Anne Ye
BIOC 218 – Final Project

A Critical Review of Computational Protein Design Strategies: Progress, Limitations, and Improvements

INTRODUCTION

The question of how a protein's 3D structure is encoded by its amino acid sequence, and how that structure in turn dictates its biological function, has been a long-standing mystery. With the advent of faster, more accurate algorithms for predicting protein structure based on primary sequence, growing interest and effort has been devoted to exploring the inverse problem, that of designing novel protein sequences to take on specified structures. Part of the interest is purely intellectual – computational protein design is the ultimate test of our understanding of processes such as protein folding and catalysis. By granting us control over all variables, protein design allows us to systematically probe the biochemical principles that underlie the complex and elegant molecular designs found in nature. At the same time, from a practical standpoint, protein design offers the tantalizing opportunity to unlock the potentially unlimited structural and functional space not sampled by nature. Proteins, both natural and evolved, have demonstrated tremendous versatility, specificity, and robustness in binding and catalysis that are often unparalleled by small molecules or synthetic processes. Thus, the ability to design novel protein systems and machines from first principles would pave the way for unprecedented advances in medicine, energy, bioremediation, and many other areas.

Similar to computational protein structure prediction, there are two general approaches to protein design. The first is "template-based," where an existing protein with a known sequence is redesigned to confer additional stability, specificity, functionality, etc. Unsurprisingly, this approach has been comparatively more successful, as by definition we have more information on which to base the new design. Because the new protein is still functionally relatively similar to the original protein, the new sequence and structure often retain some memory of their predecessors. The second approach is *de novo* design, in which a completely novel protein is generated with no prior knowledge other than the desired shape or function, and fundamental physicochemical principles governing interactions between amino acids and their surrounding environment. This is particularly challenging in the case of computational enzyme design for reactions not catalyzed in nature, where even the desired global fold is unknown, and must be extrapolated based on the kinetics of the desired chemical reaction. This paper will review both template-based and *de novo* approaches, discuss progress that has been achieved and limitations that still exist, and outline potential improvements to overcome these limitations.

ELEMENTS OF PROTEIN DESIGN METHODOLOGIES

Over the last several decades, computational protein designers have succeeded in redesigning existing proteins or constructing novel proteins to perform a diverse range of functions. However, regardless of whether the design aim involves engineering an active site, a protein-protein interface, or a completely new protein fold, these approaches all rely on the same broad methodology. Thus, rather than describe a number of similar design algorithms, this section will outline the underlying basic methodology common to all of them, highlighting critical steps and where aim-specific variation and tailoring can be incorporated.

Figure 1 depicts the various stages of designing a protein sequence to adopt a specified topology. The first step is to delineate the target structure for the design, a set of coordinates for the backbone atoms. Once the target conformation has been defined, a library of candidate amino acid sequences is generated and passed through a computational sieve, an energy function used to measure sequence-structure compatibility, to filter out high-energy sequences unlikely to adopt the desired topology. Because the number of potential sequences for even a protein of moderate size can rapidly become computationally unfeasible, a variety of sampling strategies have been applied to increase the efficiency of searching such a large sequence space. Once the initial library of sequences has been narrowed down to a manageable number of candidates, more computationally laborious but rigorous techniques, such as molecular dynamics, can be used to individually evaluate and refine each design. Finally, depending on the results of computational design, complementary experimental approaches, such as directed evolution, are often used to further improve the stability or function of the designed protein.

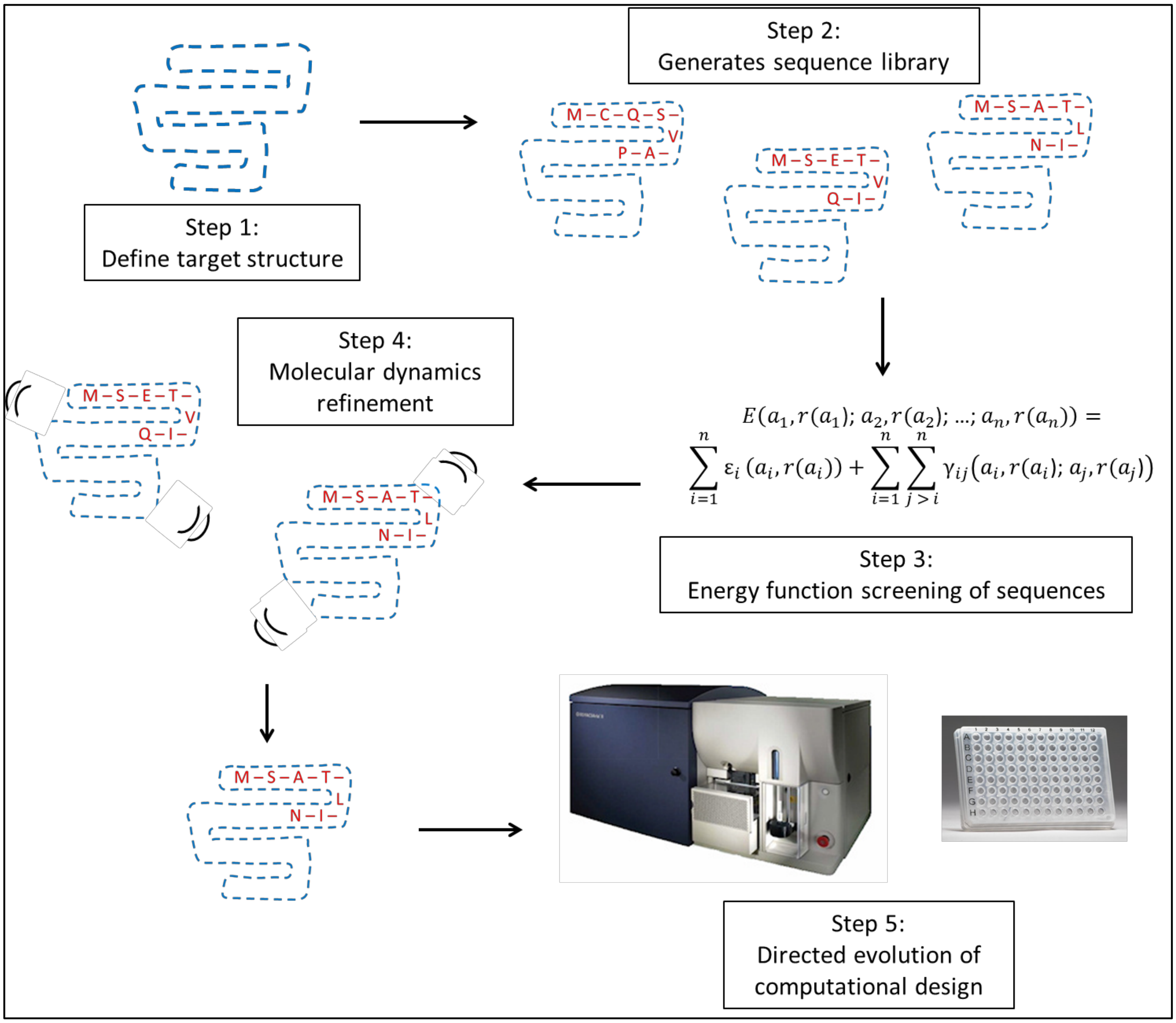
Step 1:
Define target structure
Step 2:
Generates sequence library
M – C – Q – S – V – P – A –
M – S – E – T – V – Q – I –
M – S – A – T – L – N – I –
$E(a_1, r(a_1); a_2, r(a_2); \ldots; a_n, r(a_n)) = \sum_{i=1}^{n} \varepsilon_i (a_i, r(a_i)) + \sum_{i=1}^{n} \sum_{j>i}^{n} \gamma_{ij}(a_i, r(a_i); a_j, r(a_j))$
Step 3:
Energy function screening of sequences
Step 4:
Molecular dynamics refinement
M – S – E – T – V – Q – I –
M – S – A – T – L – N – I –
Step 5:
Directed evolution of computational design

Figure 1. Schematic of the steps in computational protein design.

Defining a Target Structure

The target structure that the designed protein sequence is intended to fold into depends, naturally, on the function the protein is meant to serve. For example, in the case of redesigning a natural protein to bind the same target with greater specificity or affinity, the target structure should be similar to the existing one, so the target backbone coordinates can simply be obtained from a high-resolution crystal or NMR structure of the wild-type protein [1]. In the case of *de novo* design, a 2D diagram can be used as a template to assemble a 3D model from modular secondary structure elements or short peptide fragments from PDB that fit the desired topology [2, 3].

This latter option is particularly challenging in the case of *de novo* enzyme design, due to the high level of atomic precision in positioning the reactive functional groups that is necessary for efficient catalysis. The collaborative efforts of the Baker and Houk groups have led to the development of an “inside-out” protocol for designing novel enzyme structures [4]. Once a target reaction has been chosen, a model of the catalytic mechanism is used to pinpoint the requisite set of functional groups and propose a transition state for the reaction. Based on this, quantum mechanical calculations are used to generate a theoretical active site, or a theozyme, in which amino acids with the key functional groups are built into the geometry predicted to best stabilize the transition state. Once these parameters have been fixed, an accommodating scaffold can be selected from databases of known structures, and residues surrounding the active site are optimized to provide a stable framework for these key groups [4]. It is important to note here that the success of the designed enzyme is thus not only impacted by the computational algorithms used, but is also intrinsically linked to the accuracy of the proposed catalytic mechanism for the reaction, which is determined by biochemical studies.

Generating a Sequence Library

Given that the success of the final protein sequence design is dependent on the quality of the sequence library, the method used to enumerate candidate sequences for screening is one of the most critical features of a design algorithm. In searching for sequences to adopt a target structure, protein designers must walk a fine line, maintaining computational tractability while also widening the sequence search space to allow reasonably small deviations from the target structure. The latter consideration is particularly important during early iterations of *de novo* design, because it is uncertain whether any arbitrarily defined novel backbone is necessarily designable.

That said, in principle, the number of potential sequences for a protein comprised of *n* residues is 20^n. While protein redesign often relies on introducing mutations into a mostly fixed native sequence, in *de novo* design, it may be necessary to apply certain simplifications to constrain what is likely a vast search space. For example, the amino acid composition of secondary structural motifs have been extensively studied, beginning from Chou and Fasman’s early calculations for the propensities of the different amino acids to occur in structural elements such as α-helices and β-sheets [5-7]. These empirically determined parameters can be used to make educated guesses about which residues will be tolerated at certain positions in the protein, allowing some reduction in the sequence search space. Related to this, early success in *de novo* design was achieved using hydrophobic patterning, which is based on the premise that strategic arrangement of polar and nonpolar residues can direct folding into amphipathic secondary structures. By classifying amino acids as polar or nonpolar, hydrophobic patterning can dramatically reduce the combinatorial diversity that must be explored [8].

Besides these coarse-grained or qualitative methods for enumerating candidate sequences, more systematic search algorithms can be used to scan the sequence space. There are 2 broad categories of search algorithms: **(1)** stochastic, or probabilistic, and **(2)** deterministic. While the former relies on random sampling, and thus is faster, the latter is more time-consuming, but is exhaustive and thus guaranteed to find a global energy minimum.

A widely used stochastic sampling method, which is employed by RosettaDesign, is Monte Carlo optimization, or simulated annealing [3, 9]. In this method, a random starting design is generated and its energy calculated. Each subsequent step in the optimization involves introducing a single amino acid substitution to the starting sequence, and recalculating the energy. A given substitution is accepted automatically if it decreases the sequence's energy, but if it increases the energy, it is accepted or rejected in what is known as the Metropolis step, based on a probabilistic threshold that is a function of the Boltzmann distribution and the simulation temperature. Because the Monte Carlo method relies on stochastic sampling, it is not guaranteed to find the global energy minimum, and despite the Metropolis criterion, can still get trapped in local energy minima. To get around this, multiple runs of Monte Carlo optimization are performed with different random starting sequences in order to cover, as best as possible, a rugged energy landscape. Generally, the runs will converge to sequences of 70-80% identity, suggesting that the search is not getting trapped in a local energy minimum [9].

In contrast, ORBIT (optimization of rotamers by iterative technique), which was developed by the Mayo group, applies a deterministic search algorithm that is based on the dead-end elimination (DEE) theorem [10]. Rather than relying on random sampling, DEE cuts down on the sequence space that must be explored by iteratively pruning out amino acids or rotamer states that cannot be present in the global minimum energy conformation (GMEC). Briefly, for each amino acid at a given position, two rotamers, or side chain conformations, are compared with respect to the sum of the side chain-backbone energy (for that particular amino acid) and the minimum side chain-side chain energy that can be achieved with all other possible combinations of rotamers (i.e., at other positions). The side chain-backbone energetic contributions of the other residues in the sequence are ignored, because they are independent of, and thus assumed constant between, the two rotamer states for the specified residue [10, 11]. DEE is functionally exhaustive, and so is guaranteed to converge to the GMEC. However, due to computational limitations, DEE has generally been restricted to designing relatively small proteins. More recent modifications, such as a generalized DEE algorithm proposed by Looger and Hellinga, where clusters of rotamers are compared rather than individual rotamers, show promise in expanding the application of DEE to larger proteins [12].

Energy functions for ranking candidate sequences

Besides the search algorithm used to sample the sequence space, the other most important feature of a design methodology is the method used to evaluate and filter candidate sequences. Putative sequences are scored using an energy function to determine their compatibility with the target structure. The most successful energy functions are “hybrid” functions that employ both knowledge-based and physics-based potentials to calculate side chain-side chain and side chain-backbone interaction energies, which are then summed to give the total energy of a particular protein sequence and conformation. Examples of hybrid energy functions include those used by ORBIT and RosettaDesign, two of the most successful and widely used design softwares [9, 10]. Knowledge-based terms are statistical parameters derived from databases of known protein structures (i.e., Protein Databank, or PDB), and thus exploit the design principles that nature has used to generate robust and versatile proteins. For example, due to the hydrophobic effect, residues in the core of a protein tend to be more hydrophobic, while solvent-exposed residues on the surface tend to be hydrophilic. Thus, a candidate design would be heavily penalized for having a solvent-exposed tryptophan [13]. Another example of knowledge-based implementation is the use of rotamer libraries to model the side chain conformations of amino acids. Based on solved protein structures, residue side chains are observed to prefer a limited set of conformations [14]. The backbone-dependent probabilities for each rotamer can be converted to pseudo-energies, to bias residues towards adopting lower-energy conformations in the design.

The major physics-based energy function components include: **(1)** Lennard-Jones potential, **(2)** orientation-dependent hydrogen bond potential, **(3)** electrostatic interactions, and **(4)** an implicit solvation model [9, 15]. Van der Waals interactions, which are measured by Lennard-Jones potential, favor close packing of atoms that is constrained by sterics at short distances, and are thus necessary to model the packing of the protein core. Second, modeling of the costs of solvation/ desolvation is critical for recapitulating the hydrophobic effect. Because it is prohibitively expensive to explicitly model the interactions between amino acids and individual water molecules, the surrounding solvent is generally approximated as a continuous medium in an implicit solvation model, which is based on the accessible surface areas of atoms and empirically derived vapor-to-water free energies of transfer of amino acid side chains [15, 16]. Balancing out the implicit solvation model are the hydrogen bond potentials and electrostatic interaction energies, which lessen the penalty for buried hydrophilic residues if they are able to form stabilizing hydrogen bonds or salt bridges. Because naturally occurring protein sequences are generally expected to reside in local energy minima, energy functions can be optimized by parameterizing them to reflect this [17].

LANDMARKS AND PROGRESS IN COMPUTATIONAL PROTEIN DESIGN

Although successful design of novel proteins was reported as early as the 1970s, early designs were based on a qualitative understanding of protein biochemistry, and mainly consisted of stitching together simple, modular secondary structures such as α-helices [18] and β-sheets [19], or hydrophobic patterning [20]. In the 1990s, the advent of powerful computers, as well as the rapid expansion of structural databases from which improved force fields for modeling interaction energies could be derived, enabled the development of the first structure-based computational protein design softwares. In 1997, the Mayo group described the first fully automated algorithm for *de novo* protein design, and used it to design a sequence to fit the ββα motif found in zinc finger domains [13]. In 2003, RosettaDesign was used to engineer Top7, a 93-residue α/β protein with a completely novel protein fold [3]. In both cases, the computational designs were experimentally tested, and structural characterization showed that the synthesized protein sequences folded into compact, stable structures that were highly similar to the design model, thus validating the computational methodologies.

Since these landmark proof-of-concept successes, the field of computational design has focused on redesign and *de novo* design of proteins to perform a diverse range of functions, including catalysis, binding of other proteins or small molecules, and assembly into multimeric structures. Examples of particular interest are noted here. In 2008, a pioneering success in *de novo* enzyme design was achieved when Rosetta was used to generate the first ever retro-aldolase [21], for which there is no natural counterpart. *De novo* enzymes to carry out Kemp elimination and Diels-Alder reactions followed in quick succession [4, 22]. Enzyme redesign also achieved notable successes, including the transplantation of new catalytic activity into a crotonase by replacing two key amino acids in the active site and mutationally stabilizing the surrounding scaffold [23]. Generalizing their inside-out approach for enzyme design to the design of protein binding affinity, the Baker group has more recently described the engineering of a protein against a conserved cluster of surface residues on the stem of H1N1 influenza hemagglutinin (HA), which plays a critical role in viral entry of host cells [24]. After affinity maturation, the designed antiviral protein bound (HA) with low nanomolar affinity, and was able to inhibit fusogenic conformational changes in the target, showing great promise for therapeutic application. An inverse problem to engineering binding affinity is engineering binding specificity, or the ability to bind tightly to one target, but not structurally similar targets. The Keating group has done extensive work in developing CLASSY, a computational framework for engineering binding specificity, by converting a structure-based interaction model into a sequence-based scoring function, thus enabling the incorporation of penalties against interactions between the designed protein and undesired targets during optimization of the design-target interaction [25]. As proof of concept, CLASSY was used to tailor the peptide-binding specificities of multiple members of the bZIP transcription factor family, and the authors estimated that it would be possible to engineer >1,900 unique interaction profiles. Finally, an example of designing proteins to self-assemble into a desired oligomeric structure was the Baker group's construction of 12-mer and 24-mer nanocages by symmetrically docking naturally trimeric protein subunits together to identify an optimal packing framework, then redesigning the interfaces between the subunits to minimize the energy of self-assembly [26].

In addition to successes achieved in designing new proteins, recent years have also seen advances in protein design algorithms and softwares, although ORBIT and RosettaDesign remain widely used. One example is the Keating group's CLASSY. The Donald group at Duke University has also developed OSPREY (open-source protein redesign for you), a new suite of protein design programs that incorporates the use of continuous rotamers, as well as explicitly incorporating continuous protein backbone flexibility [27, 28]. To compensate for the increase in computational cost that results from adding degrees of freedom to the protein model, OSPREY uses improved DEE algorithms to enhance the efficiency of searching the larger sequence space.

CURRENT LIMITATIONS AND POTENTIAL IMPROVEMENTS

Despite the great progress that has been made in computational protein design since its birth, there still remain a number of limitations in currently used algorithms that prevent wider use and greater success. This is most obviously reflected by the fact that, in many cases, experimental optimization subsequent to computational design still often leads to dramatic improvements in stability and function, and sometimes, directed evolution is necessary to improve designed proteins even to the point where they are comparable to natural proteins. For example, in the case of the *de novo* Kemp eliminase, while it was remarkable that the computationally designed enzyme was able to achieve unprecedented functionality, subsequent directed evolution *in vitro* further improved the catalytic efficiency of the designed enzyme by >200-fold by incorporating as few as 4 additional mutations [4]. In another example, retrospective examination of an inactive first-generation design using molecular dynamics (MD) and x-ray crystallography highlighted several problems with the design. Essentially, the computational design was not borne out in real life – the designed active site side chains were highly mobile and did not maintain the necessary catalytic geometry upon exposure to solvent. Furthermore, stable alternate conformations competed with the catalytically active one. Based on these findings, a second round of design was initiated that yielded multiple active mutants with catalytic efficiencies comparable to those generated via directed evolution [29].

The work that has been done to examine why unsuccessful designs are unsuccessful reveals weaknesses in the current force fields used to model designs, most clearly in the modeling of long-range electrostatic interactions and the use of crude implicit solvation models that are less accurate than explicit models for the interaction of designed proteins with surrounding solvent molecules. However, in order for these issues with the energy function to be properly resolved, significantly greater computational power is needed, and this is a limiting step in the refinement of design algorithms. At present, the most successful designs will likely be achieved by using computational design to generate a weakly functional protein that can then be subjected to time- and labor-intensive optimization methods, such as directed evolution and molecular dynamics.

Rather than modularly applying computational and experimental approaches, however, it may be possible to integrate the two to greater effect. In 2006, Saraf *et al*. described the development of a new computational protein redesign algorithm called IPRO, which iteratively optimizes designs by identifying and propagating mutations that are found to improve computational design stability, affinity, etc. [30]. Essentially, IPRO serves as an *in silico* implementation of directed evolution. Since then, advances in DNA synthesis and next-generation sequencing techniques have made library construction and sequencing dramatically faster and cheaper, enabling deep sequencing of libraries after each round of evolution. This suggests the possibility of merging directed evolution and IPRO, as opposed to the current approaches that strictly alternate between rounds of *in silico* design and *in vitro* evolution [31]. After an initial design has been computed, a library of sequences can be generated *in vitro* from this template and subjected to directed evolution. Deep sequencing between rounds of evolution will allow parsing out and tracking of advantageous vs. deleterious vs. neutral hitchhiker mutations without the need for laborious and time-consuming experimental characterization of large numbers of mutants in between each round. Then, this information can be fed back into the computational design algorithm to construct the next-generation design. By reducing the time and labor involved in *in vitro* testing, while still functionally yielding the same information, deep sequencing combined with computational design should also enable more efficient, targeted scanning of the candidate sequence space.

Another interesting development in protein biochemistry that may be applied to improve computational design algorithms is the Ranganathan group's investigation of putative protein "sectors," or co-evolving groups of amino acids that are contiguous in structural space but not necessarily sequence space, that can be correlated to specific protein phenotypes [32]. The thought is that statistically significant covariance between residues likely indicates that these residues are part of a physically connected network that plays a distinct functional role. Similar to how sequence similarity search engines use position-specific residue probabilities derived from multiple sequence alignments of known sequences, in the case of protein redesign, especially redesign of binding or enzymatic activity, it may be useful to perform statistical coupling analysis (SCA) to identify residues that co-vary across the target and proteins known to bind to that target. Co-variance between residues in this case would likely correspond to sites of physical interaction between the target and binder, and this information could be used to fix or constrain these residues as pseudo-anchors when designing the new sequence, with optimization focusing on surrounding residues that may improve the packing of the interface or stabilize long-range electrostatic interactions.

Finally, another way in which current design methodologies might be improved is the incorporation of explicit negative design. As the studies on the inactive Kemp eliminase showed, it is not necessarily sufficient to try to minimize the energy of the desired conformation. Rather, deliberate destabilization of alternate conformations that are energetically close to the target structure may be required to eliminate nonspecificity or competitive nonproductive interactions. In other cases, rather than negative design, multi-state optimization may be needed to design enzymes that catalyze reactions involving multiple transition states. While stabilization of a single transition state may be sufficient to allow enzymatic catalysis to proceed, optimizing the design to promote stabilization of multiple states may lead to computational designs with greater catalytic activity prior to directed evolution.

The field of computational protein design has seen phenomenal advances in the mere 1-2 decades since its birth. As design algorithms are further refined and computing power continues to increase, the number, functional activity, and structural fidelity of successful designs will likely increase as well. In the future, we can anticipate that increasingly intertwined *in silico* and experimental approaches will enable deeper, more systematic probing into the fundamental biochemical principles driving processes such as protein folding and catalysis, while also allowing ever bolder forays into the awesome, vast protein space uninhabited by nature and undiscovered by science.

REFERENCES

[1] Malisi C, Schumann M, Toussaint NC, *et al*. Binding pocket optimization by computational protein design. *PLoS One* 2012; **7**(12): e52505.

[2] McAllister KA, Zou H, Cochran FV, *et al*. Using α-helical coiled-coils to design nanostructured metalloporphyrin arrays. *JACS* 2008, **130**(36): 11921-11927.

[3] Kuhlman B, Dantas G, Ireton GC, *et al*. Design of a novel globular protein fold with atomic-level accuracy. *Science* 2003; **302**(5649): 1364-1368.

[4] Röthlisberger D, Khersonsky O, Wollacott AM, *et al*. Kemp elimination catalysts by computational enzyme design. *Nature* 2008; **453**: 190-195.

[5] Chou PY and Fasman GD. Conformational parameters for amino acids in helical, β-sheet, and random coil regions calculated from proteins. *Biochemistry* 1974; **13**(2): 211-222.

[6] Richardson JS and Richardson DC. Amino acid preferences for specific locations at the ends of alpha helices. *Science* 1988; **240**(4859): 1648-1652.

[7] Engel DE and Degrado WF. Amino acid propensities are position-dependent throughout the length of alpha-helices. *J Mol Biol* 2004; **337**(5): 1195-1205.

[8] Regan L and Degrado WF. Characterization of a helical protein designed from first principles. *Science* 1988; **241**(4868): 976-978.

[9] Dantas G, Kuhlman B, Callender D, *et al*. A large scale test of computational protein design: folding and stability of nine completely redesigned globular proteins. *J Mol Biol* 2003; **332**(2): 449-460.

[10] Bolon DN and Mayo SL. Enzyme-like proteins by computational design. *PNAS* 2001; **98**: 14274-14279.

[11] Goldstein RF. Efficient rotamer elimination applied to protein side-chains and related spin glasses. *Biophys J* 1994; **66**(5): 1335-1340.

[12] Looger LL and Hellinga HW. Generalized dead-end elimination algorithms make large-scale protein side-chain structure prediction tractable: implications for protein design and structural genomics. *J Mol Biol* 2001; **307**(1): 429-445.

[13] Dahiyat BI and Mayo SL. *De novo* protein design: fully automated sequence selection. *Science* 1997; **278**(5335): 82-87.

[14] Dunbrack RL and Cohen FE. Bayesian statistical analysis of protein side-chain rotamer preferences. *Protein Sci* 1997; **6**(8): 1661-1681.

[15] Gordon DB, Marshall SA, and Mayo SL. Energy functions for protein design. *Curr Opin Struct Biol* 1999; **9**: 509-513.

[16] Wesson L and Eisenberg D. Atomic solvation parameters applied to molecular dynamics of proteins in solution. *Protein Sci* 1992; **1**(2): 227-235.

[17] Kuhlman B and Baker D. Native protein sequences are close to optimal for their structures. *PNAS* 2000; **97**(19): 10383-10388.

[18] Eisenberg D, Wilcox W, Eshita SM, *et al*. The design, synthesis, and crystallization of an alpha-helical peptide. *Proteins* 1986; **1**(1): 16-22.

[19] Moser R, Thomas RM, and Gutte B. An artificial crystalline DDT-binding polypeptide. *FEBS Letters* 1983; **2**(4): 247-251.

[20] Kamtekar S, Schiffer JM, Xiong H, *et al*. Protein design by binary patterning of polar and nonpolar amino acids. *Science* 1993; **262**: 1680-1685.

[21] Jiang L, Althoff EA, Clemente FR, *et al*. *De novo* computational design of retro-aldol enzymes. *Science* 2008; **319**(5868): 1387-1391.

[22] Siegel JB, Zanghellini A, Lovick HM, *et al*. Computational design of an enzyme catalyst for a stereoselective bimolecular Diels-Alder reaction. *Science* 2010; **329**(5989): 309-313.

[23] Xiang H, Luo L, Taylor KL, *et al*. Interchange of catalytic activity within the 2-enoyl-coenzyme A hydratase/isomerase superfamily based on a common active site template. *Biochemistry* 1999; **38**: 7638-7652.

[24] Fleishman SJ, Whitehead TA, Ekiert DC, *et al*. Computational design of proteins targeting the conserved stem region of influenza hemagglutinin. *Science* 2011; **332**: 816-821.

[25] Grigoryan G, Reinke AW, and Keating AE. Design of protein-interaction specificity gives selective bZIP-binding proteins. *Nature* 2009; **458**: 859-864.

[26] King NP, Sheffler W, Sawaya MR, *et al*. Computational design of self-assembling protein nanomaterials with atomic level accuracy. *Science* 2012; **336**(6085): 1171-1174.

[27] Gainza P, Roberts KE, Georgiev I, *et al*. OSPREY: protein design with ensembles, flexibility, and provable algorithms. *Methods Enzymol* 2013; **523**: 87-107.

[28] Gainza P, Roberts KE, and Donald BR. Protein design using continuous rotamers. *PLoS Comp Biol* 2012; **8**(1): e1002335.

[29] Privett HK, Kiss G, Lee TM, *et al*. Iterative approach to computational enyzme design. *PNAS* 2012; **109**(10): 3790-3795.

[30] Saraf MC, Moore GL, Goodey NM, *et al*. IPRO: an iterative computational protein library redesign and optimization procedure. *Biophys J* 2006; **90**(11): 4167-4180.

[31] Street AG and Mayo SL. Computational protein design. *Structure* 1999; **7**(5): R105-109.

[32] Halabi N, Rivoire O, Leibler S, *et al*. Protein sectors: evolutionary units of three-dimensional structure. *Cell* 2009; **138**(4): 774-786.