Chapter 1

Optimization Strategies for the Design of Protein Sequences and Combinatorial Mutation Libraries

Some language in this chapter was adapted from manuscripts coauthored with Christina L. Vizcarra, Oscar Alvizo, and Stephen L. Mayo.

Alvizo, O.; Allen, B. D.; Mayo, S. L., Computational protein design promises to revolutionize protein engineering. *Biotechniques* **2007,** *42* (1), 31–39.

Vizcarra, C. L.; Allen, B. D.; Mayo, S. L., Progress and challenges in computational protein design. Submitted **2008**.

High-throughput protein engineering

Proteins are linear heteropolymers, built from 20 standard amino acid monomers. Synthesized inside cells, they perform a vast majority of the structural, catalytic, sensory, and regulatory functions that characterize living systems as we understand them today. These myriad roles are made possible by the ability of proteins to self-assemble into well-defined structures specified by their amino acid sequences. Although only a small fraction of all possible protein sequences can assume a folded, functional form,[1, 2] the modular nature of the protein platform allows existing functions to be altered and enhanced for new or fluctuating requirements by selection for fitness from a heterogeneous population. The requisite diversity develops though copying errors and recombination of nucleic acid sequences that encode the proteins expressed by members of the population. This process is slow and relies on serendipity to discover beneficial variants. Nevertheless, it represents the only plausible description of how the complexity of life we see today could have ultimately arisen from simpler chemical systems.

Before sufficiently powerful tools for genetic manipulation were even available, it was postulated that existing biological systems for protein fabrication could be harnessed to produce nano-scale molecular machines with designed functions.[3] Major successes along these lines have been achieved via directed evolution, in which screening or *in vivo* selection is applied to isolate active variants from populations of 10^0 to 10^{15} members. Molecular diversity for directed evolution can be produced, *in vitro* or *in vivo*, using methods including error-prone DNA polymerization, recombination, gene shuffling, combinatorial libraries made using synthetic oligonucleotides, host organism mutator strains, and the humoral immune system.[4, 5] Active variants can then be isolated from

these pools with techniques such as screening with various levels of automation, selection for survival in auxotrophic strains, and affinity-based or cytometric separation of individual proteins linked to or compartmentalized with the nucleic acids that encode them.[4, 5] Notable protein engineering feats facilitated by directed evolution include enhancement of the substrate specificity, thermostability, selectivity, and solvent tolerance of enzymes,[6–9] as well as the engineering of metabolic pathsteelyways.[10]

Despite the potential of laboratory evolution to cull through vast numbers of sequences, experimental concerns, such a lack of appropriate high-throughput assays, selection systems, or instrumentation, can dramatically limit the diversity that may practically be addressed. Furthermore, the largest libraries that could conceivably be approached by experimental methods are miniscule compared to the total possible diversity of even modestly sized proteins. The inherent limitations of experimental protein engineering methods, and the hope that sequence design might eventually be completely automated, have motivated the development of computational tools for the virtual screening of astronomical pools of diversity. In some cases, the use of computational protein design (CPD) methods has allowed stable, well-folded, and functional proteins variants with many mutations away from their wild-type counterparts to be designed directly *in silico* and experimentally validated. In most situations, however, the successful design of proteins using CPD alone has been inconsistent.

This thesis describes the conception, implementation, and testing of optimization methods meant to improve the frequency with which useful and interesting sequences can be predicted by computational techniques. The initial goal was to facilitate the discovery of sequences of minimum energy given a standard model of computational protein

design. Such sequences are desirable to the extent that the design model is accurate, and are expected to help suggest possible improvements when it is not. However, the enhanced optimization methods also proved to be directly applicable to more sophisticated design models that treat multiple conformational states simultaneously. In order to better assess the utility of these more realistic methods, a method was developed to allow the design of combinatorial mutation libraries to be driven by the results of computational protein design calculations in a model-independent manner. The synergistic application of the methods described here enabled the first experimental test of computational design based on large structural ensembles.

Computational protein design by inverse folding

CPD was first conceived as the inverse of the protein-folding problem, since its most basic goal is to generate amino acid sequences that preferentially adopt a specific three-dimensional structure.[11] At its core, the inverse-folding design model consists of a search for optimal amino acid side chains at one or more positions in a fixed structural model of the protein main chain. Various amino acid types are modeled at each designed position, and potential mutations are evaluated based on their pairwise interaction energies. The continuous flexibility available to each amino acid side chain is approximated using a discrete set of low-energy conformations called rotamers.[12, 13] The goal is thus to find an optimal choice of rotamer, of any allowed amino acid type, at each designed position. A configuration of the virtual protein system described by the inverse folding model corresponds to a set of atomic coordinates for some particular amino acid sequence. To the extent that the approximations inherent in the scoring functions and

design model are appropriate, amino acid sequences specified by low-energy configurations of the system are expected to stabilize the required fold and thereby facilitate the desired activity.

Enthusiasm for computational protein design by inverse folding was piqued when an algorithm for the generation of provably optimal solutions was reported. This method, based on the dead-end elimination (DEE) theorem, specifies criteria by which rotamers at particular positions can be definitively excluded from the global minimum energy configuration (GMEC) of the system.[14] The availability of a rigorous framework for combinatorial optimization in the inverse folding model promoted confidence that protein design was computationally tractable. Accordingly, many of the initial successes in CPD were achieved via DEE-based optimization methods.[15–19] A comparison of DEE with stochastic optimization routines such as genetic algorithms (GA) and Monte Carlo with simulated annealing (MC) indicated that the inexact routines often failed to find GMEC solutions when applied with equivalent computational effort.[20] During this time, DEE-based methods were improved with a number of additional elimination strategies and heuristics that rendered them amenable to CPD problems with more variable positions and more rotamers per position.[21]

Despite these advances, the poor performance scaling of DEE has caused the field to shift toward stochastic optimization methods such as MC[22–24] and FASTER[25] so that larger and more complex designs, such as those involving enzyme substrates and transition state models, could be attempted.[23, 26–30] Rather than eliminate particular rotamers from consideration until only a single sequence remains, these stochastic methods sample sequence space by choosing perturbations to make to a fully instantiated

configuration of the system, and accepting or rejecting the perturbations based on their energetic consequences. Although these methods do not produce solutions that are provably optimal in a global sense, they can be relied upon to find local minima quickly, and their running times can be extended for as long as desired in an attempt to improve existing solutions.

In Chapter 2, I describe enhancements to the FASTER optimization procedure that dramatically improve its ability to converge to a single lowest-energy solution. In every case tested, this FASTER-based solution was either identical to the solution produced by DEE, or was the lowest-energy solution ever found (by any method) if DEE was not able to converge. In some cases, these FASTER-derived optimal solutions could not be found with extremely long runs of MC, suggesting that the improved FASTER procedure is preferable to currently available alternatives. Our experiences with FASTER strongly indicate that the combinatorial optimization problem for design by inverse folding is essentially solved: FASTER is able to quickly converge to low-energy solutions for any problem that could be meaningfully addressed with pure inverse folding simulations.

Beyond single-state inverse folding: multi-state design

The inverse-folding design model has made practical the *in silico* screening of more than 10^{200} amino acid sequences in a single design. However, it also presents significant challenges to the development of atomic scoring functions that accurately predict the fitness of particular sequences. Although initial explorations of CPD evaluated interactions between rotamers with energy functions such as Leonard-Jones

and bonded-term potentials used in molecular dynamics simulations,[31, 32] researchers soon introduced significant complications to the scoring model in an effort to make it applicable to a broader range of design goals. Some of these supplementary energy terms, such as orientation-dependent hydrogen bonding functions and implicit solvation models, have clear physical justifications and are generally accepted in the wider protein simulation community.[23, 33–36] Others, such as penalties for the exposure of nonpolar groups and unfolded state energies based on amino acid composition,[23, 37] are seldom used outside the realm of CPD. These heuristic negative design terms have been adopted primarily to solve problems peculiar to the comparison of different sequences by their molecular mechanics energies, and to overcome the rigidity of the inverse-folding design model.

Ultimately, the viability of any particular sequence depends on the degree to which it populates an entire ensemble of conformational states, including active/native states, misfolded and unfolded states, and aggregated states. While native states can be understood through high-resolution structures derived from experiment, general and tractable atomic-resolution models of alternate conformational states have not yet been developed. Implicit negative design terms like those mentioned above help the design procedure to assess how potential amino acid substitutions might affect the tendency of a sequence to assume poorly defined nonnative states. Nevertheless, these terms are often insufficient to allow selection of reasonable sequences in the context of large designs. For example, RosettaDesign, a CPD procedure based on a highly parameterized forcefield with many heuristic and implicit negative design terms, cannot effectively produce reasonable amino acid compositions when applied to the surfaces of β-sheets.[23]

For this reason, hydrophobic amino acids had to be excluded from β-sheet surface positions during the design of top7, a sequence successfully designed to assume a novel fold.[23]

The inverse-folding paradigm can be extended to allow issues in negative design to be explored in a more systematic manner and offer an alternative to the heuristic terms discussed above. Most notably, Harbury and coworkers have applied explicit negative design algorithms to directly engineer specificity into coiled-coil systems and recapitulate sequences that bind small-molecule ligands with high affinity.[38, 39] In each case, the explicit modeling of alternate states, such as undesired associated and unbound states in the coiled-coil case, and unbound and unfolded states in the ligand-binding case, was crucial for the computational design of variants exhibiting the desired functional properties. Computational multi-state design (MSD) procedures can also be used in a purely positive-design sense to find sequences able to assume several distinct conformations. For example, Ambroggio and Kuhlman used MSD to design a protein switch that assumes completely different folds and association states in the presence versus the absence of zinc.[40]

Given that specificity is crucial to the proper folding, stability, solubility, and activity of proteins, it would seem natural for explicit multi-state modeling to be applied frequently to structure-based computational protein design. Surprisingly, experience with MSD in the CPD community consists essentially of only those investigations just mentioned. Several complications have thus far limited the general utility of MSD in computational protein design.

The first major impediment to the application of MSD in CPD relates to how the various relevant states are specified. In cases with well-characterized target and competing states, one can use high-resolution experimental structures to model each of the desired states. For example, competition between homodimer and heterodimer coiled coils has been modeled by threading the relevant sequences onto identical main chain models from a crystal structure.[39] Similarly, the unbound state in a ligand-binding system was modeled by removing the ligand from a crystal structure of the bound complex.[38] Two different crystal structures were used to model the two target states in the molecular switch design.[40] In general, the astronomical range of conformations available to a sequence must be approximated with a much smaller, computationally tractable set devised to represent the entire ensemble. Unfortunately, methods for the construction of general and accurate models for some important alternative states are not yet available. Although aggregated and unfolded states have been treated in MSD using native conformations in low-dielectric media and random chain ensembles, respectively,[38, 39] the degree to which these simplified models can realistically capture the relevant properties of these states is unclear. Because unfolded and aggregated ensembles are likely quite varied and diverse, explicit treatment of them in MSD will require accurate and efficient methods to generate structural models that adequately represent them, as well as powerful MSD optimization methods that can efficiently sample sequence space given a large number of states.

The second major problem in the general application of MSD relates to how sequences should be evaluated given an ensemble of structural states. Because a single amino acid sequence can assume completely different conformations in each relevant

state, MSD methods require a two-level optimization procedure in which an outer routine samples amino acid sequences and an inner routine evaluates the energy of a sequence in the context of each state separately by rotamer optimization. The energies that result from these individual rotamer optimization calculations must be combined to yield a single fitness score that can be used to evaluate the sequence. No consensus has yet been reached on what energy combination function should be used for this purpose; in fact, different functions may be appropriate for different design goals. One attractive approach is to assess fitness according to the probability, P, that one of the desired target conformations would be fulfilled. P can be computed using basic statistical mechanics, given a finite set of desired target states, undesired competing states, and their energies:

$$P = \sum_{i \in S_T} e^{-E_i / RT} \Big/ \sum_{i \in (S_T \cup S_C)} e^{-E_i / RT} \qquad (1)$$

where S_T is the set of target states, S_C is the set of competing states, and E_i is the energy of state i.[38, 39]

This strategy cannot be used when competing states are not explicitly modeled, because the probability computed with equation 1 would always be unity in this case. When competing states are not considered, one possibility would be to simply average or sum the energies of a sequence on each state.[40] This is appropriate when the design goals require that all specified states be satisfied, as in the design of a protein switch; however, biases can arise if the magnitudes of the energies in different states are significantly different. One could also evaluate the fitness of a sequence by computing the free energy, A, of the system based on all modeled states and their energies:

$$A = -kT \log(\sum_i e^{-E_i / kT}) \qquad (2)$$

This strategy is applicable when the target state ensemble consists of similar structures intended to approximate realistic conformational flexibility, and the incompatibility of a sequence with a small fraction of the available states is relatively inconsequential. MSD scoring schemes like those based on equations 1 and 2 are expected to provide better accuracy as the number of modeled states increases. Because each individual state relies on energy calculations in the context of a rigid main chain, atomic clashes in a few states can unrealistically effect sequence selection when the total number of modeled states is small.

The final major issue in the wider adoption of multi-state design is simply that it presents a more taxing optimization problem than standard single-state design (SSD). The greater difficulty arises because a single amino acid sequence might assume completely different conformations in each relevant MSD state. This prohibits the amino-acid-ignorant rotamer optimization strategies that accelerate convergence in single state design, and requires the two-level optimization procedure described above. Because MSD must perform what essentially amounts to multiple small, independent design calculations in order to assess the fitness of a single amino acid sequence, the diversity of sequences that may be effectively sampled in MSD is dramatically limited relative to SSD. Furthermore, whereas SSD sampling in the inverse folding model is made significantly more efficient by precomputing all possible pairwise energies between rotamers at different positions and using this energy matrix as a lookup table during rotamer optimization, current limitations on physical memory render simple adaptations of this strategy untenable for MSD problems with more than a few states. Unfortunately, as discussed above, issues with the specification of representative conformational states

and the aggregation of state energies into a single fitness score should be better ameliorated when the total number of modeled states increases. Thus, both technical and scientific concerns necessitate more sophisticated and powerful optimization methodologies for acceptable sampling performance in MSD to be achieved. Although DEE-based methods have begun to be adapted to MSD problems,[41] our experiences with SSD suggest that such methods will not provide a "silver bullet" for MSD.

In Chapter 3, I present an optimization framework for multi-state CPD that can easily handle hundreds of states, and whose running time scales linearly with the number of states that are treated. Furthermore, I describe the development of an MSD-capable version of the FASTER optimization algorithm within this framework. The test calculations I report indicate that MSD-FASTER offers significant performance enhancements compared to an MSD-enabled implementation of Monte Carlo with simulated annealing (MSD-MC), that MSD-FASTER finds low-energy sequences more quickly, and that, in some cases, the lowest-energy sequences found by MSD-FASTER cannot necessarily be found at all by MSD-MC during a sampling run of reasonable length. The simulation tools developed in Chapter 3 provide a robust framework on which to base future investigations of ensemble design, explicit negative design, and new atomic-resolution models of unfolded, misfolded, and aggregated states in CPD.

MSD might be used to help overcome the inaccuracies inherent to the application of inverse folding to a single, fixed, main-chain structure. By designing sequences to satisfy an ensemble of related main-chain conformations, a MSD procedure can account, at least partially, for both the tendency of real proteins to relax in order to accommodate mutations, and the contribution of conformational entropy to protein stability. The most

obvious sources of input structural data for this purpose are nuclear magnetic resonance (NMR) experiments, for which results are widely available in the protein data bank (PDB), and molecular dynamics (MD) simulations starting from crystallographic conformations, which can be performed using a variety of accessible commercial and open-source software packages.

In Chapter 4, I describe the computational design and experimental stability assessment of several combinatorial libraries based on different sources of input structural information for the same protein. The input models include a crystal structure, an NMR ensemble, a constrained, minimized average NMR structure, and constrained and unconstrained MD ensembles. Experimental analysis of these libraries indicates that the use of an MD ensemble may help to mitigate design failures that occur due to energy function inaccuracies and the approximations of conformational discretization, but also that care must be taken in constructing an ensemble to use for this purpose.

Beyond pure computational protein design: library design

Approximations in the molecular mechanics and heuristic energy functions used in CPD, a lack of accurate structural models for all the relevant conformational states, incomplete sequence and conformational sampling, and failures to model dynamics and chemical transformations all contribute to render extremely challenging the direct *in silico* design of functional proteins. Towards this goal, progress in algorithms, physical chemical models, and computing hardware must be coupled with the frequent and rigorous comparison of computational predictions with experimental reality. Furthermore, continuing development of CPD will not be sustained without evidence that

current methods can facilitate or expedite real-world protein engineering efforts. For these reasons, recent investigations in the field have begun to focus on the synergistic coupling of CPD calculations with experimental screening and selection methods developed for use in directed evolution.

The results of protein design simulations have been used to help determine particular residues that might be especially amenable to site-saturation mutagenesis or site-directed recombination,[42–44] and have facilitated the creation of combinatorial mutation libraries.[45–47] Given appropriate laboratory automation hardware, lists of CPD-derived sequences can also be individually encoded, expressed, and assayed in high-throughput fashion.[48] Laboratory evolution procedures have also been applied to improve the lower levels of activity found in *de novo* computationally designed enzymes.[28]

For the purposes of validating and improving CPD, library design methods that maintain a closer relationship between the sequences actually tested and the sequences produced by the calculations are preferred. Thus, it might seem that simply constructing the top *n* sequences produced by a design calculation would be ideal in this case. However, practical considerations often prohibit this strategy. Few academic researchers have the resources necessary to construct and test more than tens of individual sequences for a given design problem. Furthermore, the availability of an efficient high-throughput screen or selection vastly increases the diversity that can be assayed far beyond what would be possible through gene assembly of individual sequences at any cost. In these cases, a designed combinatorial gene library can provide a more appropriate match, because libraries with arbitrary numbers of members can be synthesized economically and easily, even without laboratory automation.

Although several reported methods allow the results of CPD calculations to drive the design of combinatorial mutation libraries,[45, 47, 49] each suffers from several drawbacks that limit its generality or reduce the clarity with which the libraries it produces reflect on the predictions of the original design calculations. In Chapter 4, I describe the development and implementation of a new algorithm for the computational design of combinatorial mutation libraries based on arbitrary lists of scored amino acid sequences, such as those generated by CPD. In contrast to any competing method suggested so far, this method fulfills all of the following desired qualities: (1) it considers CPD energies explicitly; (2) it allows the user to directly specify the range of viable library sizes; (3) it allows complete control over which sets of amino acids can be considered; (4) it does not rely on heuristics to reduce the computational complexity of the problem by eliminating potentially viable libraries. This combinatorial library design algorithm was used to generate the sets of sequences that we tested for each of the designs based on different sources of structural information as described in Chapter 4.

Our results indicate that this method allows CPD to extend directly to the design of combinatorial libraries that exhibit a high proportion of stable, well-folded members. In addition to validating the new library design method, our results provide a stronger basis on which to recommend library design than was allowed by previous reports, which focused on larger libraries and displayed less obvious connections between the contributions of the computational design and the experimental results.

Conclusions

The work described here illustrates how the development of enhanced sampling and optimization procedures can crucially aid the progress of method refinement and improvement in CPD. The discovery of more efficient optimization procedures, originally intended for single-state design, prompted their application to multi-state design methods that allow many conformational states to be modeled simultaneously. The availability of these MSD methods and a general procedure for the automated design of combinatorial mutation libraries together allowed an investigation of the dependence of design results on the type and quality of input structural data. The results of these experiments provide important clues about how CPD methodology improvements should proceed. As CPD simulations become more realistic, we expect the development of more efficient sampling methods to become more central to the success of CPD, and energy function development to become less so. As more aspects of protein structure and stability begin to be modeled explicitly, the implicit and heuristic negative design terms intended to account for them can be discarded. With additional advances in computational power, conformational sampling methods, multi-state design sequence optimization algorithms, and general representations of alternate states, the set of theoretically defensible energy functions used in other types of protein simulation may one day be sufficient for the accurate computational design of protein sequences.

References

1. Keefe, A. D.; Szostak, J. W., Functional proteins from a random-sequence library. *Nature* **2001,** *410* (6829), 715–718.

2. Taylor, S. V.; Walter, K. U.; Kast, P.; Hilvert, D., Searching sequence space for protein catalysts. *Proceedings of the National Academy of Sciences of the United States of America* **2001,** *98* (19), 10596–10601.

3. Drexler, K. E., Molecular Engineering — an Approach to the Development of General Capabilities for Molecular Manipulation. *Proceedings of the National Academy of Sciences of the United States of America* **1981,** *78* (9), 5275–5278.

4. Bershtein, S.; Tawfik, D. S., Advances in laboratory evolution of enzymes. *Current Opinion in Chemical Biology* **2008,** *12* (2), 151–158.

5. Jackel, C.; Kast, P.; Hilvert, D., Protein design by directed evolution. *Annual Review of Biophysics* **2008,** *37*, 153–173.

6. Arnold, F. H., Combinatorial and computational challenges for biocatalyst design. *Nature* **2001,** *409* (6817), 253–257.

7. Hult, K.; Berglund, P., Engineered enzymes for improved organic synthesis. *Current Opinion in Biotechnology* **2003,** *14* (4), 395–400.

8. Krishna, S. H., Developments and trends in enzyme catalysis in nonconventional media. *Biotechnology Advances* **2002,** *20* (3–4), 239–267.

9. Turner, N. J., Directed evolution of enzymes for applied biocatalysis. *Trends in Biotechnology* **2003,** *21* (11), 474–478.

10. Chatterjee, R.; Yuan, L., Directed evolution of metabolic pathways. *Trends in Biotechnology* **2006,** *24* (1), 28–38.

11. Pabo, C., Molecular Technology — Designing Proteins and Peptides. *Nature* **1983,** *301* (5897), 200.

12. Dunbrack, R. L.; Cohen, F. E., Bayesian statistical analysis of protein side-chain rotamer preferences. *Protein Science* **1997,** *6* (8), 1661–1681.

13. Ponder, J. W.; Richards, F. M., Tertiary Templates for Proteins — Use of Packing Criteria in the Enumeration of Allowed Sequences for Different Structural Classes. *Journal of Molecular Biology* **1987,** *193* (4), 775–791.

14. Desmet, J.; Demaeyer, M.; Hazes, B.; Lasters, I., The Dead-End Elimination Theorem and Its Use in Protein Side-Chain Positioning. *Nature* **1992,** *356* (6369), 539–542.

15. Bolon, D. N.; Mayo, S. L., Enzyme-like proteins by computational design. *Proceedings of the National Academy of Sciences of the United States of America* **2001,** *98* (25), 14274–14279.

16. Dahiyat, B. I.; Mayo, S. L., De novo protein design: Fully automated sequence selection. *Science* **1997,** *278* (5335), 82–87.

17. Dwyer, M. A.; Looger, L. L.; Hellinga, H. W., Computational design of a Zn2+ receptor that controls bacterial gene expression. *Proceedings of the National Academy of Sciences of the United States of America* **2003,** *100* (20), 11255–11260.

18. Looger, L. L.; Dwyer, M. A.; Smith, J. J.; Hellinga, H. W., Computational design of receptor and sensor proteins with novel functions. *Nature* **2003,** *423* (6936), 185–190.

19. Malakauskas, S. M.; Mayo, S. L., Design, structure and stability of a hyperthermophilic protein variant. *Nature Structural Biology* **1998,** *5* (6), 470–475.

20. Voigt, C. A.; Gordon, D. B.; Mayo, S. L., Trading accuracy for speed: A quantitative comparison of search algorithms in protein sequence design. *Journal of Molecular Biology* **2000,** *299* (3), 789–803.

21. Gordon, D. B.; Hom, G. K.; Mayo, S. L.; Pierce, N. A., Exact rotamer optimization for protein design. *Journal of Computational Chemistry* **2003,** *24* (2), 232–243.

22. Kirkpatrick, S.; Gelatt, C. D.; Vecchi, M. P., Optimization by Simulated Annealing. *Science* **1983,** *220* (4598), 671–680.

23. Kuhlman, B.; Dantas, G.; Ireton, G. C.; Varani, G.; Stoddard, B. L.; Baker, D., Design of a novel globular protein fold with atomic-level accuracy. *Science* **2003,** *302* (5649), 1364–1368.

24. Metropolis, N.; Rosenbluth, A. W.; Rosenbluth, M. N.; Teller, A. H.; Teller, E., Equation of State Calculations by Fast Computing Machines. *Journal of Chemical Physics* **1953,** *21* (6), 1087–1092.

25. Desmet, J.; Spriet, J.; Lasters, I., Fast and accurate side-chain topology and energy refinement (FASTER) as a new method for protein structure optimization. *Proteins* **2002,** *48* (1), 31–43.

26. Jiang, L.; Althoff, E. A.; Clemente, F. R.; Doyle, L.; Rothlisberger, D.; Zanghellini, A.; Gallaher, J. L.; Betker, J. L.; Tanaka, F.; Barbas, C. F.; Hilvert, D.; Houk, K. N.; Stoddard, B. L.; Baker, D., De novo computational design of retro-aldol enzymes. *Science* **2008,** *319* (5868), 1387–1391.

27. Lassila, J. K.; Privett, H. K.; Allen, B. D.; Mayo, S. L., Combinatorial methods for small-molecule placement in computational enzyme design. *Proceedings of the National Academy of Sciences of the United States of America* **2006,** *103* (45), 16710–16715.

28. Rothlisberger, 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.; Baker, D., Kemp elimination catalysts by computational enzyme design. *Nature* **2008,** *453* (7192), 190–U4.

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

30. Shah, P. S.; Hom, G. K.; Ross, S. A.; Lassila, J. K.; Crowhurst, K. A.; Mayo, S. L., Full-sequence computational design and solution structure of a thermostable protein variant. *Journal of Molecular Biology* **2007,** *372* (1), 1–6.

31. Desjarlais, J. R.; Handel, T. M., De-Novo Design of the Hydrophobic Cores of Proteins. *Protein Science* **1995,** *4* (10), 2006–2018.

32. Hellinga, H. W.; Richards, F. M., Construction of New Ligand-Binding Sites in Proteins of Known Structure .1. Computer-Aided Modeling of Sites with Predefined Geometry. *Journal of Molecular Biology* **1991,** *222* (3), 763–785.

33. Dahiyat, B. I.; Gordon, D. B.; Mayo, S. L., Automated design of the surface positions of protein helices. *Protein Science* **1997,** *6* (6), 1333–1337.

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

35. Kortemme, T.; Morozov, A. V.; Baker, D., An orientation-dependent hydrogen bonding potential improves prediction of specificity and structure for proteins and protein-protein complexes. *Journal of Molecular Biology* **2003,** *326* (4), 1239–1259.

36. Lazaridis, T.; Karplus, M., Effective energy function for proteins in solution. *Proteins—Structure Function and Genetics* **1999,** *35* (2), 133–152.

37. Dahiyat, B. I.; Mayo, S. L., Probing the role of packing specificity in protein design. *Proceedings of the National Academy of Sciences of the United States of America* **1997,** *94* (19), 10172–10177.

38. Boas, F. E.; Harbury, P. B., Design of protein-ligand binding based on the molecular-mechanics energy model. *Journal of Molecular Biology* **2008,** *380* (2), 415–424.

39. Havranek, J. J.; Harbury, P. B., Automated design of specificity in molecular recognition. *Nature Structural Biology* **2003,** *10* (1), 45–52.

40. Ambroggio, X. I.; Kuhlman, B., Computational design of a single amino acid sequence that can switch between two distinct protein folds. *Journal of the American Chemical Society* **2006,** *128* (4), 1154–1161.

41. Yanover, C.; Fromer, M.; Shifman, J. M., Dead-end elimination for multistate protein design. *Journal of Computational Chemistry* **2007,** *28* (13), 2122–2129.

42. Endelman, J. B.; Silberg, J. J.; Wang, Z. G.; Arnold, F. H., Site-directed protein recombination as a shortest-path problem. *Protein Engineering Design & Selection* **2004,** *17* (7), 589–594.

43. Voigt, C. A.; Martinez, C.; Wang, Z. G.; Mayo, S. L.; Arnold, F. H., Protein building blocks preserved by recombination. *Nature Structural Biology* **2002,** *9* (7), 553–558.

44. Voigt, C. A.; Mayo, S. L.; Arnold, F. H.; Wang, Z. G., Computational method to reduce the search space for directed protein evolution. *Proceedings of the National Academy of Sciences of the United States of America* **2001,** *98* (7), 3778–3783.

45. Hayes, R. J.; Bentzien, J.; Ary, M. L.; Hwang, M. Y.; Jacinto, J. M.; Vielmetter, J.; Kundu, A.; Dahiyat, B. I., Combining computational and experimental screening for rapid optimization of protein properties. *Proceedings of the National Academy of Sciences of the United States of America* **2002,** *99* (25), 15926–15931.

46. Mena, M. A.; Treynor, T. P.; Mayo, S. L.; Daugherty, P. S., Blue fluorescent proteins with enhanced brightness and photostability from a structurally targeted library. *Nature Biotechnology* **2006,** *24* (12), 1569–1571.

47. Treynor, T. P.; Vizcarra, C. L.; Nedelcu, D.; Mayo, S. L., Computationally designed libraries of fluorescent proteins evaluated by preservation and diversity of function. *Proceedings of the National Academy of Sciences of the United States of America* **2007,** *104* (1), 48–53.

48. Cox, J. C.; Lape, J.; Sayed, M. A.; Hellinga, H. W., Protein fabrication automation. *Protein Science* **2007,** *16* (3), 379–390.

49. Mena, M. A.; Daugherty, P. S., Automated design of degenerate codon libraries. *Protein Engineering Design & Selection* **2005,** *18* (12), 559–561.