Protein Design

Hagen Fritsch

Rostlab, TU München

See `http://itooktheredpill.dyndns.org/2011/protein-design/` for slides and further illustrations.

Advisor: Marc Offman

ABSTRACT

Protein Design is one of the major frontiers for bioinformatics and will have huge impact on any chemistry-related industry. This paper aims to give an introduction to the topic and its challenges, by covering concepts crucial to computational protein design and providing. prospects of recent progress and future developments.

1 INTRODUCTION

Protein design has been attempted for quite some time, with the earliest results dating back to 1979 (Gutte *et al.*). It is fascinating scientist since, due to the enormous opportunities that become accessible if the field were mastered. Being able to catalyze arbitrary chemical reactions would be of great use, as complicated energy-consuming processes could be made more efficient. New chemical compounds that could not yet be synthesized efficiently may become available promising new materials, environmentally friendly methods as well as new and more efficient synthesis of drugs (cf. Jiang *et al.*, 2008). The medical industry also benefits from research in protein design that constantly improves the understanding of how proteins work.

On a larger scale, organisms may be altered for specific functionality. Bacteria digesting plastic or oil are only examples for the potential biology has to offer and that science can enhance and extend.

Unfortunately, these results are not in direct reach yet, although our understanding of proteins, their folding and catalytic mechanisms has been greatly increased since the early approaches (cf. Gutte *et al.*, 1979; Hellinga and Richards, 1991; Dahiyat and Mayo, 1997). Protein design is complicated, limited by our knowledge and computational capabilities. The following section will introduce the major challenges protein design is facing or has already mastered to some degree.

2 CHALLENGES AND CONCEPTS

2.1 Computational Complexity

One of the major challenges governing computational protein design is the huge combinatorial searchspace: Already for a very short length, i.e. a 30-peptide protein there exist 20^{30} different sequences. This number of sequences is roughly equal to a complexity of 2^{128}, which is a size considered computationally infeasible (i.e. cryptographically secure). As such, it is a more than difficult task for an algorithm, to find a sequence that fulfills the design criteria in this searchspace. Consequently, smart methods and algorithms are needed to reduce the complexity. As there are no methods to directly construct proteins residue after residue, the algorithms are faced with this huge complexity and therefore have to make compromises between computational speed and thoroughness (Desjarlais and Clarke, 1998).

2.2 Structure Prediction

Besides the complexity problem there are further obstacles to overcome. One is the unsolved issue of accurate structure prediction for a given sequence (there is exciting related progress though, e.g. cf. Bonneau *et al.*, 2001; Bowers *et al.*, 2000). But considering, that one cannot even *accurately* calculate the three-dimensional structure of the protein, how should one design one from scratch? This directly leads to the so-called *Inverse Folding Problem*, the essence of which is: *Given an existing three-dimensional structure of a protein backbone, find any sequence that folds to this structure.* Since a huge number of sequences results in the same fold (cf. Kuhlman and Baker, 2000), this problem is easier to solve, since there is not only one sequence folding to the structure, but a seemingly infinite number that folds to a very similar structure with only minor differences. It is not necessary that there is a 100% match between the fixed template backbone and the one of the designed protein: small deviations are tolerated and there are efforts to model such *backbone flexibility* (cf. Street and Mayo, 1999). The construction of a sequence that can be threaded on the given backbone follows certain rules that closely relate to energy functions (as discussed in Section 2.4).

2.3 Rotamer Libraries

An essential tool for constructions on fixed protein backbones are rotamer libraries. If one had to consider any potential conformation of an amino-acid residue, the possibilities would again be too high to be computationally feasible. Additionally, the conformational space is not discrete, despite the need for discretization for the algorithms presented in Section 3 (Street and Mayo, 1999). As it turned out, not all of these conformations, that are referred to as rotamers, are equally likely. Thus a way to reduce this part of the complexity, is to restrict the possible conformations to the most frequently observed ones in nature. Accordingly rotamer libraries, are build using statistical analysis on PDB data (illustration in Figure 1) and come in several flavors and sizes, from small libraries of only 67 residue conformations (Ponder and Richards, 1987) to huge ones in the range multiple 10’000 conformations.

The drawback of using rotamer libraries (esp. smaller ones) is the increased likelihood of missing a working conformation during the search Street and Mayo (1999).

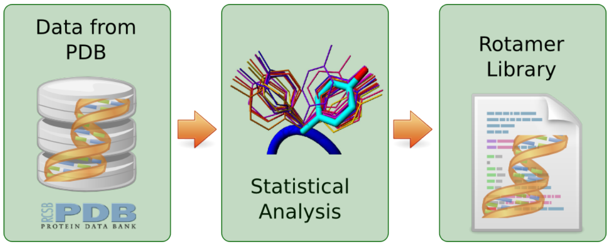

Figure 1. Rotamer library creation workflow

Since certain amino-acids prefer certain parts of the protein (i.e. core, boundary or surface), specific rotamer libraries have been built to take this into account (cf. Street and Mayo, 1999). Further extensions are based on secondary structure classification, that additionally restrict the applicable residues and help reduce the computational complexity.

2.4 Energy Functions

To evaluate the stability of a protein, energy functions are used, that are designed to correlate with the experimental stability (Street and Mayo, 1999). They predict the enthalpy of a specific conformation of the protein which is a predictor for its stability (Dahiyat, 1999). Early energy functions are based on simple rules that govern the stability of the protein's core: In short they boil down to avoiding steric clashes while filling all space (i.e. minimizing "holes") (Ponder and Richards, 1987), which are simple enough rules so that they can be used in constructing initial prototypes (Hellinga and Richards, 1991) according to the *Inverse Folding Problem*. Mathematically these rules are simulated through a so-called *van-der-Waals* potential, consisting of the *Lennard-Jones* function for approximation of the van-der-Waals forces and Coulomb's law (Boas and Harbury, 2007). It has been used in various experiments (i.e. Dahiyat and Mayo, 1997). Further forces (the bonded interactions in Figure 2) are also considered in these Molecular Mechanics Potential Energy Functions (MM-PEF; Boas and Harbury, 2007).

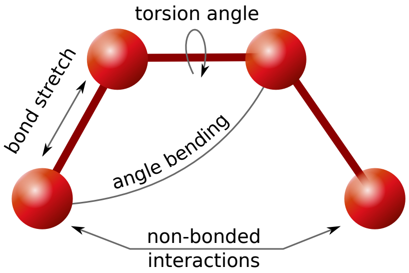

Figure 2. Basic forces considered for energy functions

A lot of progress is being made in the enhancement of energy functions, which are not only important for protein design. Being based on the same physical principles, these functions are also important to structure prediction (cf. Boas and Harbury, 2007). Unfortunately, the rules stated above are only reasonably accurate for the core of the protein. Other forces are more dominant in solvent accessible parts, which are still hard to model, but are also target of the early enhancements of energy functions (c.f. Dahiyat and Mayo, 1996): For example Dahiyat and Mayo (1997) used an additional atomic solvation potential "favoring the burial and penalizing the exposure of non-polar surface area".

For specific protein interactions, specific energy functions have to be developed. According to Lippow and Tidor (2007), recent progress has been made concerning DNA-protein interactions (Morozov *et al.*, 2005) as well as metal center interactions (Spiegel *et al.*, 2006) that play a crucial role in many catalytic activities. Solvent and solvent-mediated effects were once more tackled by Marshall *et al.* (2005), who also designed a pairwise approximation (this is important for the DEE-algorithm, see 3.1). Notable is also the idea to include individual water molecules in the rotamer library (Jiang *et al.*, 2005), as water molecules play a huge role due to their ability to act as hydrogen-bond donors and acceptors (Jiang *et al.*, 2008).

The lower the enthalpy of the protein, the more likely it is to actually adopt the desired fold. In the end, however, energy functions only estimate the enthalpy in the protein and as such the quality of the designed result sequence is directly dependent on the quality of the energy functions used.

3 ALGORITHMS AND METHODS

Another crucial aspect of protein design is the efficient construction of low-energy sequences, which is unfortunately all but trivial due to the huge search space complexity (see Section 2.1). A variety of approaches exists. The most frequently used algorithms are the *Dead End Elimination*, the *simulated-annealing Monte Carlo Method* (both of which will be briefly described in the following sections), as well as *Self-consistent mean field theory* and *Genetic Algorithms* the latter of which seeks to optimize a population of solutions inspired by biological operators such as random mutations, selection and recombination (Voigt *et al.*, 2000). More detailed explanations and comparisons towards speed and accuracy of the different algorithms are found in Voigt *et al.* (2000).

3.1 Dead End Elimination

Dead End Elimination (Desmet *et al.*, 1992) is a deterministic method that converges to the Global Minimum Energy Configuration (GMEC). The basic principle is to exclude rotamers that cannot be part of the GMEC due to mathematical properties. For this to work, the energy function has to be defined pairwise, i.e. consist solely of pairwise rotamer interaction energies, such that the whole energy is the sum of all pairwise interactions. The condition for a rotamer to be rejected is expressed in Equation 1.

$$E(i_r) + \sum_{j \neq i}^{N} \min_{s} E(i_r, j_s) > E(i_t) + \sum_{j \neq i}^{N} \max_{s} E(i_t, j_s) \tag{1}$$

If best energy one rotamer (r) can achieve at position i with any other rotamer (j_s) is higher than the maximum energy another rotamer (t) can achieve with any other rotamer, than i_r can be eliminated. Further considerations include pairs or higher order combinations that cannot possibly be part of the GMEC (Desjarlais and Clarke, 1998). Dead End Elimination is only feasible for small sequences (i.e. up to 60 residues, Dahiyat, 1999) and as such not suited for larger problems, where randomized methods come into play.

3.2 Monte Carlo Method

The Monte Carlo Method is one of these randomized methods. It is based on the monte carlo principle of repeated random sampling, but has also been applied to protein design (e.g. in Dahiyat and Mayo, 1996). The algorithm works as follows (Voigt *et al.*, 2000):

- Pick an initial random sequence of rotamers
- Replace a randomly picked residue by a different rotamer and calculate the new energy: E_{new}, accepting any move with lower energy (i.e. $E_{new} < E_{old}$).
- Moves with higher energy are sometimes accepted based on the Boltzman probability

$$\rho = e^{\frac{E_{new}-E_{old}}{kT}} \tag{2}$$

Accepting moves with higher energy is necessary for not getting trapped in local minima and the temperature influences the acceptance-likelihood based on the idea of simulated annealing (as applied in a monte carlo method for thermodynamic systems by Metropolis *et al.*, 1953): high temperatures allow the sequence to overcome energy barriers and low temperatures force it to converge to some optimum. The algorithm is then run in cycles in each of which the temperature is raised and lowered. To get an impression of the numbers, Voigt *et al.* (2000) state, that they typically set the number of cycles to 1000 with 10^6 substitution attempts each, while Kuhlman and Baker (2000) also have 10^6 substitutions, but for a whole run of the method, although they do five different runs to get some sequence diversity.

Eventually the monte carlo method yields sequences, that have no guarantee to be optimal or even close to the GMEC, but are typically well enough starting points. Kuhlman and Baker (2000) compared such designed sequences to native sequences and point out, that “the sequences obtained were nearly identical [...] and had similar energies”.

3.3 Directed Evolution

The sequences generated *in silico* have been optimized using energy functions, i.e. on assumptions made, that are neither completely accurate nor do they include all aspects of protein stability. Consequently, such sequences can only be considered as educated guesses. Currently, protein design experiments are already considered a success if initial catalytic activity is present for some of the designed sequences. Such initial activity is a required starting point for directed evolution experiments that seek to optimize the catalytic activity *in vitro* (Röthlisberger *et al.*, 2008) by allowing the fine-tuning of the sequence through evolution by repeated rounds of mutation and selection without having to rely on the limited energy models. Therefore Röthlisberger *et al.* (2008) argue that these experiments provide a way to potentially remedy these shortcomings. The results are typically enzymes with highly improved catalytic rates (200-fold in the case of Röthlisberger *et al.*, 2008) that may also provide further insight into which improvements can be made to the computational design process to achieve more accurate results.

3.4 Protein Design Cycle

This idea of improving the design process through experiments was already propsed in 1996 by Dahiyat and Mayo. Their protein design cycle consists roughly of the following steps:

1. Predict a couple of protein sequences likely to achieve a desired fold using the available protein design methods.
2. Synthesize these peptides for experimental classification.
3. Using quantitative analysis methods (predicted vs. actual stability), derive new terms that allow to improve the estimated free energy of the protein.

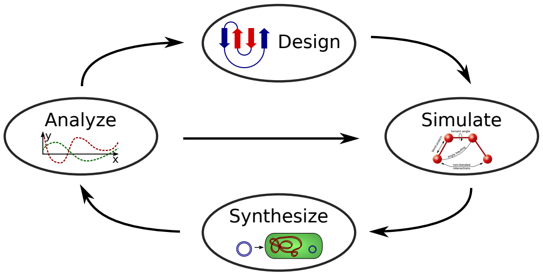

Figure 3. Protein design automation cycle (derived from Dahiyat and Mayo, 1996) Plasmid image based on Wikipedia:Plasmid_replication_(english).svg

Using this method, experiments can be gradually improved without being biased by subjective interpretation of the data. For example, Dahiyat and Mayo (1996) found that the effect of solvation (which had been missing from their simulation) is crucial, which has also been emphasized in later studies (c.f. Dahiyat and Mayo, 1997; Street and Mayo, 1999; Lippow and Tidor, 2007). Incorporating the suggested terms into the energy functions thus allowed them to improve their design performance, allowing to select better targets for synthesis. As there is not only one method (energy function) to estimate the quality of a target design, alternative ones can be used to verify or further select from the set of generated target sequences to reduce the probability of bad design due to deficiencies of the specific energy function.

4 WORKFLOW

After clarifying fundamental concepts, the description of a likely workflow shall paint a more complete picture:

The protein design process starts with defining, what the protein shall be able to do. Typically proteins are enzymes, catalyzing chemical reactions that require a high activation energy. By supporting transition states, proteins are able to lower this activation energy and thus either enhance reaction speed or even make this reactions possible in the first place. Consequently the first steps are the analysis of the chemical reaction, defining the catalysis mechanism and identifying possible realizations through a protein (Jiang *et al.*, 2008; Röthlisberger *et al.*, 2008). For example, Röthlisberger *et al.* (2008) require the deprotonation of a carbon atom by a general base as a key step in their reaction, for which the carboxyl group of Glu or Asp can be used.

Site identification. The next step in the protein design process is the identification of existing folds that have a geometry supporting the appropriate placement of the catalytic residues identified previously. Here, a fold is the three-dimensional structure of the protein backbone (of all amino acids in the protein, only the backbone atoms ($\mathrm{N} - \mathrm{C}^{\alpha} - \mathrm{CO}$) of each residue remains). This search can already be computationally challenging and a variety of approaches have been proposed. An early approach for this problem was the `DEZYMER` program (Hellinga and Richards, 1991) with a forward-search based approach. A similar, but smarter and more efficient method is known as RosettaMatch (Zanghellini *et al.*, 2006) that builds each catalytic side chain idependently from the backbone to the ligand identifying placements compatible with the ligand / transition state and each catalytic residue. The most intuitive idea however, is the inverse-rotamer tree approach (Zanghellini *et al.*, 2006), that takes a rotamer library and instead of building all residue conformations on a predetermined backbone (`DEZYMER`), it builds all conformations for the given catalytic residue starting from the catalytic end of the amino acid (see Figure 4). The result of it is a huge set of combinations of backbone-positions that would support the catalytic site. Known protein folds are then searched for compatibility with any of the combinations. This search can be accelerated by using geometric hashing techniques techniques (cf. Zanghellini *et al.*, 2006; Röthlisberger *et al.*, 2008). Typically this already produces a huge number of supporting scaffolds. Sanity checks (such as for substrate accessibility of the active site) can be used to filter this set initially at this stage and/or after the algorithmic optimization (see below). Such manual curation is still critical to the success of protein design experiments (Lippow and Tidor, 2007) and even visual inspections of the model are not rare and uncover problems not considered by the algorithms.

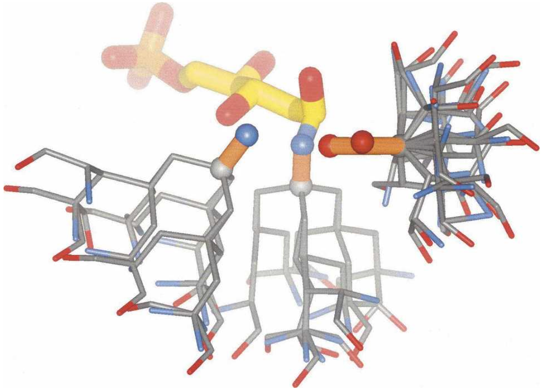

Figure 4. Inverse rotamer approach. Illustration by Zanghellini *et al.* (2006): *"The transition state is colored in yellow, and the key functional groups of the catalytic residues are in gold. The remainder of the side chains in the rotamer trees are shown using thinner lines in CPK coloring."*

A notable exception to the approach based on existing folds has been accomplished by Kuhlman *et al.* (2003), who where able to design a whole protein backbone from scratch by iterating between sequence optimization for their fixed (designed) backbone and optimization of the three-dimensional structure (i.e. the backbone coordinates) for a likely optimal sequence. However, this additional effort is typically not required as above methods already produce more than sufficient numbers of potential target scaffolds.

Threading. In the third step, the active site has to be placed on the backbone and the rest of the protein has to be filled to stabilize the structure, so that the protein achieves the desired fold (see Section 2.2). An alternative is to take the sequence of the selected template protein and to place the active site residues on the backbone according to the previously specified geometry. Ideally most parts of the protein remain in optimal condition, requiring only repacking of the newly designed active site and its surroundings, which apparently will be non-optimal packed in the beginning. But as the rest of the protein is packed in an optimal fashion, this may already be a good enough starting point for the optimization phase.

Optimization. After an initial sequence has been proposed, the algorithms from Section 3 come into play, optimizing the sequence based on appropriate energy functions. A set of target sequences is eventually produced and can be validated or further restricted using alternative energy functions. Since the protein stability is not the only important aspect in protein design and in fact the aspects governing efficient enzymes are not yet completely understood (Lippow and Tidor, 2007), other means, such as the predicted transition state binding energy or the extent of satisfaction of the catalytic geometry (Jiang *et al.*, 2008), may be used as well.

Synthesis & Directed Evolution. Analog to the protein design cycle (see Section 3.4), the best of these sequences are selected for experimental classification. If initial catalytic activity is found, the catalytic rate may be drastically enhanced through directed evolution experiments (see Section 3.3).

5 SUMMARY

Even simplistic protein design faces major challenges concerning a variety of issues, such as computational complexity, energy function design or precise atomic-level simulations. Though far from being easy-to-use, complete or accurate, proteins have already been successfully designed *de novo*. The process matured and new proteins continue to be designed for evermore complex chemical reactions (Lippow and Tidor, 2007), although they do not yet reach the catalytic rates of naturally occurring ones (Jiang *et al.*, 2008).

Protein design is not in its infancy anymore, but major open questions still need to be addressed and due to the limitless possibilities protein design may be used for, it is likely that the opportunities for further research and improvements will be taken: processes will become easier, more reliable and expand the realm possibilities (Lippow and Tidor, 2007).

Not every enzyme functionality has to be designed, many are already available in nature. Re-engineering of existing proteins is already making such functionality available to other contexts. As synthetic biology continues to benefit from understanding of biological networks, gaps can be filled through the design of specific proteins (Pleiss, 2011) proceeding with evermore technical solutions for pending medical, environmental and energy problems of our planet.

REFERENCES

Boas, F. E. and Harbury, P. B. (2007). Potential energy functions for protein design. *Curr. Opin. Struct. Biol.*, **17**(2), 199–204.

Bonneau, R., Tsai, J., Ruczinski, I., Chivian, D., Rohl, C., Strauss, C. E., and Baker, D. (2001). Rosetta in CASP4: progress in ab initio protein structure prediction. *Proteins*, **Suppl 5**, 119–26.

Bowers, P. M., Strauss, C. E. M., and Baker, D. (2000). De novo protein structure determination using sparse NMR data. *Journal of Biomolecular NMR*, **18**(4), 311–318.

Dahiyat, B. I. (1999). In silico design for protein stabilization. *Current opinion in biotechnology*, **10**(4), 387–390.

Dahiyat, B. I. and Mayo, S. L. (1996). Protein design automation. *Protein Science*, **5**(5), 895–903.

Dahiyat, B. I. and Mayo, S. L. (1997). De novo protein design: fully automated sequence selection. *Science*, **278**(5335), 82.

Desjarlais, J. R. and Clarke, N. D. (1998). Computer search algorithms in protein modification and design. *Current opinion in structural biology*, **8**(4), 471–475.

Desmet, J., Maeyer, M. D., Hazes, B., and Lasters, I. (1992). The dead-end elimination theorem and its use in protein side-chain positioning. *Nature*, **356**(6369), 539–542.

Gutte, B., Däumigen, M., and Wittschieber, E. (1979). Design, synthesis and characterisation of a 34-residue polypeptide that interacts with nucleic acids. *Nature*, **281**, 650–655.

Hellinga, H. W. and Richards, F. M. (1991). Construction of new ligand binding sites in proteins of known structure* 1:: I. Computer-aided modeling of sites with pre-defined geometry. *Journal of molecular biology*, **222**(3), 763–785.

Jiang, L., Kuhlman, B., Kortemme, T., and Baker, D. (2005). A “solvated rotamer” approach to modeling water-mediated hydrogen bonds at protein–protein interfaces. *Proteins: Structure, Function, and Bioinformatics*, **58**(4), 893–904.

Jiang, L., Althoff, E. A., Clemente, F. R., Doyle, L., Röthlisberger, D., Zanghellini, A., Gallaher, J. L., Betker, J. L., Tanaka, F., Barbas, C. F., Hilvert, D., Houk, K. N., Stoddard, B. L., and Baker, D. (2008). De novo computational design of retro-aldol enzymes. *Science*, **319**(5868), 1387–91.

Kuhlman, B. and Baker, D. (2000). Native protein sequences are close to optimal for their structures. *Proceedings of the National Academy of Sciences*, **97**(19), 10383.

Kuhlman, B., Dantas, G., Ireton, G. C., Varani, G., Stoddard, B. L., and Baker, D. (2003). Design of a novel globular protein fold with atomic-level accuracy. *Science*, **302**(5649), 1364.

Lippow, S. M. and Tidor, B. (2007). Progress in computational protein design. *Curr. Opin. Biotechnol.*, **18**(4), 305–11.

Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H., and Teller, E. e. a. (1953). Equation of state calculations by fast computing machines. *The journal of chemical physics*, **21**(6), 1087.

Pleiss, J. (2011). Protein design in metabolic engineering and synthetic biology. *Curr. Opin. Biotechnol.*, **22**(5), 611–7.

Ponder, J. W. and Richards, F. M. (1987). Tertiary templates for proteins* 1:: Use of packing criteria in the enumeration of allowed sequences for different structural classes. *Journal of molecular biology*, **193**(4), 775–791.

Röthlisberger, D., Khersonsky, O., Wollacott, A. M., Jiang, L., DeChancie, J., Betker, J., Gallaher, J. L., Althoff, E. A., Zanghellini, A., Dym, O., Albeck, S., Houk, K. N., Tawfik, D. S., and Baker, D. (2008). Kemp elimination catalysts by computational enzyme design. *Nature*, **453**(7192), 190–5.

Street, A. G. and Mayo, S. L. (1999). Computational protein design. *Structure*, **7**(5), –105.

Voigt, C. A., Gordon, D. B., and Mayo, S. L. (2000). Trading accuracy for speed: A quantitative comparison of search algorithms in protein sequence design. *J. Mol. Biol.*, **299**(3), 789–803.

Zanghellini, A., Jiang, L., Wollacott, A. M., Cheng, G., Meiler, J., Althoff, E. A., Röthlisberger, D., and Baker, D. (2006). New algorithms and an in silico benchmark for computational enzyme design. *Protein science*, **15**(12), 2785–2794.