



Published in final edited form as:

Proteins. 2020 January ; 88(1): 206–226. doi:10.1002/prot.25790.

Comparison of Rosetta flexible-backbone computational protein design methods on binding interactions

Amanda L. Loshbaugh^{1,2}, Tanja Kortemme^{1,2,3,4}

¹Department of Bioengineering and Therapeutic Sciences, University of California San Francisco, San Francisco, California

²Biophysics Graduate Program, University of California San Francisco, San Francisco, California

³Quantitative Biosciences Institute, University of California San Francisco, San Francisco, California

⁴Chan Zuckerberg Biohub, San Francisco, California

Abstract

Computational design of binding sites in proteins remains difficult, in part due to limitations in our current ability to sample backbone conformations that enable precise and accurate geometric positioning of side chains during sequence design. Here we present a benchmark framework for comparison between flexible-backbone design methods applied to binding interactions. We quantify the ability of different flexible backbone design methods in the widely used protein design software Rosetta to recapitulate observed protein sequence profiles assumed to represent functional protein/protein and protein/small molecule binding interactions. The CoupledMoves method, which combines backbone flexibility and sequence exploration into a single acceptance step during the sampling trajectory, better recapitulates observed sequence profiles than the BackrubEnsemble and FastDesign methods, which separate backbone flexibility and sequence design into separate acceptance steps during the sampling trajectory. Flexible-backbone design with the CoupledMoves method is a powerful strategy for reducing sequence space to generate targeted libraries for experimental screening and selection.

Keywords

algorithms; amino acid sequence; binding sites; computational biology/*methods; molecular models

1 | INTRODUCTION

Computational protein design searches for sequences that adopt desired structures and functions. Most generally, computational design methods require (a) algorithms to

Correspondence: Amanda L. Loshbaugh and Tanja Kortemme, Department of Bioengineering and Therapeutic Sciences, University of California San Francisco, San Francisco, CA., alosbau@gmail.com (A. L. L.) and tanjakortemme@gmail.com (T. K.).

SUPPORTING INFORMATION

Additional supporting information may be found online in the Supporting Information section at the end of this article.

efficiently search the vast sequence and conformational space accessible to proteins, and (b) effective energy functions to rank the solutions. Both of these requirements necessitate approximations. Design energy functions are often simplified while considering atomic detail,^{1,2} and the search space of sequences and conformations is typically limited by reducing degrees of freedom in a design simulation. One early approximation was to leave the backbone fixed while sampling rotameric side chain conformations during sequence design.^{3,4} While the fixed backbone approximation is useful for computational efficiency, it is rarely sufficiently accurate as flexibility is a hallmark of naturally occurring functional proteins and backbones shift to accommodate side chain mutations arising during evolution or design.^{5–8} Highly stable, idealized folds can be designed de novo,^{9,10} but design of proteins with new functions remains challenging. In most cases where new functions have been designed computationally, the designed protein is modeled on natural “scaffold” proteins with minimal changes in backbone conformation,^{11–15} although there are recent notable exceptions of functions designed into de novo proteins.^{16,17} Moreover, attaining sufficiently high activities typically requires optimization of the desired function by directed evolution.^{14,15,18} Function often depends on hydrogen bonds, which require precise backbone and side chain geometry, which remains difficult to design¹⁹ especially when a novel function requires “reshaping” of an existing protein conformation.²⁰

Various strategies have been proposed to model backbone flexibility in design, such as small random perturbations in torsional or Cartesian space,^{7,21–23} normal mode analysis,²⁴ or “backrub” motions shown to underlie commonly observed protein structural heterogeneity in high resolution crystal structures⁶ that have proven useful to model structural changes in response to mutations.^{8,25,26} Several strategies have used protein backbone ensembles for design, which are typically generated computationally employing various sampling methods^{21,22,27} including backrub motions^{28–31} but can also be taken from crystal structures.^{23,28,31}

Within the structure modeling and design program Rosetta,³² backbone flexibility has been treated in a number of ways. These include (a) generation of new protein backbones by assembly from peptide fragments which demonstrated success in ab initio structure prediction,^{33,34} (b) iterating between sequence design via Monte Carlo search and structure optimization via minimization,^{35,36} (c) a robotics-inspired kinematic closure (KIC) algorithm³⁷ shown to model loop conformations with sub-Angstrom accuracy,³⁸ and (d) the Backrub algorithm benchmarked on recapitulation of known sequences.^{28,30,39–41} Most flexible backbone design methods iterate between sequence design on a fixed backbone and structural optimization on a fixed sequence,³⁵ which effectively uncouples sequence changes from direct influence on backbone structure. In contrast, the “CoupledMoves” method in Rosetta,⁴² combines side chain and backbone moves using Rosetta backrub sampling²⁶ in a single design step.

While Rosetta flexible backbone design has been successfully applied to forward engineering of new functions,^{16,43,44} different methods have not been directly compared for accuracy using common benchmark datasets. Here, we describe such a benchmark comparison of three different flexible-backbone design methods in Rosetta: CoupledMoves,⁴² BackrubEnsemble,^{39–41} and FastDesign, which combines sequence design with the

Rosetta FastRelax method^{45,46} to move the backbone. We focus on methods within the openly available Rosetta framework because they use the same energy function, which allows us to directly compare different methods of sampling backbone flexibility. We evaluate each of the methods on its ability to recapitulate “tolerated sequence space” for binding interactions. We define tolerated sequence space as experimentally selected or naturally occurring sequences consistent with a functional binding interaction with a small molecule or protein binding partner.

We find that CoupledMoves recapitulates tolerated sequence space and individual stabilizing mutations more accurately than FastDesign or BackrubEnsemble. We introduce an updated version of the CoupledMoves algorithm (CM-KIC) that uses KIC in place of the original backrub backbone mover, which leads to further small improvements in performance. The coupled algorithm allows subtle conformational shifts in backbone torsions, which accommodate favorable side chain rotamers, in turn leading to more accurate prediction of side chain interactions. We also analyze shortcomings of the design methods that highlight areas for improvement.

2 | MATERIALS AND METHODS

2.1 | Design methods

Design protocols used Rosetta revision number 60351 and energy function ref2015.^{1,47} For each method, we used standard parameters and settings previously reported in benchmarks or design applications, except for the new CoupledMoves (CM) methods (CM-FKIC and CM-WKIC, see below) reported here. Command lines for each method can be found in the supplement.

2.1.1 | FastDesign—FastDesign is based on the FastRelax protocol in Rosetta described in Reference 45 and 46. Briefly, FastRelax consists of inner cycles of rotamer repacking and backbone and side chain torsion minimization with progressively higher weight on the repulsive part of the van der Waals energy function component (from 2% to 100% of its total value). FastDesign uses an analogous protocol but allows side chain design in addition to repacking. During FastDesign, we used harmonic coordinate constraints to keep backbone heavy atoms close to their starting position, and the weight of the constraints was ramped down from 1.0 to 0.0 during the course of each inner simulated annealing cycle. Constraint and repulsive weights were ramped five times, during five outer cycles. For each input protein structure, 400 designs were generated in independent design trajectories. Command line arguments are provided in the Supplement.

2.1.2 | BackrubEnsemble—The BackrubEnsemble method is described in Reference 40 and 41. Briefly, the method generates a structural ensemble with backbone conformational variation using the backrub algorithm implemented in Rosetta,²⁶ and then carries out fixed-backbone side chain design on each member of the ensemble. 400 ensemble members were generated using 10 000 backrub trials, a temperature of 1.2, and a backbone segment length of 3–12 residues. Command line arguments are shown in the Supplement.

2.1.3 | Forced BackrubEnsemble design—“Forced” BackrubEnsemble design forces sequence design to choose the known consensus side chain at certain positions. Forced design was applied to Glutathione Reductase positions E50 and D331, and DIG10 positions Y34, Y101, and Y115. For each protein, 100 forced simulations were run, using as input the first 100 members of the same BackrubEnsemble on which typical design was performed.

2.1.4 | CoupledMoves—The CoupledMoves method was used as described in Reference 42. Briefly, each coupled move had a 90% probability of being a backbone and side-chain move, and a 10% probability of being a ligand move. Each simulation was run for 1000 moves and 400 simulations were run for each protein-ligand or protein-protein complex. All unique amino acid sequences accepted during each simulation were output into a FASTA file, and the resulting 400 FASTA files were pooled, including redundancy, for analysis. Command line arguments are provided in the Supplement.

2.1.5 | CoupledMoves with kinematic closure—Two different methods of modeling backbone flexibility are implemented in CoupledMoves. The first method uses the Backrub algorithm²⁶ and was originally described in Reference 42. The second method uses KIC^{37,38} and is implemented in CoupledMoves here. Kinematic closure in Rosetta³⁸ generates conformations of backbone segments by sampling nonpivot torsions in the segment and then analytically determining values for six pivot torsions to close the loop. For the CM-FKIC (KIC using fragments) method, nonpivot torsions are sampled from peptide fragments³⁴ taken from the protein structure database (PDB). For the CM-WKIC (KIC using a “walking” perturbation) method, nonpivot torsions are adjusted by a random value from a Gaussian distribution centered around zero and with a SD of 3°. In each case, the remaining six pivot torsions are then solved analytically to close the loop. Command line arguments are provided in the Supplement.

2.1.6 | Ligand handling—Rosetta requires ligands to be described by a params file, which contains information defining the ligand’s atom types, bond geometry, and chemical connectivity. We generated params files from PDB structures using Rosetta’s molfile_to_params.py utility script. We did not model multiple ligand conformers except for DIG, for which the DIG ligand conformer library used during DIG10 design¹⁵ was obtained from the authors.

CoupledMoves samples ligand rigid-body translation and rotation in all cases. FastDesign minimizes ligand torsional degrees of freedom in addition to backbone torsion angles during its minimization step. BackrubEnsemble and FixBB do not sample ligand movement.

2.2 | Benchmark datasets

2.2.1 | Cofactor binding sites—This dataset is described in detail in Reference 42. Briefly, the dataset is comprised of seven protein families, each containing a conserved small molecule cofactor binding site (Table S1). The crystal structure with the highest resolution available was chosen as the starting point for design. As in Reference 42, positions with a side-chain heavy atom within 6 Å of any heavy atom in the cofactor ligand were allowed to

design to any amino acid identity, and positions that could clash with designable positions were allowed to repack (change conformation but not identity) (Table S2). Known profiles were obtained from natural sequences of these binding sites as described in Reference 42.

2.2.2 | Enzyme specificity—This dataset is described in detail in Reference 42. Briefly, the dataset is comprised of 10 enzymes for which there are experimentally validated specificity-altering mutations in the ligand binding sites (Table S3). As in Reference 42, design was carried out with either the native or the non-native substrate/substrate analog. Positions with heavy atoms within 4.5 Å of any ligand atoms differing between the native and non-native substrate were allowed to design to any amino acid identity, and positions that could clash with designable positions (as described in Reference 42) were allowed to repack (Table S4). Structures were prepared as described in Reference 42. Briefly, for each enzyme four types of structures were prepared: (a) the native enzyme with the native ligand, (b) the mutant enzyme(s) with the nonnative ligand, (c) the native enzyme with the non-native ligand, and (d) the mutant enzyme with the native ligand.

2.2.3 | DIG10—The DIG10 dataset was taken from Reference 15. Briefly, DIG10 is a computationally designed protein that has been engineered to bind the small molecule digoxigenin (DIG).¹⁵ A computational design, DIG10, was subjected to selection by yeast surface display, first of a single-site saturation mutagenesis library, then of a combinatorial library of beneficial mutations identified in the first selection, yielding variant DIG10.1. DIG10.1 was then subjected to site saturation mutagenesis (SSM) and selections using yeast surface display to probe the effect of mutations, which our computational design protocol seeks to recapitulate. For input, we used the crystal structure of wild-type protein (PDB: 1Z1S) on which DIG10 was designed, the sequence of DIG10.1 (which we placed onto the 1Z1S scaffold using the Rosetta fixed backbone [FixBB] design protocol), and the digoxigenin conformation from the DIG10.2/digoxigenin complex (PDB: 4J8T), where DIG10.2 is a DIG10.1 variant containing additional mutations from the SSM selection). Digoxigenin was placed into the 1Z1S scaffold by using PyMOL to align 4J8T and 1Z1S, then combining the digoxigenin molecule from 4J8T and the protein structure from 1Z1S into a new PDB file. The known profile represents the frequency equivalent (F_{equiv} , described below) of the selection experiment on the DIG10.1 SSM library. The 39 positions selected for experimental site saturation in Reference 15 were allowed to design to only those amino acid identities with high enough sequencing counts to be included in the enrichment and depletion calculations in Reference 15 (Tables S5 and S6). We note that the experimental screen mutated 1–2 position at a time, whereas we design multiple positions simultaneously. In CoupledMoves design, 30 positions were allowed to repack based on the possibility of clashes with designed positions; in design by noncoupled methods (FastDesign, BackrubEnsemble, FixBB), all positions were allowed to repack (Table S5).

2.2.4 | Fen49—The Fen49 dataset was taken from Reference 48. Fen49 is a computationally designed protein that has been engineered to bind the small molecule fentanyl (Fen). The original computational design, Fen49, has an affinity of 6.9 μM for a fentanyl-bovine serum albumin conjugate. After four rounds of selection, starting from Fen49, a combination of two substitutions, A78V and A172I, was identified to produce a

variant with a 100-fold improved affinity of 64 nM. We used the wild-type protein (PDB: 2QZ3), on which the sequence of Fen49 was modeled using Rosetta FixBB as input to our design simulations. The fentanyl conformation from designed fentanyl binder Fen49*/fentanyl complex (PDB: 5TZO, where Fen49* is a Fen49 Y88A point mutant that was more suitable for complex structure determination⁴⁸ was placed into the 2QZ3 scaffold using PyMOL. Fentanyl was placed into the 2QZ3 scaffold by using PyMOL to align 2QZ3 and 5TZO, then combining the fentanyl molecule from 5TZO and the protein structure from 2QZ3 into a new PDB file. While all positions of Fen49 were subjected to SSM, for our study we designed only the 18 residues defined as binding site in Reference 48 (Table S7). Design was allowed only to those amino acids with high enough sequencing counts to be included in the enrichment and depletion calculations in Reference 48 (Table S8). Finally, four positions (37, 64, 69, 71) in the input structure were set to alanine (using Rosetta's FixBB protocol), because the wild-type residue was disallowed due to low counts (Table S8). In CoupledMoves design, 22 positions were allowed to repack based on the possibility of clashes with designed positions; in design by noncoupled methods, all positions were allowed to repack (Table S7). The known profile represents the frequency equivalent (F_{equiv} , described below) of the final round of selection (obtained from the authors). Note that the experimental screen mutated one position at a time, whereas we design multiple positions simultaneously.

2.2.5 | Frequency equivalent—Experimental data from the DIG10 and Fen49 datasets are deep sequencing counts before and after selection. To allow direct comparison between the experimental data and sequence profiles from Rosetta design for each mutation x at position i , we derived a frequency equivalent (F_{equiv}) from the experimental frequency data:

$$F_{equiv} = \frac{\frac{f_i^{x,sel}}{f_i^{x,unsel}} / \frac{f_i^{orig,sel}}{f_i^{orig,unsel}}}{\sum \left(\frac{f_i^{x,sel}}{f_i^{x,unsel}} / \frac{f_i^{orig,sel}}{f_i^{orig,unsel}} \right)}$$

where $f_i^{X, Sel}$ and $f_i^{X, unsel}$ are the frequency of that mutation, and $f_i^{orig, sel}$ and $f_i^{orig, unsel}$ are the frequency of the original amino acid identity, in the selected and unselected populations, respectively, and F_{equiv} is normalized by dividing over the sum across all amino acid identities found in the sequencing results. F_{equiv} is then used in comparison to Rosetta design results.

2.2.6 | hGH/hGHR—The hGH/hGHR dataset was taken from Reference 30. The protein-protein interface between human growth hormone (hGH) and human growth hormone receptor (hGHR) is high affinity, with a K_D reported as 0.9 nM⁴⁹ and 1.56 nM⁵⁰. As input for design, we used a crystal structure (PDB: 1A22). The known sequence profiles were taken from a phage display selection experiment, wherein 35 key residues from the ~1300Å² hGH/hGHR interface were divided into six combinatorial libraries of five or six positions.⁵⁰ To minimize potential cooperative interactions, positions were grouped into libraries that

maximized the three-dimensional distance between residues. Our computational workflow mimicked this strategy, using the same designable residues and running independent design trajectories for each of the six libraries. As in Reference 30, residues within 4 Å of designed residues were allowed to repack (Table S9).

2.2.7 | Herceptin-HER2—The Herceptin-HER2 dataset was taken from Reference 51. The protein-protein interface between therapeutic antibody Herceptin and its target, human epidermal growth factor 2 (HER2), is high affinity⁵² ($K_D = 0.35$ nM). We used a crystal structure (PDB: 1N8Z) as input structure for design, truncated as in Reference 51 to include only chain A positions 1–106, chain B positions 1–119, and chain C positions 511–607. The known sequence profiles were taken from phage display selection experiments that used five combinatorial libraries containing five to seven positions each after four rounds of selection.⁵² We mimicked the experimental strategy in our computation, with five separate design runs, one for each experimental library, and allowing repacking of residues within 4 Å of designed residues, as in Reference 51 (Table S10). Herceptin/HER2 sidechains were repacked from the crystal structure before design.

2.3 | Performance metrics

2.3.1 | Position profile similarity—Position profile similarity (PPS) was computed as described in Reference 42. Briefly, PPS represents the similarity in the side chain amino acid identity distributions between the predicted and known sequences at a given position:

$$\text{position profile similarity} = 1 - D^{JS}(p_{\text{known},i}, p_{\text{design},i})$$

where $p_{\text{known},i}$ and $p_{\text{design},i}$ are the probability distributions over the 20 amino acids for the known (natural or experimental) and designed sequences, respectively, at position i and $D^{JS}(x, y)$ is the Jensen-Shannon divergence between two distributions x and y , as in Reference 42.

2.3.2 | RankTop—For the profile datasets, mutations were ranked according to their frequency in the predicted and known (experimental/natural) sequence profile. RankTop is the rank, in the predicted profile, of the top ranked amino acid from the known profile. If the amino acid is not found, its rank is set to 20.

2.3.3 | Percent Enrichment—As in Reference 42, the percent enrichment (PE) for each specificity-altering mutation in the enzyme specificity dataset was calculated as follows:

$$PE(WT \rightarrow MUT) = \%_{\text{non - native}} - \%_{\text{native}}$$

$$PE(MUT \rightarrow WT) = \%_{\text{native}} - \%_{\text{non - native}}$$

where $\%_{\text{native}}$ is the percent occurrence of the mutation in sequences designed for the native ligand and $\%_{\text{non - native}}$ is the percent occurrence of the mutation in sequences designed for

the non-native ligand. $PE(WT \rightarrow MUT)$ was used for predictions that start with the wild-type structure and $PE(MUT \rightarrow WT)$ was used for predictions that start with the mutant structure. As in Reference 42, a prediction was considered correct if it obtained a positive percent enrichment value.

2.3.4 | Rank—For the enzyme specificity dataset, mutations were ranked by descending order of their percent enrichment values, as described in Reference 42.

2.3.5 | Sequence Entropy—The sequence entropy H_i was computed as in Reference 42:

$$H_i = - \sum_x P_x \log_2 P_x$$

where P_x is the percent of sequences with amino acid x at position i .

2.3.6 | Distance from input sequence—Distance from input sequence is a variation of the profile similarity metric, where distance is calculated as:

$$distance = 1 - D^{JS}(p_{input, i}, p_{design, i})$$

where $p_{input, i}$ and $p_{design, i}$ are the probability distributions of the single input side chain and the designed sequence profiles, respectively, at position i .

3 | RESULTS

3.1 | Design methods

We set out to compare four flexible-backbone design methods (Figure 1, Methods) using a common set of benchmarks (described below): (a) FastDesign utilizing the Rosetta FastRelax method^{45,46} for backbone flexibility, (b) BackrubEnsemble Design,^{40,41} (c) CoupledMoves with backrub (CM-BR),⁴² and (d) the new CoupledMoves with kinematic closure (CM-KIC) method introduced here. We also compare to fixed-backbone design (FixBB) and a null model where all amino acid frequencies are set to 5%.

The main algorithmic differences between the methods are illustrated in Figure 1A. FastDesign (Figure 1A, left) iterates between two steps. In the first step (fixed-backbone sequence design), amino acid side chain identities and rotameric conformations are optimized using Monte-Carlo simulated annealing but the backbone is kept fixed. In the second step (fixed-sequence torsion minimization), the entire structure is minimized using backbone and side chain torsion degrees of freedom while keeping the sequence fixed. These steps are iterated through cycles of simulated annealing, during which the weight of the repulsive component of the Lennard-Jones potential is increased stepwise to enable amino acid changes that may introduce unfavorable clashes but can be subsequently relaxed in the minimization step. FastDesign has been used in a variety of application to design new functions.^{16,17,43}

The BackrubEnsemble method^{40,41} (Figure 1A, middle) also proceeds in two steps. The first step generates an ensemble of backbones through application of Backrub moves. Each Backrub move²⁶ selects two pivot backbone C α atoms and rotates the entire segment between them (2–11 residues) as a rigid body. Backrub moves are made throughout the protein structure (or a predefined region) by randomly selecting pivot points. The second step performs fixed-backbone sequence design on each member of the ensemble using Monte-Carlo simulated annealing. The BackrubEnsemble method in Rosetta has been shown to recapitulate protein conformational fluctuations^{28,29,53} and tolerated sequence space^{30,40,41} and has been successfully applied to the redesign of protein recognition specificity.⁴⁴

In contrast to FastDesign and BackrubEnsemble that separate fixed-backbone sequence design from fixed-sequence backbone sampling, CoupledMoves combines backbone and side chain moves, which can include sequence changes, into a single “coupled” Monte-Carlo step (Figure 1A, right). In this fashion, the backbone can respond to a designed sequence change more directly than in the noncoupled FastDesign and BackrubEnsemble methods. However, coupling backbone and side chain moves could artificially collapse designed structures. Because replacing a larger with a smaller amino acid side chain is less likely to lead to clashes, the change is more likely to be accepted. In subsequent steps it is harder to recover from such a collapse, as the backbone will have moved to accommodate the smaller side chain. To alleviate this problem, each side chain move in CoupledMoves considers all rotamers for allowed amino acids and chooses a likely side-chain rotamer and identity based on its Boltzmann-weighted Rosetta score. This change led to a considerable decrease in the number of designed small side chains (alanine or glycine).⁴² Finally, coupled moves can also be performed for the ligand, where rotation and translation of the ligand can be combined with ligand conformer changes. CoupledMoves has been shown to better recapitulate amino acid preferences in small molecule binding sites and mutations that switch enzyme specificity,⁴² but has not yet been tested in a forward-engineering application.

The original version of the CoupledMoves method uses Backrub moves to sample backbone degrees of freedom. Here we introduce an updated version of the CoupledMoves algorithm that performs backbone moves with the KIC algorithm³⁸ (Figure 1B). KIC selects two pivot C α atoms that define a segment, and a third pivot C α atom within the segment. The algorithm next perturbs the backbone torsion angles around all nonpivot C α atoms in the segment, breaking the loop. Finally, the torsion angles of the three pivot atoms are solved analytically to close the loop. The original implementation of KIC samples backbone phi/psi torsion angles at the nonpivot C α atoms probabilistically from Ramachandran space.³⁸ Our implementation here allows phi/psi sampling by substitution of peptide fragments derived from the protein structure databank (FKIC) or random “walk” perturbation of backbone torsion angles by values from a Gaussian distribution centered around zero with a SD of 3° (WKIC) (see Methods).

3.2 | Benchmark datasets

We evaluate the performance of the different methods on six benchmark datasets (Table 1, Figure 2). Each benchmark contains information on functional sequence variants. We chose binding as a proxy for function because the engineering of binding interactions is a common task with many important applications. Moreover, the stability of a binding interaction is a functional constraint that can be more easily explicitly modeled and scored by Rosetta than for example requirements for efficient enzyme catalysis that are often incompletely understood. The datasets comprise both small molecule binding sites and protein-protein interaction interfaces.

Four of the datasets contain small molecule binding sites (Table 1, Figure 2). The first two datasets were taken from Reference 42. Dataset 1 comprises evolutionary sequence alignments for seven naturally occurring protein families (Figure 2A shows one structure as a representative) that each bind a specific cofactor (“cofactor” set, Tables S1 and S2). Dataset 2 was curated from experimentally-characterized substrate specificity-altering point mutations for ten different enzymes (“enzyme specificity” set, Figure 2B, Table S3 and S4). Datasets 3 and 4 were compiled from site saturation mutagenesis (SSM) experiments performed on two different proteins designed by Rosetta to bind small molecules (sets “DIG10” (digoxigenin),¹⁵ Figure 2C and Table S5 and S6, and “Fen49” (fentanyl),⁴⁸ Figure 2D and Table S7 and S8). The SSM libraries were screened for binding to the target small molecule (digoxigenin or fentanyl, respectively) using yeast display followed by deep sequencing of naive and selected populations.

The two protein-protein interface datasets contain sequences selected from combinatorial libraries (allowing all 20 naturally occurring amino acids at 5 to 7 sequence positions) by phage display and subsequent sequencing of individual clones (Table 1). Dataset 5 comprises sequences from five phage display libraries of Herceptin (17 positions total) selected for binding to HER2 (“Herceptin/HER2” set,⁵² Figure 2E and Table S9). Dataset 6 comprises sequences from six libraries of hGH (35 positions total) selected for binding to hGHR (“hGH/hGHR” set,⁵⁰ Figure 2F and Table S10).

3.3 | Performance metrics

Five of the datasets contain sequences from either experimental selection (DIG10, Fen49, Herceptin-HER2, hGH/hGHR) or natural sequence alignments of evolutionary families (cofactor), reflecting the diversity of amino acids at each position compatible with the protein’s function (tolerated sequence space).³⁰ We refer to this diversity as the “known sequence profile” for each position. We evaluate the ability of our design methods to recapitulate these known sequence profiles by quantifying two metrics used previously,^{40,42} profile similarity and rank top, both calculated per position (see Methods). Position profile similarity (PPS) measures the similarity of the probability distribution of amino acid frequencies between the known profile and the profile generated by Rosetta design at each position. Rank top measures the rank, in the design profile, of the amino acid most frequently observed at a given position in the known profile.

The enzyme specificity benchmark⁴² contains individual point mutations (rather than sequence profiles) experimentally characterized to switch enzyme substrate specificity. In this case, in contrast to the analysis for the sequence profile datasets, we do not assume knowledge of positions mutated in the experiment. Instead, we evaluate how the experimentally characterized specificity switching mutation ranks across designed mutations at all positions in the vicinity of the changed substrate, to approximate an actual design project where it is not clear a priori which position should be mutated. In addition to the absolute rank we also evaluate the percentile,⁴² of the experimentally characterized mutation among all design predictions (see Methods).

Each metric has a different experimental interpretation. The tolerated sequence space captured by the PPS metric is useful for the design of libraries, which can be screened for criteria in addition to binding affinity and specificity, such as protein stability and solubility. RankTop is useful for cases where a few mutations or design sequences are selected for individual experimental tests. Percentile gives information on how many predictions would need to be tested in order to find a successful mutation when making predictions for a range of positions.

3.4 | CoupledMoves improves prediction of tolerated sequence space

We first evaluated the overall performance of each flexible backbone design method on the five sequence profile datasets. Figure 3A shows the distributions of position profile similarities across all designed positions in each benchmark, with the median indicated by a white dot. CoupledMoves attains higher median PPS values than FastDesign and FixBB for all datasets, and higher median PPS values than BackrubEnsemble for all datasets except Fen49 and DIG10 where CoupledMoves and BackrubEnsemble perform similarly. Somewhat surprisingly, using this global metric CoupledMoves is worse than the null model for the DIG10 set. Even more strikingly, FastDesign and FixBB do not attain a higher median PPS values than the null model for any of the datasets (except cofactor), and are considerably worse than the null model for the hGH/hGHR and DIG10 datasets. As discussed below, the comparatively poor overall PPS of all methods for the hGH/hGHR, DIG10, and Fen49 datasets is due to low similarity between the input sequence and the known profile. In these cases, the null model scores as well or better than the design methods; of the flexible-backbone design methods, CoupledMoves performs best.

We next evaluated the RankTop values for all five datasets (Figure 3B). Here, all flexible backbone methods (except FastDesign for the hGH/hGHR dataset) perform better than fixed backbone design, which in the majority of the cases misses the most frequent amino acid side chain from the known profiles (the null model by definition ranks all amino acids the same so is not relevant here). The rank top values are lowest (best) for the Herceptin/HER2 and cofactor sets. CoupledMoves performs better than BackrubEnsemble and FastDesign for the Herceptin/HER2, hGH/hGHR and cofactor datasets, similar to FastDesign for the DIG10 set and similar to BackrubEnsemble for the Fen49 set. Moreover, for several benchmarks (hGH/hGHR, Herceptin/HER2, Fen49), CM-WKIC leads to small but noticeable improvement in RankTop values over CM-BR. Taken together, when considering both PPS

and RankTop over all datasets, CoupledMoves and in particular CM-WKIC perform best overall.

We also considered PPS and RankTop for each protein family comprising the Cofactor dataset (Figure S1), and found that CoupledMoves outperforms FastDesign for all families, and outperforms BackrubEnsemble for six of the seven families, with the exceptions of the flavin binding site of Flavodoxins. Performance for individual libraries of the Herceptin/HER2 (Figure S2) and hGH/hGHR (Figure S3) leads to similar conclusions.

To determine if methods were more predictive for different groups of positions, we plotted the PPS values for the different methods against each other (Figure 4A,B). CoupledMoves achieves similar or better PPS for nearly all positions when compared to the noncoupled methods (Figure 4A, CM-KIC shown as example). BackrubEnsemble achieves PPS values better or similar than FastDesign (Figure 4B, left), and better than FixBB (Figure 4B, middle), for almost all positions. FastDesign, compared to FixBB, achieves better PPS for some positions, but worse PPS for others (Figure 4B, right). Figure 4C quantifies the number of positions for which CoupledMoves is better, worse, or similar to the noncoupled methods. A prediction for a position is classified as “better” or “worse” by a given method relative to a comparison method when the difference in performance is above or below, respectively, a threshold of ± 0.1 for PPS or ± 5 for RankTop. When the difference is within the threshold, the predictions are classified as “similar.” CoupledMoves achieves better PPS values than BackrubEnsemble for 65 ± 1 positions, better than FastDesign for 119 ± 2 positions, and better than FixBB for 143 ± 2 positions. The standard deviation (SD) represents the average across CM-BR, CM-FKIC, and CM-WKIC. CoupledMoves also achieves better RankTop for more positions than BackrubEnsemble, FastDesign, and FixBB (39 ± 3 , 67 ± 3 and 126 ± 4 positions, respectively), (Figure 4C). Moreover, CoupledMoves performs worse than noncoupled methods for very few positions (Figure 4C, red bars).

3.5 | CoupledMoves outperforms the other methods at recapitulating key affinity-determining side chains

We next sought to evaluate the ability of the different methods to predict amino acid preferences for the positions that are most functionally important in the five profile datasets. Sequence logo representations of the tolerated sequence space for each of our datasets (Figures S4–S8) indicated considerable differences in sequence entropies between individual positions, and we reasoned that conserved side chain residues at low sequence-entropy positions are more likely to be important for protein function than residues at position with higher entropy. We hence split the positions in each dataset into three sequence entropy groups (see Methods) and evaluated median PPS and RankTop for the cofactor and Herceptin/HER2 datasets, which have the most consensus positions (Figure 5, Figure S9). Positions with low (entropy < 0.33) or medium ($0.33 < \text{entropy} < 0.67$) entropy were defined as consensus positions. The top known side chain for these positions was defined as the consensus side chain. We find that CoupledMoves achieves better PPS than the null model for consensus positions in the Herceptin/HER2 and cofactor datasets. FastDesign is better than the null model for only low-entropy positions for both datasets. BackrubEnsemble is better than the null model for low entropy positions in the cofactor dataset, but not

Herceptin/HER2. In contrast, the null model has the highest PPS for the high entropy bin, which might be expected for positions with high mutational tolerance.

Similar to PPS, CoupledMoves achieves the best (lowest) RankTop values for consensus positions, predicting the correct amino acid residue with at least some frequency at most positions, as opposed to noncoupled methods which frequently do not identify the consensus side amino acid identity at all (rank of 20) (Figure 5). CoupledMoves predictions typically have the highest entropy (Figure S10), which leads on average to higher similarity at variable positions. Nevertheless, PPS and RankTop at low-entropy positions (Figure 5, Figure S9), and energetic rankings of consensus positions (see Discussion) indicate that CoupledMoves is the most accurate method for functionally relevant interactions.

In addition to low-entropy positions determined from known sequence profiles, we also considered experimentally-characterized affinity-improving mutations, which were available for the Herceptin/HER2, Fen49, and enzyme specificity datasets (the latter set is discussed below). For Herceptin, the most important affinity-improving mutation, D V_H98 W, resulted in 3-fold improvement of binding affinity and was found in 23% of sequences resulting from phage display.⁵² Contrary to previous findings⁵¹ where BackrubEnsemble recapitulated D V_H98 W as the top mutation, the noncoupled methods tested in this study did not identify tryptophan (Figure S7), but CoupledMoves methods selected the tryptophan mutation at low frequency (CM-BR 1.1%, CM-FKIC 1.3%, CM-WKIC 1.5%). We note that this position is surface exposed in the original structure, leading to high entropy in the design profiles where many side chains are tolerated. It is possible that a structural rearrangement in the D V_H98 W mutant adds additional interactions across the interface but that these structural changes are not correctly modeled in our simulations.

For the Fen49 dataset, the authors identified two key mutations, A78V and A172I, that led to ~100-fold improvement in binding affinity to fentanyl, but none of the design methods tested here found both mutations (Figure S6). These two positions are located in the binding pocket and enriched in larger hydrophobic residues in the selection, presumably to provide additional surface complementarity with fentanyl.⁴⁸ While all design methods did substitute larger hydrophobic side chains, only FastDesign ranked 172I highly, and only BackrubEnsemble ranked 78V highly. CoupledMoves selected 78V at a lower frequency. No method identified the combination of A78V and A172I. While there is no crystal structure with these mutations, we hypothesize that packing I172 against the phenyl ring of fentanyl may be inaccessible to the fentanyl conformer of Fen49, and modeling ligand flexibility might enable design to converge on I172. Unlike position 172, which is at a suitable distance for van der Waals interactions with fentanyl, there is an almost 6 Å distance between the closest heavy atoms of position 78 and the ligand, and the position has a relatively large solvent-accessible surface area. It is therefore unsurprising that Rosetta is unable to arrive at a consensus for this position. The inability of all methods to find the key mutations in Fen49 may represent shortcomings in modeling ligand flexibility. In addition, the Fen49 deep sequencing results are incomplete due to experimental limitations (eg, the original Fen49 side chains were present in the selection but did not have frequency counts⁴⁸).

3.6 | CoupledMoves improves prediction of substrate specificity-altering mutations

The enzyme specificity dataset provides an opportunity to analyze functionally important mutations, as the dataset is made up of pairs of structures where individual point mutations have been experimentally characterized that switch ligand-binding specificity between two ligands.⁴² To determine to what extent the different flexible backbone methods can recapitulate these experimentally characterized specificity-switching mutations, we carried out design simulations on structures with either the original or the new ligand in the binding pocket and designing positions in the vicinity of the ligand substructure change, as described previously⁴² (Tables S3 and S4). To design for mutations switching specificity to the new ligand, we prepared the input structure by computationally substituting the new ligand into the binding pocket of the wild-type protein crystal structure. For the inverse, we swapped the wild-type ligand into the binding pocket of the mutant crystal structure (see Methods).

Some enzymes in this dataset have multiple experimentally-characterized mutations, either a single position to multiple identities (Protein Data Bank (PDB) codes: 1 K70, 3KZO), or multiple positions (PDB: 1A80, 3HG5), for a total of 29 cases (12 wild-type and 17 mutant side chains). The CoupledMoves methods (CM-BR, CM-FKIC, CM-WKIC) correctly identify (positive percent enrichment, see Methods) 14, 11, and 12 specificity-determining mutations, respectively, while the noncoupled methods (FastDesign, BackrubEnsemble, FixBB) identify only 7, 7, and 5 mutations, respectively (Table 2). All CoupledMoves methods identify specificity-altering mutations with a better percentile and rank than the noncoupled methods (Tables 2 and 3), with the original CM-BR attaining the best median and quartile performance, and FastDesign and BackrubEnsemble performing similarly poorly.

3.7 | Noncoupled methods more frequently make incorrect predictions where correct side chain residues are lost, while CoupledMoves most frequently makes predictions where correct residues are gained

We next considered how the sequence of the input structure influences method performance. Only positions with low and medium entropy (< 0.67) in the known profile are considered. Three broad scenarios can be distinguished (Figure 6, top panels). In the first scenario (“loss”), the input side chain (the residue in the starting structure used for design) is present or even preferred in the known sequence profile but is depleted in the design simulations. In the second scenario (“gain”), the input side chain and the known position profile are dissimilar, but preferred side chains are enriched by design. The third scenario occurs when design results in little change of similarity to the known profile (“neutral”). When plotting the PPS values for each method as a function of profile similarity to the input, loss occurs more frequently for positions designed by BackrubEnsemble, FastDesign, and FixBB, whereas gain occurs more frequently for positions designed by CoupledMoves and BackrubEnsemble (Figure 6A, middle and bottom panels, Table S11).

We also performed a similar analysis for the RankTop values. We defined “loss” as the case where a correct starting amino acid side chain is ranked worse than 5 in the final profile and gain as the case when the known top amino acid side chain is not present in the starting sequence and design models it with a rank of 15 or better (Figure 6B, top panel). We only

observed loss for positions designed by BackrubEnsemble, FastDesign, and FixBB (Figure 6B, middle and bottom panels). CoupledMoves achieves gain with the best median and quartile RankTop values (Figure 6B, middle panel), and for the greatest number of positions (Figure 6B, bottom panel). Positions are more likely to remain neutral than to experience gain or loss (Figure 6, bottom panels, Table S11), thus positions with near-correct input sequence tend to maintain higher PPS values. This observation offers an explanation for the comparatively poor PPS and RankTop values of all methods for the DIG10, Fen49 and hGH/hGHR datasets (Figure 3), which are characterized by low similarity between each dataset's input sequence and known profile (Figure S11).

We then asked which methods best predict side chains at positions deemed both functionally relevant (consensus) and difficult (requiring gain). We find that CoupledMoves is more likely than noncoupled methods to enrich for correct side chains not present in the input, with 1.2- and 1.5-fold increase in number of positions experiencing gain, compared to BackrubEnsemble and FastDesign, respectively (Table S11). In addition, CoupledMoves most consistently avoids loss (0.22- and 0.30-fold decrease in number of positions experiencing loss, compared to BackrubEnsemble and FastDesign, respectively), and retention of correct input side chains (neutral scenario) contributes to overall performance. Taken together, the overall best performance of CoupledMoves arises both from increasing the number of positions with gain and decreasing the number of positions experiencing loss.

We also classified positions as polar/charged or hydrophobic based on the most preferred side chain in the known sequence profile, and use this classification to evaluate performance in recapitulating polar contacts vs hydrophobic packing. CoupledMoves outperforms BackrubEnsemble and FastDesign in discovering and retaining correct side chains at both polar/charged and hydrophobic positions (Table S11).

3.8 | Selected structural examples

At the Herceptin/HER2 interface, arginine at position V_H50 (RV_H50) is one of four positions (the other three are YV_H56 , WV_H95 , and YV_H100a) where CoupledMoves maintains a consensus side chain that is completely lost by one or more noncoupled methods (Figure S7). In the crystal structure, RV_H50 forms a hydrogen bond network across the Herceptin/HER2 interface by interacting with Herceptin TV_L94 and HER2 E273 and D275. CoupledMoves is able to retain RV_H50 , while FastDesign and BackrubEnsemble replace this residue with hydrophobic residues, predominantly methionine and glycine, respectively (Figure 7A).

Hydrogen bonds between digoxigenin and the designed protein are most frequently retained by CoupledMoves. In the crystal structure of DIG10.2 (the digoxigenin binder designed with knowledge from the results of the experimental library screen¹⁵), tyrosines 34, 101, and 115 hydrogen bond with digoxigenin, as designed.¹⁵ CoupledMoves frequently chooses tyrosine at all three positions (Figure 7B, top), whereas FastDesign often models only one interaction correctly (Figure 7B, middle), and BackrubEnsemble models two (Figure 7B, bottom). At position 115, BackrubEnsemble most frequently models asparagine, which is too short to hydrogen bond with digoxigenin. FastDesign most frequently models leucine, not tyrosine, at position 115, and instead models tyrosine at nearby position 11 (alanine consensus in

experiment), forming an alternative hydrogen bond with the ester oxygen rather than carbonyl oxygen of the nearby digoxigenin ring.

A third structural example for loss is found in the binding site for cofactor flavin-adenine dinucleotide (FAD) in glutathione reductase (Figure 7C). The majority of natural glutathione reductases coordinate FAD with glutamate at position 50 (E50) and aspartate at position 331 (D331). These side chains are frequently maintained by CoupledMoves, but not by FastDesign or BackrubEnsemble (Figure 7D). Models generated by CoupledMoves agree with the input crystal structure (3DK9), in which E50 forms a hydrogen bond network with two hydroxyl groups of the 3–4-dihydroxy-furan moiety of FAD. CoupledMoves also predicts a hydrogen bond between evolutionarily conserved residue D331 and a hydroxyl group of FAD. The noncoupled design methods frequently replace both polar side chains with apolar side chains, valine at position E50, and alanine or methionine at D331, eliminating the hydrogen bonds between the protein and the ligand.

3.9 | Computational time

We also evaluated the relative compute time for each of the different methods. We first analyzed how performance depended on the number of trajectories run (Figure S12). This analysis suggested that performance is optimal for Coupled Moves, BackrubEnsemble and FastDesign at 400, 200 and 100 trajectories, respectively, with slight variation between datasets (Figure S12). Since each BackrubEnsemble and FastDesign trajectory takes approximately 2-fold and 20-fold more time than CoupledMoves, respectively, CoupledMoves requires substantially less compute time than FastDesign and about equal compute time to BackrubEnsemble (Table S12).

4 | DISCUSSION

We demonstrate that CoupledMoves recapitulates known sequence profiles at designed positions more accurately than the FastDesign and BackrubEnsemble methods. We consider two conceptual categories of positions: (a) important for function and (b) difficult to design. For the first category, we classify positions as important for function (in this case binding) either by proxy of low sequence entropy in the known sequence profile, or if specific mutations have been experimentally determined to be important, as in the enzyme specificity dataset. CoupledMoves most accurately predicts low entropy consensus positions for all profile benchmarks (Figure 5) and outperforms the other methods in correctly identifying specificity-switching mutations in the enzyme specificity set (Tables 2 and 3). For the second category, we designate positions as difficult to design if the most frequent amino acid side chain in the known profile is not present in the structure used as input for design. Considering both low and medium entropy positions, CoupledMoves is more likely than BackrubEnsemble and FastDesign to correctly identify both charged/polar and hydrophobic side chain residues at higher frequency than in the input sequence (gain), while FastDesign is least likely to model a preferred side chain residue present in the input sequence (loss) (Figure 6, Table S11, Figure S11). We conclude that CoupledMoves is best able to predict both residues that are important for function and difficult to design in our datasets.

To provide insights into why the different methods model consensus side chains with different frequencies, despite using the same energy function, we analyzed how the correct amino acid at these positions was ranked by energy for each of the different methods. Figure 8A shows distributions of percentiles for predicted total Rosetta energy of instances where a method models the known top ranked amino acid side chain. These distributions are shifted towards higher percentiles for CoupledMoves compared to the other methods. CM-FKIC predicts the consensus side chain for 51 positions with total energy above the 75th percentile, while BackrubEnsemble and FastDesign predict 37 and 27 positions in the same category. CoupledMoves models the consensus side chain for a total of 132 designable positions in the datasets, compared to 111 and 95 positions for BackrubEnsemble and FastDesign, respectively. The high sequence entropy of CoupledMoves design compared to other methods (Figure S10) makes it even more remarkable that CoupledMoves ranks the energetics of consensus side chains so favorably among many options. We conclude that, for side chains modeled with >0.33 frequency and >75 th energy percentile, CoupledMoves predictions are likely correct.

In cases where the BackrubEnsemble method does model the consensus side chain during design, the energetics rank favorably (Figure 8A). One possible reason for the overall worse performance of BackrubEnsemble over CoupledMoves is that cases correctly predicted by BackrubEnsemble might be derived from only a subset of ensemble members whose backbone conformations are compatible with energetically favorable placement of the consensus side chain. In these cases, the input/consensus side chain is compatible with the ensemble, but during sequence design another amino acid side chain has more favorable Rosetta energy. Indeed, forcing the consensus side chains onto all ensemble members results in a greater proportion of models with unfavorable (positive) Rosetta energy, and a smaller proportion of models with highly favorable energy (Figure S13, shown are glutathione reductase and digoxigenin binder, which are examples of loss by the BackrubEnsemble method). This behavior suggests that ensemble members are not uniformly compatible with consensus sidechains, highlighting a limitation of the BackrubEnsemble method. Backbone moves are sampled only once, at the beginning of the trajectory during ensemble creation (Figure 1A). Sidechains are subsequently modeled onto each ensemble member by finding an energetically favorable rotamer for the pre-determined backbone conformation. In contrast, the CoupledMoves design trajectory cycles small backbone adjustments in response to sequence change moves, which allows switching from nonconsensus to consensus side chains. Without cycles of backbone and sidechain sampling, the BackrubEnsemble method is limited to snapshots of the allowed backbone conformational diversity defined by the initial ensemble members.

For CoupledMoves, the design frequency increases with energy percentile for consensus side chains (Figure 8B), which is expected-side chains with a higher (more favorable) energy percentile should be chosen more frequently. However, this trend is less pronounced for both BackrubEnsemble and FastDesign. For BackrubEnsemble, this behavior is possibly due to the limitations enforced by the backbone conformations of the ensemble. In the case of FastDesign, it is possible that the minimization step in FastDesign is prone to trapping the design simulations in local minima and hence that the frequency of chosen amino acids poorly reflect their actual fitness rank. This hypothesis is supported by the low entropy of

FastDesign design sequence profiles (Figure S10). FastDesign may be less likely to escape local minima than the other methods, despite the use of a reduced Lennard-Jones repulsive term in the early cycles of the simulation (Figure 1A).

In addition to limitations in sampling methods (as well as the energy function used to rank designs), there are potential limitations related to the benchmark datasets. While this work considers protein/protein and protein/small molecule interactions, the selection of benchmarks could be expanded to include protein/peptide interactions, such as those described in References 40, 54, and 55. There are also potential limitations inherent to some datasets included in this study. For example, in the case of the enzyme specificity dataset, we can only compare to the point mutations that were experimentally tested, but we do not have sequence profiles. The enzymes have not been subject to saturation mutagenesis, so it is unknown whether additional specificity-altering mutations could exist.

Sequence profiles in the cofactor dataset result from natural evolution, rather than experimental screening. Natural evolution includes selection pressures beyond affinity (catalysis, specificity, kinetics etc.), so that the sequence profiles for natural binding site positions may be influenced by factors beyond those modeled here. In addition, our analysis does not evaluate covariation between residue positions. However, evolutionary sequence profiles have the advantage of clearly identifying consensus binding positions, and we observe considerable agreement between Rosetta predicted and known sequence profiles for this set.

Finally, all methods tested perform most poorly at consensus positions in the deep sequencing datasets, DIG10 and Fen49, and the design methods perform worse than the null model on DIG10. Initiating design from the crystal structure corresponding to the result of the library selection (PDB: 4J8T) did not improve performance. It is possible that the selection experiments report on additional considerations such as expression and display on the yeast surface that are not considered in the design simulations, or that the sensitivity range of the selection is tuned to primarily differentiate between functional vs deleterious mutations but is less capable of quantitatively ranking binding affinity. Alternatively, critical adjustments of both backbones and the ligand, in addition to ligand strain and ligand flexibility, are not correctly captured in the Rosetta simulations.

Apart from suggesting individual point mutations such as in the enzyme specificity set, our results on recapitulating position-specific sequence profiles highlight the utility of CoupledMoves for generating libraries. CoupledMoves will be most useful in design cases where protein backbones are supplied with existing side chains, such as natural or previously-characterized designed proteins (rather than the de novo design of new structures). Computation has long been used to reduce the sequence space queried by library screens,⁵⁶⁻⁵⁸ and it is well established that flexible-backbone protein design can generate sequences similar to observed natural and experimental sequences.^{27,28,30,31,36,41,59,60} As the design results obtained with CoupledMoves most accurately reflect tolerated sequence space in comparison to other methods using the same energy function, CoupledMoves represents a powerful flexible backbone strategy for generating combinatorial libraries for screening and selection, and optimizing proteins for new and useful functions.

Computational design of binding sites remains difficult, in part due to limitations in current ability to realistically sample backbone conformations that enable side chains to make realistic contacts during sequence design. The benchmarking framework presented here can be adapted to different types of design applications, such as sequence design on parametrically-generated rather than natural protein backbones, or de novo designed rather than reengineered binding sites.

Supplementary Material

Refer to Web version on PubMed Central for supplementary material.

ACKNOWLEDGMENTS

This work was supported by grants from the National Institutes of Health (NIH R01-GM110089) and the National Science Foundation (NSF DBI-1564692) to TK. The authors would like to thank Kale Kundert and Noah Ollikainen for helpful discussions, and Christy Tinberg and Matt Bick for sharing data.

Funding information

National Science Foundation, Grant/Award Number: DBI-1564692; National Institutes of Health, Grant/Award Number: R01-GM110089

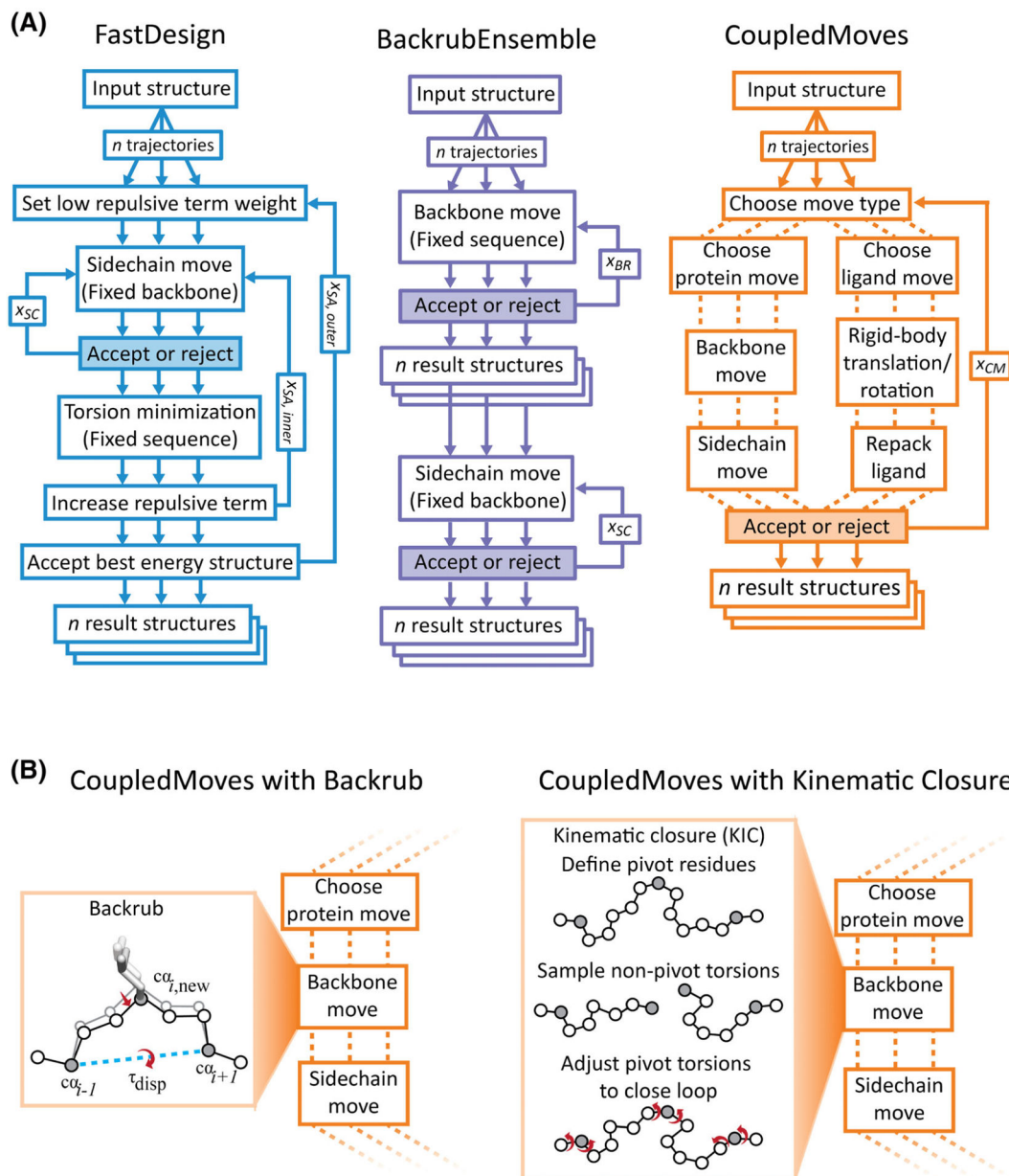
REFERENCES

1. Alford RF, Leaver-Fay A, Jeliaskov JR, et al. The Rosetta all-atom energy function for macromolecular Modeling and design. *J Chem Theory Comput.* 2017;13(6):3031–3048. [PubMed: 28430426]
2. Gordon DB, Marshall SA, Mayo SL. Energy functions for protein design. *Curr Opin Struct Biol.* 1999;9(4):509–513. [PubMed: 10449371]
3. Dahiyat BI, Mayo SL. De novo protein design: fully automated sequence selection. *Science.* 1997;278(5335):82–87. [PubMed: 9311930]
4. Ponder JW, Richards FM. Tertiary templates for proteins. Use of packing criteria in the enumeration of allowed sequences for different structural classes. *J Mol Biol.* 1987;193(4):775–791. [PubMed: 2441069]
5. Baldwin EP, Hajiseyedjavadi O, Baase WA, et al. The role of backbone flexibility in the accommodation of variants that repack the core of T4 lysozyme. *Science.* 1993;262(5140):1715–1718. [PubMed: 8259514]
6. Davis IW, Arendall WB 3rd, Richardson DC, et al. The backrub motion: how protein backbone shrugs when a sidechain dances. *Structure.* 2006;14(2):265–274. [PubMed: 16472746]
7. Desjarlais JR, Handel TM. Side-chain and backbone flexibility in protein core design. *J Mol Biol.* 1999;290(1):305–318. [PubMed: 10388574]
8. Keedy DA, Georgiev I, Triplett EB, et al. The role of local backrub motions in evolved and designed mutations. *PLoS Comput Biol.* 2012; 8(8):e1002629. [PubMed: 22876172]
9. Koga N, Tatsumi-Koga R, Liu G, et al. Principles for designing ideal protein structures. *Nature.* 2012;491(7423):222–227. [PubMed: 23135467]
10. Rocklin GJ, Chidyausiku TM, Goresnik I, et al. Global analysis of protein folding using massively parallel design, synthesis, and testing. *Science.* 2017;357(6347):168–175. [PubMed: 28706065]
11. Correia BE, Bates JT, Loomis RJ, et al. Proof of principle for epitope-focused vaccine design. *Nature.* 2014;507(7491):201–206. [PubMed: 24499818]
12. Jiang L, Althoff EA, Clemente FR, et al. De novo computational design of retro-aldol enzymes. *Science.* 2008;319(5868):1387–1391. [PubMed: 18323453]

13. Procko E, Berguig GY, Shen BW, et al. A computationally designed inhibitor of an Epstein-Barr viral Bcl-2 protein induces apoptosis in infected cells. *Cell*. 2014;157(7):1644–1656. [PubMed: 24949974]
14. Rothlisberger D, Khersonsky O, Wollacott AM, et al. Kemp elimination catalysts by computational enzyme design. *Nature*. 2008;453 (7192):190–195. [PubMed: 18354394]
15. Tinberg CE, Khare SD, Dou J, et al. Computational design of ligand-binding proteins with high affinity and selectivity. *Nature*. 2013;501 (7466):212–216. [PubMed: 24005320]
16. Dou J, Vorobieva AA, Sheffler W, et al. De novo design of a fluorescence-activating beta-barrel. *Nature*. 2018;561(7724): 485–491. [PubMed: 30209393]
17. Silva DA, Yu S, Ulge UY, et al. De novo design of potent and selective mimics of IL-2 and IL-15. *Nature*. 2019;565(7738):186–191. [PubMed: 30626941]
18. Fleishman SJ, Whitehead TA, Ekiert DC, et al. Computational design of proteins targeting the conserved stem region of influenza hemagglutinin. *Science*. 2011;332(6031):816–821. [PubMed: 21566186]
19. Dou J, Doyle L, Jr Greisen P, et al. Sampling and energy evaluation challenges in ligand binding protein design. *Protein Sci*. 2017;26(12): 2426–2437. [PubMed: 28980354]
20. Kundert K, Kortemme T. Computational design of structured loops for new protein functions. *Biol Chem*. 2019;400(3):275–288. [PubMed: 30676995]
21. Davey JA, Chica RA. Improving the accuracy of protein stability predictions with multistate design using a variety of backbone ensembles. *Proteins*. 2014;82(5):771–784. [PubMed: 24174277]
22. Davey JA, Chica RA. Multistate computational protein design with backbone ensembles. *Methods Mol Biol*. 2017;1529:161–179. [PubMed: 27914050]
23. Larson SM, England JL, Desjarlais JR, Pande VS. Thoroughly sampling sequence space: large-scale protein design of structural ensembles. *Protein Sci*. 2002;11(12):2804–2813. [PubMed: 12441379]
24. Fu X, Apgar JR, Keating AE. Modeling backbone flexibility to achieve sequence diversity: the design of novel alpha-helical ligands for Bcl-xL. *J Mol Biol*. 2007;371(4):1099–1117. [PubMed: 17597151]
25. Georgiev I, Keedy D, Richardson JS, Richardson DC, Donald BR. Algorithm for backrub motions in protein design. *Bioinformatics*. 2008;24 (13):i196–i204. [PubMed: 18586714]
26. Smith CA, Kortemme T. Backrub-like backbone simulation recapitulates natural protein conformational variability and improves mutant side-chain prediction. *J Mol Biol*. 2008;380(4): 742–756. [PubMed: 18547585]
27. Ding F, Dokholyan NV. Emergence of protein fold families through rational design. *PLoS Comput Biol*. 2006;2(7):e85. [PubMed: 16839198]
28. Friedland GD, Lakomek NA, Griesinger C, Meiler J, Kortemme T. A correspondence between solution-state dynamics of an individual protein and the sequence and conformational diversity of its family. *PLoS Comput Biol*. 2009;5(5):e1000393. [PubMed: 19478996]
29. Friedland GD, Linares AJ, Smith CA, Kortemme T. A simple model of backbone flexibility improves modeling of side-chain conformational variability. *J Mol Biol*. 2008;380(4):757–774. [PubMed: 18547586]
30. Humphris EL, Kortemme T. Prediction of protein-protein interface sequence diversity using flexible backbone computational protein design. *Structure*. 2008;16(12):1777–1788. [PubMed: 19081054]
31. Humphris-Narayanan E, Akiva E, Varela R, Ó Conchúir S, Kortemme T. Prediction of mutational tolerance in HIV-1 protease and reverse transcriptase using flexible backbone protein design. *PLoS Comput Biol*. 2012;8(8):e1002639. [PubMed: 22927804]
32. Leaver-Fay A, Tyka M, Lewis SM, et al. ROSETTA3: an object-oriented software suite for the simulation and design of macromolecules. *Methods Enzymol*. 2011;487:545–574. [PubMed: 21187238]
33. Simons KT, Bonneau R, Ruczinski I, et al. Ab initio protein structure prediction of CASP III targets using ROSETTA. *Proteins*. 1999;3: 171–176. [PubMed: 10526365]

34. Simons KT, Kooperberg C, Huang E, Baker D. Assembly of protein tertiary structures from fragments with similar local sequences using simulated annealing and Bayesian scoring functions. *J Mol Biol.* 1997; 268(1):209–225. [PubMed: 9149153]
35. Kuhlman B, Dantas G, Ireton GC, Varani G, Stoddard BL, Baker D. Design of a novel globular protein fold with atomic-level accuracy. *Science.* 2003;302(5649):1364–1368. [PubMed: 14631033]
36. Saunders CT, Baker D. Recapitulation of protein family divergence using flexible backbone protein design. *J Mol Biol.* 2005;346(2): 631–644. [PubMed: 15670610]
37. Coutsias EA, Seok C, Jacobson MP, Dill KA. A kinematic view of loop closure. *J Comput Chem.* 2004;25(4):510–528. [PubMed: 14735570]
38. Mandell DJ, Coutsias EA, Kortemme T. Sub-angstrom accuracy in protein loop reconstruction by robotics-inspired conformational sampling. *Nat Methods.* 2009;6(8):551–552. [PubMed: 19644455]
39. Ollikainen N, Kortemme T. Computational protein design quantifies structural constraints on amino acid covariation. *PLoS Comput Biol.* 2013;9(11):e1003313. [PubMed: 24244128]
40. Smith CA, Kortemme T. Structure-based prediction of the peptide sequence space recognized by natural and synthetic PDZ domains. *J Mol Biol.* 2010;402(2):460–474. [PubMed: 20654621]
41. Smith CA, Kortemme T. Predicting the tolerated sequences for proteins and protein interfaces using RosettaBackrub flexible backbone design. *PLoS One.* 2011;6(7):e20451. [PubMed: 21789164]
42. Ollikainen N, de Jong RM, Kortemme T. Coupling protein side-chain and backbone flexibility improves the re-design of protein-ligand specificity. *PLoS Comput Biol.* 2015;11(9):e1004335. [PubMed: 26397464]
43. Chevalier A, Silva DA, Rocklin GJ, et al. Massively parallel de novo protein design for targeted therapeutics. *Nature.* 2017;550(7674):74–79. [PubMed: 28953867]
44. Kapp GT, Liu S, Stein A, et al. Control of protein signaling using a computationally designed GTPase/GEF orthogonal pair. *Proc Natl Acad Sci USA.* 2012;109(14):5277–5282. [PubMed: 22403064]
45. Khatib F, Cooper S, Tyka MD, et al. Algorithm discovery by protein folding game players. *Proc Natl Acad Sci U S A.* 2011;108(47): 18949–18953. [PubMed: 22065763]
46. Tyka MD, Keedy DA, Andre I, et al. Alternate states of proteins revealed by detailed energy landscape mapping. *J Mol Biol.* 2011;405(2): 607–618. [PubMed: 21073878]
47. Park H, Bradley P, Greisen P Jr, et al. Simultaneous optimization of biomolecular energy functions on features from small molecules and macromolecules. *J Chem Theory Comput.* 2016;12(12): 6201–6212. [PubMed: 27766851]
48. Bick MJ, Greisen PJ, Morey KJ, et al. Computational design of environmental sensors for the potent opioid fentanyl. *Elife.* 2017;6: e28909. [PubMed: 28925919]
49. Cunningham BC, Wells JA. Comparison of a structural and a functional epitope. *J Mol Biol.* 1993;234(3):554–563. [PubMed: 7504735]
50. Pal G, Kouadio JL, Artis DR, et al. Comprehensive and quantitative mapping of energy landscapes for protein-protein interactions by rapid combinatorial scanning. *J Biol Chem.* 2006;281(31): 22378–22385. [PubMed: 16762925]
51. Babor M, Mandell DJ, Kortemme T. Assessment of flexible backbone protein design methods for sequence library prediction in the therapeutic antibody Herceptin-HER2 interface. *Protein Sci.* 2011;20(6):1082–1089. [PubMed: 21465611]
52. Gerstner RB, Carter P, Lowman HB. Sequence plasticity in the antigen-binding site of a therapeutic anti-HER2 antibody. *J Mol Biol.* 2002;321(5):851–862. [PubMed: 12206766]
53. Ollikainen N, Smith CA, Fraser JS, et al. Flexible backbone sampling methods to model and design protein alternative conformations. *Methods Enzymol.* 2013;523:61–85. [PubMed: 23422426]
54. King CA, Bradley P. Structure-based prediction of protein-peptide specificity in Rosetta. *Proteins.* 2010;78(16):3437–3449. [PubMed: 20954182]

55. Rubenstein AB, Pethe MA, Khare SD. MFPred: rapid and accurate prediction of protein-peptide recognition multispecificity using self-consistent mean field theory. *PLoS Comput Biol*. 2017;13(6): e1005614. [PubMed: 28650961]
56. Hayes RJ, Bentzien J, Ary ML, et al. Combining computational and experimental screening for rapid optimization of protein properties. *Proc Natl Acad Sci U S A*. 2002;99(25):15926–15931. [PubMed: 12446841]
57. Treynor TP, Vizcarra CL, Nedelcu D, et al. Computationally designed libraries of fluorescent proteins evaluated by preservation and diversity of function. *Proc Natl Acad Sci U S A*. 2007;104(1):48–53. [PubMed: 17179210]
58. Voigt CA, Mayo SL, Arnold FH, Wang ZG. Computational method to reduce the search space for directed protein evolution. *Proc Natl Acad Sci U S A*. 2001;98(7):3778–3783. [PubMed: 11274394]
59. Koehl P, Levitt M. Protein topology and stability define the space of allowed sequences. *Proc Natl Acad Sci U S A*. 2002;99(3):1280–1285. [PubMed: 11805293]
60. Larson SM, Garg A, Desjarlais JR, Pande VS. Increased detection of structural templates using alignments of designed sequences. *Proteins*. 2003;51(3):390–396. [PubMed: 12696050]

**FIGURE 1.**

Design methods. A, Design method comparison. The FastDesign (left, blue) and BackrubEnsemble (middle, purple) methods separate sequence design steps (using a fixed backbone) from backbone optimization steps (using a fixed sequence). CoupledMoves (right, orange) evaluates combined moves that sample both backbone conformation and amino acid sequence (or, alternatively, combine ligand translations/rotations with changes of ligand conformers). FastDesign performs five outer ($x_{SA,outer}$) simulated annealing cycles, during which the weight of the Lennard-Jones repulsive energy term is ramped from 2% to 100%. For each ramped weight, an inner cycle ($x_{SA,inner}$) consists of a complete round of sequence design with x_{SC} steps on a fixed backbone, followed by a step that minimizes backbone, sidechain, and ligand torsion angles. BackrubEnsemble performs 10,000 (x_{BR})

Backrub moves to generate each ensemble member. For both FastDesign and BackrubEnsemble, x_{SC} scales with the number of possible moves, and is equal to 10 times the number of possible rotamers at all designable or repackable positions. CoupledMoves performs 1,000 trials (x_{CM}) per trajectory. B, Original and updated backbone mover in CoupledMoves. The original CoupledMoves method⁴² uses the Backrub algorithm to make backbone moves. A backrub move^{6,26} rotates a segment as a rigid body by displacement angle τ_{disp} around an axis between two pivot C α atoms 2–11 residues apart (shown is a 2-residue move). In the updated versions of CoupledMoves introduced here, backbone moves are made using a Kinematic Closure algorithm.³⁸ Backbone torsion angles for nonpivot C α atoms are perturbed either using fragment insertion (FKIC) or by small perturbations away from the existing angles (WKIC), then the loop is closed by analytical determination of Φ and Ψ angles (red) at three pivot C α atoms (gray). FKIC, Kinematic Closure using fragments; WKIC, Kinematic Closure using “walking” perturbation

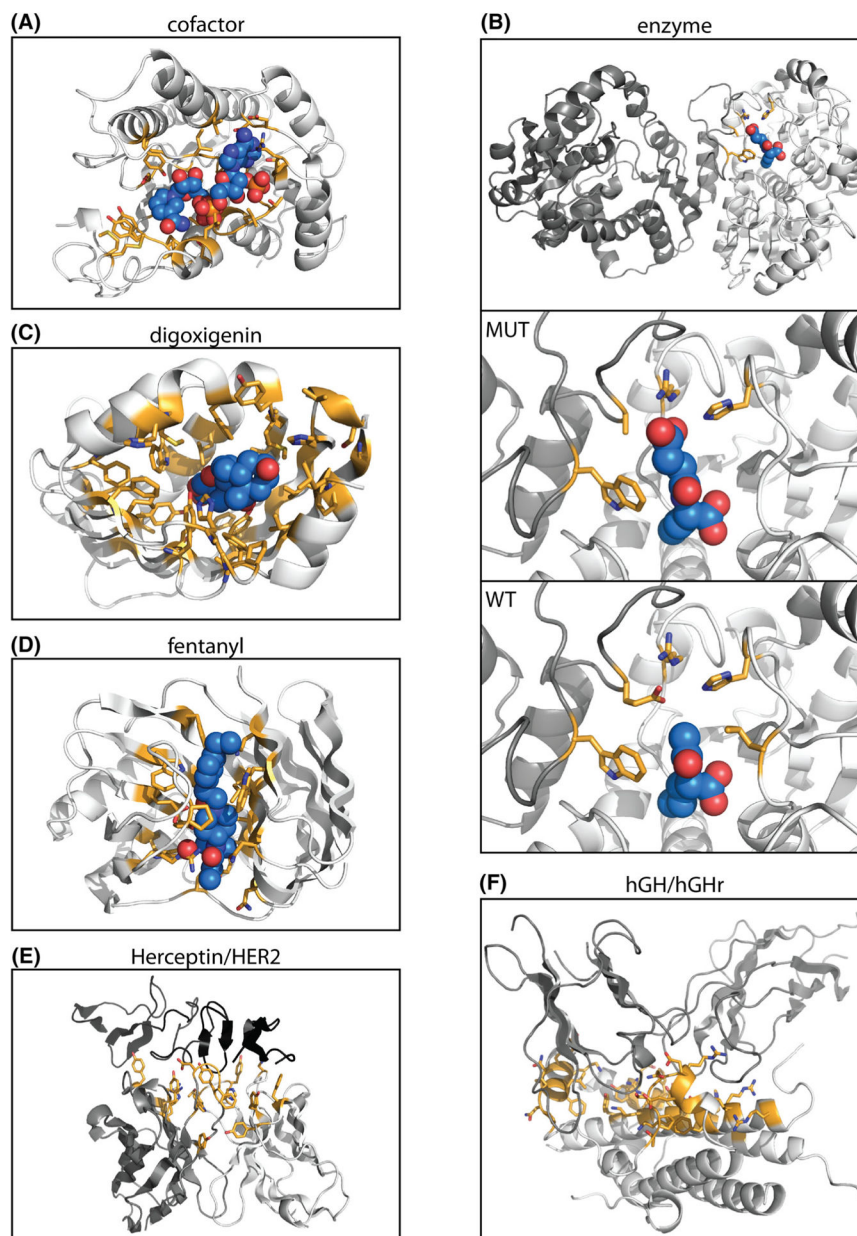
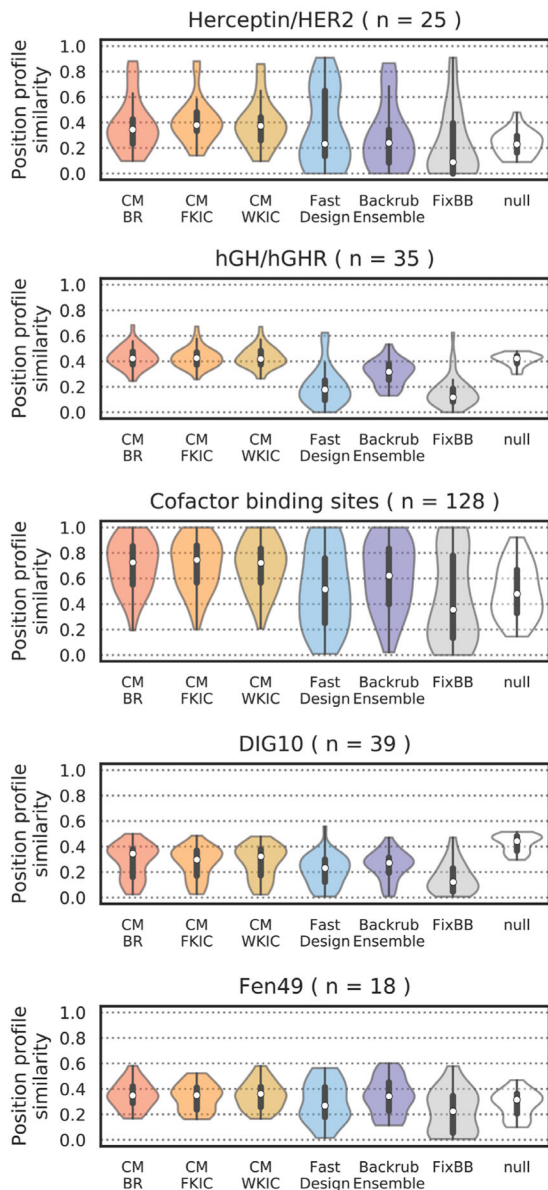
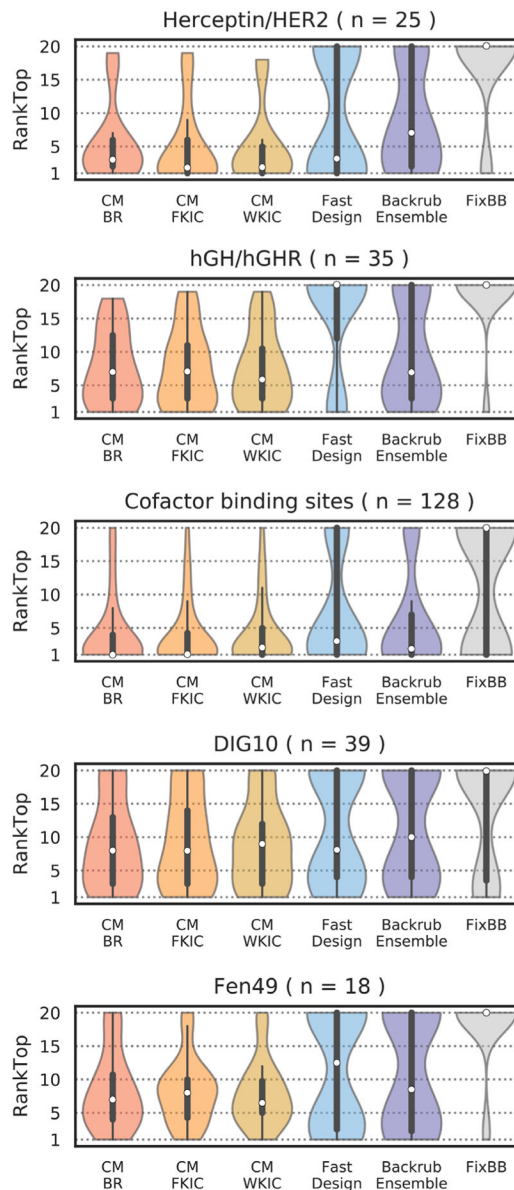


FIGURE 2. Benchmark dataset structures. Side chains at designed positions are highlighted in orange and shown as sticks. Ligands are colored light blue and shown in sphere representation. The structures shown are those used as input for design, as described in Methods. Nitrogen atoms are shown in dark blue, and oxygen atoms are shown in red. A, Representative structure from the cofactor dataset, alcohol dehydrogenase with cofactor NAP. Structures for other six protein families are not shown. B, Representative structure from the enzyme dataset, N-acetylornithine carbamoyltransferase. The full structure of the mutant enzyme, with ligand N-(3-carboxypropanoyl-L-norvaline (SN0), is shown in the top panel. The middle panel shows the binding site. The bottom panel shows the binding site of the wild-type protein, with ligand N-acetyl-L-norvaline (AN0). The other nine enzymes are not shown. C,

DIG10.1, the designed digoxigenin binder on which the SSM library was generated and selected, with digoxigenin. D, The wild-type protein used for design of fentanyl binding protein, with fentanyl placed in the binding pocket. E, Herceptin/HER2. Designable positions on the Herceptin antibody light chain (light gray) and heavy chain (dark gray) interact with target HER2 (black). The combination of designable positions from all libraries is shown. F, hGH/hGHR. Designable positions on hGH (light gray) interact with target hGHR. The combination of designable positions from all libraries is shown

(A) Position profile similarity**(B) RankTop****FIGURE 3.**

Comparison of design method performance on sequence profile datasets. PPS (A) and RankTop (B) distributions. A RankTop of 1 means that the design method correctly identified the most frequent amino acid side chain observed in the experimental/natural profile, whereas a RankTop of 20 means that the side chain was observed with zero frequency, or that all other side chains were modeled with some frequency and the top known side chain was the least frequent. The median of the distributions is marked with a white dot. Second and third quartiles are marked by the thick black bar, and the thin bar marks 1.5 times the inter-quartile range. The width of the violins is determined by the number of observations in each bin, and bins are defined using Scott's normal reference rule. The number of sequence positions in each set is described by n

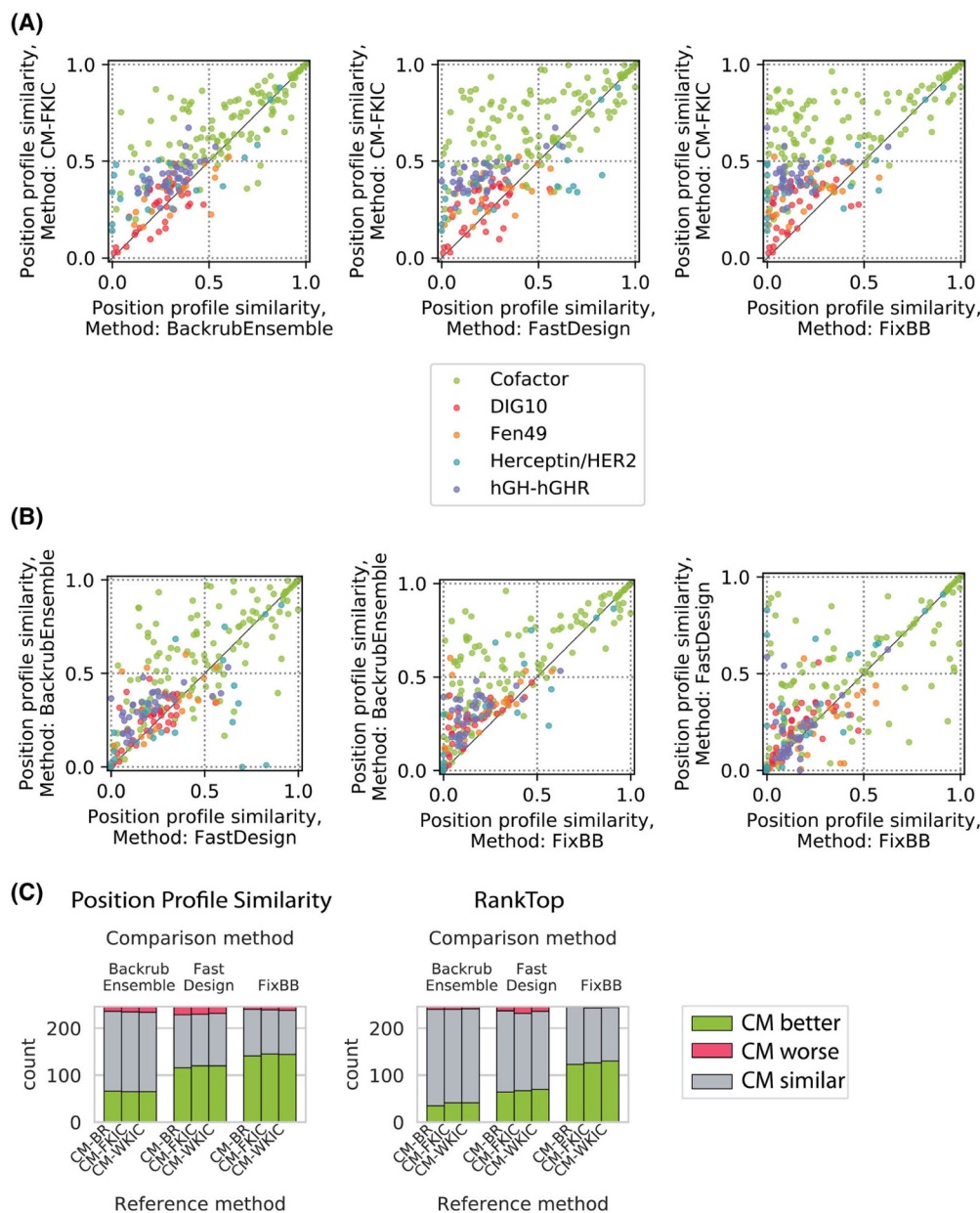


FIGURE 4. Method performance comparison for profile datasets by sequence position. Shown are the same data as in Figure 3, but plotting individual sequence positions instead of distributions. Colors indicate different datasets. A, Comparison between CM-FKIC and noncoupled methods. Points above the diagonal represent positions where CM-FKIC outperforms the noncoupled method. B, Comparison between noncoupled methods, where points above the diagonal represent positions where BackrubEnsemble outperforms FastDesign (left) or FixBB (middle), or where FastDesign outperforms FixBB (right). C, Summary of position counts classified by whether CoupledMoves (“Reference method”) performs better (green), worse (red) or similar (gray) compared to coupled methods (“Comparison method”). The CoupledMoves reference method is “better” or “worse” than the comparison method when

the difference in performance is above or below, respectively, a threshold of ± 0.1 for PPS or ± 5 for RankTop. When the difference is within the threshold, the methods are classified as “similar.” CM-FKIC, CoupledMoves with Kinematic Closure using fragments

Author Manuscript

Author Manuscript

Author Manuscript

Author Manuscript

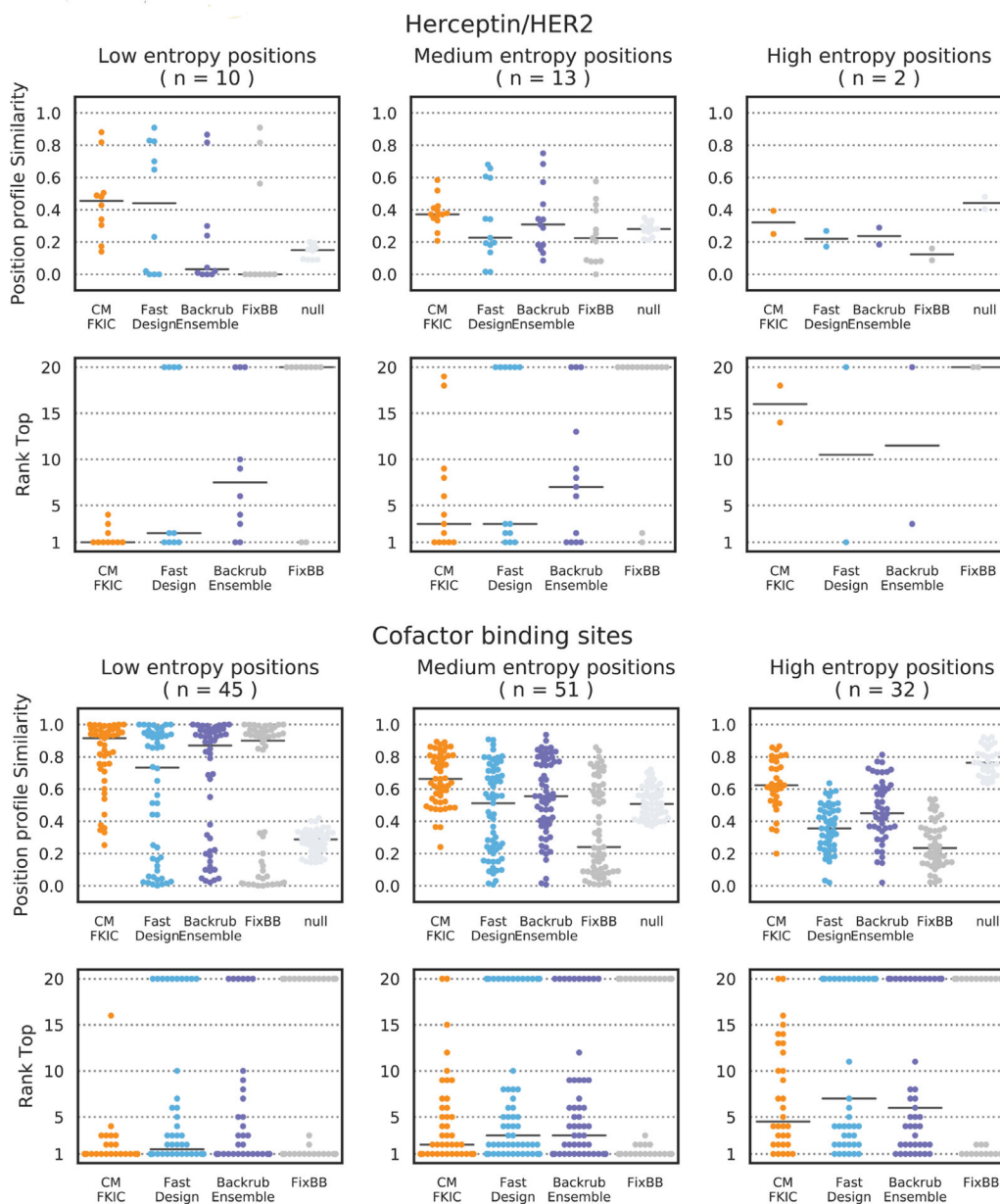
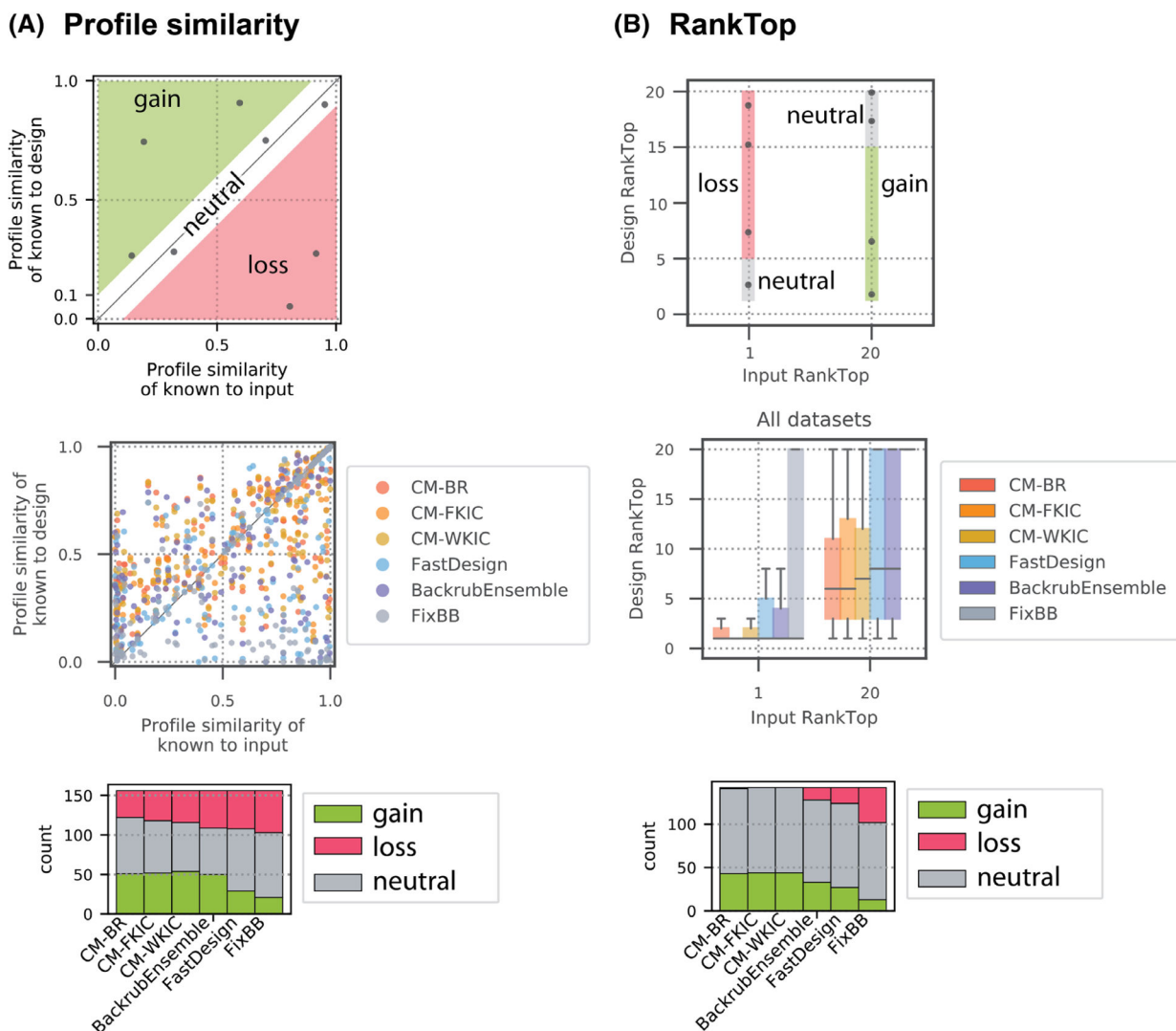


FIGURE 5. PPS and RankTop as a function of known sequence entropy. Each point represents one sequence position. Shown here are the Herceptin/HER2 (top) and Cofactor (bottom) datasets, which have the highest number of medium and low entropy positions. The remaining datasets are shown in Figure S9. For each dataset, PPS and RankTop are binned by entropy of the known sequence profile at each position (low: entropy ≤ 0.33 , medium: $0.33 < \text{entropy} \leq 0.67$, and high: entropy > 0.67). The median is marked with a horizontal black line

**FIGURE 6.**

PPS and RankTop as a function of similarity to input. Gain (green) and loss (red) as defined in the main text. Only positions with low and medium entropy (< 0.67) are considered. This figure combines all datasets; individual datasets are shown in Figure S11. A, PPS as a function of similarity to the input sequence for all profile datasets. Top: Gain and loss zones are defined by a threshold of 0.1 difference between input-known PPS and design-known PPS. Middle: Each point represents one position in the protein sequence, colored by design method. CoupledMoves results (yellow, orange, red) are enriched in the gain zone and FastDesign (blue) and FixBB (gray) results are enriched in the loss zone. Bottom: Quantifications of number of designed sequence positions in gain, loss, and neutral zones for each method. B, RankTop as a function of similarity to the input sequence for all profile datasets, except Fen49, which is omitted because the fentanyl deep sequencing data do not include the input sequence. The top amino acid from the known profile is assigned a rank of 1 if it is present in the input sequence, or a rank of 20 if it is not. Top: A threshold of 5 in the difference in RankTop between input and designed sequences defines the gain and loss zones. Middle: Box plots represent all positions in all datasets, except fentanyl. The median

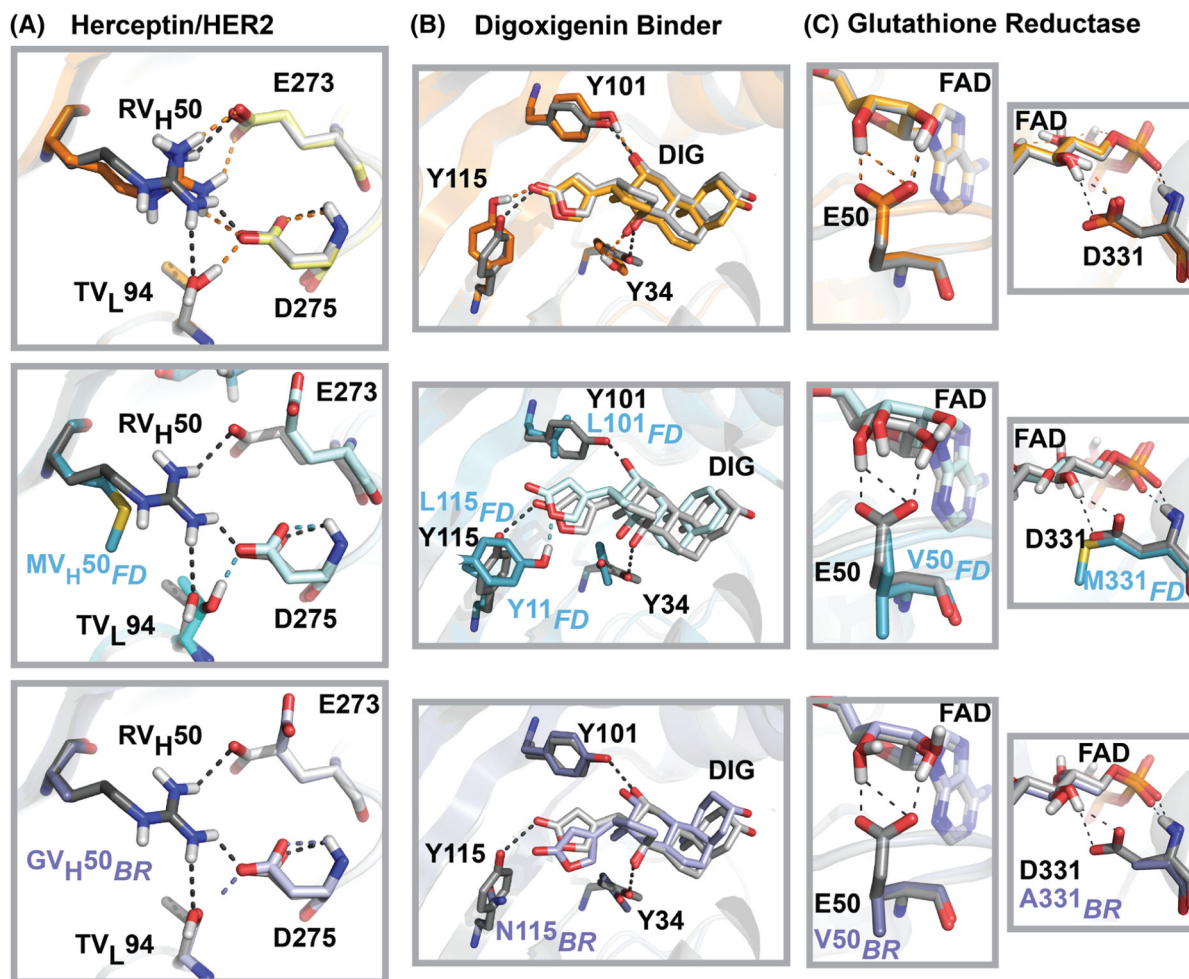
of the distributions is marked with a horizontal line. Second and third quartiles are marked by the box, and the whiskers extend to 1.5 times the inter-quartile range. Bottom: Quantification of sequence positions in gain, loss, and neutral zones for RankTop values

Author Manuscript

Author Manuscript

Author Manuscript

Author Manuscript



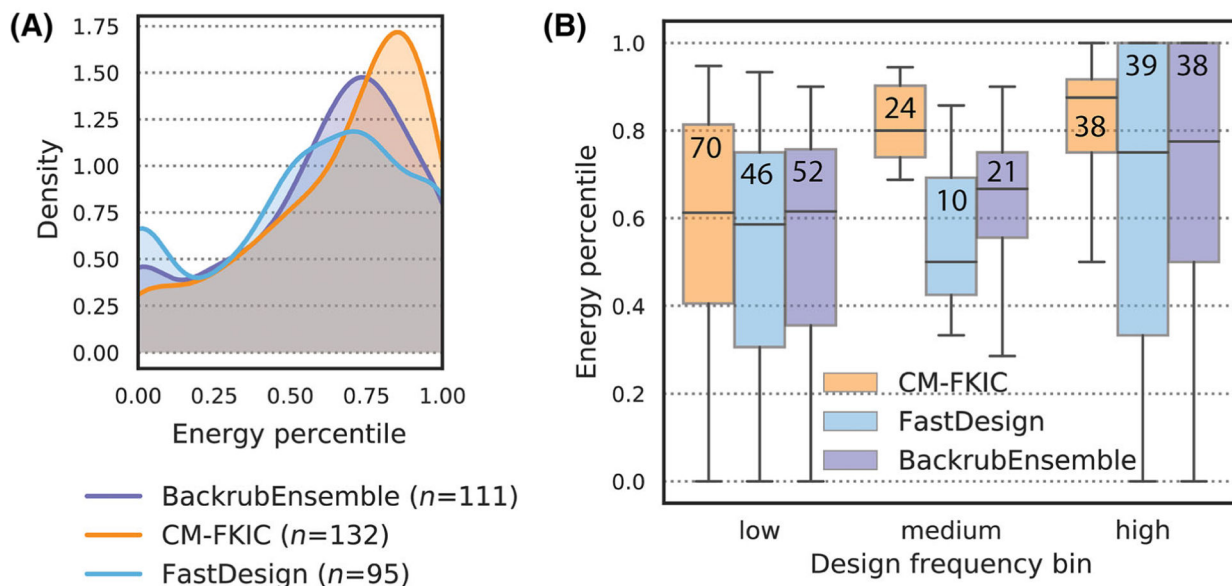
(D) Design frequencies

	Herceptin	Digoxigenin binder			Glutathione Reductase	
	RVH50	Y34	Y101	Y115	E50	D331
CoupledMoves	0.06 ± 0.02	0.39 ± 0.03	0.68 ± 0.05	0.32 ± 0.14	0.90 ± 0.09	0.27 ± 0.11
FastDesign	0.00	0.44	0.12	0.00	0.06	0.00
BackrubEnsemble	0.00	0.44	0.42	0.02	0.07	0.01

FIGURE 7.

Examples of structural models generated by different design methods. Comparison of crystal structures used as input for design (gray) to models generated by CoupledMoves (top, orange), FastDesign (middle, blue), and BackrubEnsemble (bottom, purple). A, The crystal structure of Herceptin/HER2 (PDB: 1N8Z) shows a hydrogen bond network (black dashed lines) spanning the interface between Herceptin residues RV_H50 (dark color) and TV_L94 (medium color), and HER2 residues E273 and D275 (light color). Key designable residue RV_H50 is retained by CoupledMoves, which models a native-like hydrogen bond network (orange dashed lines). In contrast, FastDesign and BackrubEnsemble model reduced networks (blue and purple dashed lines, respectively). Hydrogen atoms for 1N8Z were added using Rosetta. B, Three tyrosines (Y34, Y101, Y115) form a hydrogen bond network

(black dashed lines) with digoxigenin (DIG) in the crystal structure of digoxigenin binder DIG10.2 (PDB: 4J8T). CoupledMoves most frequently retains all three tyrosines and forms a similar network (orange dashed lines). FastDesign frequently models leucines at positions 101 and 115, and instead frequently models tyrosine at position 11, forming a hydrogen bond with the ester oxygen rather than carbonyl oxygen of the nearby digoxigenin ring (blue dashed line). BackrubEnsemble most frequently models asparagine at position 115, while retaining the other two contacts (purple dashed lines). C, In crystal structures, glutamate E50 (left column, PDB: 3DK9) and aspartate D331 (right column, PDB: 6FTC) form a hydrogen bond network with flavin-adenine dinucleotide (FAD) (black dashed lines). CoupledMoves retains E50 and D331 in geometries that maintain the network (orange dashed lines). FastDesign and BackrubEnsemble frequently model hydrophobic residues at these positions, abolishing the network. Hydrogen atoms for 3DK9 were added using Rosetta. D, The frequencies of the top known side chain for each position as designed by the different methods. Values for CoupledMoves represent averages and standard deviations across CM-BR, CM-FKIC, and CM-WKIC. CM-BR CoupledMoves with Backrub; CM-FKIC, CoupledMoves with Kinematic Closure using fragments; CM-WKIC, CoupledMoves with Kinematic Closure using “walking” perturbation

**FIGURE 8.**

Distribution of energy percentiles for correctly modeled positions. “Energy percentile” refers to the percentile of the average total Rosetta energy of the correctly modeled side chain compared to that of all other side chains modeled by the design method at that position. Energy percentile was calculated for consensus (entropy > 0.67) positions for which a method modeled the consensus at least once. A, Distribution of energy percentiles. Count n indicates the number of positions for which each method modeled the consensus side chain at least once. B, Energy percentiles as a function of design frequency are shown as boxplots. Values from (A) are binned by design frequency (low: frequency < 0.33 , medium: $0.33 < \text{frequency} < 0.67$, and high: frequency > 0.67). The number of values in each bin is shown on each boxplot. The median of the distributions is marked with a horizontal line. Second and third quartiles are marked by the box, and the whiskers extend to 1.5 times the interquartile range.

TABLE 1

Benchmark datasets

Binding interaction	Benchmark name	Proteins in the set	Known functional sequences
Protein/small molecule	Cofactor	Seven protein families each binding a different cofactor	Natural sequence alignments, taken from Reference 42
	Enzyme specificity	10 enzymes	Experimentally-characterized point mutations that switch substrate specificity, taken from Reference 42
	DIG10	Computationally designed protein binding digoxigenin	Amino acid frequencies derived from deep sequencing of a site saturation mutagenesis library after three rounds of selection for binding to digoxigenin, from Reference 15
	Fen49	Computationally designed protein binding fentanyl	Amino acid frequencies derived from deep sequencing of a single site saturation mutagenesis library after four rounds of selection for binding to fentanyl, from Reference 48
Protein/protein	hGH/hGHR	Protein-protein complex of human growth hormone (hGH) binding to its receptor (hGHR)	Sequences of clones selected from five combinatorial libraries of hGH (average of 180 sequenced clones per library) after two rounds of selection for binding to hGHR, from Reference 50
	Herceptin/HER2	Protein-protein complex of Herceptin binding to HER2	Sequences of clones selected from four combinatorial libraries of Herceptin (average of 70 sequenced clones per library) after four rounds of selection for binding to HER2, from Reference 52

TABLE 2

Performance summary for the enzyme specificity benchmark

	Find mutant amino acid starting from wild-type structure and mutant ligand			Find mutant amino acid starting from mutant structure and wild-type ligand		
	Median Percentile	Median Rank	Number of mutations identified	Median Percentile	Median Rank	Number of mutations identified
CM-BR	79	15	14	76	22	13
CM-FKIC	70	25	10	76	22	13
CM-WKIC	78	19	12	80	22	12
BackrubEnsemble	0	60	7	74	32	9
FastDesign	0	60	7	0	100	4
FixBB	0	80	5	0	80	5

Abbreviations: CM-BR CoupledMoves with Backrub; CM-FKIC, CoupledMoves with Kinematic Closure using fragments; CM-WKIC, CoupledMoves with Kinematic Closure using “walking” perturbation.

Note: Shown are median values across the benchmark. Percentile and rank were computed from sorted percent enrichment values (as defined in Methods). If the correct amino acid is not sampled, rank is the maximum possible number of amino acids for designed positions (number of designable positions*20). A mutation is considered correctly identified if a method identifies it with a positive percent enrichment.

TABLE 3

Detailed performance for enzyme specificity benchmark

Find mutant amino acid starting from wild-type protein structure & mutant ligand													
Wild-type PDBID	Mutant PDBID	Wild-type Ligand	Mutant Ligand	Desired Mutation	# Design Positions	CM-BR	CM-FKIC	CM-WKIC	Backrub Ensemble	Fast Design	FixBB control	Rank	Percentile
1A80	1M9H	NDP	NAD	K232G	5	79	-	64	-	-	-	100	100
1A80	1M9H	NDP	NAD	R238H	5	66	-	-	-	-	-	100	100
1FCB	ISZE	PYR	173	L230A	5	98	3	100	1	89	12	100	1
1K70	IRA0	HPY	FPY	D314A	4	100	0	99	2	98	3	99	2
1K70	IRA5	HPY	FPY	D314G	4	98	3	99	2	98	3	80	3
1K70	IRAK	HPY	FPY	D314S	4	-	80	-	-	94	6	-	80
1PK7	IOUM	ADN	TAL	M64V	3	77	15	-	-	-	60	60	60
1ZK4	IZK1	NAP	NAD	G37D	7	99	2	100	1	100	1	100	1
2FZN	3E2Q	PRO	HYP	Y540S	2	95	3	100	2	-	40	-	40
2H6F	2H6G	FAR	GER	W602T	9	63	68	50	91	93	14	-	180
2O7B	2O78	HC4	TCA	H89F	4	79	18	70	25	78	19	80	80
3HG5	3LX9	GLA	A2G	E203S	7	140	140	-	-	-	140	140	140
3HG5	3LX9	GLA	A2G	L206A	7	72	40	93	11	98	4	100	1
3KZ0	3L02	ANO	SNO	E92A	5	99	2	99	2	100	1	100	100
3KZ0	3L04	ANO	SNO	E92P	5	-	100	54	47	-	100	-	100
3KZ0	3L05	ANO	SNO	E92S	5	94	7	90	11	-	100	-	100
3KZ0	3L06	ANO	SNO	E92V	5	86	15	61	40	82	19	92	9

Find mutant amino acid starting from mutant protein structure & wild-type ligand													
Wild-type PDBID	Mutant PDBID	Wild-type Ligand	Mutant Ligand	Desired Mutation	# Design Positions	CM-BR	CM-FKIC	CM-WKIC	Backrub Ensemble	Fast Design	FixBB control	Rank	Percentile
1A80	1M9H	NDP	NAD	K232G	5	100	100	100	100	-	-	100	100
1A80	1M9H	NDP	NAD	R238H	5	100	100	100	100	-	-	100	100
1FCB	ISZE	PYR	173	L230A	5	97	5	96	6	88	18	100	100

Author Manuscript

Author Manuscript

Author Manuscript

Author Manuscript

1K70	IRA0	HPY	FPY	D314A	4	99	2	95	6	98	3	93	8	83	18	-	80
1K70	IRA5	HPY	FPY	D314G	4	73	23	74	22	74	22	-	80	-	80	-	80
1K70	IRAK	HPY	FPY	D314S	4	-	80	-	80	-	80	-	80	-	80	-	80
1PK7	IOUM	ADN	TAL	M64V	3	80	13	85	10	82	12	-	60	-	60	-	60
1ZK4	IZK1	NAP	NAD	G37D	7	100	1	100	1	100	1	99	2	-	140	99	2
2FZN	3E2Q	PRO	HYP	Y54OS	2	-	40	-	40	-	40	-	40	88	22	-	40
2H6F	2H6G	FAR	GER	W602T	9	59	75	69	57	57	79	85	28	-	180	-	180
2O7B	2O78	HC4	TCA	H89F	4	85	13	74	22	-	80	-	80	-	80	-	80
3HG5	3LX9	GLA	A2G	E203S	7	80	29	76	35	80	29	92	12	91	8	97	5
3HG5	3LX9	GLA	A2G	L206A	7	59	59	58	60	59	58	-	140	88	13	-	140
3KZ0	3L02	AN0	SN0	E92A	5	90	9	94	6	95	5	98	3	-	100	100	1
3KZ0	3L04	AN0	SN0	E92P	5	89	12	88	13	92	9	91	10	-	100	95	6
3KZ0	3L05	AN0	SN0	E92S	5	84	17	85	16	86	15	95	6	-	100	-	100
3KZ0	3L06	AN0	SN0	E92V	5	86	18	93	10	89	14	74	32	-	100	95	7

Abbreviations: CM-BR CoupledMoves with Backrub; CM-FKIC, CoupledMoves with Kinematic Closure using fragments; CM-WKIC, CoupledMoves with Kinematic Closure using “walking” perturbation.

Note: Percentile and rank are defined as in Table 2.