- IBM Blue Gene team (2001): Blue Gene: a vision for protein science using a petaflop supercomputer. *IBM Sys J* 40:310–27.
- Krieger E, Koraimann G, Vriend G (2002): Increasing the precision of comparative models with YASARA NOVA—a self-parameterizing force field. *Proteins* 47:393–402.
- Liu H, Elstner M, Kaxiras E, Frauenheim T, Hermans J, Yang W (2001): Quantum mechanics simulation of protein dynamics on long timescale. *Proteins* 44:484–9.
- Morris AL, MacArthur MW, Hutchinson EG, Thornton JM (1992): Stereochemical quality of protein structure coordinates. *Proteins* 12:345–64.

- Novotny J, Rashin AA, Bruccoleri RE (1988): Criteria that discriminate between native proteins and incorrectly folded models. *Proteins* 4:19–30.
- Pearson WR (1990): Rapid and sensitive sequence comparison with FASTP and FASTA. *Methods Enzymol* 183:63–98.
- Peitsch MC, Schwede T, Guex N (2000): Automated protein modelling—the proteome in 3D. *Pharmacogenomics* 1:257–66.
- Rost B (1999): Twilight zone of protein sequence alignments. *Protein Eng* 12:85–94.
- Sali A, Blundell TL (1993): Comparative protein modelling by satisfaction of spatial restraints. *J Mol Biol* 234:779–815.
- Sanchez R, Sali A (1997): Evaluation of comparative protein structure modeling by MODELLER-3. *Proteins* (Suppl. 1):50–8.
- Sanchez R, Sali A (1999): ModBase: a database of comparative protein structure models. *Bioinformatics* 15:1060–1.
- Sander C, Schneider R (1991): Database of homology-derived protein structures and the structural meaning of sequence alignment. *Proteins* 9:56–68.
- Simons KT, Bonneau R, Ruczinski I, Baker D (1999): Ab initio structure prediction of CASP III targets using ROSETTA. *Proteins* (Suppl. 3):171–6.
- Sippl MJ (1990): Calculation of conformational ensembles from potentials of mean force. *J Mol Biol* 213:859–83.
- Sippl MJ (1993): Recognition of errors in three dimensional structures of proteins. *Proteins* 17:355–62.
- Stites WE, Meeker AK, Shortle D (1994): Evidence for strained interactions between side-chains and the polypeptide backbone. *J Mol Biol* 235:27–32.
- Tappura K (2001): Influence of rotational energy barriers to the conformational search of protein loops in molecular dynamics and ranking the conformations. *Proteins* 44:167–79.
- Taylor WR (1986): Identification of protein sequence homology by consensus template alignment. *J Mol Biol* 188:233–58.
- Thompson JD, Higgins DG, Gibson TJ (1994): ClustalW: improving the sensitivity of progressive multiple sequence alignments through sequence weighting, position-specific gap penalties and weight matrix choice. *Nucleic Acids Res* 22:4673–80.
- Vriend G (1990): WHAT IF—A molecular modeling and drug design program. *J Molec Graphics* 8:52–6.
- Vriend G, Sander C (1993): Quality control of protein models: directional atomic contact analysis. *J Applied Crystallogr* 26:47–60.
- Wang J, Cieplak P, Kollman PA (2000): How well does a restrained electrostatic potential (RESP) model perform in calculating conformational energies of organic and biological molecules? *J Comput Chem* 21:1049–74.
- Xiang Z, Honig B (2001): Extending the accuracy limits of prediction for side-chain conformations. *J Mol Biol* 311:421–30.