PROTEINS
STRUCTURE ■ FUNCTION ■ BIOINFORMATICS

Automating human intuition for protein design

Lucas G. Nivón,[1] Sinisa Bjelic,[1] Chris King,[1] and David Baker[1,2*]

[1] Department of Biochemistry, University of Washington, Seattle, Washington 98195

[2] Howard Hughes Medical Institute (HHMI), University of Washington, Seattle, Washington 98195

ABSTRACT

In the design of new enzymes and binding proteins, human intuition is often used to modify computationally designed amino acid sequences prior to experimental characterization. The manual sequence changes involve both reversions of amino acid mutations back to the identity present in the parent scaffold and the introduction of residues making additional interactions with the binding partner or backing up first shell interactions. Automation of this manual sequence refinement process would allow more systematic evaluation and considerably reduce the amount of human designer effort involved. Here we introduce a benchmark for evaluating the ability of automated methods to recapitulate the sequence changes made to computer-generated models by human designers, and use it to assess alternative computational methods. We find the best performance for a greedy one-position-at-a-time optimization protocol that utilizes metrics (such as shape complementarity) and local refinement methods too computationally expensive for global Monte Carlo (MC) sequence optimization. This protocol should be broadly useful for improving the stability and function of designed binding proteins.

Proteins 2013; 00:000–000.

Key words: algorithm; benchmark; enzyme; protein design; protein engineering; small-molecule binding.

INTRODUCTION

Computational protein design has been used to design proteins with new structures or functions. The new functions range from small-molecule binding to specific protein binding to catalytic activity.[1–4] The computational design of proteins that bind reaction transition state models, and ligands more generally, often starts from a set of naturally occurring protein scaffolds of known structure. It proceeds by first identifying placements of the ligand in the scaffolds and second, optimizing the surrounding residues for favorable interactions with the ligand without compromising the overall stability of the protein. The resultant designed proteins are usually inspected by a researcher and modified before they are experimentally tested. These modifications are based on human intuition about protein stability, aggregation, and binding interactions. Sequence changes far from, or facing away from, the designed site are often reverted, and larger residues substituted for smaller ones (very small clashes during fixed-backbone computations may disfavor larger residues, with better packing, from being selected). Automation of these human intervention steps is desirable for systematically optimizing the design process, for reducing the human time required for design, and more generally, for making protein design more broadly accessible.

Automation of a process requires a benchmark for evaluation of performance. Several types of benchmarks have previously been described for protein–small-molecule interaction modeling. These include small-molecule docking, prediction of small-molecule–protein interaction affinity,[5–7] and amino acid sequence recovery at natural protein–small-molecule interfaces.[8,9] The problem of how to alter the sequence of a naturally occurring protein to bind a new small-molecule is much more challenging, and not directly addressed by existing benchmarks. For example, it is necessary to consider whether an amino acid substitution that increases the apparent binding affinity for a new ligand overly compromises the stability of the protein scaffold.

Additional Supporting Information may be found in the online version of this article.

Nivón L.G. and Bjelic S. contributed equally to this work.

*Correspondence to: David Baker, Department of Biochemistry; Howard Hughes Medical Institute (HHMI), University of Washington, Seattle, WA 98195. E-mail: dabaker@u.washington.edu

Received 27 June 2013; Revised 25 September 2013; Accepted 21 October 2013
Published online 30 October 2013 in Wiley Online Library (wileyonlinelibrary.com).
DOI: 10.1002/prot.24463

To guide the automation of human intuition in the manual stages of protein design, we assembled a benchmark set of 51 proteins that tests the ability of a method to recapitulate mutation decisions made by human protein designers in realistic novel-design situations. We also developed a new local sequence optimization procedure that uses a greedy algorithm and allows multiple sampling methods to be carried out in serial using metrics too computationally expensive for global sequence design. We show that the new protocol improves on traditional design methods on the human designer benchmark. Monte Carlo (MC) based Rosetta design together with the novel greedy optimization provide a fully automated pipeline for computational design of enzymes and ligand-binding proteins with minimal human intervention.

MATERIALS AND METHODS

Match-design-order benchmark: human design interventions on Rosetta designed proteins

The match–design–order (MDO) benchmark consists of proteins gathered from our protein engineering efforts: design of a *de novo* Morita Baylis Hillman catalyst (MBH prefix)[10]; design of a phosphorylated-ester binding protein (1kux1 prefix; Nivón unpublished); design of a binding protein for digoxigenin (DIG prefix; Tinberg *et al.*)[4]; design of a *de novo* galactosidase (GA and GF prefixes; Bjelic unpublished); design of a binder for the fluorophore 3,5-difluoro-4-hydroxybenzylidene imidazolinone (DFHBI; MB prefix; Bick unpublished); design of a beta-lactamase (BL prefix; Khersonsky unpublished); and design of *de novo* chorismate mutase (dCM prefix; Richter unpublished). The MDO benchmark is available via the link: http://robetta.bakerlab.org/downloads/ligand_design_benchmarks/MDOBENCH/

Overall the design set differs from the native (match) set by a mean of 19.0 (SD 5.5) mutations (Fig. S1A, Supporting Information). The design set differs from the order (human modified) set by mean of 10.3 (SD 4.0) mutations (Fig. S1A, Supporting Information). Human designers place fewer mutations than Rosetta does, and have less variance in the number of mutations introduced.

The design and order set of structures differ by a total of 527 mutations, of which 62.4% (328) are reversions to the native sequence identity. The mutations in the order set are not strongly weighted toward hydrophobic or polar residues, with 22.4% going from hydrophobic in the design to polar in the order, and 25.2% going in the opposite direction. Mutations from the design to order set are slightly more likely to increase amino acid size (54.5%) rather than decrease it (45.5%). The most frequent type of change was a slight size increase of 10–20 Da, for example, adding a methyl group (Fig. S1B, Supporting Information, mass distribution). Mutations made by human designers in the order set range from adjacent to the ligand (3–4 Å distance from residue CA to the ligand) up to second shell (12–13 Å distant) with a small minority of mutations over 13 Å (Fig. S1C, Supporting Information, distance distribution).

The greedy protocol performs sidechain repacking with a stochastic MC algorithm, and therefore it is not deterministic. To estimate sample variation we tested the best greedy Protocol (see Results section, below, ES10_broad2) over five independent runs, giving a mean of 8.75 with SEM 0.03. Because the SEM is small, we report the results of single runs over the full benchmark set, and only draw conclusions from differences at least three times the SEM (0.1 mutations).

Native sequence recovery benchmark for protein–ligand complexes

We chose a representative member from each protein class (binding, immunological, transport, etc.) in the Binding Mother Of All Databases (MOAD) to construct the sequence recovery benchmark.[11] The proteins in MOAD are well resolved (<2.5 Å) with biologically relevant ligands (small organic molecules and cofactors, but not crystallographic additives, salts, etc.) and binding data derived from the literature. The proteins in each class were inspected manually, curated to include only binders of natural ligands in the affinity range of 10 μM or lower. Structures were prepared for Rosetta calculations as described in Supporting Information Appendix A. Our data set was directed specifically toward natural small-molecule binders and excludes enzymes and catalytic antibodies. Small-molecule binding proteins should be evolutionarily optimized only for binding and overall stability, which we can effectively model. Enzyme modeling would require additional information about the functional residues, such as the requirement for a catalytic triad at a specified set of distances from a substrate peptide in a protease. Catalytic antibodies were also excluded from the benchmark as they did not have an evolutionary timeframe over which to evolve, and have less converged sequences. Transition metal-binding proteins were also removed from the benchmark, as these require additional metal-specific interactions with amino acids to be included for optimal performance. The resulting set consists of 51 proteins with ligands, as summarized in Supporting Information Appendix B. The protein–ligand native sequence recovery benchmark is available as part of the standard Rosetta package on github at: Rosetta/main/tests/scientific/biweekly/enzdes_benchmark

Sampling and algorithm

The protein–ligand native sequence recovery benchmark enabled evaluation of new scoring terms and new sampling algorithms for protein–ligand interaction design. These were tested by adding the modifications to the standard Rosetta energy terms one at a time.[12,13]

Evolutionary information from a Multiple Sequence Alignment (MSA) was introduced via a position-specific scoring matrix (PSSM) to give a likelihood score to each residue at each position. The MSA implementation uses a PSSM generated by sequence alignment of homologs with a maximum *E*-value cutoff of 0.0009 using blastpgp. This gives a log-odds score derived from the relative proportion of each amino acid and the prior probability of observing each amino acid.[14] The influence of PSSM score on sequence recovery was investigated by iterating the PSSM weight over a set of 11 discrete values: 1, 2, 3, 4, 5, 10, 20, 50, 100, 200, and 300 in the native recovery benchmark.

Energy terms for sidechain repacking are represented in a graph-like data structure with connections between each residue describing their pairwise interactions.[15] The enzyme design protocol[16] is typically run with a higher weight on protein–ligand interactions in the energy graph, so that alterations in these energies will play a larger role during MC sidechain repacking steps. This up-weighting is only applied to residues that change identity, and not applied during minimization, when residue identity is static. However, finding an appropriate value for this protein–ligand interaction adjustment is problematic without a large training set. Here protein–ligand interactions were up-weighted between one- and threefold in increments of 0.2. The default benchmark was always run with a weight of 1.8.

Modulation of the repulsive part of the Lennard–Jones (LJ) potential has been demonstrated to improve sampling and free energy calculations,[17] and a reduced-repulsion "soft" LJ term allows for higher sequence recovery of natives during design. Here we test different methods for applying the "soft" LJ potential during design.

Deeper sampling of rotamers can be accomplished by increasing the number of MC cycles within a trajectory or by running multiple trajectories in parallel. Rotamer sampling is carried out using MC while temperature is slowly decreased, in a simulated annealing method. Each step in the temperature cycle is an "inner" iteration, with a set number of rotamer sampling steps. The "outer" iterations carry out each "inner" cycle a set number of times while the temperature is varied. A set number of "inner" quench cycles can optionally be performed *N* times with the multi-cool annealer (MCA) (after *N* independent runs of temperature annealing, the best individual final score is passed on as the output structure). The MCA may allow for better sampling in many cases, due to the stochasticity of an MC trajectory. Here we tested the effect of outer iteration scaling and MCA sampling on sequence recovery in the native sequence benchmark (Supporting Information).

In the enzyme design protocol applied here a "design cycle" is a set of MC rotamer substitution (as described above) and a gradient minimization. We determined the effect on sequence recovery of increasing the number of design cycles up to five. The number of cycles defaults to two for the enzyme design protocol and always allows at least one round of sampling with a soft LJ potential while the last cycle is performed with a hard potential. A higher number of design cycles will increase overall sampling, but may lock the structure into any energy minima encountered during the earliest cycles, or minimization steps may perturb the backbone and introduce errors.

Rotamer sampling can be improved by utilizing the existing sidechain rotamers from the input structure. These rotamers are used until they are swapped for lower energy ones, which eventually leads to the loss of these particular rotamers from the set of allowed conformations.

Code version and availability

Rosetta deposited SVN revision **51912** was used throughout the study to enable the reproducibility of the results presented here.[15] Sequence recovery calculations over the protein–ligand benchmark were carried out with the enzyme design application (which is used for any protein–ligand design problem, "enzyme" design being accomplished by introducing a set of extra geometric constraints in addition to Rosetta scoring).[16] The final greedy optimization Rosetta protocol (in RosettaScripts format[18] with an example run in Supporting Information Appendix C) for recapitulation of human design intervention is available in the standard Rosetta package: Rosetta/main/source/src/apps/public/enzdes/ES10_broad2.xml

RESULTS AND DISCUSSION

Match-design-order benchmark: recapitulating human design interventions

Native protein sequences have been optimized over an evolutionary timescale to have optimal stability and function. Protein design algorithms that optimize overall protein stability can correctly recover many of the native residue identities.[8] Alternative design methods can be evaluated based on the extent of recovery of native sequence over a set of monomeric proteins, and a similar approach can be used to optimize ligand-binding design methods.

However, native sequence recovery is an imperfect measure of the performance of a method for designing new small-molecule binding sites. The protein backbone is pre-configured for ligand binding, the second-shell (and further) sidechain interactions are also pre-configured to buttress first-shell interactions, and the ligand is already placed in the optimal conformation and orientation. In contrast, in a novel design scenario, neither the backbone nor the surrounding sidechains are likely to be precisely configured to support the new binding site. A native sequence recovery approach also cannot be used to assess the utility of a bias toward the native sequence, which is often used to reduce the incidence of potentially destabilizing mutations from the native sequence.

We have developed a new benchmark of raw Rosetta designs along with the final human-designer modified sequences to test design algorithms in a more realistic context, design of a novel function into a protein backbone structure previously lacking that function. We call this the MDO benchmark. The benchmark consists of 51 protein-triplets: (a) a native PDB structure (here the output from the matcher, or "match," that has all native sequence except at important catalytic or binding positions specified in advance), (b) the raw output from Rosetta with designed residues around the ligand of interest ("design"), (c) and the final human modified sequence, which is often substantially different from the raw design output ("order"). The MDO benchmark consists of proteins gathered from protein engineering efforts in the Baker lab (see Materials and Methods section for details of the proteins and mutations made by human designers).

Algorithm choice and development

We sought an algorithm to recapitulate the changes introduced by human designers over the 51 protein MDO set, essentially a piece of software that would produce an output design as similar as possible to a human designer's sequence. This algorithm should be as general as possible, allowing for hypothesis testing; for example, does filtering using shape complementarity[19] measures between protein and ligand help imitate human behavior? It should allow for complex scoring and sampling methods that require long computation times. Since human designers typically consider residue positions one-by-one, we chose an algorithm for sequence optimization that tests mutations one-by-one around the active site (with adjustable sampling and scoring methods) and then incorporates those changes in rank order by score.

This protocol for navigating a tree of decisions in a multi-parameter search space is a greedy algorithm, as it evaluates each possibility independently, sorts them by a selection function, and then takes the best options first. Greedy algorithms may not be able to locate a global optimum, instead getting stuck in a local minimum, but in some cases they very quickly converge on a global optimum. One would expect a greedy algorithm to perform well when the starting sequence is already close to the optimum sequence, but not to do well in an overall sequence optimization problem starting from a random sequence. For the late-stage design optimization problem considered here the input is already MC optimized and somewhat close to a global optimum. Greedy algorithms have been applied to many problems in computational biology including sequence alignment,[20] fragment selection,[21] RNA structure building via a stepwise approach,[22] protein–peptide specificity prediction,[23] and a recent study using a greedy algorithm after a MC rotamer search for sidechain placement.[24] The greedy algorithm applied to the protein design problem is most similar to the Self-Consistent Mean-Field method[25,26] but with mutations applied in rank order, and without iteration or a check for self-consistency.

The protocol uses a variety of easily swappable structure assessment conditions, called filters, and sampling methods, referred to as movers.[18] The protocol operates on a designed structure and examines each position that has been altered from the native structure. Every amino acid point mutant and rotamer state at every position is sampled independently as follows:

- After rotamer optimization, gradient minimization of all neighbor sidechains within an 8 Å sphere, and a user-defined further optimization (termed the "mover"; e.g., rotamer optimization in a larger sphere, ligand torsion-angle minimization, and others as detailed below), the total energy is stored.
- Substitutions that fail any user-defined quality filters (e.g., shape complementarity) are eliminated.
- After all point mutants and/or rotamers have been evaluated, substitutions at each position are sorted by energy, and positions are rank-ordered by the energy of the optimal substitution at each position.
- Substitutions are combined by first attempting placement of the optimal substitution at the optimal position, evaluating the total energy, and accepting if the total score improves. The substitution at the second ranked position is then attempted, accepted, or rejected, and the process is continued until substitutions at all positions have been attempted.

Due to the deterministic nature of the algorithm, this approach converges reliably to nearly identical solutions, with slightly more variation if a more aggressive mover is applied.

We do not know whether human intuition systematically improves designs, and a difficulty in even formulating an answer to this question is that every human has somewhat different preferences in design. The MDO framework allows us to begin to frame and rigorously test hypotheses about how to improve protein design; without such a benchmark, evaluation of new algorithms is largely anecdotal.

Recapitulation of designer interventions in the match-design-order set

We ran a series of tests using the MDO benchmark to find the optimal mover, filter and native sequence-favoring weight to use with the greedy protein design

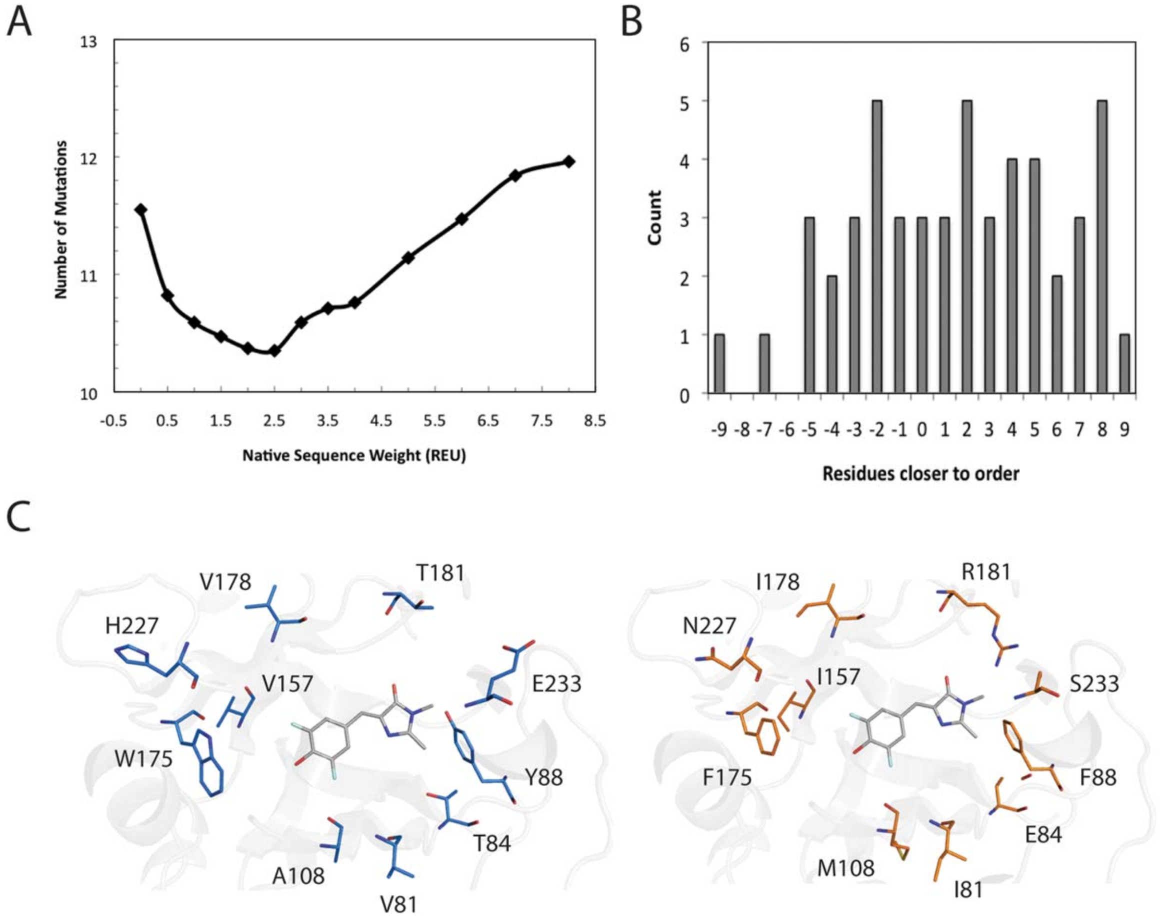

Figure 1

Design improvements from a greedy scanning algorithm. (**a**) The number of mutations to order as a function of the native sequence favoring weight. Lower numbers are better, indicating a closer sequence to the order set. The optimal native weight is in the range 2–2.5. (**b**) The number of residue improvements (fewer mutations to the order set) for the best greedy protocol, for each of the 51 proteins in the MDO test set. (**c**) A comparison of the raw Rosetta design output with positions adjusted by the greedy algorithm indicated as sticks (**left**, *blue*) and the greedy protocol output with those same positions indicated (**right**, *orange*) for MB11, the most improved case.

refinement protocol. We ran the different protocols on the design set to produce an output set of structures, calculated the average number of mutations between this output set and the order (human modified) set, and used this as the scoring metric. Lower numbers are better, and zero indicates that the protocol has perfectly recapitulated all of the human design decisions in the order set. We also calculated the number of mutations in the output set to the native sequence (match), and the number of mutations to the input design set (designs) to keep track of how many sequence changes the protocol is making. The starting point is 10.3 mutations to the order set, 19.0 mutations to the natives (and 0.0 to the design set, which is the input).

We experimented with the use of a favorable weight on native residues through the favor sequence profile (FSP) mover in Rosetta. For example, an FSP weight of 2 gives a bonus of −2 Rosetta Energy Units (REU) to the native residue at each position, while any other residue has no added bonus. The number of mutations to the order set achieves a broad minimum from 1.5 to 3 as FSP weight is adjusted, centered around 2–2.5 [Fig. 1(a)]. Optimization of a native-sequence favoring weight with a traditional sequence recovery benchmark is impossible. The recovery of native sequence would simply increase monotonically with increasing native-sequence weight. The MDO benchmark makes this test possible.

We tested a number of different movers in the greedy algorithm for energy minimization upon introduction of each mutation at each position (Table I). The same filter is used in all cases, the shape complementarity (SC) filter with a weight of −5 in addition to the total energy. These movers range from a relatively simple mover (local repack around the mutated residue followed by a minimization of the protein–ligand interface) to more complex movers with multiple cycles of repacking and minimization (Table I). A mover of (repack interface

Table I
Optimal Mover in the Greedy Protocol

Name	Mover	Order	Match	Designs
Designs	*NA (input set, unmodified)*	*10.33*	*18.96*	*0.00*
ES10_broad2	cut 10/12/13/15 and ES10 mover	8.78	12.51	8.2
ES10_broad3	cut 11/13/14/16 and ES10 mover	8.96	12.39	8.82
ES10_broad	cut 8/10/11/13 and ES10 mover	8.96	12.71	7.55
ES9_broad2	cut 10/12/13/15 and ES9 mover	9.02	13.29	7.82
ES9	softpackLOC/minINT	10.24	15.39	4.29
ES13b	softpackLOC/softenzpackINT/minINT/hardenzpackINT/minINT	10.35	15.49	4.31
ES10	softpackLOC/minINT/hardpackINT/minINT	10.47	15.39	4.29
ES13a	softpackLOC/enzpackINT/minINT	10.51	15.12	4.65
ES14	Nativescan followed by enzscan with ES10 mover	10.59	14.51	5.51
ES10_broad2_nofsp	ES10_broad2 with no native sequence bias	17.65	25.67	9.76

The greedy protocol was run over the MDO test set with the mover specified in the Table, always using the ES10_baseline (-5SC) filter.
Definitions: INT, all interface around ligand; LOC, only the region around the mutated residue; softpack, repack the specified region with a lower vdW repulsion; cut a/b/c/d, Values for automatic determination of the design and repack region around the ligand. Residues with atom CA < a Angstroms from the ligand are designable; those with CA -> CB vector pointing toward the ligand are designable if CA < b. Residues with CA < c from the ligand are repackable; those with CA -> CB vector pointing toward the ligand are repackable if CA < d. All other residues are left with natural identity and rotamer (neither repackable nor designable); hardpack, repack the specified region with standard vdW; enzpack, repack the ligand pocket, allowing ligand repacking, rigid body moves; broad, larger design shall (with broad 2 and 3 each larger, respectively); min, sidechain minimization; nativescan, only allow native residue or designed residue at each position, while enzscan allows all residues at each position.

with low LJ repulsion → minimize interface → repack with normal LJ repulsion → minimize) has the best performance. More complex movers are not able to improve the number of mutations to the order (Table I).

Larger design shells give a lower number of mutations to order. The standard Rosetta design protocol optimized the identity of residues within 6 Å of the ligand, or 8 Å if the residue Cα–Cβ vector is pointing toward the ligand. Expanding the design shell to 10/12 Å allows the protocol to alter 3312 residues in 51 proteins, versus 1049 residues in the standard design shell. Human designers tend to make changes outside of the standard design shell, for example, adding backing-up interactions to keep first-shell residues in place. The smaller design shell by definition cannot recapitulate human design decisions outside of the sphere of residues it examines.

The best mover produces on average 8.8 mutations from the order set (vs. 10.3 in mutations in the starting structures; Table I). For comparison, the same protocol over an expanded shell without any bias toward native sequence produces 17.7 mutations from the order set.

We tested the greedy protocol with a variety of filters to find the optimal behavior in the MDO benchmark set for recapitulation of human designed sequences and found similar behavior for total energy alone or total energy plus a SC filter with a weight of −5 (Table S1, Supporting Information). All other filter combinations gave worse behavior, such as a heavy negative weight on SC or any weight on the SASA filter.

Sequence analysis of outputs from the best greedy protocol

Most sequences in the MDO benchmark are slightly improved, with 0 to 8 fewer substitutions [Fig. 1(b)]. In some cases there is actually an increase in the number of mutations to the ordered sequence [negative numbers in Fig. 1(b)], but in most of these cases the method has simply placed a number of reversions to native that were not placed by the human designer. The best case is MB11 with 15 mutations from the ordered sequence in the input design and only 6 in the output from the greedy protocol [Fig. 1(c); residue identities in the design on the left in blue, residues after the greedy protocol on the right, in orange]. More than half of the residues that are altered have a Cα–Cβ vector pointing away from the ligand. These mutations from wild type are unlikely to favorably impact ligand–protein interaction energies. In this case all nine of the correctly altered amino acid positions are reversions to the amino acid identity in the original scaffold. The case with the least improvement is BL23 with only 5 mutations from the ordered sequence in the input design, and 14 in the output from the greedy protocol. Again most of the changes introduced by the greedy protocol are reversions to native, but in this case those changes do not agree with those made in the ordered sequence.

Native sequence recovery benchmark

With results from the MDO benchmark and the new greedy protocol in hand, we sought to optimize the MC-based design protocol used before manual modification or the greedy protocol. This MC-based protocol has previously been optimized for monomeric native proteins, not for protein–ligand interaction design. Our results from the MDO indicated that the preservation of native sequence is important to maintain the stability of engineered proteins. We aimed to optimize the MC enzyme design protocol in Rosetta to minimize the need for sequence reversion in subsequent greedy optimization, and, more generally, to improve the quality of design outputs.

To optimize the existing MC-based computational design protocol for the protein–ligand design problem, we assembled a protein–small-molecule benchmark set of wild-type (as opposed to computationally designed) protein structures (as described in more detail in the Materials and Methods section and Supporting Information Appendices A and B) with biological ligands, high structure resolution, and measured binding affinity better than 10 μM. The benchmark samples 1041 amino acid positions in 51 proteins, which gives an average of 20 designable residues per protein active site. We used this benchmark to assess protein sequence recovery with different design algorithms or score functions. The overall sequence recovery for the benchmark set with the standard Rosetta enzyme design protocol is 44%.[16]

Monte Carlo design algorithm improvement with the native sequence benchmark

General features of sequence recovery benchmark

To quantitatively evaluate how the MC design algorithm behaves with different scoring and sampling methods we first examine the complexes with highest and lowest sequence recovery. The best case is the 1DB1[27] complex in which 22 residues out of 35 are recovered [Fig. 2(a)], with a total sequence recovery of 63%. In the case of the 2PFY[28] complex only 2–3 residues out of 11 are correctly predicted [Fig. 2(b)], with a resulting sequence recovery of only 24%. In general the sequence recovery is correlated with the chemical composition of the active site and the ligand, as the energy function performs better with more hydrophobic amino acids.[8] 1DB1 is a nuclear receptor in complex with vitamin D, which is large and relatively hydrophobic [Fig. 2(a)]. 2PFY is an extra-cytoplasmic receptor bound to pyroglutamic acid, which is quite small and polar [Fig. 2(b)].

Incorporating evolutionary information with a position specific scoring matrix

Protein design onto an existing protein structure can benefit from knowledge of the close evolutionary homologs encoded in a PSSM[29] and included in the energy function as an additional term. Including a PSSM term provides a relatively large increase in sequence recovery (15%) with only a very small increase in total Rosetta score [Fig. 2(c)]. We observe an optimal PSSM weight above which sequence recovery deteriorates; this behavior is different from a native-sequence favoring weight, which would simply produce perfect recovery at a high level [Fig. 2(c)]. Sequence design for a novel function might benefit less from a PSSM term, although conserved residues that are vital to stability would be preserved using this method.

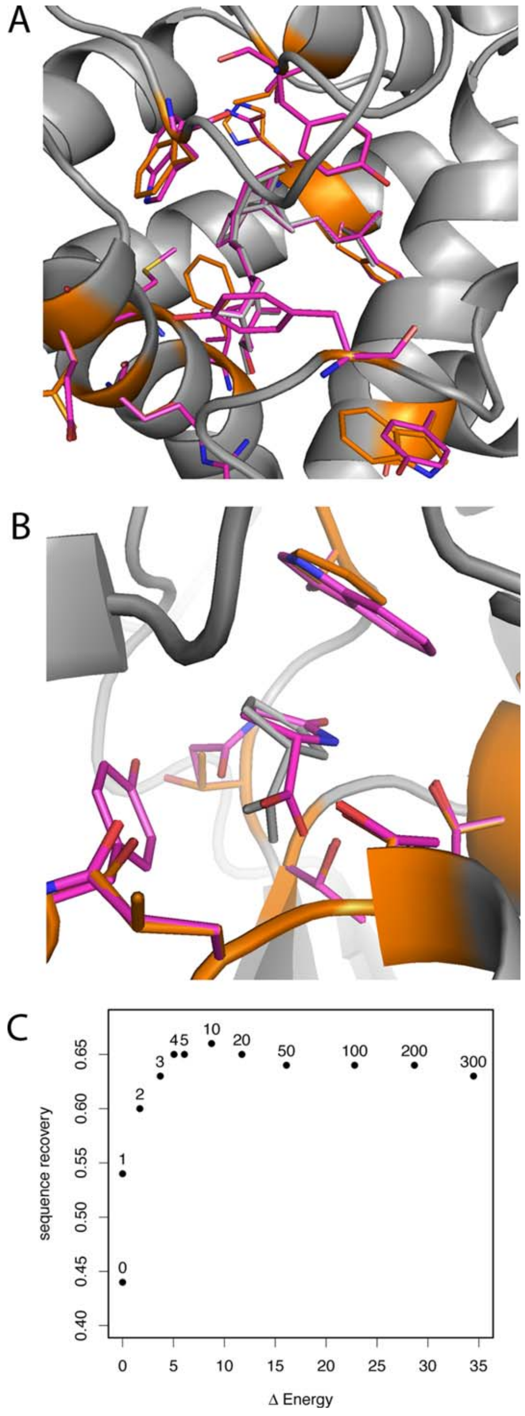

Figure 2

Native sequence recovery examples and incorporation of evolutionary information in design through a PSSM. (**a**) Comparison of the most successful case in the benchmark, 1DB1 complex, and (**b**) the most diverged sequence, 2PFY complex. *Purple* is the WT structure and *orange* are the mutations introduced during the design stage. (**c**) Sampling of the WT sequence can be improved by including information from homolog structures with a PSSM. A moderate weight on the PSSM score increases the sequence recovery significantly while the overall energy is unchanged or moderately increased. PSSM weight is varied between 0 and 300.

Table II
Optimization of Sampling and Energy Evaluation for the Native Sequence Recovery Benchmark Set

Method	Details	Sequence recovery (%)
M1	No probability of amino acid given backbone phi/psi,[a] no pairwise statistical residue–residue contact term,[b] Song *et al.* corrections[c]	44.7
M2	M1 + design with standard Lennard–Jones potential[d]	44.8
M3	Deeper rotamer sampling during packing[e]; 10 parallel trajectories of the packer, choosing the best single run[f]	44.8
M4	M1 + M3	45.0
M5	M3 with PSSM (weight = 1)	55.4
M6	M3 with wild-type residue rotamer added[g]	50.9

Each test was run with Rosetta options as indicated in the legend.
List of corresponding Rosetta flag names: [a]p_aa_pp, [b]fa_pair, [c]correct, [d]no soft_rep_design, [e]outeriterations_scaling 4, [f]multi_cool_annealer 10, [g]use_input_sc.

Optimal scoring and sampling methods for native recovery

We used the sequence recovery benchmark to test alternative energy function parameterization and sampling methods. These include recent force field modifications from Song *et al.*[12] and Leaver-Fay *et al.*,[30] modulation of unfavorable repulsive interactions, upweighting of protein–ligand interactions, altering the number of annealing cycles during MC rotamer packing, and modifying rotamer-inclusion schemes. Increasing the number of cycles in annealing, and employing an expanded “multi-cool-annealing” (MCA)[31] protocol performed well, as did the energy function term changes of Song *et al.* (SOM).

We next explored the combination of the force field and algorithm improvements with each other and with the native sequence and rotamer bias terms. The combination of the best sampling method, MCA, with the Song *et al.* force field corrections yielded an additional small improvement in sequence recovery (Table II; other combinations did not generally lead to improvements). The best sequence recovery was achieved with a PSSM score term, MCA sampling, and the Song energy corrections (55.4%), and we recommend this combination for most protein–ligand design cases (Table II). The scoring behavior of Song *et al.*[12] and Leaver-Fay *et al.*[30] is default in Rosetta as of git tag @2fac63a via the “talaris2013” scoring function (Supporting Information). Native rotamer inclusion leads to an even better sequence recovery of 56.0%, but as soon as one needs to redesign the active site to introduce a new function (instead of recapitulating the native ligand-binding site as we do here) it is advantageous to use the more general PSSM information instead with weight set to one.

CONCLUSION

The human-design benchmark (MDO) is uniquely suited to evaluating the ability of algorithms to recapitulate human intuition during the design of novel function into protein scaffolds. It formalizes a test system for design algorithms, allowing for rigorous hypothesis testing without resorting to individual design examples. Of course we do not know if human intuition improves upon computationally designed proteins. Now, with the greedy algorithm and the MDO benchmark, we can systematically evaluate different human-imitating algorithms (e.g., one emphasizing shape complementarity, another emphasizing solvent accessible surface area).

The optimal greedy protocol combines the best mover, FSP weight, and filters. The sampling in this protocol is local—an attempted mutation is introduced at a given position and only nearby residues are optimized—and not global over the entire designed interface, as is the case in standard Rosetta MC-based sampling. This protocol should perform well for small-molecule binding proteins. A separate optimization would be required for other problems, such as protein–protein interaction design, with an appropriate benchmark set.

The two primary bottlenecks in the production of high numbers (hundreds) of computational designs are the human time required to evaluate and refine each structure, and the cost and complexity of synthesizing large numbers of genes. The greedy protocol reduces the amount of time required to produce each design, while increasing the likelihood that individual designs are stable and functional. The recent developments in array-based DNA synthesis will increase the number of testable independent sequences.[32] In many instances of computational protein design a very small fraction, approximately 1%–2% of designs, is folded and active. The combination of the greedy optimization protocol and array-based DNA synthesis could significantly increase the chance of success for difficult design challenges.

ACKNOWLEDGMENTS

L.G. Nivón acknowledges support from an NSF Minority Post-Doctoral Fellowship. S. Bjelic acknowledges support from The Swedish Research Council (623-2008-497) and Foundation BLANCEFLOR Boncompagni-Ludovisi, née Bildt. D. Baker thanks DARPA/DTRA for financial support.

REFERENCES

1. Kiss G, Celebi-Olcum N, Moretti R, Baker D, Houk KN. Computational enzyme design. Angew Chem Int Ed Engl 2013;52(22):5700–5725.
2. Whitehead TA, Baker D, Fleishman SJ. Computational design of novel protein binders and experimental affinity maturation. Methods Enzymol 2013;523:1–19.
3. Koga N, Tatsumi-Koga R, Liu G, Xiao R, Acton TB, Montelione GT, Baker D. Principles for designing ideal protein structures. Nature 2012;491(7423):222–227.
4. Tinberg CE, Khare SD, Dou J, Doyle L, Nelson JW, Schena A, Jankowski W, Kalodimos CG, Johnsson K, Stoddard BL, Baker D. Computational design of ligand-binding proteins with high affinity and selectivity. Nature 2013;50:212–216.
5. Meiler J, Baker D. ROSETTALIGAND: protein-small molecule docking with full side-chain flexibility. Proteins 2006;65(3):538–548.
6. Davis IW, Baker D. RosettaLigand docking with full ligand and receptor flexibility. J Mol Biol 2009;385(2):381–392.
7. Malisi C, Schumann M, Toussaint NC, Kageyama J, Kohlbacher O, Hocker B. Binding pocket optimization by computational protein design. PLoS One 2012;7(12).
8. Kuhlman B, Baker D. Native protein sequences are close to optimal for their structures. Proc Natl Acad Sci U S A 2000;97(19):10383–10388.
9. Zanghellini A, Jiang L, Wollacott AM, Cheng G, Meiler J, Althoff EA, Rothlisberger D, Baker D. New algorithms and an in silico benchmark for computational enzyme design. Protein Sci 2006; 15(12):2785–2794.
10. Bjelic S, Nivon LG, Celebi-Olcum N, Kiss G, Rosewall CF, Lovick HM, Ingalls EL, Gallaher JL, Seetharaman J, Lew S, Montelione GT, Hunt JF, Michael FE, Houk KN, Baker D. Computational design of enone-binding proteins with catalytic activity for the Morita-Baylis-Hillman reaction. ACS Chem Biol 2013;8(4):749–757.
11. Hu LG, Benson ML, Smith RD, Lerner MG, Carlson HA. Binding MOAD (Mother of All Databases). Proteins: Struct Funct Bioinf 2005;60(3):333–340.
12. Song Y, Tyka M, Leaver-Fay A, Thompson J, Baker D. Structure-guided forcefield optimization. Proteins 2011;79(6):1898–1909.
13. Leaver-Fay MOM, Tyka M, Jacak R, Song Y, Kellog, E, Thompson J, Davis I, Pache R, Lyskov S, Gray J, Kortemme T, Richardson J, Havranek J, Snoeyink J, Baker D, Kuhlman, B. Scientific benchmarks for guiding macromolecular energy function improvement. Methods Enzymol 2013;523:109–143.
14. Altschul SF. Amino-acid substitution matrices from an information theoretic perspective. J Mol Biol 1991;219(3):555–565.
15. Leaver-Fay A, Tyka M, Lewis SM, Lange OF, Thompson J, Jacak R, Kaufman K, Renfrew PD, Smith CA, Sheffler W, Davis IW, Cooper S, Treuille A, Mandell DJ, Richter F, Ban YE, Fleishman SJ, Corn JE, Kim DE, Lyskov S, Berrondo M, Mentzer S, Popovic Z, Havranek JJ, Karanicolas J, Das R, Meiler J, Kortemme T, Gray JJ, Kuhlman B, Baker D, Bradley P. ROSETTA3: an object-oriented software suite for the simulation and design of macromolecules. Methods Enzymol 2011;487:545–574.
16. Richter F, Leaver-Fay A, Khare SD, Bjelic S, Baker D. De novo enzyme design using Rosetta3. PLoS One 2011;6(5):e19230.
17. Beutler TC, Mark AE, Vanschaik RC, Gerber PR, Vangunsteren WF. Avoiding singularities and numerical instabilities in free-energy calculations based on molecular simulations. Chem Phys Lett 1994; 222(6):529–539.
18. Fleishman SJ, Leaver-Fay A, Corn JE, Strauch EM, Khare SD, Koga N, Ashworth J, Murphy P, Richter F, Lemmon G, Meiler J, Baker D. RosettaScripts: a scripting language interface to the rosetta macromolecular modeling suite. PLoS One 2011;6(6):e20161.
19. Lawrence MC, Colman PM. Shape complementarity at protein/protein interfaces. J Mol Biol 1993;234(4):946–950.
20. Zhang Z, Schwartz S, Wagner L, Miller W. A greedy algorithm for aligning DNA sequences. J Comput Biol 2000;7(1-2):203–214.
21. Tuffery P, Derreumaux P. Dependency between consecutive local conformations helps assemble protein structures from secondary structures using Go potential and greedy algorithm. Proteins 2005; 61(4):732–740.
22. Sripakdeevong P, Kladwang W, Das R. An enumerative stepwise ansatz enables atomic-accuracy RNA loop modeling. Proc Natl Acad Sci U S A 2011;108(51):20573–20578.
23. King CA, Bradley P. Structure-based prediction of protein-peptide specificity in Rosetta. Proteins: Struct Funct Bioinf 2010;78(16): 3437–3449.
24. Miao ZC, Cao Y, Jiang TJ. RASP: rapid modeling of protein side chain conformations. Bioinformatics 2011;27(22):3117–3122.
25. Lee C. Predicting protein mutant energetics by self-consistent ensemble optimization. J Mol Biol 1994;236(3):918–939.
26. Wang C, Schueler-Furman O, Baker D. Improved side-chain modeling for protein-protein docking. Protein Sci 2005;14(5):1328–1339.
27. Rochel N, Wurtz JM, Mitschler A, Klaholz B, Moras D. The crystal structure of the nuclear receptor for vitamin D bound to its natural ligand. Mol Cell 2000;5(1):173–179.
28. Rucktooa P, Antoine R, Herrou J, Huvent I, Locht C, Jacob-Dubuisson F, Villeret V, Bompard C. Crystal structures of two Bordetella pertussis periplasmic receptors contribute to defining a novel pyroglutamic acid binding DctP subfamily. J Mol Biol 2007;370(1): 93–106.
29. Altschul SF, Madden TL, Schaffer AA, Zhang JH, Zhang Z, Miller W, Lipman DJ. Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. Nucleic Acids Res 1997;25(17): 3389–3402.
30. Leaver-Fay A, O'Meara MJ, Tyka M, Jacak R, Song YF, Kellogg EH, Thompson J, Davis IW, Pache RA, Lyskov S, Gray JJ, Kortemme T, Richardson JS, Havranek JJ, Snoeyink J, Baker D, Kuhlman B. Scientific benchmarks for guiding macromolecular energy function improvement. Methods Protein Design 2013;523:109–143.
31. Leaver-Fay A, Jacak R, Stranges PB, Kuhlman B. A generic program for multistate protein design. PLoS One 2011;6(7):e20937.
32. Kosuri S, Eroshenko N, Leproust EM, Super M, Way J, Li JB, Church GM. Scalable gene synthesis by selective amplification of DNA pools from high-fidelity microchips. Nat Biotechnol 2010; 28(12):1295–1299.