Development and benchmarking of methods for computational design, and experimental characterization, of proteins that bind small-molecule ligands.

Loshbaugh, Amanda Lynne

Development and benchmarking of methods for computational design, and experimental characterization, of proteins that bind small-molecule ligands.

by

Amanda Lynne Loshbaugh

DISSERTATION

Submitted in partial satisfaction of the requirements for degree of

DOCTOR OF PHILOSOPHY

in

Biophysics

in the

GRADUATE DIVISION
of the
UNIVERSITY OF CALIFORNIA, SAN FRANCISCO

Approved:

DocuSigned by:

Tanja Kortemme ______________________ Tanja Kortemme

FA555ADA28F1439...

Chair

DocuSigned by:

Jim Wells ______________________ Jim Wells

D72E4D1406B4C2...

DocuSigned by:

James fraser ______________________ James fraser

430BBB9A04D24A3...

____

____

Committee Members

Dedication

I would like to dedicate this dissertation to my mother, who made it clear from day one that I can be an astronaut or a zoologist, or anything in between, as long as I have fun.

Acknowledgements

I would like to thank just a few of the many people who have made this work possible.

Firstly, I am indebted to those talented scientists who contributed many years of research to the biosensors screened in chapter 3. Dr. Dan Mandell conceived the original biosensor concept of a ligand-induced heterodimer linked in a modular fashion to a split protein reporter, Drs. Alex Zanghellini, Lin Jiang and David Baker created the matcher protocol, Drs. Roland Pache and Kyle Barlow created the computational design pipeline described in section 3.4.1, and Drs. Roland Pache, Noah Ollikainen, Yao-Ming Huang, Kyle Barlow, and Tanja Kortemme visually inspected and made sequence changes to designs generated by Rosetta. For this last point, I especially thank Tanja Kortemme, whose manual design led to one of the two best ibuprofen biosensors described in this thesis. I am also indebted to Yao-Ming, who developed the *in vivo* screen described in sections 3.4.2 and 3.4.3, and Dr. Anum Azam-Glasgow who was an equal contributor to development of the cell-free luciferase screen described in section 3.4.5. Noah conceived of and implemented the CoupledMoves protocol described in chapter 2, and answered my many questions as I continued to develop his protocol. I could not have asked for better colleagues during graduate school, and I am glad to have worked with every member of the Kortemme lab, and especially Anum, with whom every conversation made the science a little more clear.

I would like to thank the many mentors and teachers who have guided me along the way. Firstly, the faculty of the Oberlin College Biochemistry department imparted a deep appreciation for the bio-organic structures and physical forces that shape life on this Earth. I thank Prof. Brian Kuhlman for encouraging my foray into atomic-level protein engineering. Drs. Andrew Leaver-Fay, Steven Lewis, and Kale Kundert were instrumental in my entry to the world of programming proteins by programming Rosetta. I am grateful for the knowledge and dedication of the entire Rosetta community. I also thank my committee members, Profs. Jim Wells and Jaime Fraser, for sharing their perspectives on protein engineering. And finally, I am deeply grateful for the support and scientific guidance I received from my thesis advisor, Prof. Tanja Kortemme. Tanja allowed me great intellectual freedom, while invariably providing the most precise biophysical perspective on every experiment and idea. Chapter 2 of this dissertation is a manuscript authored jointly by Loshbaugh and Kortemme, and will be submitted for publication.

Finally, I would like to thank my parents for their unwavering support, for inspiring me by example, and for sharing their lifelong passion for intellectual growth.

Abstract

Development and benchmarking of methods for computational design, and experimental characterization, of proteins that bind small-molecule ligands.

Author: Amanda Loshbaugh

I present computational and experimental methods relating to the design of binding interactions involving proteins, including interactions of protein/small molecule, dimeric protein/protein, and tertiary protein/small molecule/protein systems. The precise geometric design of atomic contacts necessary for binding interactions is an unsolved problem in the field of protein engineering, yet the design of binding interactions is essential for the furtherance of medicine, manufacturing, and basic science research. In chapter 2, compare computational algorithms for flexible backbone protein design in the Rosetta software suite. Design protocols were benchmarked for their ability to recapitulate observed protein sequence profiles assumed to represent the fitness landscapes of protein/protein and protein/small molecule binding interactions. We found that the CoupledMoves protocol, which combines backbone flexibility and sequence exploration into a single acceptance step during the sampling trajectory, better recapitulates sequence profiles than the BackrubEnsemble and FastDesign protocols, which separate backbone flexibility and sequence design into separate acceptance steps during the sampling trajectory. In chapter 3, I describe a method for efficiently screening and characterizing chemically induced dimers (CID) that detects and responds to the presence of small molecules. I screen a library of engineered biosensors, each of which is composed of a CID sensor module and a reporter module, which can be interchanged. The sensor module is a heterodimer whose interface contains a ligand binding site transplanted by computational design from a monomeric protein, such that ligand binding induces heterodimerization. The reporter module is a protein complementation system whose complementation is induced by dimerization of the sensor domain. I present two methods to individually screen hundreds of designed CIDs targeting various proteins, (1) using a growth-based reporter module in *E coli*, and (2) using a luminescent reporter in a cell-free protein expression system. Finally, the screen successfully identified a CID that responds to ibuprofen, and this system could be adapted for therapeutic application. This dissertation presents methodological advances for both the computational and experimental design of protein binding interactions.

Table of Contents

List of Figures

List of Tables

Chapter 1: Introduction

Proteins are the foundational machinery on which life runs. Enzymes catalyze chemical reactions that convert nutrients and light to usable energy. This energy is then used by countless proteins to perform the work of life, from motor and structural proteins that define the motility and shape of cells, to proteins that duplicate and epigenetically modify DNA for transmission of genetic information down the lineage of cell division. The structure-function relationship, which relates the three dimensional structure of a protein to its biological function, is a powerful paradigm that enables researchers to use structural representations to design function. Protein design occupies an essential role within the interdisciplinary field of synthetic biology, which aims to fabricate or design new biological components and systems. I address two levels of protein engineering: (1) atomic-level design of binding interactions, and (2) experimental screening and validation of binding between protein dimers and small molecules.

Protein function typically involves binding other proteins or molecules, yet designing such binding interactions remains challenging. While adding binding functionality to an existing protein could enable new synthetic biology tools, there is only one known example using computational design to add a ligand binding site into an existing protein in a *de novo* fashion, meaning at a location that did not previously bind a small molecule.[1] Instead, success designing ligand-binding proteins more frequently relies on adapting pre-existing ligand binding sites to bind a target ligand,[2-5] experimentally generating selective antibodies that recognize pre-existing ligand binding sites,[6] or making chimeras of modular proteins to take advantage of allosteric mechanisms in response to target ligand binding.[7, 8] Precise contacts remain difficult to predict.[9] Computational protein modeling software involves representation, sampling, and scoring of macromolecular conformations. The work presented here addresses the sampling component, specifically sampling that includes both sequence design and backbone flexibility in the Rosetta software suite.[10] I present a methodology to benchmark flexible-backbone design protocols against each other, by quantifying performance on multiple experimental and evolutionary sequence datasets. Additionally, I discuss aspects of design algorithms that may contribute to differences in performance.

Designed proteins, once generated by computation, must be screened experimentally, yet screening individual designed proteins is typically labor intensive. In some cases, high-throughput screening techniques such as directed evolution may be appropriate. In other cases, individual screening of many designs may call for lower throughput techniques. I present a methodology to rapidly and efficiently screen tens to hundreds of computationally designed ligand-inducible protein heterodimers. The cell-free protein expression protocol presented here screens for enzyme activity that occurs when split enzymes are reconstituted by designed heterodimeric proteins. The protocol could be adapted to screen for protein function that does not require dimerization, such as monomeric enzyme activity, or transcriptional activation.

The work presented here represents methodological advances in both computational and experimental design of ligand binding sites. These methods could be applied to make synthetic biology tools for studying biology, designing therapeutics, or biological manufacturing.

Chapter 2: Comparison of Rosetta flexible-backbone computational protein design methods on binding interactions

2.1 Introduction

Computational protein design searches for sequences that adopt desired structures and functions. Most generally, computational design methods require (i) algorithms to efficiently search the vast sequence and conformational space accessible to proteins, and (ii) effective energy functions to rank the solutions. Both of these requirements necessitate approximations. Design energy functions are often simplified while considering atomic detail,[11, 12] 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.[13, 14] 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.[15-17] Highly stable, idealized folds can be designed *de novo*,[18-21] 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,[3, 5, 22, 23] and typically requires optimization of the desired function by directed evolution.[3-5, 9, 24, 25] Function often depends on hydrogen bonds, which require precise backbone and side chain geometry, which remains difficult to design[9] especially when a novel function requires "reshaping" of an existing protein conformation.[26]

Various strategies have been proposed to model backbone flexibility, such as small random perturbations in torsional or Cartesian space,[27-30] normal mode analysis,[31] backbone ensembles from crystal structures[28] or from computational simulations,[32-34] or backbone parameterization, in particular for helical bundles.[19] Within the structure modeling and design program Rosetta,[10] backbone flexibility has been treated in a number of ways. These include (i) generation of new protein backbones by assembly from peptide fragments which demonstrated success in *ab initio* structure prediction,[35, 36] (ii) cycling between sequence design via Monte Carlo search and structure optimization via minimization,[37-39] which led to the first *de novo* protein fold not observed in nature,[39] (iii) a robotics-inspired kinematic closure (KIC) algorithm[40] shown to model loop conformations with sub-Angstrom accuracy,[41] and (iv) the Backrub algorithm, developed to describe structural changes underlying protein structural heterogeneity commonly observed in high resolution crystal structures[17] and benchmarked on recapitulation of known sequences.[33, 34, 42, 43] Most flexible backbone design methods iterate between sequence design on a fixed backbone and structural optimization on a fixed sequence, 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 sampling[43, 44] in a single design step.

While Rosetta flexible backbone design has been successfully applied to forward engineering,[25, 45-48] 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,[43, 44] and FastDesign, which combines sequence design with the Rosetta FastRelax method [49, 50] 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 direct 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 kinematic closure (KIC) in place of the original backrub backbone mover, which leads to further marginal 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.

2.2 Results

2.2.1 Design methods

We set out to compare four flexible-backbone design methods (**Figure 2.1**) using a common set of benchmarks (described below): (i) FastDesign utilizing the Rosetta FastRelax method[49, 50] for backbone flexibility (see Methods), (ii) BackrubEnsemble Design,[43, 44] (iii) CoupledMoves with Backrub (CM-BR),[42] and (iv) 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 2.1A**. FastDesign (**Fig. 2.1A, left**) iterates between two steps. In the first step, 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, 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. Simulated annealing first enables amino acid changes that introduce unfavorable clashes, which can be subsequently relaxed in the minimization step. FastDesign has been used in a variety of design applications [18, 46, 47, 51-53].

The BackrubEnsemble method[54] (**Figure 2.1A, middle**) also proceeds in two steps. The first step generates an ensemble of backbones through application of Backrub moves. Each Backrub move[17] selects two pivot backbone Cα atoms and rotates the entire segment between them (2-11 residues) as a rigid body. Backrub moves are made throughput 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. Incorporating backrub moves into Rosetta simulations led to considerable improvements in modeling structural changes upon point mutation,[43, 55] conformational fluctuations,[34, 44, 56] and molecular recognition specificity,[54, 57] and successful application to the redesign of recognition specificity.[45]

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 (**2.1A, right**). In this fashion, the backbone can respond to a designed sequence change more directly than in the non-coupled 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 designed alanine or glycine side chains.[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. Coupled moves 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 Coupled moves 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 kinematic closure (KIC) algorithm[41] (**Figure 2.1B**). KIC selects two pivot Cα atoms that define a segment, and a third pivot Cα atom within the segment. The algorithm next perturbs the backbone torsion angles around all non-pivot Cα 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 non-pivot Cα atoms probabilistically from Ramachandran space.[41] 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 standard deviation of 3° (WKIC) (see Methods).

2.2.2 Benchmark datasets

We evaluate the performance of the different methods on six benchmark datasets (**Table 2.1, 2.2**). 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, such as engineering of therapeutic proteins or small molecule sensors. 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 2.1, Figure 2.2, Tables S2.1-10**). The first two datasets were taken from [42]. Dataset 1 comprises evolutionary sequence alignments for eight naturally occurring protein families that each bind a specific cofactor ("cofactor" set, **Figure 2.2A**). Dataset 2 was curated from experimentally-characterized substrate specificity-altering point mutations for ten different enzymes ("enzyme specificity" set, **Figure 2.2D**). 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)[3], **Figure 2.2C**, and “Fen49” (fentanyl)[2], **Figure 2.2B**). 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 2.1**). Dataset 5 comprises sequences from 5 phage display libraries of Herceptin (17 positions total) selected for binding to HER2 (“Herceptin/HER2” set [58], **Figure 2.2E**). Dataset 6 comprises sequences from 6 libraries of human growth hormone (hGH) (35 positions total) selected for binding to human growth hormone reception (hGHR) (“hGH/hGHR” set [59], **Figure 2.2F**).

2.2.3 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).[33] 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,[42, 57] profile similarity and rank top, both calculated per position. 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 (see Methods). 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 benchmark[42] 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.

2.2.4 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 2.3A** shows the distributions of position profile similarities across all designed positions in each benchmark, with the median indicated by a white dot. CoupledMoves and BackrubEnsemble attain higher median PPS values than the null model for the Herceptin, Cofactor, and Fen49 datasets, although BackrubEnsemble does so by a lower margin. Somewhat surprisingly, using this global metric FastDesign and FixBB do not attain a higher median PPS values than the null model for most 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 2.3B**). 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 2.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 2.S2**) and hGH/hGHR (**Figure 2.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 2.4A,B**).

CoupledMoves achieves similar or better PPS for nearly all positions when compared to the non-coupled methods (**Figure 2.4A**, CM-KIC shown as example). BackrubEnsemble achieves PPS values better or similar than FastDesign (**Figure 2.4B, left**), and better than FixBB (**Figure 2.4B, middle**), for almost all positions. FastDesign, compared to FixBB, achieves better PPS for some positions, but worse PPS for others (**Figure 2.4B, right**). **Figure 2.4C** quantifies the number of positions for which CoupledMoves is better, worse, or similar to the non-coupled 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 classed 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. Standard deviation 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 2.4C**). Moreover, CoupledMoves performs worse than non-coupled methods for very few positions (**Figure 2.4C**, red bars).

2.2.5 CoupledMoves is the accurately predicts 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 5 profile datasets. Sequence logo representations of the tolerated sequence space for each of our datasets (**Figures S2.4-8**) 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 2.5, Figure S2.9**). Positions with low (entropy ≤ 0.33) or medium (0.33 < 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 non-coupled methods which frequently do not identify the consensus side amino acid identity at all (rank of 20) (**Figure 2.5**). CoupledMoves predictions typically have the highest entropy (**Figure S2.10**), which leads on average to higher similarity at variable positions. Nevertheless, PPS and RankTop at low-entropy positions (**Figure 2.5, Figure S2.9**), and energetic rankings of consensus positions (see Discussion, **Figure 2.8**) indicates 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.[58] Contrary to previous findings[60] where BackrubEnsemble recapitulated D V_H98 W as the top mutation, the non-coupled methods tested in this study did not identify tryptophan (**Figure S9**), 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, A77V and A171I, that led to ~100-fold improvement in binding affinity to fentanyl, but none of the design methods tested here found both mutations (**Figure S2.8**). 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.[2] While all design methods did substitute larger hydrophobic side chains, only FastDesign ranked 171I highly, and only BackrubEnsemble ranked 77V highly. CoupledMoves selected 77V at a lower frequency. No method identified the combination of A77V and A171I. While there is no crystal structure with these mutations, we hypothesize that packing I171 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 I171. Unlike position 171, which is an ideal distance for van der Waals interaction with fentanyl, there is an almost 6 Å distance between the closest heavy atoms of position 77 and the ligand and has a 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. For example, the original Fen49 side chains were present in the selection but did not have frequency counts.[2]

2.2.6 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 previously[42] (**Table S2.3, Table S2.4**). 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: 1K70, 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 mutations specificity-determining mutations, respectively, while the non-coupled methods (FastDesign, BackrubEnsemble, FixBB) identify only 7, 7, and 5 mutations, respectively (**Table 2.2**). All CoupledMoves methods identify specificity-altering mutations with a better percentile and rank than the non-coupled methods (**Tables 2.2, 2.3**), with the original CM-BR attaining the best median and quartile performance, and FastDesign and BackrubEnsemble performing similarly poorly.

2.2.7 Gain and loss

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 2.6, 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 2.6a**, middle and bottom panels, **Table S2.11**).

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 below 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 higher (**Figure 2.6a, top panel**). We only observed loss for positions designed by BackrubEnsemble, FastDesign, and FixBB (**Figure 2.6b, middle and**

bottom panels). CoupledMoves achieves gain with the best median and quartile RankTop values (**Figure 2.6b, middle panel**), and for the greatest number of positions (**Figure 2.6b, bottom panel**). Positions are more likely to remain neutral than to experience gain or loss (**Figure 2.6, bottom panels, Table S2.11**), 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 2.3**), which are characterized by low similarity between each dataset's input sequence and known profile (**Figure S2.11**).

We then asked which methods best predict positions deemed both functionally relevant (consensus) and difficult (requiring gain). We find that CoupledMoves is more likely than non-coupled 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 S2.11**). 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 versus hydrophobic packing. CoupledMoves outperforms BackrubEnsemble and FastDesign in discovering and retaining both polar/charged and hydrophobic positions (**Table S2.11**).

2.2.8 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 non-coupled methods (**Figure S2.9**). 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 retains RV_H50, while FastDesign and BackrubEnsemble replace this residue with hydrophobic residues, predominantly methionine and glycine, respectively (**Figure 2.7A**).

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 screen[3]), tyrosines 34, 101, and 115 hydrogen bond with digoxigenin, as designed.[3] CoupledMoves frequently chooses Tyrosine at all three positions (**Figure 2.7B, top**), whereas FastDesign models only one interaction correctly (**Figure 2.7B, middle**), and BackrubEnsemble models two (**Figure 2.7B, 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) binding site in glutathione reductase (**Figure 2.7C**). 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 (**Table 2.6**). 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 non-coupled 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.

2.3 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: (i) important for function and (ii) 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 2.5**) and outperforms the other methods in correctly identifying specificity-switching mutations in the enzyme specificity set (**Table 2.2, Table 2.3**). 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 the iterative BackrubEnsemble and FastDesign methods 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 model a preferred side chain residue present in the input sequence (loss) (**Table S2.11, Figure 2.6, Figure S2.11**). 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 2.8a** 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 S2.10**) 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 > 75th 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 2.8a**). One possible reason for the overall worse performance of BackrubEnsembles 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 2.9**, 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, and highlights a limitation of the BackrubEnsemble method. Backbone moves are sampled only once, at the beginning of the trajectory during ensemble creation (**Figure 2.1a**). 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 non-consensus 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 2.8b**), 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 backbones 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 S2.10**). FastDesign may be less likely to escape local minima with its simulated annealing and minimization algorithm that the other methods, despite the use of a reduced Lennard-Jones repulsive term in the early cycles of the simulation (**Figure 2.1a**).

In addition to limitations in sampling methods (as well as the energy function used to rank designs), there are also potential limitations inherent in our benchmark datasets. 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 there are additional specificity-altering mutations.

Sequence profiles in the cofactor dataset result from natural evolution, rather than experimental screening. Natural evolution includes selection pressures beyond affinity (function, stability, kinetics), so that the sequence profiles for natural binding site positions may be influenced by factors beyond those modeled by Rosetta. 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 versus 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,[61-63] and it is well established that flexible-backbone protein design can generate sequences similar to observed natural and experimental sequences.[32-34, 38, 54, 64-66] 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.

2.4 Methods

2.4.1 Benchmark Datasets

2.4.1.1 Cofactor binding sites

This dataset is described in detail in [42]. Briefly, the dataset is comprised of seven protein families, each containing a conserved small molecule cofactor binding site (**Table S2.1**). The highest resolution available crystal structure was chosen as the starting point for design. As in [42], positions with a side-chain heavy atom within 6 Å of any heavy atom in the co-factor 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 [42].

2.4.1.2 Enzyme specificity

This dataset is described in detail in [42]. Briefly, the dataset is comprised of 10 enzymes for which there are experimentally validated specificity-altering mutations in the ligand binding sites (**Table S2.3**). As in [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 [42]) were allowed to repack (**Table S2.4)**. Structures were prepared as described in [42]. Briefly, for each enzyme four types of structures were prepared: 1) the native enzyme with the native ligand, 2) the mutant enzyme(s) with the non-native ligand, 3) the native enzyme with the non-native ligand, and 4) the mutant enzyme with the native ligand.

2.4.1.3 DIG10

The DIG10 dataset was taken from [3]. Briefly, DIG10 is a computationally designed protein that has been engineered to bind the small molecule digoxigenin (DIG) [3]. 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. The binding fitness landscape of DIG10.1 was then probed by SSM and selections using yeast surface display, which converged after four rounds of selection to variant DIG10.2. Our computational protocol seeks to replicate the deep sequencing library selection that led from DIG10.1 to DIG10.2. 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 FixBB protocol), and the digoxigenin conformation from the DIG10.2/digoxigenin complex (PDB: 4J8T). 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 [3] 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 [3] (**Table S5, Table 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 non-coupled methods, all positions were allowed to repack (**Table S5**).

2.4.1.4 Fen49

The Fen49 dataset was taken from [2]. 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 Fen-BSA. After four rounds of selection, 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, as a 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 determination [2]) 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 [2] (**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 [2] (**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 non-coupled 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.

2.4.1.5 Frequency equivalent

Experimental data from the DIG10 and Fen49 datasets are deep sequencing counts before and after selection, which are not directly comparable to amino acid identity frequencies from computational design. We therefore derived a frequency equivalent (F_{equiv}) from the fitness score described in [67], to allow direct comparison between the experimental data and sequence profiles from Rosetta design for mutation *x* at position *i*:

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

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.

2.4.1.6 hGH/hGHR

The hGH/hGHR dataset was taken from [33]. 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 nM[68] and 1.56 nM.[59] 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 ~1300 $Å^2$ hGH/hGHR interface were divided into six combinatorial libraries of five or six positions.[59] 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 [33], residues within 4 Å of designed residues were allowed to repack (**Table S9**).

2.4.1.7 Herceptin-HER2

The Herceptin-HER2 dataset was taken from [60]. The protein-protein interface between therapeutic antibody Herceptin and its target, human epidermal growth factor 2 (HER2), is high affinity (K_D = 0.35nM[58]). We used a crystal structure (PDB: 1N8Z) as input structure for design, truncated as in [60] 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.[58] 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 [60] (**Table S2.10**). Herceptin/HER2 sidechains were repacked from the crystal structure before design.

2.4.2 Rosetta Design protocols

Design protocols used Rosetta revision number 60351 and score function ref2015.[11, 69] For each method, we used standard parameters and settings previously reported in benchmarks or design applications, except for the new CM-FKIC and CM-WKIC methods reported here. Command lines for each method can be found in the supplement.

2.4.2.1 CoupledMoves

The CoupledMoves method was used as described in [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 1,000 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.

2.4.2.2 CoupledMoves with Kinematic Closure

Two different methods of modeling backbone flexibility are implemented in CoupledMoves. The first method uses the Backrub algorithm [43, 70] and was originally described in [42]. The second method uses kinematic closure [40, 41, 71] and is implemented in

CoupledMoves here (**Figure 2.1b**). Kinematic closure in Rosetta [41] generates conformations of backbone segments by sampling non-pivot torsions in the segment and then analytically determining values for 6 pivot torsions to close the loop. For CM-FKIC, non-pivot torsions are sampled from peptide fragments taken from the PDB.[35] For CM-WKIC, non-pivot torsions are adjusted by a random value from a Gaussian distribution centered around zero and with a standard deviation 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.

2.4.2.3 FastDesign

FastDesign is based on the FastRelax protocol in Rosetta described in [49, 50]. 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 is ramped down from 1.0 to 0.0 during the course of each inner simulated annealing cycle. Constraint and repulsive weights are 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.

2.4.2.4 BackrubEnsemble

The BackrubEnsemble method is described in [57]. Briefly, the method generates a structural ensemble with backbone conformational variation using the backrub algorithm,[43] 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 atoms. Command line arguments are shown in the Supplement.

2.4.2.5 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 trajectories were run, using as input the first 100 members of the same BackrubEnsemble on which typical design was performed.

2.4.2.6 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 design [3] 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.

2.4.2.7 Computational performance

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 S2.12**). 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 S2.12**). Since each BackrubEnsemble and FastDesign trajectory takes approximately 2-fold and 20-fold more time than CoupledMoves, respectively (**Figure S2.13**), CoupledMoves requires substantially less compute time than FastDesign and about equal compute time to BackrubEnsemble (**Table 2.4**).

2.4.3 Performance Metrics

2.4.3.1 Position profile similarity

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

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

where $p_{known,i}$ and $p_{design,i}$ 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 [44].

2.4.3.2 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.

2.4.3.3 Percent Enrichment

As in [42] the percent enrichment (PE) for each specificity-altering mutation in the enzyme specificity dataset was calculated as follows:

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

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

where $\%_{native}$ is the percent occurrence of the mutation in sequences designed for the native ligand and $\%_{non-native}$ is the percent occurrence of the mutation in sequence 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 [42], a prediction was considered correct if it obtained a positive percent enrichment value.

2.4.3.4 Rank

For the enzyme specificity dataset, mutations were ranked by descending order of their percent enrichment values, as described in [42].

2.4.3.5 Entropy

Sequence entropy was computed as in [42]. Briefly, the sequence entropy H_i for each position was calculated as follows:

$$H_i = -\sum_x P_x \log_{20} P_x$$

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

2.4.3.6 Distance from input sequence

Distance from input sequence is a variation of 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.

2.5 Figures

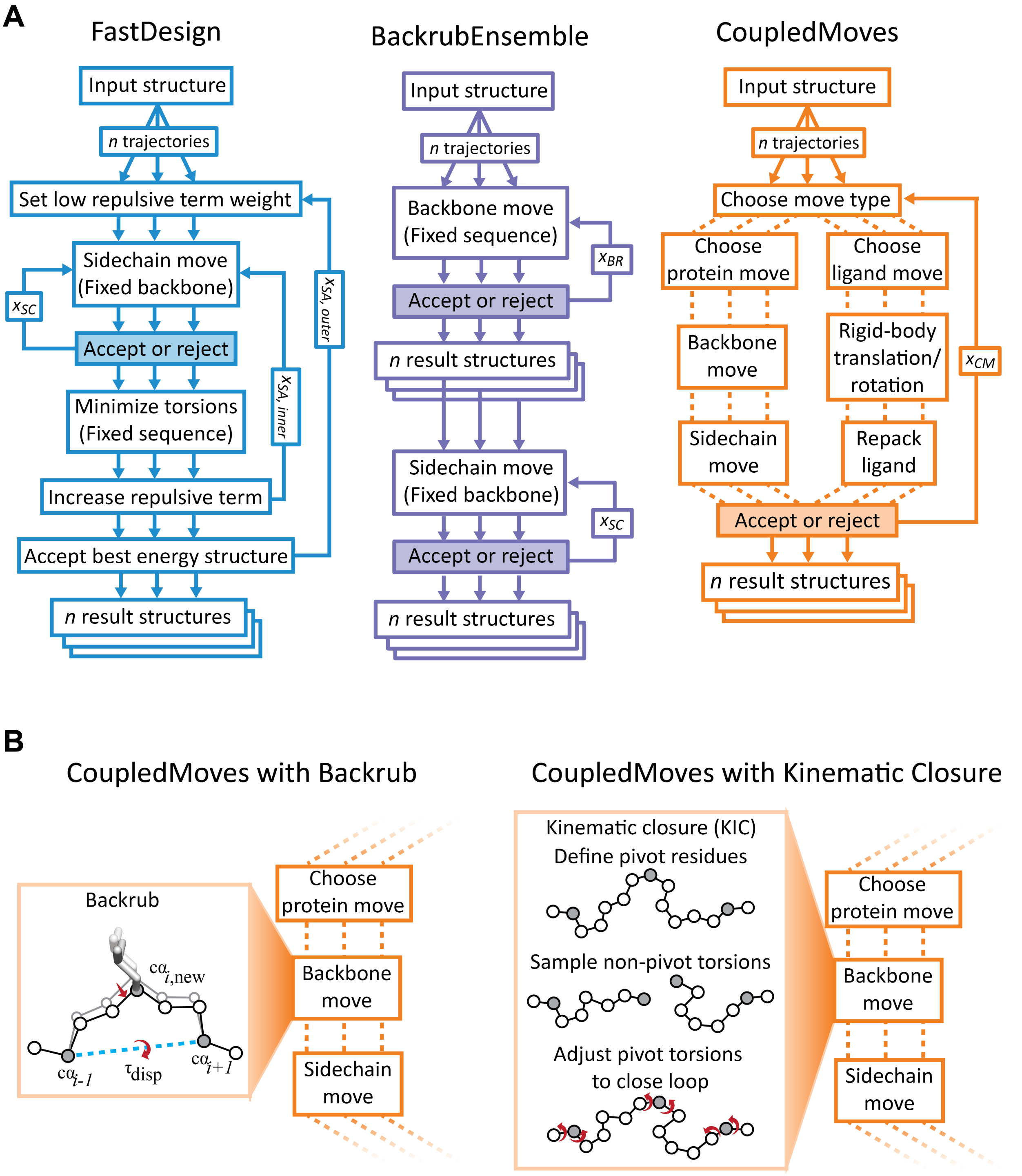


Figure 2.1: Design methods.
(A) Design method comparison. The FastDesign (left, blue) and BackrubEnsemble (middle, purple) methods separate sequence design steps (using a fixed backbone) from backbone optimization steps (using a fixed sequence). CoupledMoves (right, orange) evaluates combined

Continued on next page.

Figure 2.1, continued: Design methods.
moves that sample both backbone conformation and amino acid sequence (or, alternatively, combine ligand translations/rotations with changes of ligand conformers). CoupledMoves performs 1000 trials (x_{CM}) per trajectory. 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. **(B)** Original and updated backbone mover in CoupledMoves. The original CoupledMoves method[42] uses the Backrub algorithm to make backbone moves. A backrub move [17, 43] rotates a segment as a rigid body by displacement angle τ_{disp} around an axis between two pivot Cα 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.[41] Backbone torsion angles for non-pivot Cα 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 Cα atoms (grey).

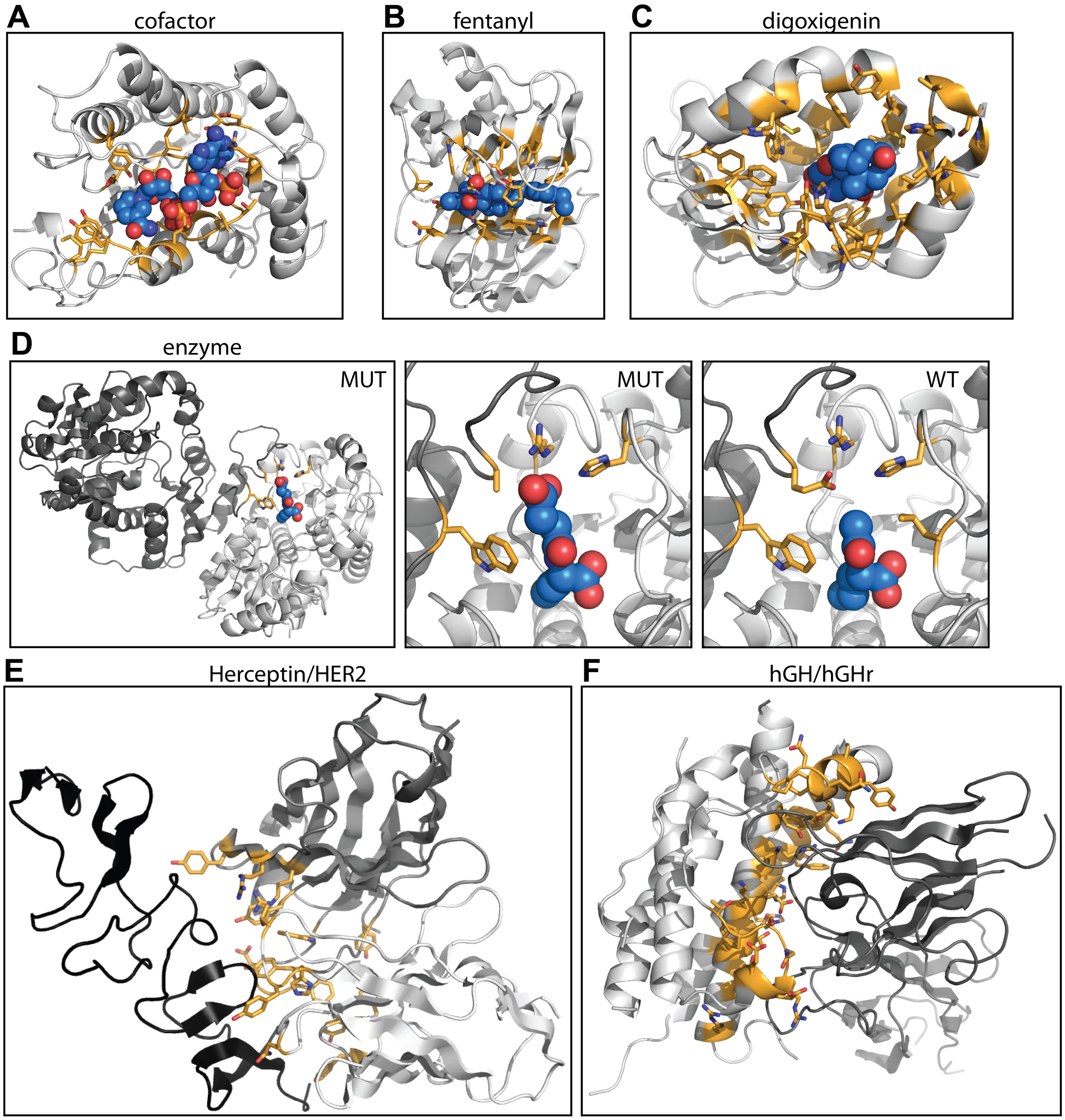


Figure 2.2: Benchmark dataset structures.
Side chains at designed positions are highlighted in orange and shown as sticks. Ligands are colored light blue and shown in sphere representation. The structures shown are those used as input for design, as described in Methods. Nitrogen atoms are shown in dark 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) The wild-type protein used for design of fentanyl binding protein, with fentanyl placed in the binding pocket. (C) DIG10.1, the designed digoxigenin binder on which the SSM library was generated and selected, with digoxigenin. (D) Representative structure from the enzyme dataset, N-acetylornithine carbamoyltransferase. The full structure of the mutant enzyme, with ligand

Continued on next page.

Figure 2.2, continued: Benchmark dataset structures.
N-(3-carboxypropanoyl-L-norvaline (SN0), is shown in the left panel. The middle panel shows the binding site. The right panel shows the binding site of the wild-type protein, with ligand N-acetyl-L-norvaline (AN0). The other nine enzymes are not shown. (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 are shown. (F) hGH/hGHr. Designable positions on hGH (light gray) interact with target hGHr. The combination of designable positions from all libraries is shown.

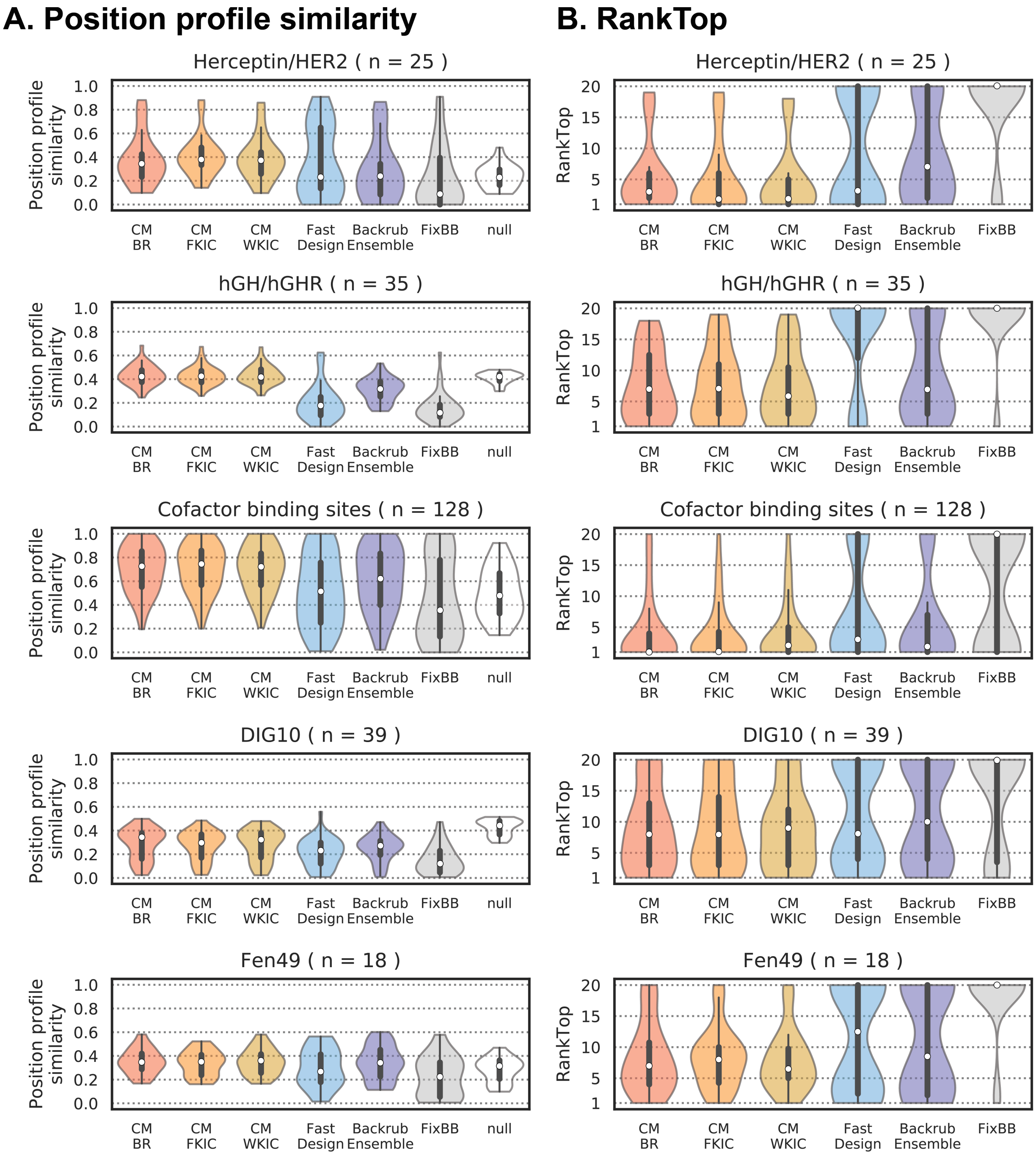


Figure 2.3: Comparison of design method performance on sequence profile datasets. PPS (A) and RankTop (B) distributions. A rank of 1 means that the design method correctly identified the most frequent amino acid side chain observed in the experimental/natural profile, whereas a RankTop of 20 means that side chain was observed with zero frequency, or that all side chains were modeled with some frequency and the top known 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.

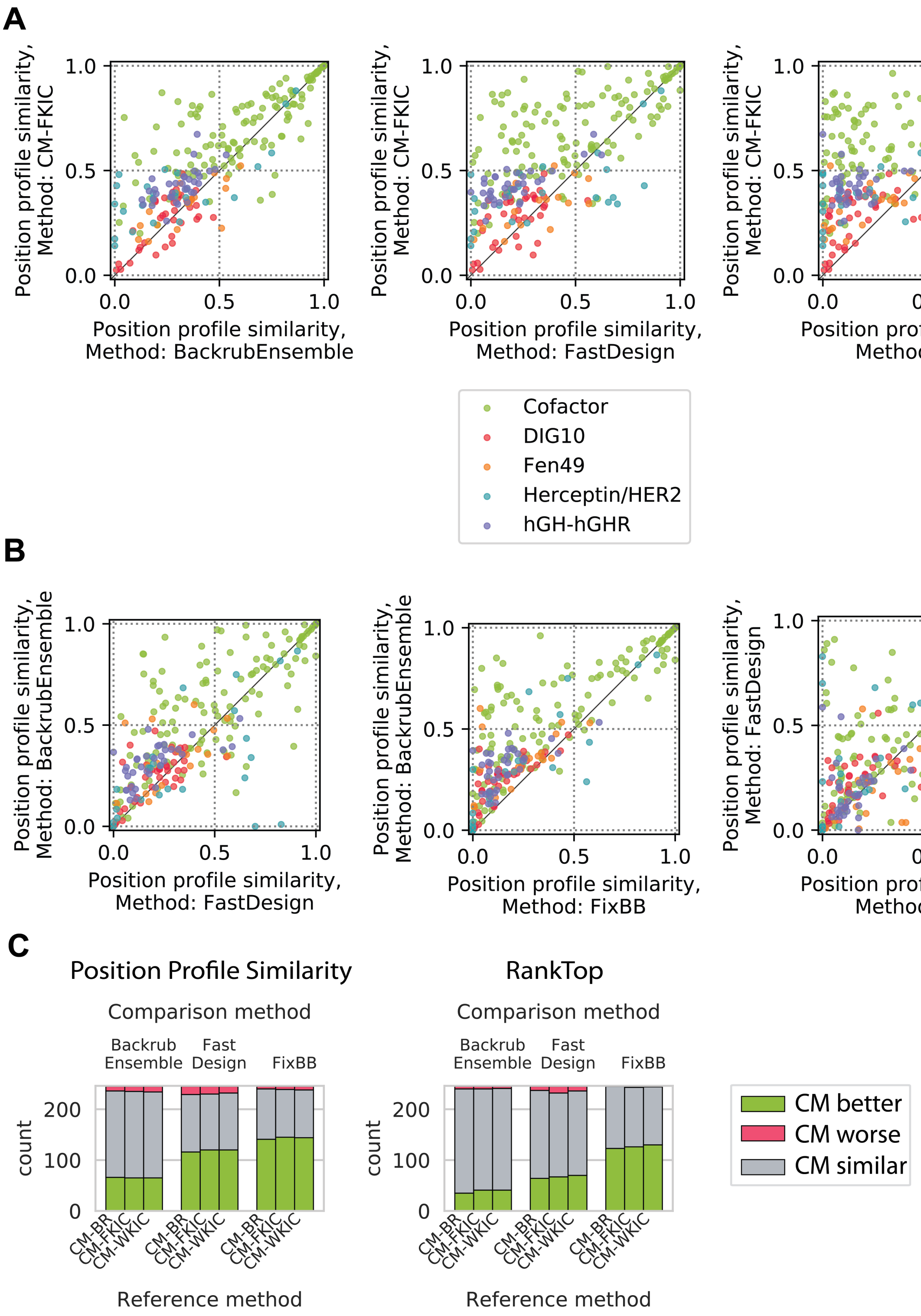


Figure 2.4: Method performance comparison for profile datasets by sequence position. Shown are the same data as in Figure 3, but plotting individual sequence positions instead of distributions. Colors indicate different datasets. (**A**) Comparison between CM-FKIC and non-coupled methods. Points above the diagonal represent positions where CM-FKIC outperforms **Continued on next page.**

Figure 2.4, continued: Method performance comparison for profile datasets by sequence position.

the non-coupled method. (**B**) Comparison between iterative 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 non-iterative 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 classed as "similar."

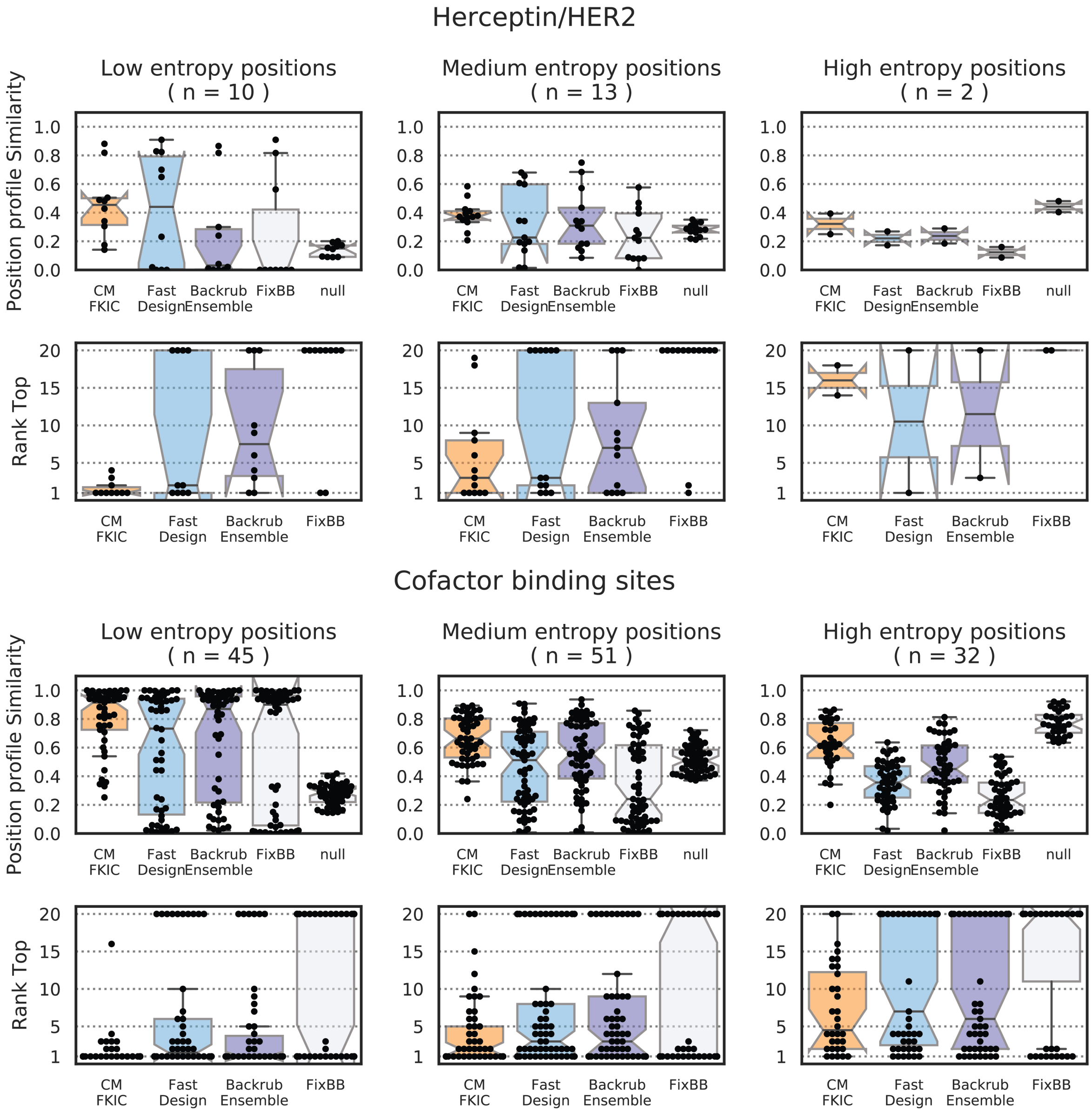


Figure 2.5: PPS and RankTop as a function of known sequence entropy.
Each point represents one sequence position. Shown here are the Herceptin/HER2 (top) and Cofactor (bottom) datasets, which have the highest number of low entropy positions. The remaining datasets are shown in **Figure S2.9**. For each dataset, PPS and RankTop are binned by entropy of the known sequence profile at each position (low: entropy ≤ 0.33, medium: 0.33 < entropy ≤ 0.67, and high: entropy > 0.67). The boxplot covers the second and third quartiles, and the vertical whiskers mark 1.5 times the inter-quartile range. Median is marked with a horizontal black line, and notches represent a 95% confidence interval (CI) around the median; when CI extends past the quartiles, notches extend beyond the box, leading to a "flipped" appearance.

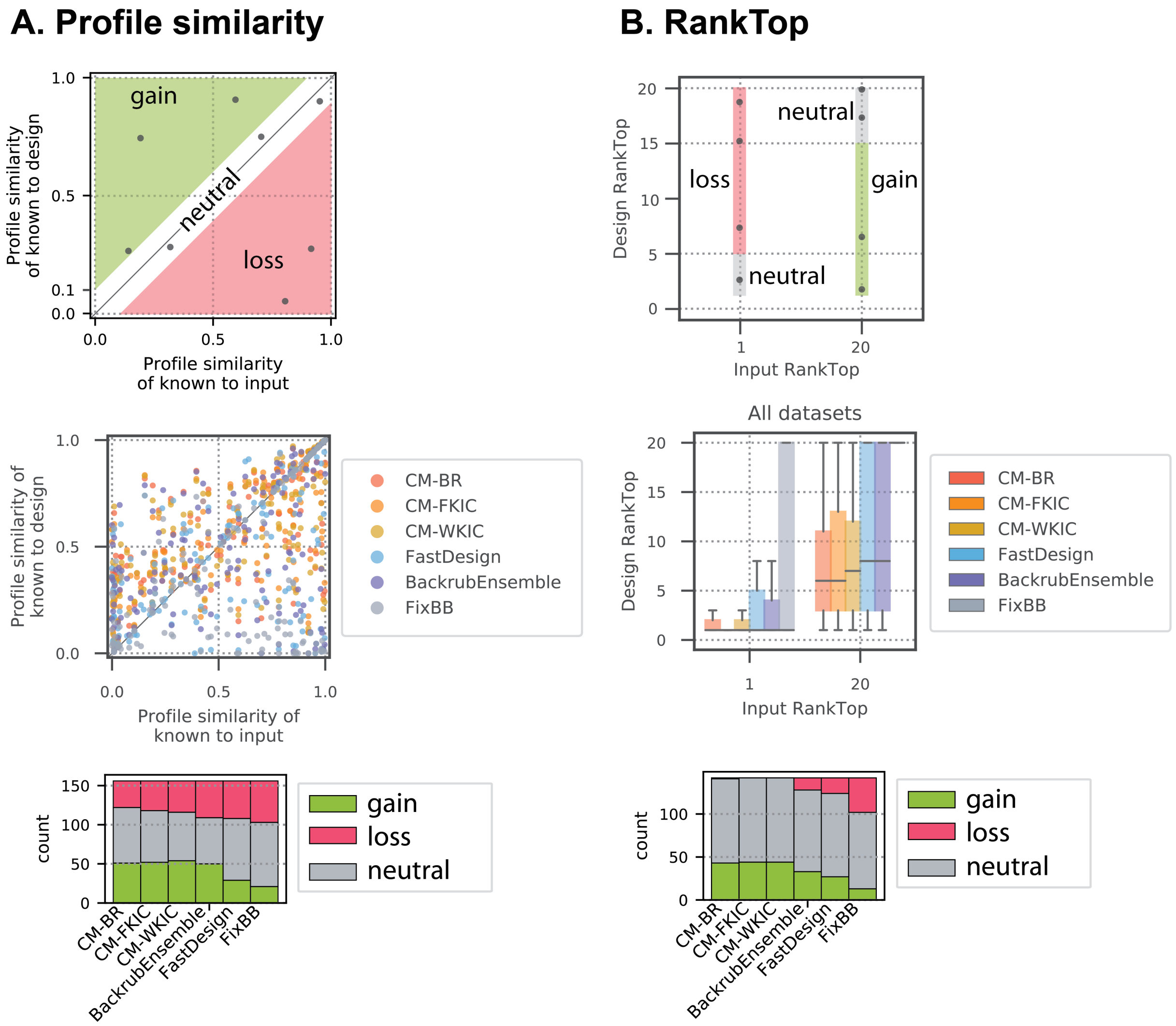
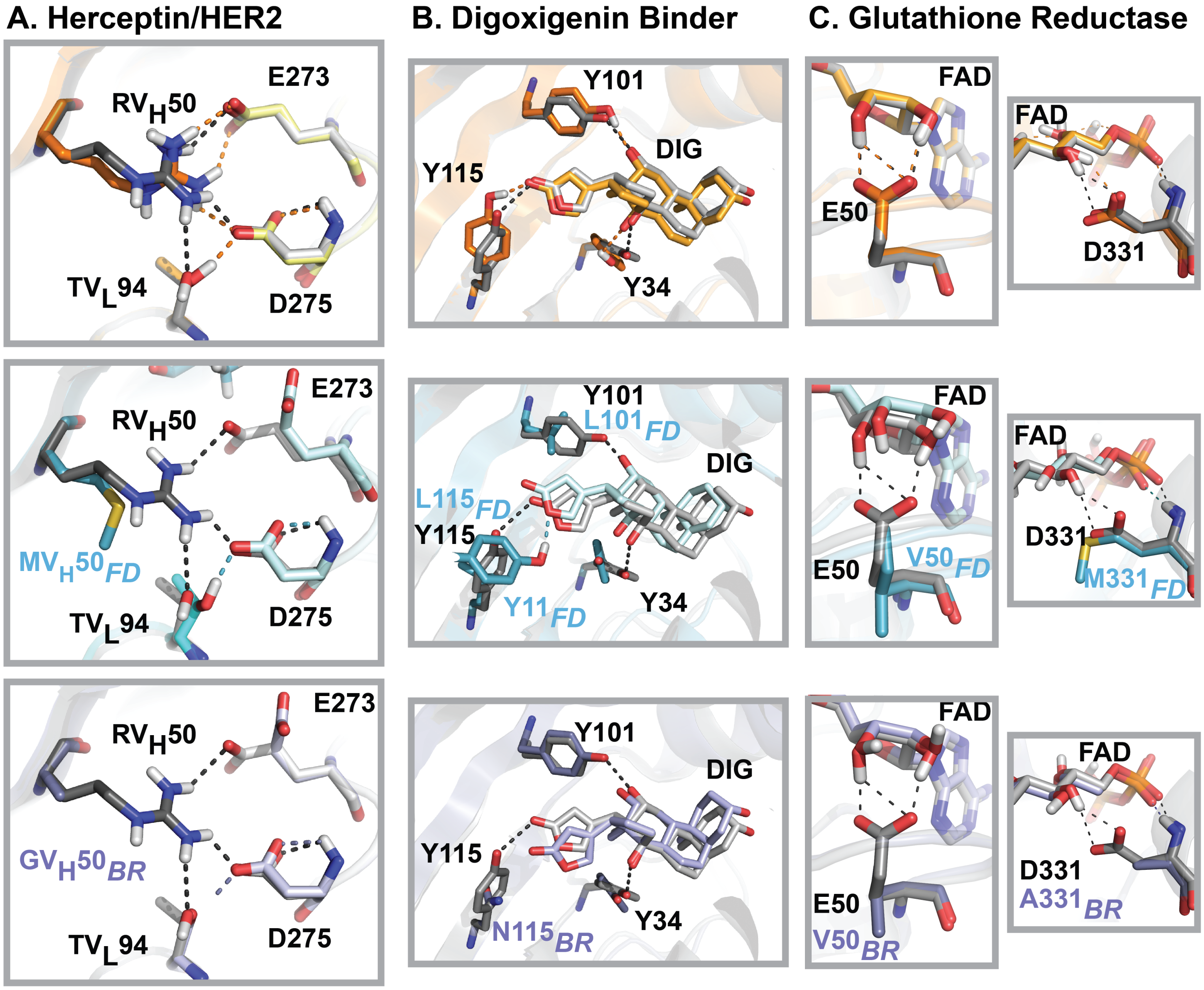


Figure 2.6: PPS and RankTop as a function of similarity to input.
Gain (green) and loss (red) as defined in the main text. Only positions with low and medium entropy (≤ 0.67) are considered. This figure combines all datasets; individual datasets are shown in **Figure S2.11**. (**A**) PPS as a function 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 (grey) results 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.



D. Design frequencies

	Herceptin	Digoxigenin binder			Glutathione Reductase	
	R VH50	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 2.7: 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$_H$50 (dark color) and TV$_L$94 (medium color), and HER2 residues E273 and D275 (light color). Key designable residue RV$_H$50 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

Continued on next page.

Figure 2.7, continued: Examples of structural models generated by different design methods.
(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 form 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 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.

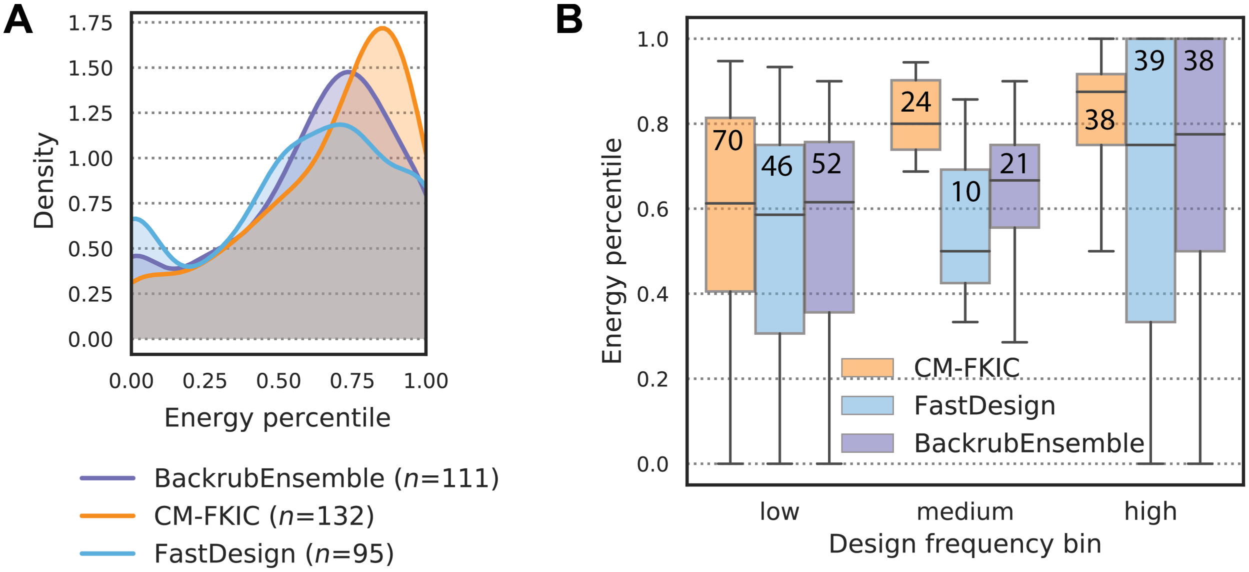


Figure 2.8: Distribution of energy percentiles for correctly modeled positions. "Energy percentile" refers to the percentile of the average total Rosetta energy of the correctly modeled side chain compared to that of all other side chains modeled by the design method at that position. Energy percentile 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 percentile 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 < 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 inter-quartile range.

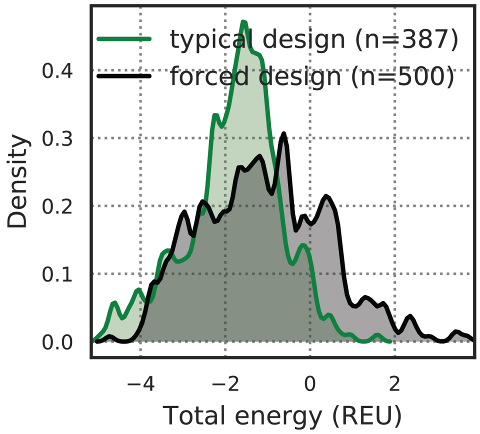


Figure 2.9: Comparison between typical and forced design of consensus side chains onto Backrub ensemble.

Distribution of Rosetta energies (REU) for consensus amino acid side chains at five positions: Glutathione Reductase positions E50 and D331, and DIG10 positions Y34, Y101, and Y115. For each position, we show 100 models forced to adopt the consensus side chain during sequence design (black), and typical models (green) that arrived at the consensus side chain though they were allowed to design to multiple side chain identities. For typical models, n corresponds to the number of models with the known consensus side chain, out of a total of 2000 (400 models for each of the five positions; design frequencies are shown in **Figure 2.7d)**. Density represents a Gaussian kernel density estimate using a bin width of 0.1 REU.

2.6 Tables

Table 2.1: Benchmark datasets.
Type of binding interaction (protein/small molecule or protein/protein) and source of known functional sequences are shown. Design position and starting amino acid residues are shown.

Binding interaction	**Benchmark name**	**Known functional sequences**
Protein / small molecule	Cofactor	Natural sequence alignments from Pfam database [72]
	Enzyme specificity	Experimentally-characterized point mutations
	DIG10	Amino acid frequencies derived from deep sequencing of a site saturation mutagenesis library after 3 rounds of selection
	Fen49	Amino acid frequencies derived from deep sequencing of a single site saturation mutagenesis library after 4 rounds of selection
Protein / protein	hGH / hGHR	Sequences of clones selected from 5 combinatorial libraries (average of 180 sequenced clones per library) after 2 rounds of selection
	Herceptin / HER2	Sequences of clones selected from 4 combinatorial libraries (average of 70 sequenced clones per library) after 4 rounds of selection

Table 2.2: Performance summary for enzyme specificity benchmark.
Shown are median values across the benchmark. Percent enrichment describes the difference in frequency of a specificity-altering mutation designed in the presence of its specific ligand, compared to its frequency when designed around the alternate ligand. A mutation is considered correctly identified if a method identifies it with a positive percent enrichment. Rank and percentile were computed from sorted percent enrichment values. 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).

	Find mutant amino acid starting from wild-type structure & mutant ligand			Find wild-type amino acid starting from mutant structure & 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

Table 2.3: Detailed performance for enzyme specificity benchmark.
Percentile and rank are defined as in Table 2.2.

						Find mutant amino acid starting from wild-type protein structure & mutant ligand											
						CM BR		CM FKIC		CM WKIC		Backrub Ensemble		Fast Design		FixBB control	
Wild-type PDB ID	Mutant PDB ID	Wild-type Ligand	Mutand Ligand	Desired Mutation	# Design Positions	Percentile	Rank	Percentile	Rank	Percentile	Rank	Percentile	Rank	Percentile	Rank	Percentile	Rank
1A80	1M9H	NDP	NAD	K232G	5	79	22	-	100	64	37	-	100	-	100	-	100
1A80	1M9H	NDP	NAD	R238H	5	66	35	-	100	-	100	-	100	-	100	-	100
1FCB	1SZE	PYR	173	L230A	5	98	3	99	2	100	1	97	4	89	12	100	1
1K70	1RA0	HPY	FPY	D314A	4	100	1	98	3	99	2	100	1	98	3	99	2
1K70	1RA5	HPY	FPY	D314G	4	98	3	99	2	98	3	-	80	-	80	98	3
1K70	1RAK	HPY	FPY	D314S	4	-	80	-	80	-	80	-	80	94	6	-	80
1PK7	1OUM	ADN	TAL	M64V	3	77	15	-	60	-	60	-	60	-	60	-	60
1ZK4	1ZK1	NAP	NAD	G37D	7	99	2	98	4	100	1	99	2	100	1	100	1
2FZN	3E2Q	PRO	HYP	Y540S	2	95	3	100	1	98	2	-	40	-	40	-	40
2H6F	2H6G	FAR	GER	W602T	9	63	68	50	91	66	63	97	6	93	14	-	180
2O7B	2O78	HC4	TCA	H89F	4	79	18	70	25	78	19	-	80	-	80	-	80
3HG5	3LX9	GLA	A2G	E203S	7	-	140	-	140	-	140	92	12	-	140	-	140
3HG5	3LX9	GLA	A2G	L206A	7	72	40	93	11	-	140	99	2	98	4	100	1
3KZO	3L02	AN0	SN0	E92A	5	99	2	99	2	99	2	-	100	100	1	-	100
3KZO	3L04	AN0	SN0	E92P	5	-	100	-	100	54	47	-	100	-	100	-	100
3KZO	3L05	AN0	SN0	E92S	5	94	7	90	11	89	12	-	100	-	100	-	100
3KZO	3L06	AN0	SN0	E92V	5	86	15	61	40	82	19	92	9	-	100	-	100

						Find wild-type amino acid starting from mutant protein structure & wild-type ligand											
						CM BR		CM FKIC		CM WKIC		Backrub Ensemble		Fast Design		FixBB control	
Wild-type PDB ID	Mutant PDB ID	Wild-type Ligand	Mutand Ligand	Desired Mutation	# Design Positions	Percentile	Rank	CMFKIC	Rank	Percentile	Rank	Percentile	Rank	Percentile	Rank	Percentile	Rank
1A80	1M9H	NDP	NAD	G232K	5	-	100	-	100	-	100	-	100	-	100	-	100
1A80	1M9H	NDP	NAD	H238R	5	-	100	-	100	-	100	-	100	-	100	-	100
1FCB	1SZE	PYR	173	A230L	5	97	5	96	7	96	6	88	18	-	100	-	100
1K70	1RA0	HPY	FPY	A314D	4	99	2	95	6	98	3	93	8	83	18	-	80
1K70	1RA5	HPY	FPY	G314D	4	73	23	74	22	74	22	-	80	-	80	-	80
1K70	1RAK	HPY	FPY	S314D	4	-	80	-	80	-	80	-	80	-	80	-	80
1PK7	1OUM	ADN	TAL	V64M	3	80	13	85	10	82	12	-	60	-	60	-	60
1ZK4	1ZK1	NAP	NAD	D37G	7	100	1	100	1	100	1	99	2	-	140	99	2
2FZN	3E2Q	PRO	HYP	S540Y	2	-	40	-	40	-	40	-	40	88	22	-	40
2H6F	2H6G	FAR	GER	T602W	9	59	75	69	57	57	79	85	28	-	180	-	180
2O7B	2O78	HC4	TCA	F89H	4	85	13	74	22	-	80	-	80	-	80	-	80
3HG5	3LX9	GLA	A2G	S203E	7	80	29	76	35	80	29	92	12	91	8	97	5
3HG5	3LX9	GLA	A2G	A206L	7	59	59	58	60	59	58	-	140	88	13	-	140
3KZO	3L02	AN0	SN0	A92E	5	90	9	94	6	95	5	98	3	-	100	100	1
3KZO	3L04	AN0	SN0	P92E	5	89	12	88	13	92	9	91	10	-	100	95	6
3KZO	3L05	AN0	SN0	S92E	5	84	17	85	16	86	15	95	6	-	100	-	100
3KZO	3L06	AN0	SN0	V92E	5	86	18	93	10	89	14	74	32	-	100	95	7

Table 2.4: Compute time.
Total compute time for each method. Values represent the mean plus or minus the standard deviation across 400 trajectories.

Method	Time (hours)		
CM-BR	52	±	27
CM-FKIC	75	±	43
CM-WKIC	73	±	27
FastDesign	1,502	±	1,377
Backrub + FixBB	117 5	± ±	115 3
= Backrub Ensemble	122	±	115

2.7 Supplemental Figures

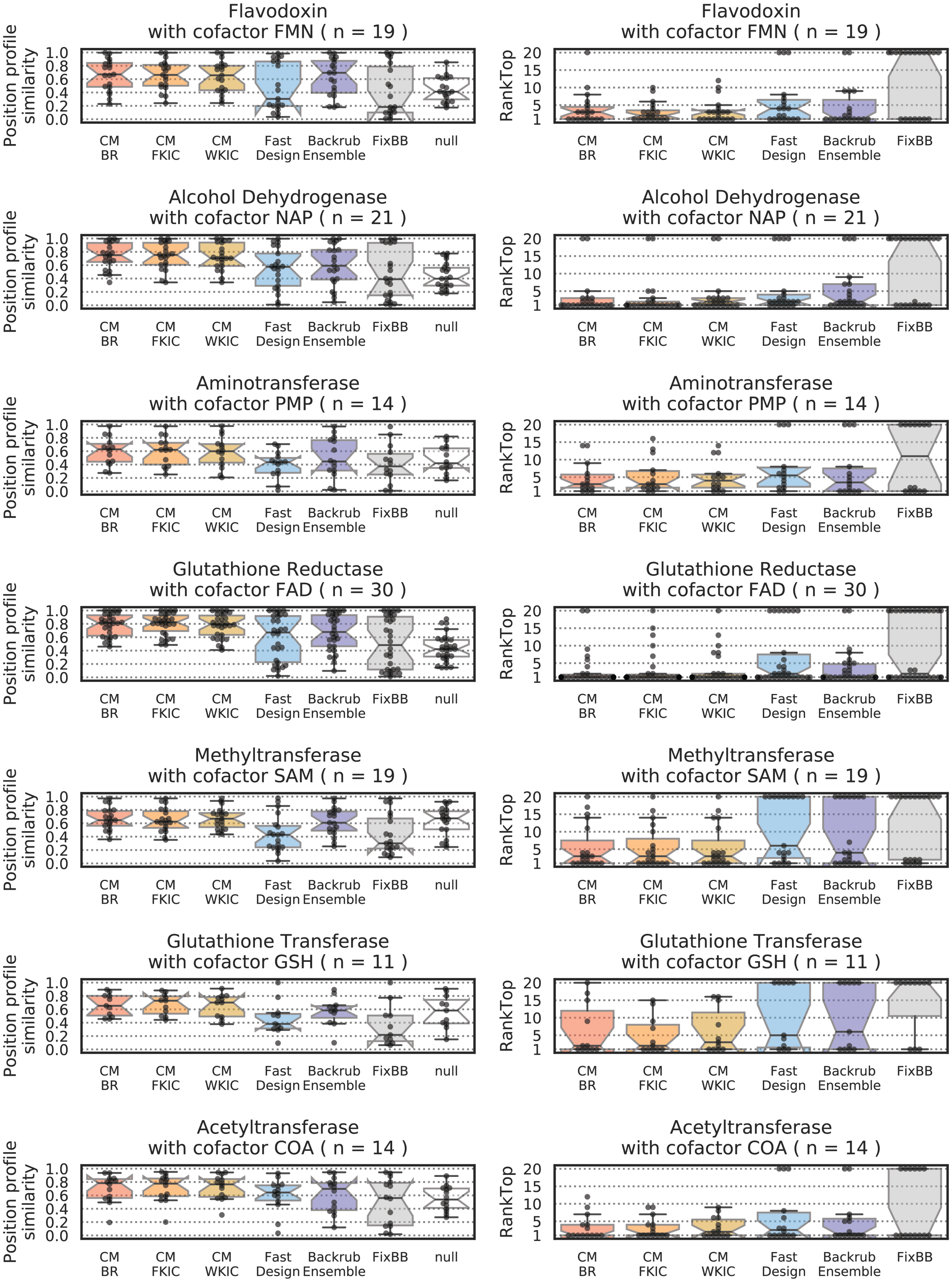


Figure S2.1: Position profile similarity and RankTop for all designed positions (n) for each protein family in the cofactor dataset. Continued on next page.

Figure S2.1, continued. Position profile similarity and RankTop for all designed positions (*n*) for each protein family in the cofactor dataset.

Distributions are shown as boxplots, while values for individual positions are overlaid as swarms of black points. For PPS (left), a value of 1 means the design method perfectly recapitulated the known sequence profile, whereas a value of zero means that the design method did not model any of the amino acid side chain identities from the known profile. For RankTop (right), a value of 1 means that the design method correctly identified the most frequent amino acid side chain observed in the known profile, whereas a RankTop of 20 means that side chain was observed with zero frequency, or that all side chains were modeled with some frequency and the top known was the least frequent. Median is marked with a horizontal black line, and notches represent a 95% confidence interval (CI) around the median; when CI extends past the quartiles, notches extend beyond the box, leading to a "flipped" appearance. The boxplot covers the second and third quartiles, and the vertical whiskers mark 1.5 times the inter-quartile range.

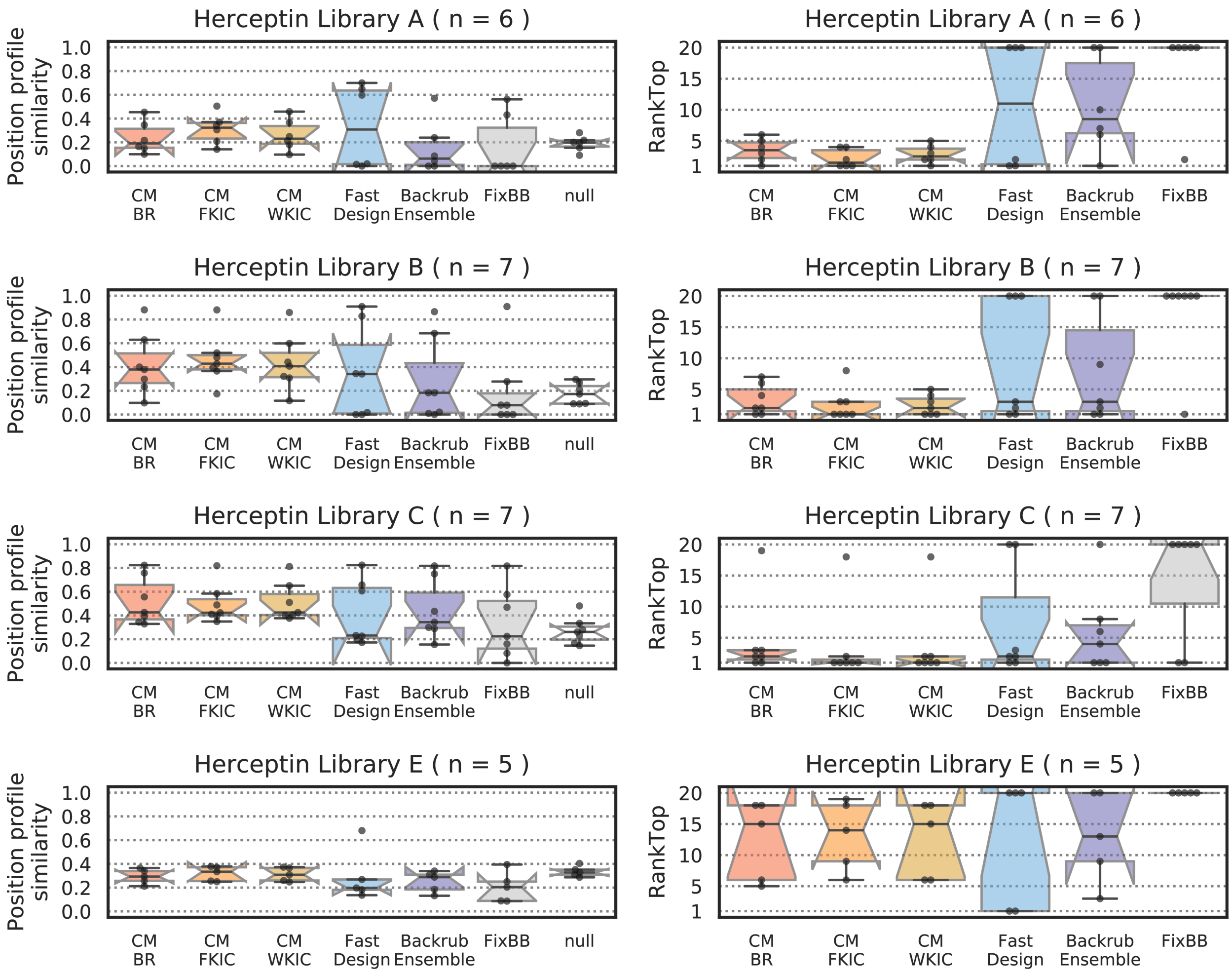


Figure S2.2: Profile similarity and rank top for all designed positions (*n*) for individual libraries of the Herceptin/HER2 dataset.

Distributions are shown as boxplots, while values for individual positions are overlaid as swarms of black points. For PPS (left), a value of 1 means the design method perfectly recapitulated the known sequence profile, whereas a value of zero means that the design method did not model any of the amino acid side chain identities from the known profile. For RankTop (right), a value of 1 means that the design method correctly identified the most frequent amino acid side chain observed in the known profile, whereas a RankTop of 20 means that side chain was observed with zero frequency, or that all side chains were modeled with some frequency and the top known was the least frequent. Median is marked with a horizontal black line, and notches represent a 95% confidence interval (CI) around the median; when CI extends past the quartiles, notches extend beyond the box, leading to a "flipped" appearance. The boxplot covers the second and third quartiles, and the vertical whiskers mark 1.5 times the inter-quartile range.

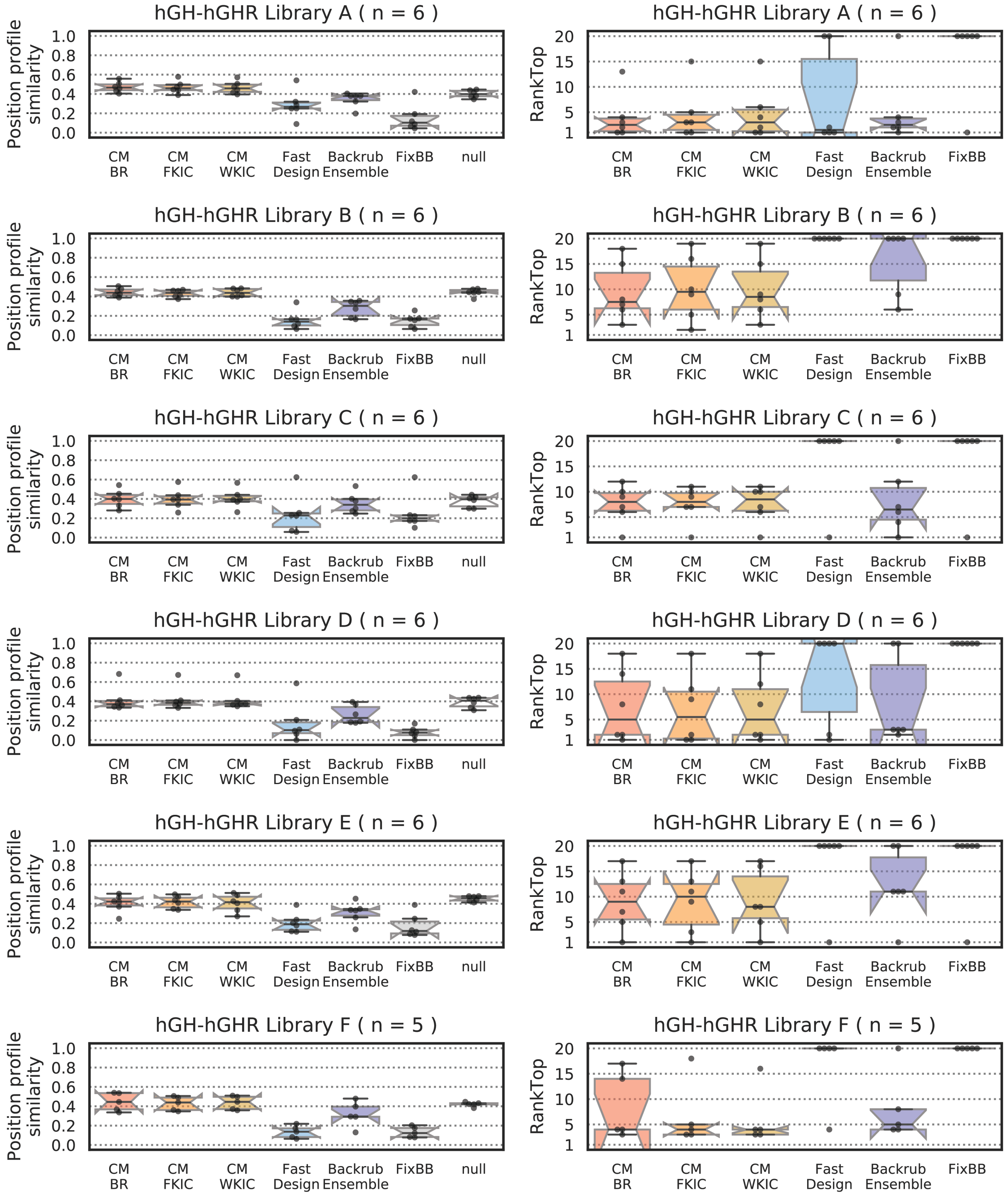


Figure S2.3: Profile similarity and rank top for all designed positions (*n*) for individual libraries of the hGH/hGHR dataset.

Distributions are shown as boxplots, while values for individual positions are overlaid as swarms of black points. For PPS (left), a value of 1 means the design method perfectly recapitulated the known sequence profile, whereas a value of zero means that the design method did not model

Continued on next page.

Figure S2.3, continued: Profile similarity and rank top for all designed positions (*n*) for individual libraries of the hGH/hGHR dataset.

any of the amino acid side chain identities from the known profile. For RankTop (right), a value of 1 means that the design method correctly identified the most frequent amino acid side chain observed in the known profile, whereas a RankTop of 20 means that side chain was observed with zero frequency, or that all side chains were modeled with some frequency and the top known was the least frequent. Median is marked with a horizontal black line, and notches represent a 95% confidence interval (CI) around the median; when CI extends past the quartiles, notches extend beyond the box, leading to a "flipped" appearance. The boxplot covers the second and third quartiles, and the vertical whiskers mark 1.5 times the inter-quartile range.

Cofactor positions

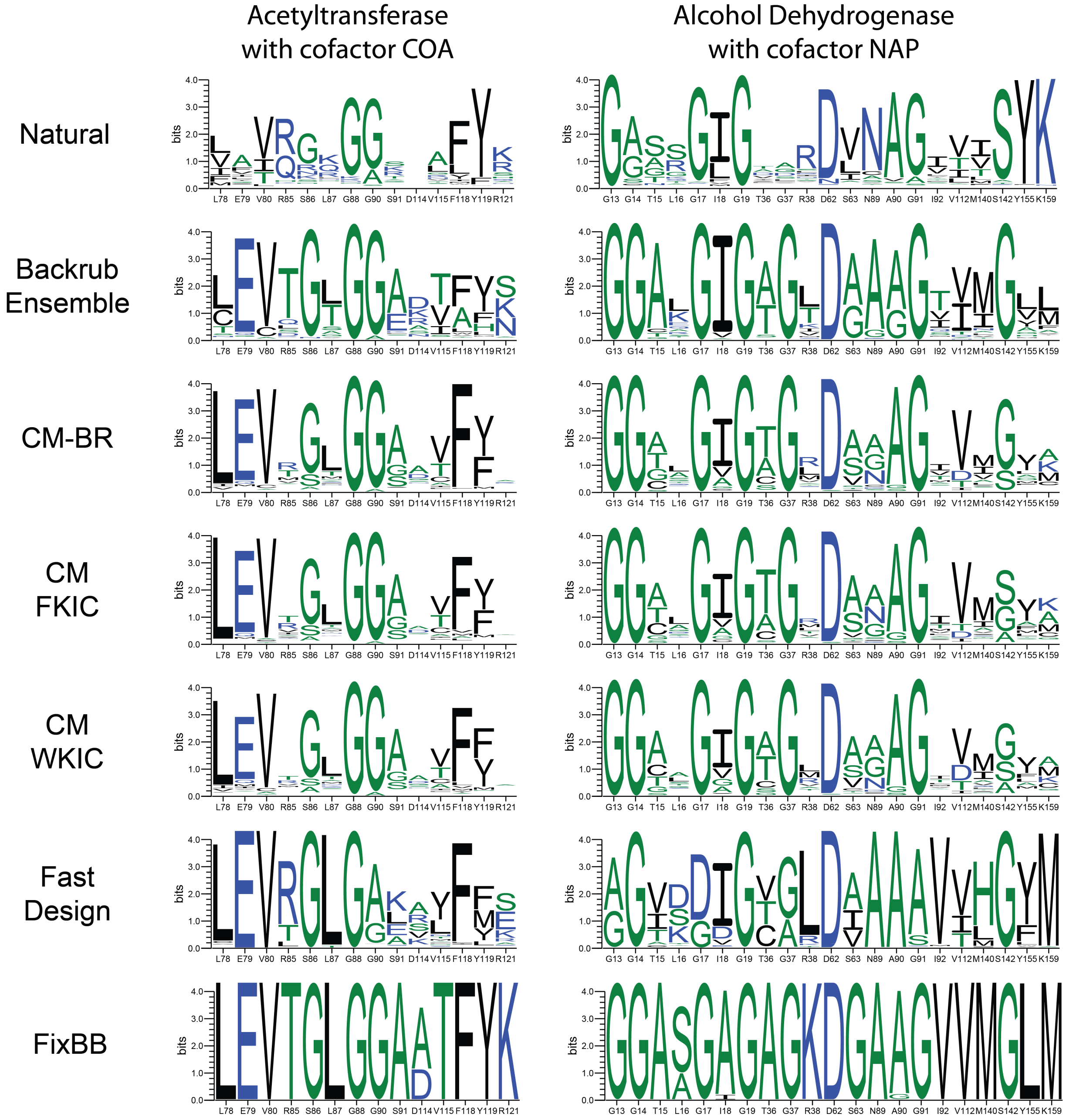


Figure S2.4: Sequence logos for predicted and known binding site sequences of the cofactor dataset.

The height of each letter is proportional to its contribution to the column's information content. The height of each column is inversely proportional to the sequence variation at that position. **Continued on next page.**

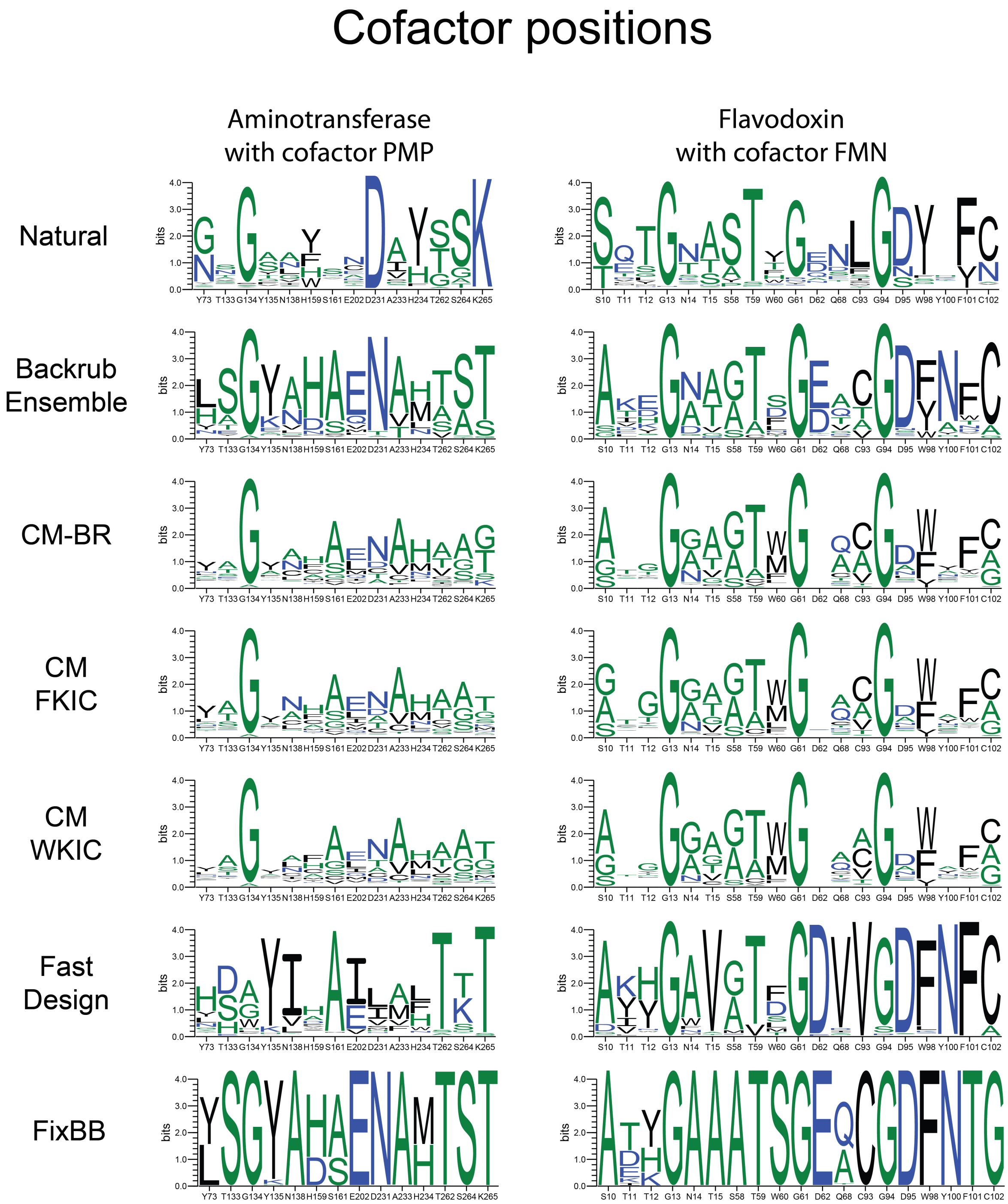


Figure S2.4, continued: Sequence logos for predicted and known binding site sequences of the cofactor dataset.

The height of each letter is proportional to its contribution to the column’s information content. The height of each column is inversely proportional to the sequence variation at that position. **Continued on next page.**

Cofactor positions

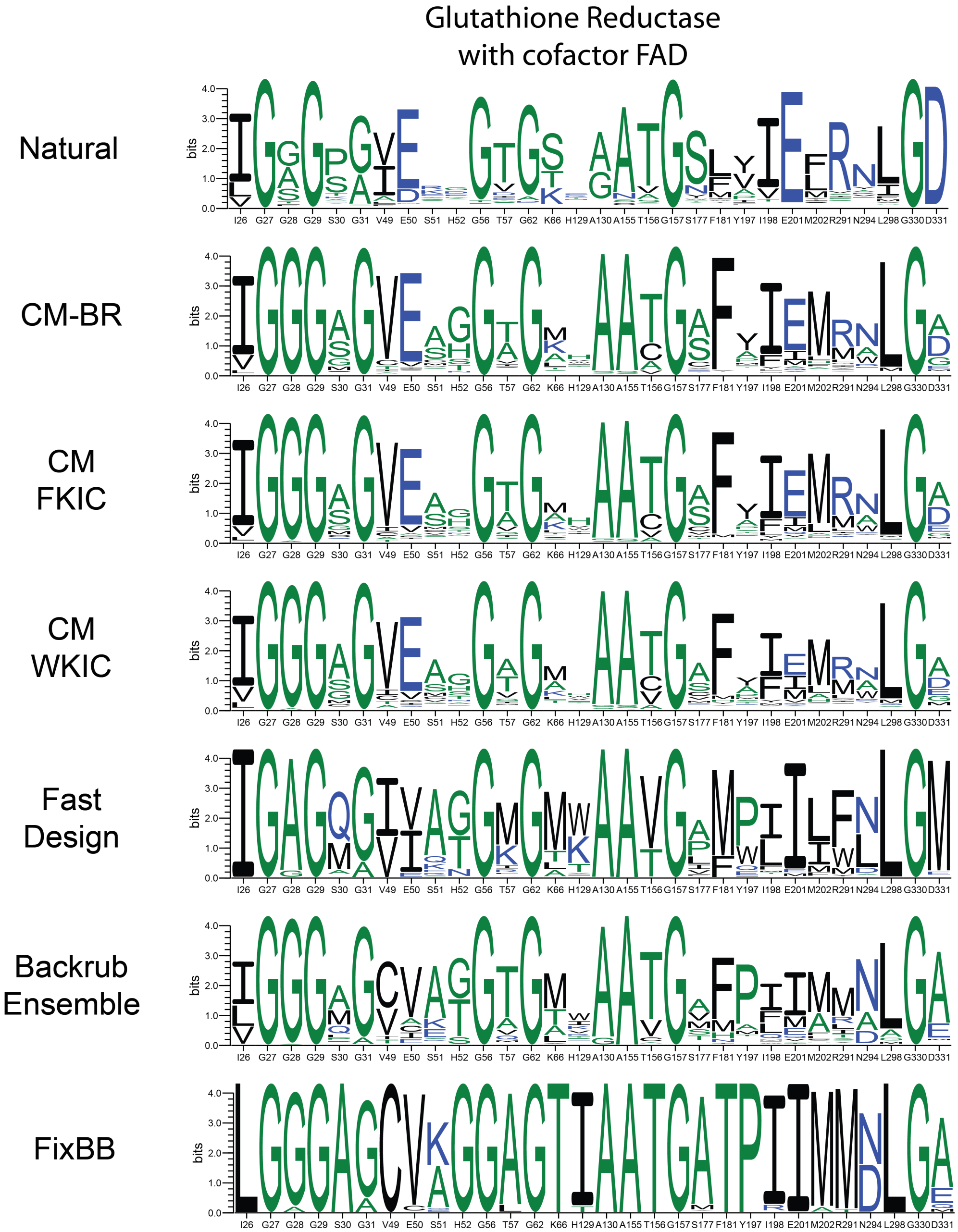


Figure S2.4, continued: Sequence logos for predicted and known binding site sequences of the cofactor dataset.
The height of each letter is proportional to its contribution to the column's information content. The height of each column is inversely proportional to the sequence variation at that position. **Continued on next page.**

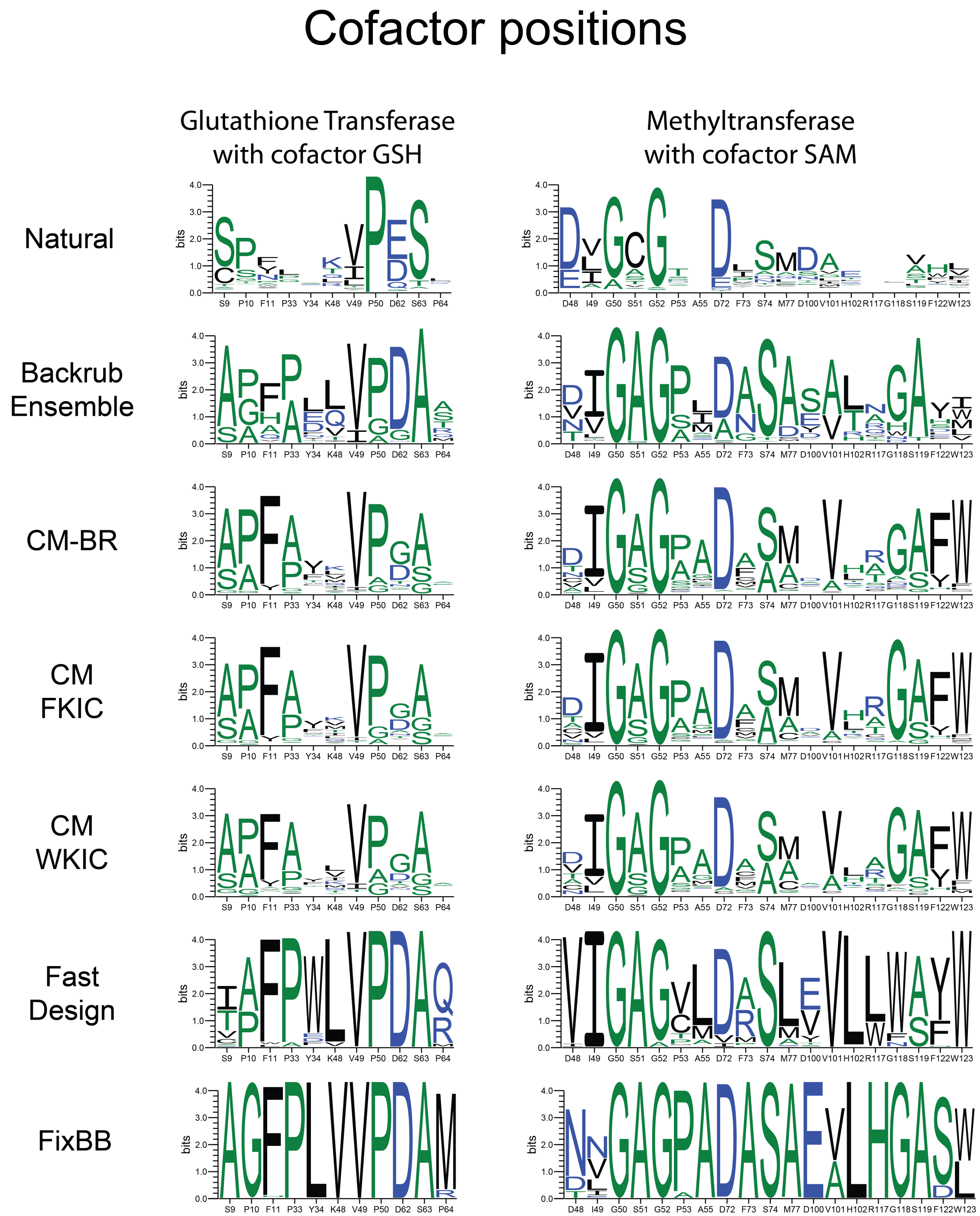


Figure S2.4, continued: Sequence logos for predicted and known binding site sequences of the cofactor dataset.

The height of each letter is proportional to its contribution to the column's information content. The height of each column is inversely proportional to the sequence variation at that position.

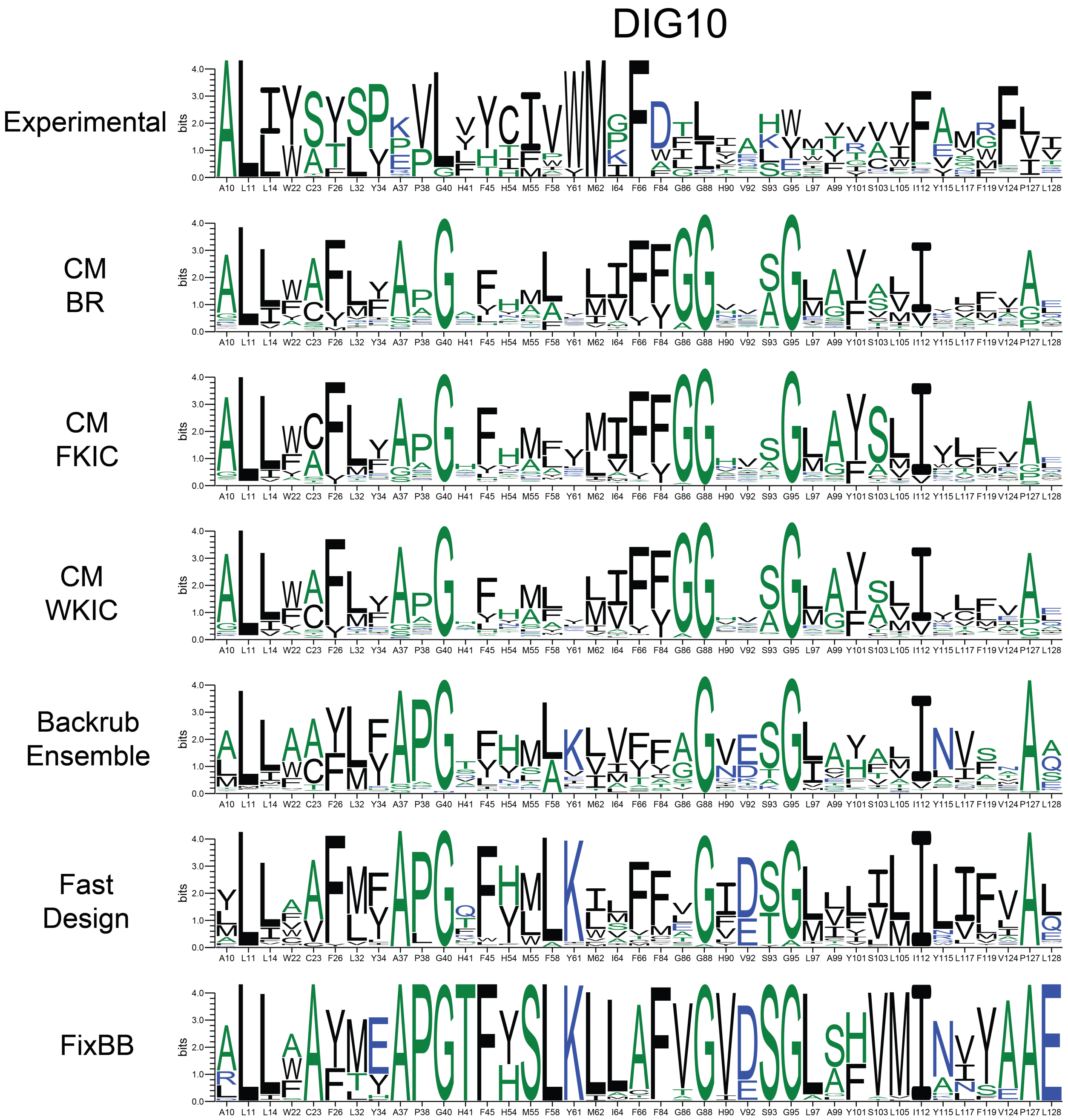


Figure S2.5: Sequence logos for predicted and known binding site sequences of the DIG10 dataset.

The height of each letter is proportional to its contribution to the information content of the column. The height of each column is inversely proportional to the sequence variation at that position. The experimental profile shows amino acid residues that were enriched in the experimental selection.

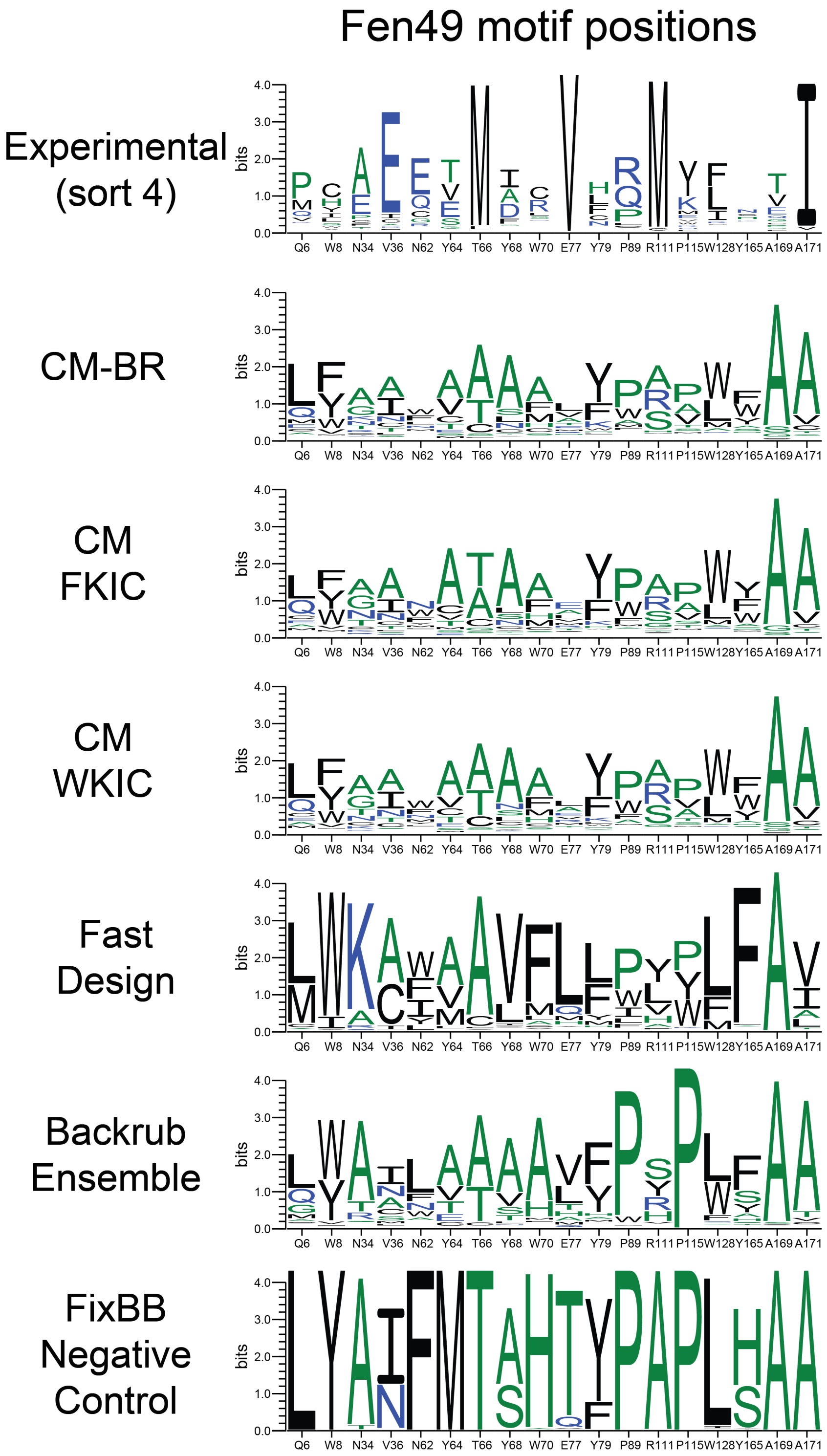


Figure S2.6: Sequence logos for predicted and known binding site sequences of the Fen49 dataset.

The height of each letter is proportional to its contribution to the information content of the column. The height of each column is inversely proportional to the sequence variation at that position. Experimental data are taken from sort 4 of the library.[2]

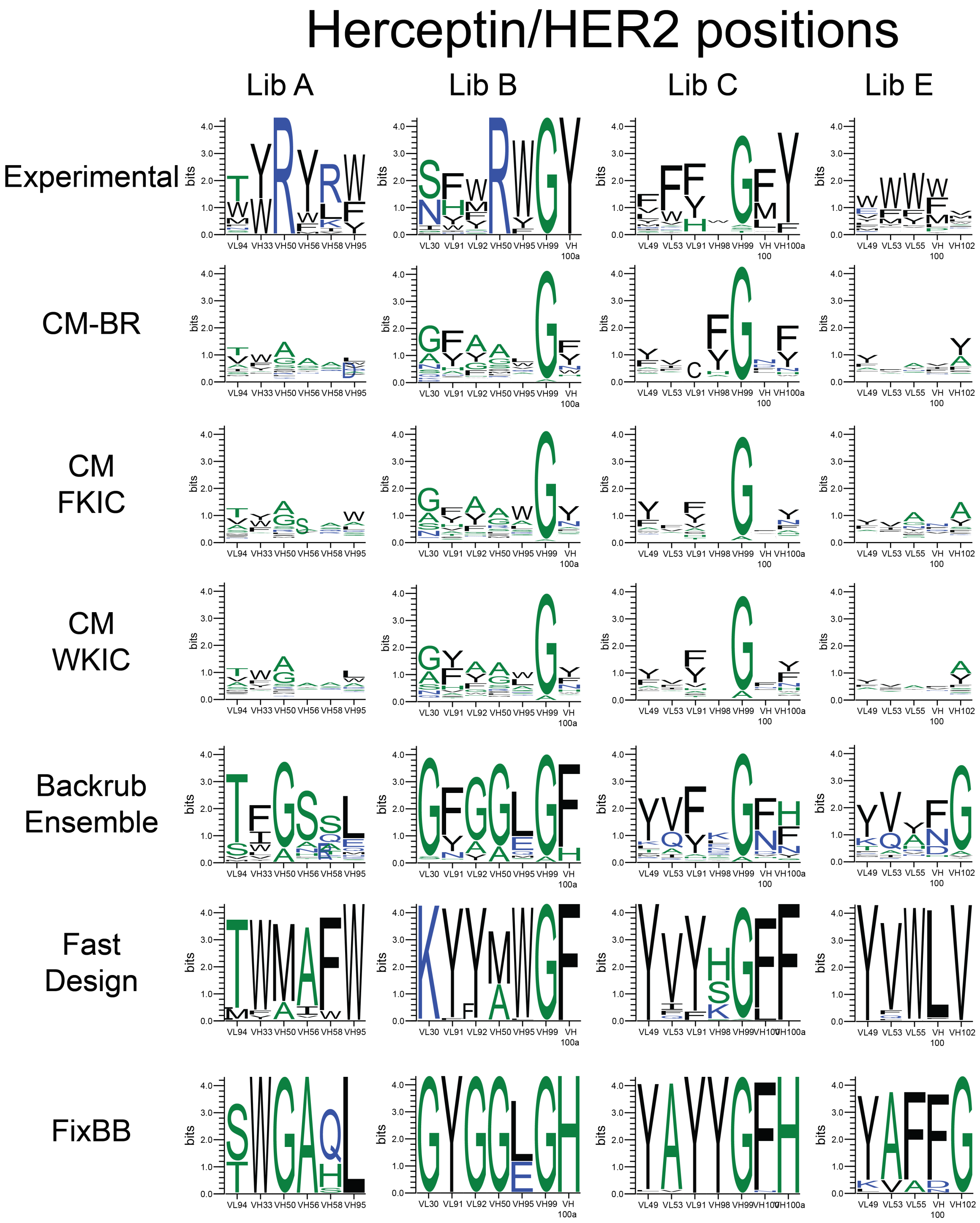


Figure S2.7: Sequence logos for predicted and known binding site sequences for the Herceptin/HER2 dataset.

The height of each letter is proportional to its contribution to the information content of the column. The height of each column is inversely proportional to the sequence variation at that position. Different experimental libraries (Lib A, B, C, E) are indicated. Library D was omitted because the experimental data were dominated by the wild-type sequence. Residues are labeled with Kabat numbering.

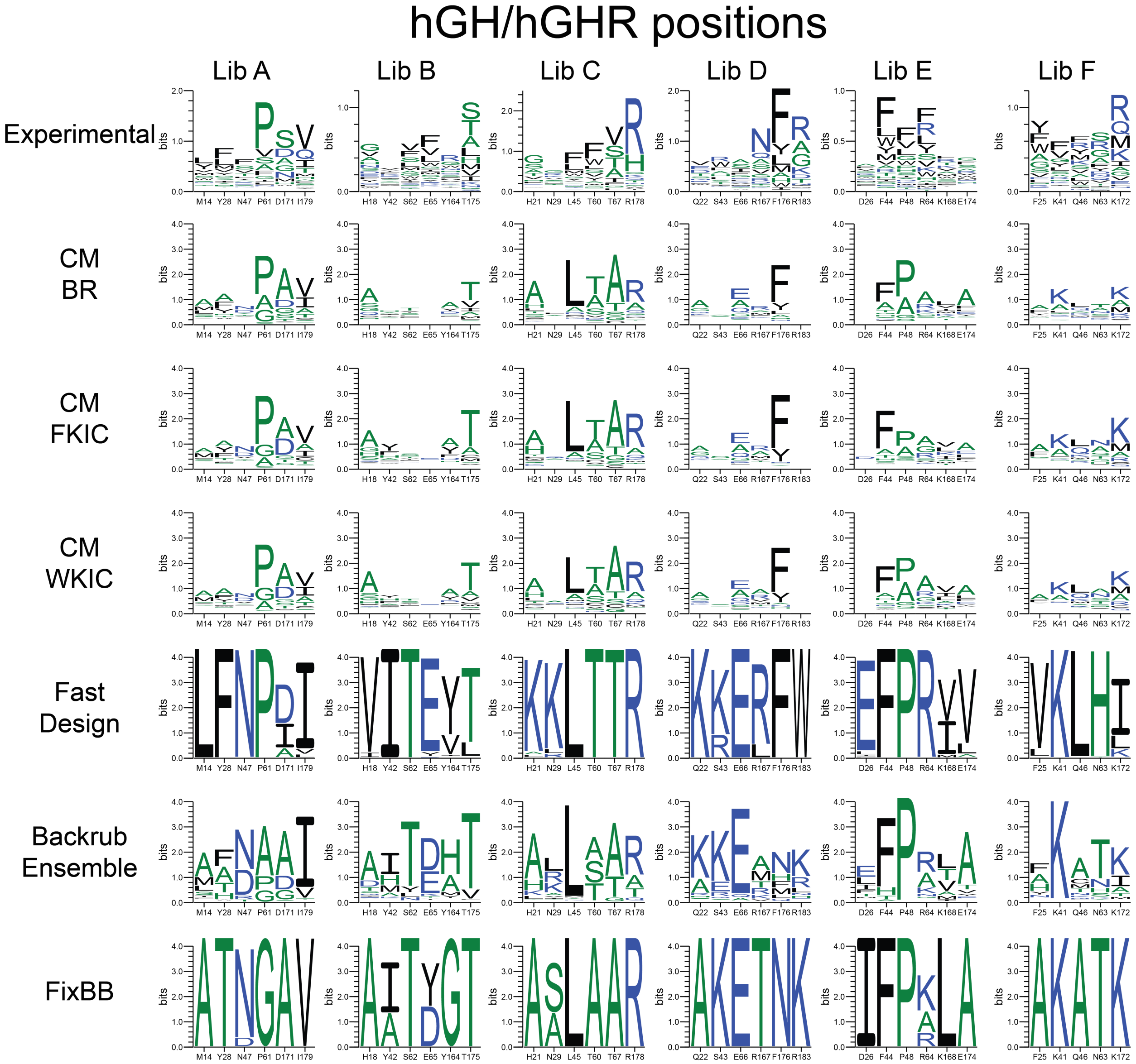


Figure S2.8: Sequence logos for predicted and known binding site sequences of the hGH/hGHR dataset.

The height of each letter is proportional to its contribution to the information content of the column. The height of each column is inversely proportional to the sequence variation at that position. Different experimental libraries (Lib A, B, C, D, E, F) are indicated.

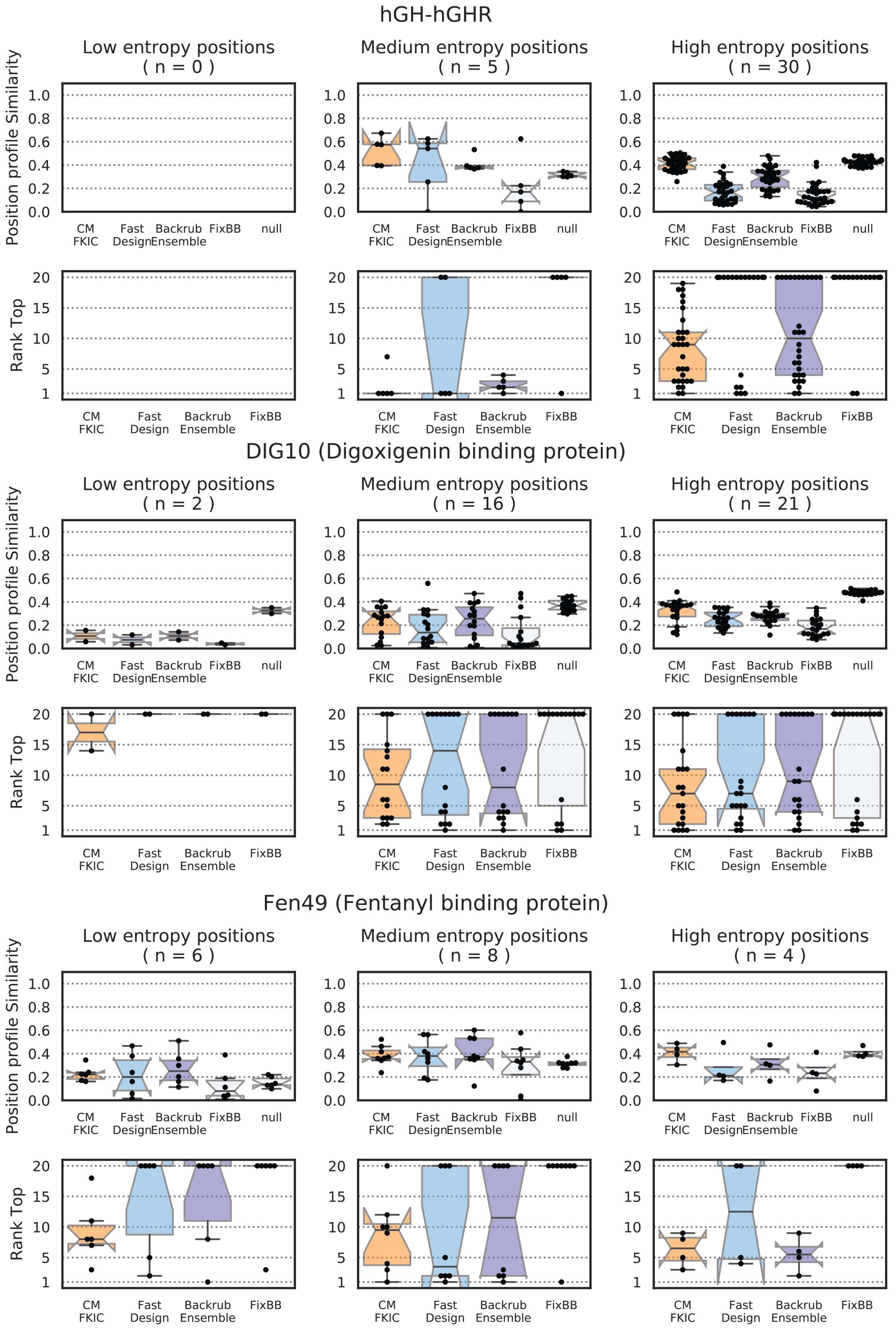


Figure S2.9: Profile similarity and RankTop as a function of known sequence entropy for the hGH/hGHR, DIG10 and Fen49 datasets. Continued on next page.

Figure S2.9, continued: Profile similarity and RankTop as a function of known sequence entropy for the hGH/hGHR, DIG10 and Fen49 datasets.

Each point represents one sequence position. The Herceptin/HER2 and Cofactor datasets are shown in Figure 2.5 in the main text. For each dataset (indicated in the header), profile similarity and RankTop are binned by entropy of the known sequence profile at each position (low: entropy ≤ 0.33, medium: 0.33 < entropy ≤ 0.67, and high: entropy > 0.67). The number of low entropy positions in these three datasets is small. The boxplot covers the second and third quartiles, and the vertical whiskers mark 1.5 times the inter-quartile range. Median is marked with a horizontal black line, and notches represent a 95% confidence interval (CI) around the median; when CI extends past the quartiles, notches extend beyond the box, leading to a "flipped" appearance.

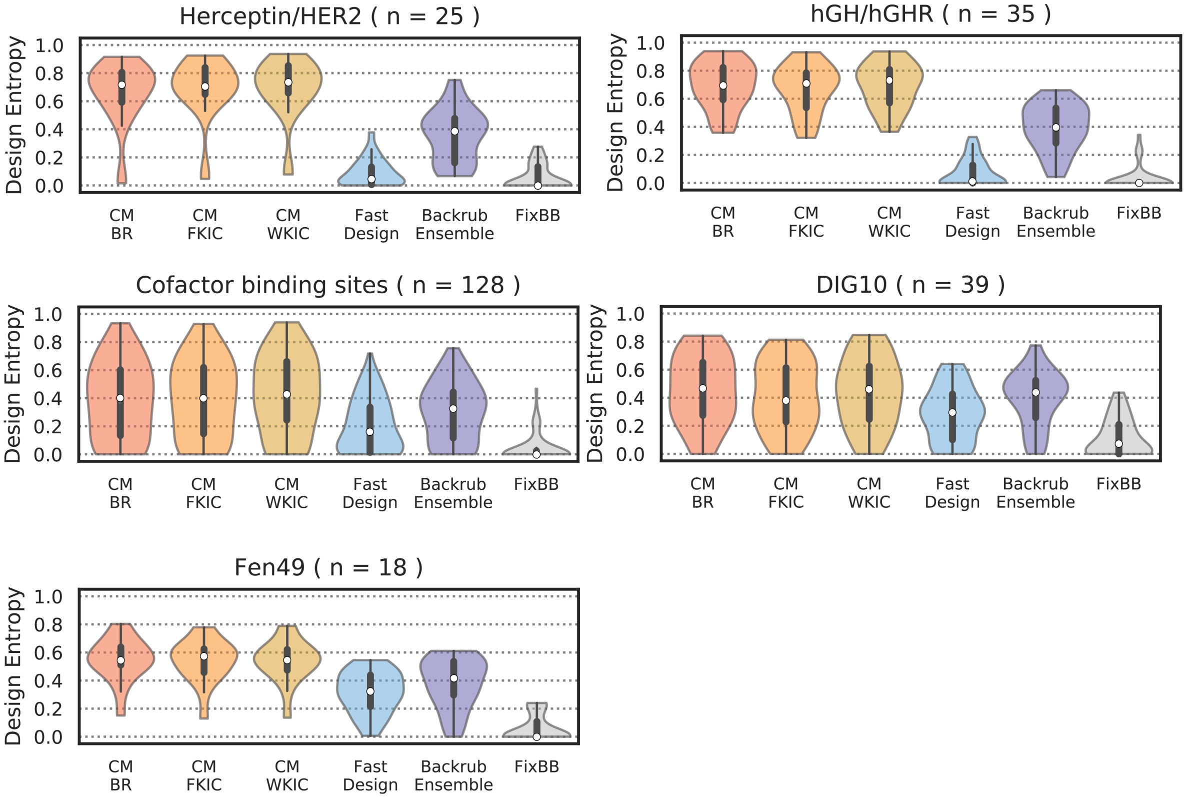


Figure S2.10: Design entropy.

Shown are the distributions of entropy of the design sequence profiles for each designed position in each benchmark. 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.

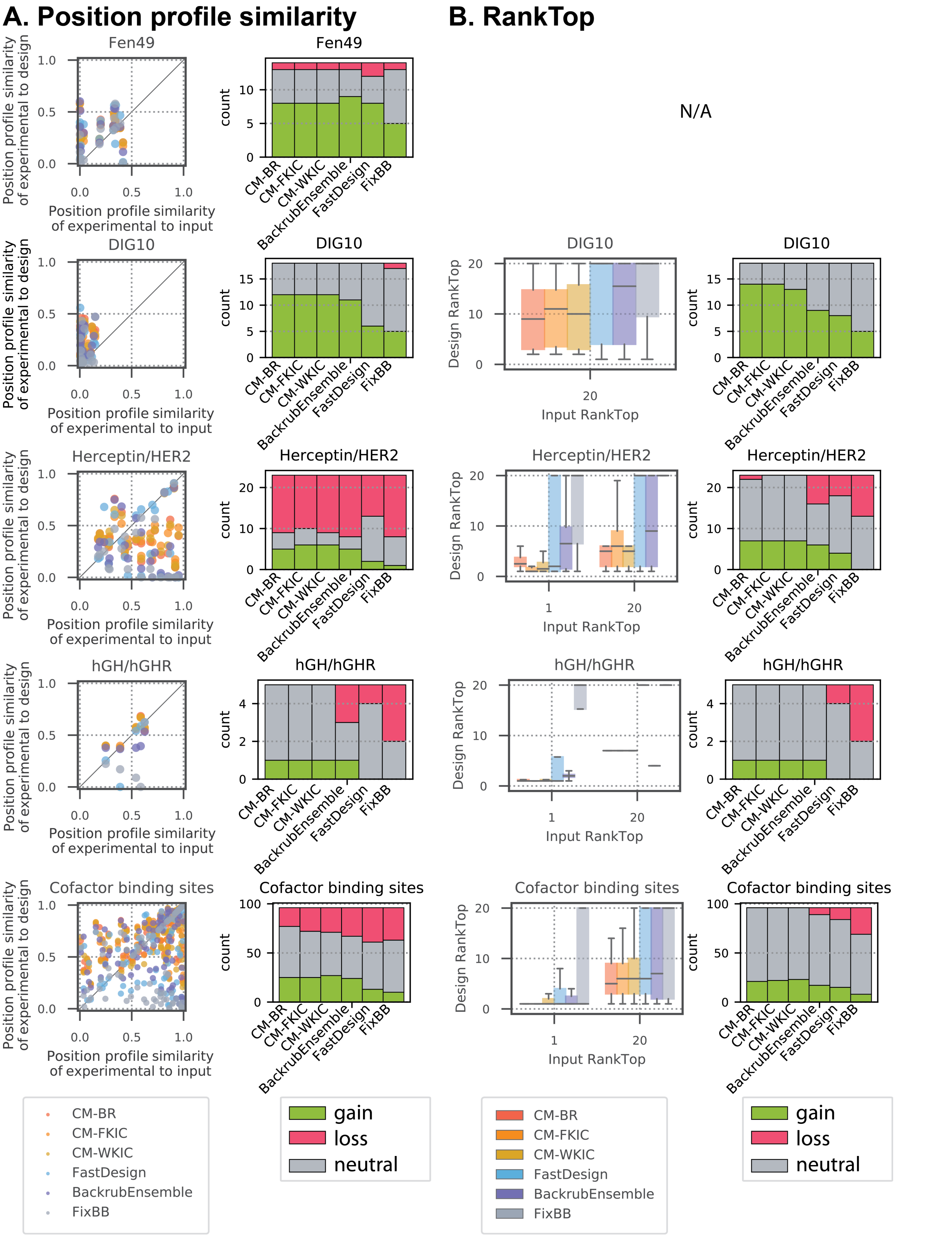


Figure S2.11: Position profile similarity and RankTop as a function of similarity between the input sequence and the known profile at each position. Continued on next page.

Figure S2.11, continued: Position profile similarity and RankTop as a function of similarity between the input sequence and the known profile at each position.
When a preferred side chain from the known sequence profiles is not present in the input sequence, methods can achieve "gain" (green) by identifying correct amino acids with high frequency or rank. Alternatively, when a preferred side chains is present in the input, inaccurate design can cause "loss" (red). Only positions with low and medium entropy (≤ 0.67) are considered. (**A**) Left: PPS as a function of similarity to the input sequence for all profile datasets. Each point represents one position in the protein sequence, colored by design method. Right: Quantifications of number of designed sequence positions in gain, loss, and neutral zones. Gain and loss zones are defined by a threshold of 0.1 difference between input-known PPS and design-known PPS. (**B**) Left: Boxplots of each method's RankTop as a function of similarity to the input sequence. 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. 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. All profile datasets are shown except Fen49, which is omitted because the fentanyl deep sequencing data do not include the input sequence. For the digoxigenin dataset, there are no consensus positions for which the top experimentally selected side chain was present in the starting sequence. Right: Quantification of sequence positions in gain, loss, and neutral zones for RankTop values.

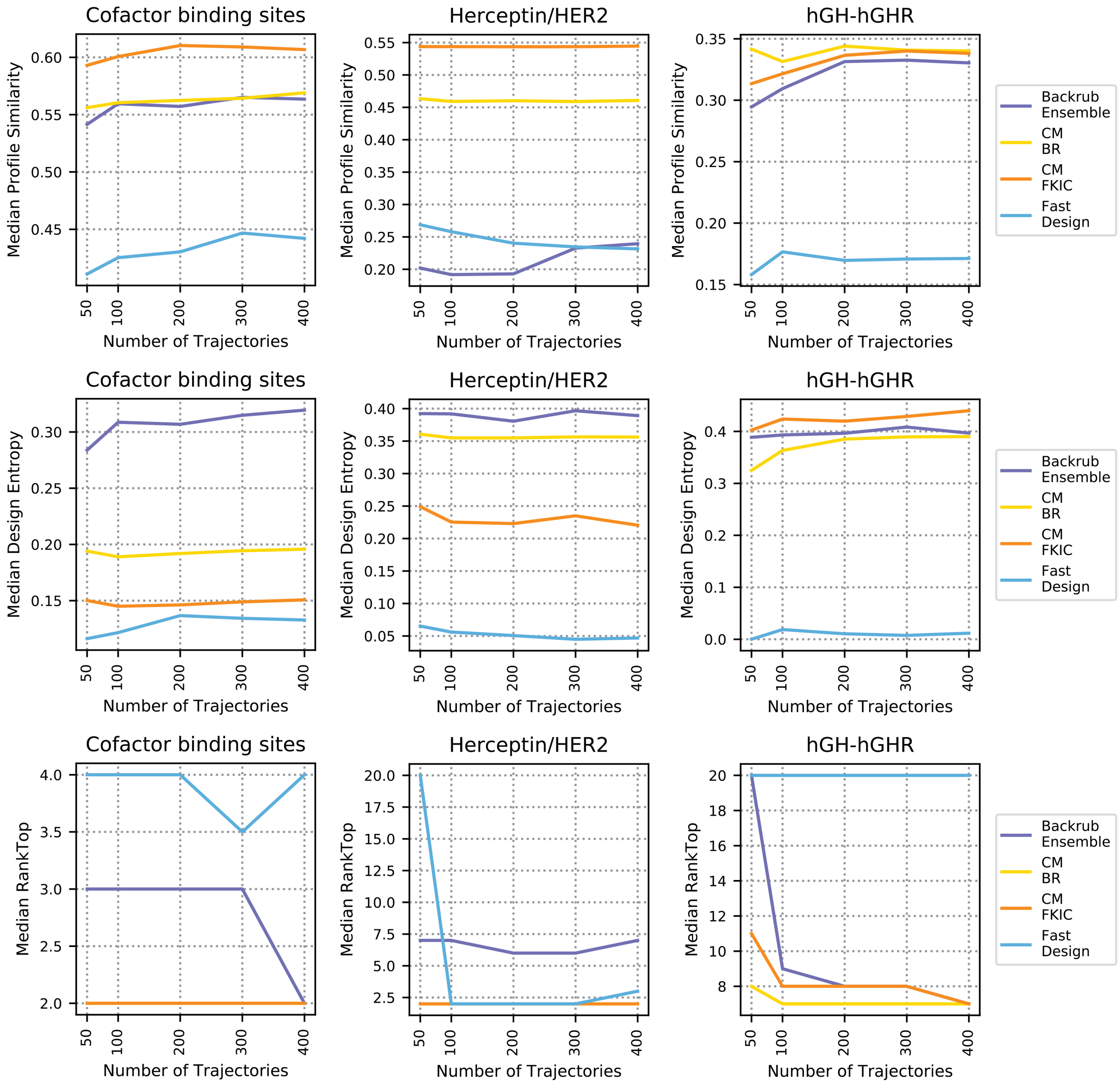
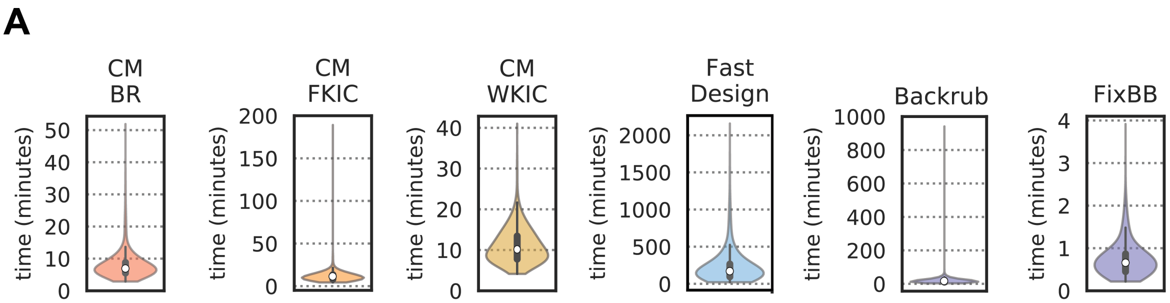


Figure S2.12: Number of trajectories.

Comparison of median PPS, design entropy and RankTop as a function of number of design trajectories (*n*) for the Cofactor, Herceptin, and hGH/hGHR datasets for each method.



B

method	mean job time (seconds)		number of trajectories, *n*		total time (job time * *n*)
CM-BR	469 ± 243	*	400	=	187,651 ± 97,227
CM-FKIC	673 ± 385	*	400	=	269,266 ± 153,884
CM-WKIC	658 ± 244	*	400	=	263,232 ± 97,452
FastDesign	13516 ± 12394	*	400	=	5,406,286 ± 4,957,416
Backrub	1056 ± 1036	*	400	=	422,374 ± 414,488
+ FixBB	44 ± 23	*	400	=	17,519 ± 9,202
= Backrub Ensemble					439,893 ± 414,590

Figure S2.13: Compute time.
(A) Distribution of individual trajectory compute times for each method. The median of the distributions is marked by the horizontal red bar. Second and third quartiles are marked by the thick black bar, and the thin bar marks 1.5 times the inter-quartile range. (B) Total time is calculated by multiplying mean job time ± standard deviation by the number of trajectories (*n*) required for consistent performance. BackrubEnsemble is the sum of two methods, Backrub (to generate conformational ensemble) and FixBB (to design sidechains). For the summed BackrubEnsemble time, standard deviation is the square root of the sum of the individual deviations squared.

2.8 Supplemental Tables

Table S2.1: Cofactor dataset ligands.
Ligand name and 3-letter PDB ligand identifier are shown for each protein family of the Cofactor dataset.

Protein	**Ligand**
Acetyl Transferase	Coenzyme A (COA)
Alcohol Dehydrogenase	nicotinamide-adenine-dinucleotide phosphate (NADP) (NAP)
Amino-transferase	4'-deoxy-4'-aminopyridoxal-5'-phosphate (PMP)
Flavodoxin	flavin mono-nucleotide (FMN)
Glutathione S-Transferase	glutathione (GSH)
Methyl-transferase	S-adenosyl-methionine (SAM)
Glutathione Reductase	flavin-adenine dinucleotide (FAD)

Table S2.2: Cofactor benchmark structures and positions.
Protein names and PDB codes are shown, along with designable and packable positions. Position numbering corresponds to PDB numbering.

Protein	PDB	Designable positions	Packable positions
Acetyl Transferase	3S6F	78, 79, 80, 85, 86, 87, 88, 90, 91, 114, 115, 118, 119, 121 (n = 14)	4, 6, 26, 27, 28, 52, 60, 62, 63, 75, 77, 81, 84, 89, 93, 94, 95, 109, 111, 112, 116, 117, 122, 124, 126, ligand (n = 26)
Alcohol Dehydrogenase	1ZK4	13, 14, 15, 16, 17, 18, 19, 36, 37, 38, 62, 63, 89, 90, 91, 92, 112, 140, 142, 155, 159, 187, 189, 190, 192, 194, 195, 205 (n = 28)	11, 12, 22, 23, 34, 35, 39, 42, 45, 46, 58, 59, 61, 64, 69, 72, 87, 93, 94, 108, 109, 113, 117, 138, 143, 144, 149, 152, 156, 158, 162, 185, 186, 191, 197, 198, 201, 202, 206, 210, 211, 216, 219, 221, 222, 244, 248, ligand (n = 48)
Aminotransferase	2XBN	73, 133, 134, 135, 138, 159, 161, 202, 231, 233, 234, 262, 264, 265 (n = 14)	75, 136, 137, 139, 141, 142, 155, 158, 162, 164, 200, 204, 205, 206, 236, 260, 270, 273, 347, 349, 390, 392, ligand (n = 23)
Flavodoxin	1F4P	10, 11, 12, 13, 14, 15, 58, 59, 60, 61, 62, 68, 93, 94, 95, 98, 100, 101, 102 (n = 19)	8, 16, 17, 18, 19, 57, 65, 66, 69, 70, 71, 91, 96, 97, 105, 106, 125, 126, 127, 130, ligand (n = 21)
Glutathione S-Transferase	3R2Q	9, 10, 11, 33, 34, 48, 49, 50, 62, 63, 64, 98, 105 (n = 13)	3, 4, 6, 12, 13, 14, 15, 31, 38, 40, 43, 44, 46, 51, 52, 61, 66, 67, 68, 91, 94, 95, 101, 102, 108, 109, 112, 157, 160, 164, 168, ligand (n = 32)
Methyltransferase	3DLC	8, 16, 20, 28, 48, 49, 50, 51, 52, 53, 55, 72, 73, 74, 77, 100, 101, 102, 117, 118, 119, 122, 123 (n = 23)	7, 13, 17, 19, 21, 24, 27, 31, 32, 46, 47, 56, 57, 58, 59, 68, 70, 71, 76, 78, 80, 81, 84, 103, 104, 114, 115, 116, 120, 121, 125, 126, 128, 129, 130, 132, 133, 144, 207, ligand (n = 40)
Glutathione Reductase	3DK9	26, 27, 28, 29, 30, 31, 49, 50, 51, 52, 56, 57, 58, 62, 63, 66, 129, 130, 155, 156, 157, 177, 181, 197, 198, 201, 202, 291, 294, 298, 330, 331, 337, 338, 339, 340, 342, 372 (n = 38)	24, 25, 33, 35, 47, 48, 54, 61, 64, 65, 67, 70, 103, 114, 125, 126, 127, 131, 132, 140, 142, 147, 153, 154, 159, 160, 180, 192, 200, 205, 206, 223, 226, 286, 288, 295, 297, 300, 329, 332, 336, 341, 343, 344, 369, 370, 371, 373, 376, 377, 441, ligand (n = 52)

Table S2.3: Enzyme dataset.

Ligand names are shown for wild-type and mutant proteins of the Enzyme specificity dataset.

Protein	**Wild-type ligand**	**Mutant ligand**
2-5-diketo-D-gluconic acid reductase A	dihydro-nicotinamide-adenine-dinucleotide phosphate (NADPH) (NDP)	nicotinamide-adenine-dinucleotide (NAD)
Alcohol dehydrogenase	nicotinamide-adenine-dinucleotide (NAD)	NADP nicotinamide-adenine-dinucleotide phosphate (NAP)
Alpha-galactosidase A	N-actyl-2-deoxy-2-amino-galactose (A2G)	alpha D-galactose (GLA)
Cytosine deaminase	(4S)-5-fluoro-4-hydroxy-3,4-dihydropyrimidin-2(1H)-one (FPY)	4-hydroxy-3,4-dihydro-1H-pyrimidin-2-one (HPY)
Farnesyltransferase	geran-8-yl geran (GER)	farnesyl (FAR)
Flavocytochrome b(2)	benzoyl-formic ACID (173)	pyruvic acid (PYR)
Histidine ammonialyase	phenylethylene-carboxylic acid (TCA)	para-coumaric acid (HC4)
N-acetylornithine carbamoyltransferase	N-(3-carboxypropanoyl-L-norvaline (SN0)	N-acetyl-L-norvaline (AN0)
Proline dehydrogenase	4-hydroxyproline (HYP)	proline (PRO)
Purine nucleoside phosphorylase	9-(6-deoxy-alpha-L-talofuranosyl)-6-methylpurine (TAL)	adenosine (ADN)

Table S2.4: Enzyme specificity benchmark structures and positions.
Proteins (wild-type and mutant) and ligand names and PDB codes are shown, along with designable and packable positions. Position numbering corresponds to PDB numbering. **Continued on next page.**

Protein	**PDB**	**Designable positions**	**Packable positions**
2-5-diketo-D-gluconic acid reductase A	1M9H (mutant)	232, 233, 234, 235, 238 (n = 5)	19, 22, 23, 24, 25, 28, 32, 41, 190, 215, 231, 237, 239, 241, 242, ligand (n = 16)
	1A80 (wild-type)	232, 233, 234, 235, 238 (n = 5)	19, 22, 23, 24, 25, 28, 32, 41, 190, 215, 231, 237, 239, 241, 242, ligand (n = 16)
Alcohol dehydrogenase	1ZK1 (mutant)	13, 14, 15, 16, 36, 37, 38, 42 (n = 8)	10, 11, 12, 23, 33, 34, 35, 39, 41, 46, 58, 59, 61, 62, 63, 64, 69, 72, 89, 90, 112, 192, 193, 194, ligand (n = 25)
	1ZK4 (wild-type)	13, 14, 15, 16, 36, 37, 38 (n = 7)	10, 11, 12, 23, 33, 34, 35, 39, 42, 46, 56, 58, 59, 61, 62, 63, 69, 72, 89, 90, 192, 193, 194, ligand (n = 24)
Alpha-galactosidase A	3LX9 (mutant)	170, 203, 206, 207, 227, 229, 231 (n = 7)	47, 92, 93, 134, 136, 137, 141, 142, 168, 172, 174, 177, 180, 184, 201, 204, 208, 209, 228, 241, 242, 245, 246, 249, 253, 264, 266, 267, ligand (n = 29)
	3HG5 (wild-type)	170, 203, 206, 207, 227, 229, 231 (n = 7)	47, 51, 92, 93, 134, 136, 137, 141, 142, 168, 172, 174, 177, 180, 184, 201, 204, 208, 209, 211, 228, 241, 242, 245, 246, 249, 264, 265, 266, 267, ligand (n = 31)
Cytosine deaminase	1K70 (wild-type)	63, 313, 314, 319 (n = 4)	61, 65, 66, 81, 85, 88, 122, 124, 154, 214, 217, 246, 273, 275, 278, 279, 282, 317, 318, 320, ligand (n = 21)
	1RA5 (mutant)	63, 313, 314, 319 (n = 4)	61, 65, 66, 81, 85, 88, 122, 124, 154, 156, 214, 217, 246, 273, 275, 278, 279, 282, 317, 318, 320, ligand (n = 22)
	1RAK (mutant)	63, 313, 314, 319 (n = 4)	61, 65, 66, 81, 85, 88, 122, 124, 154, 156, 214, 217, 246, 273, 275, 278, 279, 282, 317, 318, 320, ligand (n = 22)
	1RA0 (mutant)	63, 313, 314, 317, 319 (n = 5)	61, 65, 66, 69, 81, 85, 88, 122, 124, 154, 214, 217, 246, 273, 275, 278, 279, 282, 318, 320, ligand (n = 21)
Farnesyltransferase	2H6G (mutant)	602, 605, 606, 651, 654, 655, 706, 803, 865 (n = 9)	596, 599, 603, 609, 649, 650, 652, 658, 662, 693, 702, 703, 705, 709, 710, 748, 753, 799, 800, 802, 860, 861, 862, 864, 868, 902, 903, ligand (n = 28)
	2H6F (wild-type)	602, 605, 606, 654, 655, 705, 706, 803, 865 (n = 9)	596, 599, 603, 609, 650, 651, 658, 662, 693, 702, 703, 709, 748, 753, 754, 757, 761, 799, 800, 802, 860, 861, 862, 864, 868, 902, 903, ligand (n = 28)

Table S2.4, continued: Enzyme specificity benchmark structures and positions. Proteins (wild-type and mutant) and ligand names and PDB codes are shown, along with designable and packable positions. Position numbering corresponds to PDB numbering.

Protein	PDB	Designable positions	Packable positions
Flavocytochrome b(2)	1SZE (mutant)	143, 198, 230, 254, 286, 325, 326 (n = 7)	139, 144, 199, 202, 228, 229, 252, 256, 280, 283, 289, 292, 296, 323, 324, 373, 377, ligand (n = 18)
	1FCB (wild-type)	143, 198, 230, 254, 326 (n = 5)	139, 144, 199, 202, 228, 229, 252, 280, 283, 289, 292, 296, 323, 325, 373, 376, ligand (n = 17)
Histidine ammonia-lyase	2O78 (mutant)	89, 90, 405, 406 (n = 4)	66, 68, 69, 86, 87, 153, 154, 157, 202, 432, 381, 391, 392, 402, 408, 409, 503, ligand (n = 18)
	2O7B (wild-type)	89, 90, 405, 406 (n = 4)	66, 68, 69, 86, 87, 153, 154, 157, 202, 432, 391, 392, 402, 408, 409, 503, ligand (n = 17)
N-acetylornithine carbamoyltransferase	3L05 (mutant)	180, 296, 298, 77, 92 (n = 5)	48, 50, 51, 181, 182, 184, 252, 253, 270, 293, 301, 302, 78, 93, 98, ligand (n = 16)
	3L06 (mutant)	180, 184, 298, 302, 77, 92 (n = 6)	48, 50, 51, 112, 178, 181, 182, 251, 252, 253, 270, 291, 293, 296, 297, 301, 303, 308, 78, 93, ligand (n = 21)
	3L04 (mutant)	180, 184, 298, 77, 92 (n = 5)	48, 50, 51, 181, 182, 252, 253, 270, 296, 301, 302, 78, 93, ligand (n = 14)
	3L02 (mutant)	180, 298, 77, 92 (n = 4)	48, 50, 51, 181, 182, 184, 252, 253, 270, 293, 296, 301, 302, 78, 93, 98, ligand (n = 17)
	3KZO (wild-type)	180, 184, 298, 77, 92 (n = 5)	48, 50, 51, 181, 182, 252, 253, 270, 293, 296, 302, 78, 93, 98, ligand (n = 15)
Proline dehydrogenase	2FZN (wild-type)	513, 540 (n = 2)	259, 283, 285, 327, 370, 431, 485, 487, 511, 516, 538, 542, 552, 556, 559, 560, ligand (n = 17)
	3E2Q (mutant)	285, 513, 540 (n = 3)	259, 283, 287, 327, 329, 370, 431, 485, 487, 511, 516, 542, 552, 556, 559, 560, ligand (n = 17)
Purine nucleoside phosphorylase	1OUM (mutant)	64, 180, 181 (n = 3)	62, 69, 73, 87, 159, 179, 185, 198, ligand (n = 9)
	1PK7 (wild-type)	64, 159, 180 (n = 3)	62, 156, 160, 181, ligand (n = 5)

Table S2.5: Digoxigenin benchmark structures and positions.
PDB codes are shown for the source of the protein and ligand structure. Position and starting side chain identity are shown for designable and packable positions. Table lists positions packed by CoupledMoves methods; other methods pack all positions. Positions are numbered as in [3].

Protein	PDB	Designable positions	Packable positions (coupled methods)
Designed digoxigenin binder DIG10.1	1Z1S (protein), 4J8T (ligand)	A10, L11, L14, W22, C23, F26, L32, Y34, A37, P38, G40, H41, F45, H54, M55, F58, Y61, M62, I64, F66, F84, G86, G88, H90, V92, S93, G95, L97, A99, Y101, S103, L105, I112, Y115, L117, F119, V124, P127, L128 (n = 39)	I6, L7, V8, H9, R12, L13, E15, A19, R20, L25, P39, K42, T43, R48, E49, T50, I51, W52, L57, P59, E60, V69, F71, A80, T91, T107, P121, R123, L125, I6, L7, V8, H9, R12, L13, E15, A19, R20, L25, P39, K42, T43, R48, E49, T50, I51, W52, L57, P59, E60, V69, F71, A80, T91, T107, P121, R123, L125, DIG (n = 30)

Table S2.6: Allowed design for digoxigenin dataset.
Shown are amino acids (one letter codes) to which positions were allowed to design. Amino acids were included only if they had high enough sequencing counts to be included in the enrichment and depletion calculations in [3].

Position	Allowed amino acids
10	ACDEFGILMNPRSTVY
11	ACDFGHILMNPQRSTVY
14	AFHIKLMPQRSTVW
22	ACFGLMPQRSTVWY
23	ACDFGHILNPRSTVWY
26	CFILMSTVWY
32	FHILMPQRSTV
34	ACDEFHIKLNPQRSTVY
37	AEGIKLPQRSTV
38	AEGHKLMPQRSTVW
40	ACDEFGHILNPRSTVWY
41	ACDEFGHKLNPQRSTVY
45	ACDFGHILRSTVWY
54	ACDEFGHIKLNPQRSTVY
55	AEFGIKLMNRSTVW
58	ACDFGHILMNPRSTVWY
61	ACDEFGHIKLNPQRSVWY
62	AFGIKLMNPRSTVW
64	ADFGIKLMNPRSTVY
66	ACFILMNPRSTVY
84	ACDFGHILMNPRSTVWY
86	ACDEFGHILNPRSTVWY
88	ACDEFGHILNPRSTVWY
90	ACDEFGHIKLNPQRSTVY
92	ADEFGIKLMPQRSTVW
93	ACDFGHIKLMNPRSTVWY
95	ACDEFGHILNPRSTVWY
97	AEFGHIKLMPQRSTVWY
99	ACDEFGHILNPRSTVY
101	ACDEFGHIKLNPQRSTVWY
103	ACDFHILNPRSTVWY
105	AFGHIKLMPQRSTVW
112	ACFHIKLMNPRSTV
115	ACDEFHIKLNQRSTVWY
117	ACDFGHILMNPQRSTVY
119	ACDFGHILMNPRSTVWY
124	ACDEFGHILMNPRSTVWY
127	AGHIKLPQRSTV
128	AEFGHIKLMPQRSTVW

Table S2.7: Fentanyl specificity benchmark structures and positions.
PDB codes are shown for the source of the protein and ligand structure. Position and starting side chain identity are shown for designable positions, and position is shown for packable positions. Position numbering corresponds to PDB numbering as in [2].

Protein	PDB	Designable positions	Packable positions (coupled methods)
Designed fentanyl binder Fen49	2QZ3 (protein), 5TZO (ligand)	Q7, W9, N35, V37, N63, Y65, T67, Y69, W71, E78, Y80, P90, R112, P116, W129, Y166, A170, A172 (n = 18)	Y5, D11, T43, R73, L76, V82, W85, Y88, Y108, T110, A115, S117, I118, D121, F125, Q127, V131, A165, V168, Y174, Q175, FEN (n = 22)

Table S2.8: Allowed design for fentanyl dataset.
Amino acids were allowed in design only if they had high enough sequencing counts to be included in the enrichment and depletion calculations in [2]. Shown are the amino acid side chains (one letter codes) to which positions were allowed to design. Because Fen49 wild-type identities are disallowed during design (see Methods), positions marked with (*) were mutated to alanine with the FixBB application during preparation of the input structure for design.

Position	Allowed side chains
9	ACDEFGHIKLMNPRSTVWY
35	ACDEFGIKLMNPQRTVWY
37*	ACDEFGHILMNQRSTWY
65*	ACEGLMRSTV
67	ACDEFGIKLMNPQRSTVWY
69*	ACDFGHIKLNRSTVW
71*	ACDEFGHIKLMNPRSTV
78	ACDEFGHKLMPQRSTVWY
90	ACDEFGHILMNPQRSTVWY
112	ACDFGHILMPRSTVWY
116	AEGKLMPQRSTVWY
129	ACEFGIKLMPQRSTVW
170	ACEGLPQRSTV

Table S2.9: hGH/hGHr specificity benchmark structures and positions.
For each library, position and starting side chain identity are shown for designable positions, and position is shown for packable positions. Position numbering corresponds to PDB numbering.

Library	Designable positions	Packable positions
A	M14, Y28, N47, P61, D171, I179 (n = 6).	17, 21, 32, 41, 48, 49, 50, 60, 66, 67, 68, 70, 75, 78, 160, 163, 164, 167, 174, 175, 176, 177, 178, 181, 183, 202, 254, 276, 315, 365, (n = 30)
B	H18, Y42, S62, E65, Y164, T175 (n = 6).	22, 28, 38, 41, 44, 45, 46, 51, 53, 63, 66, 69, 160, 165, 167, 168, 171, 174, 176, 179, 202, 248, 252, 254, 255, 270, 271, 272, 277, 315, 363, 364, (n = 32)
C	H21, N29, L45, T60, T67, R178 (n = 6).	14, 24, 25, 33, 41, 42, 44, 51, 58, 61, 66, 68, 75, 78, 82, 164, 167, 170, 171, 172, 174, 176, 179, 181, 182, 189, 226, 256, 272, 315, 317, 364, 365, (n = 33)
D	Q22, S43, E66, R167, F176, R183 (n = 6).	18, 19, 21, 23, 24, 25, 26, 28, 40, 60, 61, 62, 63, 67, 72, 75, 78, 79, 82, 164, 172, 175, 179, 184, 254, 276, 277, 364, (n = 28)
E	D26, F44, P48, R64, K168, E174 (n = 6).	14, 17, 18, 21, 22, 25, 45, 47, 49, 50, 51, 52, 53, 56, 68, 157, 160, 164, 169, 172, 203, 221, 225, 226, 254, 256, 310, 313, 315, 363, 364, (n = 31)
F	F25, K41, Q46, N63, K172 (n = 5).	21, 26, 28, 29, 32, 36, 38, 42, 45, 56, 60, 62, 65, 66, 82, 160, 164, 167, 168, 169, 176, 226, 252, 254, 258, 270, 272, 277, 364, (n = 29)

.

Table S2.10: Herceptin/HER2 specificity benchmark structures and positions.
Design and packable positions are shown for each library. Design positions are listed in Kabat numbering [73]. For packable positions, numbering corresponds to consecutive renumbering of the 312 positions in combined chain A positions 1-106, chain B positions 1-119, and chain C positions 511-607. Herceptin Library D is omitted because the experimental data were dominated by the wild-type sequence.

Library	Designable positions	Packable positions
A	V_L94, V_H33, V_H50, V_H56, V_H58, V_H95 (n = 6).	93, 95, 138, 140, 141, 153, 155, 157, 158, 161, 164, 166, 176, 204, 211, 213, 272, 273, 275, 276, 287, 288 (n = 22).
B	V_L30, V_L91, V_L92, V_H50, V_H95, V_H99, V_H100a (n = 7).	28, 29, 31, 32, 66, 71, 90, 93, 94, 138, 139, 141, 153, 155, 157, 164, 165, 176, 204, 208, 210, 212, 213, 273, 275, 284, 285, 286, 287, 288, 296, 298, 301, 303, 305, 307 (n = 36).
C	V_L49, V_L53, V_L91, V_H98, V_H99, V_H100, V_H100a (n = 7).	32, 46, 48, 50, 52, 54, 90, 92, 205, 212, 214, 285, 286, 287, 288, 296, 298, 308 (n = 18).
E	V_L49, V_L53, V_L55, V_H100, V_H102 (n = 5).	46, 48, 50, 52, 54, 56, 58, 108, 110, 203, 204, 205, 211, 212, 214, 216, 308 (n = 17).

Table S2.11: Number of designed sequence positions in PPS gain/loss/neutral zones. Values for CoupledMoves represent averages ± standard deviation for CM-BR, CM-FKIC, and CM-WKIC. The charged or polar category includes arginine, histidine, lysine, aspartate, glutamate, serine, threonine, asparagine, glutamine, tyrosine, and cysteine. The hydrophobic category includes alanine, phenylalanine, glycine, isoleucine, leucine, methionine, valine, tryptophan, and proline.

		Amino acid category		
		all	charged or polar	hydrophobic
Coupled Moves	gain	43±3	16±1	28±2
	loss	32±4	25±3	6±1
	neutral	81±6	26±3	55±3
Backrub Ensemble	gain	37	13	24
	loss	41	28	13
	neutral	78	26	52
Fast Design	gain	28	10	18
	loss	46	33	13
	neutral	82	24	58
Fixed Backbone	gain	13	4	9
	loss	50	34	16
	neutral	93	29	64

2.9 Supplemental Rosetta command lines and XML scripts

2.9.1 CM-BR (with ligand)

```
Rosetta/main/source/bin/coupled_moves.default.linuxgccrelease -s pdb -
mute protocols.backrub.BackrubMover -ex1 -ex2 -extrachi_cutoff 0 -
nstruct 1 -ignore_unrecognized_res -score::weights ref2015 -
extra_res_fa ligand_name.params -resfile resfile -coupled_moves::mc_kt
2.4 -coupled_moves::boltzmann_kt 2.4 -coupled_moves::ntrials 1000 -
coupled_moves::initial_repack false -coupled_moves::ligand_mode true -
coupled_moves::ligand_weight 2 -coupled_moves::fix_backbone false -
coupled_moves::bias_sampling true -coupled_moves::bump_check true -
coupled_moves::backbone_mover backrub -
coupled_moves::exclude_nonclashing_positions true -nstruct 400
```

2.9.2 CM-BR (without ligand)

```
Rosetta/main/source/bin/coupled_moves.default.linuxgccrelease -s pdb -
mute protocols.backrub.BackrubMover -ex1 -ex2 -extrachi_cutoff 0 -
nstruct 1 -ignore_unrecognized_res -score::weights ref2015 -resfile
resfile -coupled_moves::mc_kt 2.4 -coupled_moves::boltzmann_kt 2.4 -
coupled_moves::ntrials 1000 -coupled_moves::initial_repack false -
coupled_moves::fix_backbone false -coupled_moves::bias_sampling true -
coupled_moves::bump_check true -coupled_moves::backbone_mover backrub
-coupled_moves::exclude_nonclashing_positions true -nstruct 400
```

2.9.3 CM-FKIC (with ligand)

```
Rosetta/main/source/bin/coupled_moves.default.linuxgccrelease -s
name.pdb -mute protocols.backrub.BackrubMover -ex1 -ex2 -
extrachi_cutoff 0 -nstruct 1 -ignore_unrecognized_res -score::weights
ref2015 -extra_res_fa name.params -resfile name.resfile -
coupled_moves::mc_kt 2.4 -coupled_moves::boltzmann_kt 2.4 -
coupled_moves::ntrials 1000 -coupled_moves::initial_repack false -
coupled_moves::ligand_mode true -coupled_moves::ligand_weight 2 -
coupled_moves::fix_backbone false -coupled_moves::bias_sampling true -
coupled_moves::bump_check true -
coupled_moves::exclude_nonclashing_positions true -
coupled_moves::backbone_mover kic -coupled_moves::kic_perturber
fragment -loops:frag_sizes 9 3 -loops:frag_files name.200.9mers.gz
name.200.3mers.gz -nstruct 400
```

2.9.4 CM-FKIC (without ligand)

```
Rosetta/main/source/bin/coupled_moves.default.linuxgccrelease -s
name.pdb -mute protocols.backrub.BackrubMover -ex1 -ex2 -
```

```
extrachi_cutoff 0 -nstruct 1 -ignore_unrecognized_res -score::weights
ref2015 -resfile name.resfile -coupled_moves::mc_kt 2.4 -
coupled_moves::boltzmann_kt 2.4 -coupled_moves::ntrials 1000 -
coupled_moves::initial_repack false -coupled_moves::ligand_mode false
-coupled_moves::fix_backbone false -coupled_moves::bias_sampling true
-coupled_moves::bump_check true -
coupled_moves::exclude_nonclashing_positions true -
coupled_moves::backbone_mover kic -coupled_moves::kic_perturber
fragment -loops:frag_sizes 9 3 -loops:frag_files name.200.9mers.gz
name.200.3mers.gz -nstruct 400
```

2.9.5 CM-WKIC (with ligand)

```
Rosetta/main/source/bin/coupled_moves.default.linuxgccrelease -s
name.pdb -mute protocols.backrub.BackrubMover -ex1 -ex2 -
extrachi_cutoff 0 -nstruct 1 -ignore_unrecognized_res -score::weights
ref2015 -extra_res_fa name.params -resfile name.resfile -
coupled_moves::mc_kt 2.4 -coupled_moves::boltzmann_kt 2.4 -
coupled_moves::ntrials 1000 -coupled_moves::initial_repack false -
coupled_moves::ligand_mode true -coupled_moves::ligand_weight 2 -
coupled_moves::fix_backbone false -coupled_moves::bias_sampling true -
coupled_moves::bump_check true -
coupled_moves::exclude_nonclashing_positions true -
coupled_moves::backbone_mover kic -coupled_moves::kic_perturber
walking -nstruct 400
```

2.9.6 CM-WKIC (without ligand)

```
Rosetta/main/source/bin/coupled_moves.default.linuxgccrelease -s
name.pdb -mute protocols.backrub.BackrubMover -ex1 -ex2 -
extrachi_cutoff 0 -nstruct 1 -ignore_unrecognized_res -score::weights
ref2015 -extra_res_fa name.params -resfile name.resfile -
coupled_moves::mc_kt 2.4 -coupled_moves::boltzmann_kt 2.4 -
coupled_moves::ntrials 1000 -coupled_moves::initial_repack false -
coupled_moves::fix_backbone false -coupled_moves::bias_sampling true -
coupled_moves::bump_check true -
coupled_moves::exclude_nonclashing_positions true -
coupled_moves::backbone_mover kic -coupled_moves::kic_perturber
walking -nstruct 400
```

2.9.7 FastDesign (with ligand)

```
Rosetta/main/source/bin/relax.default.linuxgccrelease -s name.pdb -
resfile name.resfile -extra_res_fa ligand_name.params -ex1 -ex2 -
extrachi_cutoff 0 -nstruct 400 -in:file:fullatom -relax:fast -
```

```
relax:respect_resfile -relax:constrain_relax_to_start_coords -
relax:coord_cst_stdev .5
```

2.9.8 FastDesign (without ligand)

```
Rosetta/main/source/bin/relax.default.linuxgccrelease -s name.pdb -
resfile name.resfile -ex1 -ex2 -extrachi_cutoff 0 -nstruct 400 -
in:file:fullatom -relax:fast -relax:respect_resfile -
relax:constrain_relax_to_start_coords -relax:coord_cst_stdev .5
```

2.9.9 BackrubEnsemble step 1: Backrub ensemble generation (with ligand)

```
Rosetta/main/source/bin/backrub.default.linuxgccrelease -
score::weights ref2015 -s name.pdb -nstruct 400 -
ignore_unrecognized_res -extra_res_fa ligand_name.params -
backrub:ntrials 10000 -mc_kt 1.2 -max_atoms 12
```

2.9.10 BackrubEnsemble step 1: Backrub ensemble generation (without ligand)

```
Rosetta/main/source/bin/backrub.default.linuxgccrelease -
score::weights ref2015 -s name.pdb -nstruct 400 -
ignore_unrecognized_res -backrub:ntrials 10000 -mc_kt 1.2 -max_atoms
12
```

2.9.11 BackrubEnsemble step 2: Design on backrub ensemble (with ligand)

```
Rosetta/main/source/bin/rosetta_scripts.default.linuxgccrelease -
parser:protocol FBBRS.xml -parser:script_vars res_file=name.resfile -s
name_ensemble_member.pdb -nstruct 400 -ignore_unrecognized_res -
extra_res_fa ligand_name.params
```

2.9.12 BackrubEnsemble step 2: Design on backrub ensemble (without ligand)

```
Rosetta/main/source/bin/rosetta_scripts.default.linuxgccrelease -
parser:protocol FBBRS.xml -parser:script_vars res_file=name.resfile -s
name_ensemble_member.pdb -nstruct 400 -ignore_unrecognized_res
```

2.9.13 BackrubEnsemble step 2: Design on backrub ensemble, file name FBBRS.xml

```
<ROSETTASCRIPTS>
    <SCOREFXNS>
    </SCOREFXNS>
    <RESIDUE_SELECTORS>
    </RESIDUE_SELECTORS>
    <TASKOPERATIONS>
     <ReadResfile name="resfile" filename="%%res_file%%" />
      <ExtraRotamers name="ex1" chi="1" />
      <ExtraRotamers name="ex2" chi="2" />
      <ExtraChiCutoff name="exchi0" extrachi_cutoff="0" />
    </TASKOPERATIONS>
    <FILTERS>
    </FILTERS>
    <MOVERS>
      <PackRotamersMover name="pack_rot" 
task_operations="resfile,ex1,ex2,exchi0" />
    </MOVERS>
    <APPLY_TO_POSE>
    </APPLY_TO_POSE>
    <PROTOCOLS>
      <Add mover="pack_rot" />
    </PROTOCOLS>
    <OUTPUT/>
</ROSETTASCRIPTS>
```

2.9.14 FixBB control (with ligand)

```
Rosetta/main/source/bin/rosetta_scripts.default.linuxgccrelease -
parser:protocol FBBRS.xml -parser:script_vars res_file=name.resfile -s 
name.pdb -nstruct 400 -ignore_unrecognized_res -extra_res_fa 
ligand_name.params
```

2.9.15 FixBB control (without ligand)

```
Rosetta/main/source/bin/rosetta_scripts.default.linuxgccrelease -
parser:protocol FBBRS.xml -parser:script_vars res_file=name.resfile -s 
name.pdb -nstruct 400 -ignore_unrecognized_res -extra_res_fa 
ligand_name.params
```

Chapter 3: Experimental methods for screening designed ibuprofen sensors

3.1 Introduction

Detection and response to signals is fundamental to living systems, and requires systems capable of both sensing a signal and actuating a response. Engineered sensor/actuator tools could play crucial roles if design can reliably generate sensors for molecules that cannot be detected by existing proteins. Applications for sensor/actuators include therapeutics, biological manufacturing, probing biology, and engineering communication. For example, we could use small molecule signals to control multiple interactions within a single cell for the purposes of studying biology, engineer new channels of cooperation and communication between cells, or regulate gene expression in response to the presence of metabolic intermediates, such as the farnesyl pyrophosphate (FPP) intermediate in the bisabolene production pathway.[1, 74] I present an experimental method for testing modular systems with sensor and actuator domains, wherein the identities of the input (target small molecule) and the output (protein complementation) can be tailored to various synthetic biology applications.[1] I also describe a system that senses the small molecule ibuprofen, and reports its presence via dimerization of a split reporter.

The ibuprofen sensor reported here was designed using a computational protocol which grafts a known binding from a naturally occurring, monomeric protein, to the interface of a heterodimer that did not previously contain a binding site, such that the ligand binding induces dimerization which is reported by complementation of a split reporter (Figure 3.1a), as in [1]. In natural biology, signals often take the form of small molecules, which are sensed by binding to a pocket in a protein, and signal transduction often proceeds via allostery or homodimerization.

Few known examples of ligand-induced heterodimerization have been characterized, and the rapamycin[75] and abscisic acid[76] systems are most well known. By combining ligand-induced heterodimerization with fusion to a split reporter, we arrive at a combination of modular input and modular output that is powerfully adaptable to various applications. In some cases, we may desire to detect a target small molecule of interest, and any convenient actuator may be used to report the presence of the small molecule (Figure 3.1b). Alternatively, if the desired application is activation of a particular actuator, any convenient small molecule can be used to activate dimerization of the sensor and actuator (Figure 3.1c). These simple conceptual examples can be built upon for more complex synthetic biology applications.

3.2 Results

Sensor domains were engineered using the macromolecular modeling and design software Rosetta[10] by a process (Figure 3.2, see Methods) which can be generalized to design sensors for various small molecules.[1] Briefly, the geometry of amino acid side chains coordinating the ligand (motif residues) were defined from an existing binding site in monomeric protein COX-1 (PDB: 1EQG). In step 2, motif residues that make key contacts with ibuprofen were matched using a protocol (adapted from [77]) to heterodimer Ultraspiracle/Ecdysone Receptor (PDB: 2NXX), referred to as a scaffold protein, which has a backbone conformation capable of placing the motif residues in the correct orientations to bind ibuprofen. In step 3, motif residues and ibuprofen were placed into the scaffold, and the surrounding region was designed to accommodate and stabilize the transplanted binding site. In step 4, designs proteins were linked to protein complementation systems for testing in *E. coli* and *in vitro*.

Small molecules of interest were selected for their possible utility and include intermediates in metabolic pathways, therapeutic agents, and toxins (Figure 3.3). Experimental Safety and ease of use were also considered, and ligands were chosen to be soluble in aqueous solution, and do not include chemicals classed as particularly hazardous or carcinogenic. The set also included a number of generally nontoxic molecules, which could be used in many applications with low risk of cytotoxic effects. Of particular interest in this category is *p*-coumaric acid, which switches between isomers in response to light. Additionally, ibuprofen (IBP) falls into this category and is of interested for therapeutic applications because it is cheap, FDA approved, and humans are regularly and safely prescribed doses as high as 2,400 mg/day.[78] In the realm of metabolic engineering, FPP is of great interest because it is the precursor to several commercially important compounds. A sensor/actuator for the toxic insecticide thiacloprid could be used for bioremediation. Cells use homoserine lactone and serotonin to communicate with each other, and sensor/actuators for these could be used to add additional channels of communication between engineered cells. The full complement of designs described in Figure 3.3 and Appendix 3.3 were previously generated by computational design and chosen for the experimental screens described here. Each design contains 8 to 22 mutated positions, and motifs for some target ligands were matched to more than one scaffold heterodimer (Figure 3.3).

We focus on experimental characterization of ibuprofen sensors. We individually screened 16 designs targeting ibuprofen, each containing 14-20 mutations from the wild-type heterodimer, Ultraspiracle and Ecdysone Receptor (PDB: 2NXX) (Figure 3.4). For two designs, #490 and 492, we observed increased signal from the split reporter proteins in the presence of ibuprofen compared to a control with no ibuprofen, as described below. Sequence changes between the wild type protein, a top-ranking Rosetta design, and the two designs 490 and 492 containing additional mutations from visual inspection, are shown in Figure 3.5.

In the first experimental screen, we used split murine dihydrofolate reductase[79] (DHFR) linked to biosensors, and expressed the fusion proteins in *E. coli* in the presence of bacterial DHFR inhibitor trimethoprim. If the biosensor functions according to the design concept, ligand-dependent dimerization of the sensor module will cause complementation of the murine DHFR actuator module, rescuing *E coli* cells from the toxicity caused by trimethoprim inhibition of bacterial DHFR. Thus, signal in the form of bacterial culture density is dependent on reconstituted murine DHFR enzymes. The second system uses reporter NanoLuc, a highly engineering split luciferase,[80] expressed in the "TXTL" cell-free transcription-translation (TXTL) protein expression system.[81] NanoLuc is an engineered heterodimer derived from a monomeric deep sea shrimp luciferase, and composed of one 18kDa domain (LgBIT) and one 1.3kDa peptide (SmBIT) that fits into a groove in the larger domain. Peptides with a variety of affinities for the larger domain are available; our constructs used SmBIT peptide 114, which has an affinity of 190µM for LgBIT.[80]

Designs were screened using systems that allowed quantification of reporter signal without purification, which can be challenging for potentially unstable computationally designed proteins. With *in vivo* experiments, ligands are added to growth media, while the sensor/actuator is expressed in the bacterial cytoplasm. This does not allow direct control over the intracellular concentration of ibuprofen, which cannot be easily quantified. In TXTL, protein is expressed in *E. coli* extract, so that ligand can be titrated directly into the extract expressing the sensor/actuator. *In vivo* experiments require relatively large volumes of media, which in turn requires large amounts of ligand. Due to the need for oxygenation via shaking, reducing ligand requirement by reducing culture size slows bacterial growth and reduces the throughput of the experiments. DHFR experiments can also be carried out by printing colonies on agar plates instead of growing culture in liquid media, which is also low-throughput. Plate printing, even using robotic automation, is laborious, and growth on agar media takes several days. Protein expression in TXTL is efficient, taking only a few hours, and occurring in volumes as small as 10µL which uses much less ligand.

Ibuprofen sensors were first screened in *E. coli* with the split DHFR reporter. Initial screens were carried out by printing colonies of each sensor on agar growth media containing either ligand, added from an ethanol stock, or an equivalent volume of ethanol without ligand. For ibuprofen sensor designs #490 and 492, colonies grew larger on media with 1mM IBP (3. 6a), which is consistent with ligand-induced dimerization. Colonies for additional designs are shown in Figures 6b-c. Design #490 and the wild type scaffold as a control were grown in liquid media with varying concentrations of ligand (Figure 3.7). After 21 hours, cells expressing design #490 and grown in media containing 1mM IBP exhibited an increase in growth measured by OD_{600} over cells grown in the same condition except without IBP (0.29 ± 0.05 compared to 0.08 ± 0.01 OD_{600}). Cells expressing a control, the wild type scaffold protein fused to DHFR, grew to 0.36 ± 0.04 OD_{600} after 19 hours, regardless of ligand concentration (Figure 3.7a), consistent with ligand-mediated dimerization at the designed binding site.

Each design described in Figure 3.3 was screened using the DHFR plate-printing experiment, and designs for ligands except ergosterol were screened using the DHFR liquid experiment. For these designs we did not observe reproducible signal. In many cases, cells did not grow once we induced expression of the design-DHFR fusion constructs. No colonies were observed in plate printing experiments for serotonin designs #131, 145, 475, 496, 497, and 498, and ibuprofen sensor designs #468, 490, 494, and 501. Cells grew robustly when expressing designs targeting farnesyl pyrophosphate and *p*-coumaric acid, but did not exhibit ligand-dependent growth differential. Cells expressing constructs for the remaining target ligands grew slowly and also did not exhibit ligand-dependent growth differential.

To demonstrate the modularity of the sensor/actuator system, we tested ibuprofen sensors #490 and #492 with a different actuator; we also sought to test whether sensor signal was dependent on the presence of motif residues by testing constructs with each motif residue mutated to alanine. We employed NanoLuc split luciferase to report on sensor/actuator dimerization, and expressed the fusion constructs using TXTL cell-free protein expression according to the schema shown in Figure 3.8 and described in Methods. Briefly, the two halves of the heterodimer construct were expressed separately, then were combined with the target ligand ibuprofen and furimazine, the luciferase substrate, for measurement. Results are shown in Figure 3.9. Constructs with the complete motif (red) luminesce in a ligand-dependent manner, while constructs with alanine in place of one of the two motif residues on the Ultraspiracle chain (orange and yellow) have a reduced response, and constructs with alanine in place of the single motif residue on the Ecdysone Receptor chain do not respond to ibuprofen (green). These data are consistent with ligand-mediated dimerization at the designed binding site, and with ibuprofen coordination by the transplanted motif residues, though structural characterization would be required to confirm.

Of the remaining designs described in Figure 3.3, the following were also tested using the NanoLuc TXTL protocol: all designs targeting ibuprofen, *p*-coumaric acid (except #505), serotonin (based on scaffold PDBs 3IA3 and 3NW0) (except #s 482, 518, 519, and 522), theophylline (except #166), and thiacloprid (except #652). For these designs, we were unable to confirm signal, except for the ibuprofen biosensors for which some designs demonstrated activity but with a smaller dynamic range than designs #490 and #492. Designs targeting caffeine, naproxen, and serotonin (based on scaffold PDB 3EAB), were not tested because constructs did not assemble during the cloning step. Designs targeting ergosterol were not tested due to the insolubility of the ligand, and designs targeting homoserine lactone were not tested because we did not previously observe growth for *E coli* expressing the designs in plate-printing DHFR screens. Designs targeting farnesyl pyrophosphate were screened in TXTL by Dr. Anum Azam-Glasgow, and those results are reported in [1].

3.3 Discussion

Experiments in *E coli* and in TXTL cell extract supported the possibility of ligand-induced dimerization for two computationally designed ibuprofen sensors, #490 and 492. Furthermore, we demonstrated the modularity of our sensor/actuator design concept with ligand-dependent actuation of two different protein complementation reporters, DHFR and NanoLuc.

De novo binding site design remains a challenge, and indeed, as discussed in the Results section of this chapter, we did not observe ligand-dependent signal when we tested designed sensors for several additional ligands. In addition to challenges surrounding structural design, ligand-related experimental factors may have contributed to lack of observed signal. Ligands can have a positive or negative effect on the chosen actuator systems, for example by influencing *E coli* cell growth. Target ligand caffeine exhibits cytotoxic effects, decreasing culture density, with effect increasing with caffeine concentration (Figure 3.10).

Low signal to noise ratio (S/N) was a confounding factor in our experiments. Signal depends on difference in affinity between the ternary protein/ligand/protein complex and the protein/protein heterodimer. When that difference is small relative to background noise, signal detection is difficult. It remains difficult to accurately predict mutations that increase ligand-mediated affinity without increasing heterodimer protein/protein affinity, or conversely to predict mutations that decrease protein/protein affinity without decreasing affinity of the tertiary protein/ligand/protein complex. An experimental technique such as directed evolution with positive and negative selection might allow discovery of mutations that improve dynamic range, and this information could be incorporated into the design pipeline to improve computational design methods.

3.4 Methods

3.4.1 Computational design of ibuprofen sensors

Sensor domains were engineered using the macromolecular modeling and design software Rosetta[10] by a process (Figure 3.2) which can be generalized to design sensors for various small molecules.[1] In step 1, the geometry of amino acid side chains coordinating the ligand (motif residues) were defined from an existing binding site for the target small molecule, typically found in a monomeric protein such as an enzyme. Three or four motif residues that make key contacts with the ligand were selected from a high-resolution crystal structure of the existing binding site by manual inspection. In step 2, a Rosetta matching protocol (adapted from [77]) was used to search heterodimer proteins for backbone conformations compatible with placing motif residues in the correct orientations to bind the ligand. Proteins on which side chains are designed are referred to as scaffolds. Motif residues were matched to both chains of the heterodimer scaffold, such that the binding site spanned the interface. If a match was found, design proceeded to step 3, wherein the motif residues and ligand were placed in the scaffold heterodimer. Then, flexible backbone and sequence design of the surrounding shell accommodated and stabilized the transplanted binding site. Designs were filtered by metrics such as pre-organization of the ligand binding site, ligand solvent-exposed surface area, and hydrogen-bond satisfaction. In step 4, designs proteins were linked to protein complementation systems for testing in *E. coli* and *in vitro*.

This method was used to generate 16 designs targeting ibuprofen, each containing 14-20 mutations from the wild-type heterodimer, Ultraspiracle and Ecdysone Receptor (PDB: 2NXX) (Figure 3.4). Sequence changes between the wild type protein, a top-ranking Rosetta design, and two designs (#490 and #492) containing additional mutations or reversions from visual inspection, are shown in Figure 3.5. The binding site motif was extracted from a crystal structure of COX-1 complexed with ibuprofen (PDB: 1EQG). In COX-1, the motif is composed of two hydrophobic residues, V317 and L327, which pack against the nonpolar portion of ibuprofen, and a third polar motif residue, R88, which coordinates the ibuprofen's carboxylic acid functional group. During design, the two hydrophobic motif residues were substituted for methionine. Two motif residues (E336R and Y343M) were grafted onto Ultraspiracle, and the third motif residue (Y322M) was grafted onto Ecdysone Receptor.

3.4.2 DHFR screen on agar plates

The plates shown in Figures 3.6 and 3.11 were prepared as follows. M9 medium was prepared with 1.5% w/v agar, 50 µg/mL spectinomycin, 2µg/mL trimethoprim and 100µM IPTG. For experiments involving ibuprofen (IBP) sensors, plates contained either 1mM IBP or an equivalent volume of ethanol for the blank; when prepared, media contained 3.3% v/v ethanol. For experiments involving ergosterol (ERG) sensors, plates contained either 1mM ERG or an equivalent volume of ethanol for the blank; when prepared media contained 1% v/v ethanol. Ethanol evaporates readily at standard temperature and pressure and plates were prepared at least one day before colony printing. 45mL growth medium was poured into rectangular plates (Rotor PlusPlates, catalog number PLU-003). Individual colonies from plasmid transformation were picked and suspended into 500µL liquid M9 medium with 50 µg/mL spectinomycin in 96-well deep well blocks, covered with a gas-permeable membrane, and grown overnight at 37°C with shaking at 220RPM in a New Brunswick Innova 44 shaker. These cultures were then printed onto the previously-prepared agar plates using a Singer Instruments Rotor HDA plate-printing robot, and the plates were stored at room temperature in a dark cabinet during growth. Plates were removed from the cabinet and photographed at 24-hour time points for three days.

3.4.3 DHFR screen in liquid culture

For ibuprofen sensor design #490 and wild-type scaffold protein data shown in Figure 3.7a, samples were prepared as follows. Three separate colonies, corresponding to biological replicates 1-3 in Figure 3.7b, were picked and grown overnight at 37°C in 200µL M9 medium with 50 µg/mL spectinomycin. In the morning, 5mL M9 with 50 µg/mL spectinomycin was taken from 4°C storage and added to the overnight cultures. Cultures were grown at 37°C for 2 additional hours to approximate log phase (measured values were 0.62, 0.73, and 0.39 OD_{600} for the three cultures of design #490, and 0.48, 0.46, and 0.49 OD_{600} for the wild-type protein, respectively) then diluted to an OD_{600} of 0.10. During the 2 hours while cultures were growing to approximate log phase, M9 medium was prepared with 50 µg/mL spectinomycin, 60mM IPTG, and 0.5µg/mL trimethoprim. The medium was then divided into three volumes, to which a stock of 30mM ibuprofen in ethanol was added to concentrations of 2000, 400, and 0µM ibuprofen, respectively. For the latter two solutions, equivalent volumes of ethanol were added such that solutions contained the same concentration of ethanol. After the medium and cultures were prepared, 500µL of ibuprofen medium and 500µL of culture medium were mixed together in 96-well deep-well blocks according to the checkerboard schema shown in Figure 3.7b. Each well received 500µL of ibuprofen medium and 500µL of culture medium. After mixture, the final concentrations of ibuprofen were 1000, 200, and 0µM, depending on the well, while the cell culture concentration was 0.05 OD_{600} for all wells. All growth media contained a 3.3% v/v ethanol. The plates were then covered with a gas-permeable membrane and placed in a shaking incubator at 30°C for 21 hours. At 21 hours, 200µL volumes of culture were transferred to a transparent-bottom 96-well plate, and OD_{600} was measured in a plate reader.

3.4.4 Preparation of *E. coli* S30 extract for cell-free protein expression

Energy buffer and *E. coli* S30 extract from Rosetta2 cells were prepared using the "TXTL" protocol, originally described in [81] and with adaptations described in [1], and stored at -80°C. For protein expression reactions, TXTL extract and energy buffer were thawed on ice and prepared by adding to final concentrations 1mM IPTG, 0.2nM T7 RNA Polymerase plasmid (pID 108 in Appendix 3.2, acquired from Zachary Sun in Richard Murray's lab). TXTL extract prepared in our lab (Figure 3.11a) produced similar amounts of control protein GFP compared to extract acquired from the authors of [81] (Figure 3.11b).

3.4.5 NanoLuc screen in TXTL

For ibuprofen sensor design #490 and wild-type scaffold protein data shown in Figure 3.9, samples were prepared as follows (see Figure 3.8). TXTL extract was prepared in November

2017, as described above. Data were collected in an experiment carried out on 1/16/2018. TXTL reaction was prepared as described above, with additional 50µM Ponasterone A in DMSO (See Appendix 3.1 for preparation), a cofactor for Ecdysone receptor which forms one half of the scaffold used to design the ibuprofen sensor. This mixture was divided into separate reactions for the expression of each protein. To initiate protein expression, DNA was added to the TXTL solution. The amount of DNA used is described in Table 3.1. Reactions were placed in closed Eppendorf tubes and placed at 30°C for 7.5 hours, during which protein expression occurs. During this time, ethanol and/or 121.194 mM ibuprofen dissolved in ethanol (see Appendix 3.1 for preparation) were transferred to a 384-well plate using an Echo acoustic liquid handler, and the ethanol was evaporated off using a GeneVac evaporator on setting "High BP" for 10 minutes. Each well received the same amount of ethanol, and the amount of ibuprofen transferred was such that, when later combined with the TXTL-expressed protein, the final concentrations would be 0.0, 15, 50, 100.0, or 200.0µM. After the 7.5 hour TXTL incubation, reactions were removed to room temperature and prepared to final concentrations of 15% volume TXTL, 1X sterile phosphate buffered saline (PBS, final composition in reaction of 137 mM NaCl, 27 mM KCl, 10 mM Na_2HPO_4, 18 mM KH_2PO_4, and a pH of 7.4), and 1mg/mL bovine serum albumin (BSA). Next, the two halves of the heterodimer biosensor, which had been individually expressed in TXTL, were combined into the previously-prepared 384-well plate with ibuprofen. 10µL of extract expressing each heterodimer half, in the 15% TXTL solution, were transferred using the Echo acoustic liquid handler into the wells with the layout described in Table 3.2. Blank samples were prepared with 1X PBS and 1mg/mL BSA. Finally, NanoLuc substrate buffer was prepared according to manufacturer instructions (Nano-Glo Luciferase Assay System, Promega catalog #N1110). NanoLuc substrate buffer was added and luminescence measured using a SpectraMax

L luminometer. Each well was measured as follows. 20 µL buffer was injected into the well with M-injection setting, the plate was shaken at a speed of 30mm/s for 1 s, then luminescence was measured with integration time of 1s and PMT sensitivity set to “photon counting.”

3.5.1 Appendix 3.1: Recipes

3.5.1.1 Ibuprofen

Ibuprofen (IBP) was dissolved in ethanol. For experiments in *E. coli*, IBP was prepared to a concentration of 30 mM. For experiments in TXTL, IBP was prepared to a concentration of 121 mM in ethanol.

3.5.1.2 Ponasterone A

Ponasterone A (PonA) was dissolved in DMSO to a concentration of 10 mM in ethanol.

3.5.1.3 Caffeine

Caffeine (CFF) was dissolved in water to a concentration of 75 mM.

3.5.2 Appendix 3.2: Sequences

3.5.2.1 pID108: T7 RNA Polymerase

aataattttgtttaactttaagaaggaggatccaaatgaacacgattaacatcgctaagaacgacttctctgacatcgaactggctgctatcccg
ttcaacactctggctgaccattacggtgagcgtttagctcgcgaacagttggcccttgagcatgagtcttacgagatgggtgaagcacgcttc
cgcaagatgtttgagcgtcaacttaaagctggtgaggttgcggataacgctgccgccaagcctctcatcactaccctactccctaagatgatt
gcacgcatcaacgactggtttgaggaagtgaaagctaagcgcggcaagcgcccgacagccttccagttcctgcaagaaatcaagccgga
agccgtagcgtacatcaccattaagaccactctggcttgcctaaccagtgctgacaatacaaccgttcaggctgtagcaagcgcaatcggtc
gggccattgaggacgaggctcgcttcggtcgtatccgtgaccttgaagctaagcacttcaagaaaaacgttgaggaacaactcaacaagcg
cgtagggcacgtctacaagaaagcatttatgcaagttgtcgaggctgacatgctctctaagggtctactcggtggcgaggcgtggtcttcgtg
gcataaggaagactctattcatgtaggagtacgctgcatcgagatgctcattgagtcaaccggaatggttagcttacaccgccaaaatgctgg
cgtagtaggtcaagactctgagactatcgaactcgcacctgaatacgctgaggctatcgcaacccgtgcaggtgcgctggctggcatctctc
cgatgttccaaccttgcgtagttcctcctaagccgtggactggcattactggtggtggctattgggctaacggtcgtcgtcctctggcgctggt
gcgtactcacagtaagaaagcactgatgcgctacgaagacgtttacatgcctgaggtgtacaaagcgattaacattgcgcaaaacaccgca

Continued on next page.

3.5.2.1 pID108, continued: T7 RNA Polymerase

```
tggaaaatcaacaagaaagtcctagcggtcgccaacgtaatcaccaagtggaagcattgtccggtcgaggacatccctgcgattgagcgtg
aagaactcccgatgaaaccggaagacatcgacatgaatcctgaggctctcaccgcgtggaaacgtgctgccgctgctgtgtaccgcaagg
acaaggctcgcaagtctcgccgtatcagccttgagttcatgcttgagcaagccaataagtttgctaaccataaggccatctggttcccttacaa
catggactggcgcggtcgtgtttacgctgtgtcaatgttcaacccgcaaggtaacgatatgaccaaaggactgcttacgctggcgaaaggta
aaccaatcggtaaggaaggttactactggctgaaaatccacggtgcaaactgtgcgggtgtcgataaggttccgttccctgagcgcatcaag
ttcattgaggaaaaccacgagaacatcatggcttgcgctaagtctccactggagaacacttggtgggctgagcaagattctccgttctgcttc
cttgcgttctgctttgagtacgctggggtacagcaccacggcctgagctataactgctcccttccgctggcgtttgacgggtcttgctctggca
tccagcacttctccgcgatgctccgagatgaggtaggtggtcgcgcggttaacttgcttcctagtgaaaccgttcaggacatctacgggattg
ttgctaagaaagtcaacgagattctacaagcagacgcaatcaatgggaccgataacgaagtagttaccgtgaccgatgagaacactggtga
aatctctgagaaagtcaagctgggcactaaggcactggctggtcaatggctggcttacggtgttactcgcagtgtgactaagcgttcagtcat
gacgctggcttacgggtccaaagagttcggcttccgtcaacaagtgctggaagataccattcagccagctattgattccggcaagggtctga
tgttcactcagccgaatcaggctgctggatacatggctaagctgatttgggaatctgtgagcgtgacggtggtagctgcggttgaagcaatg
aactggcttaagtctgctgctaagctgctggctgctgaggtcaaagataagaagactggagagattcttcgcaagcgttgcgctgtgcattgg
gtaactcctgatggtttccctgtgtggcaggaatacaagaagcctattcagacgcgcttgaacctgatgttcctcggtcagttccgcttacagc
ctaccattaacaccaacaaagatagcgagattgatgcacacaaacaggagtctggtatcgctcctaactttgtacacagccaagacggtagc
caccttcgtaagactgtagtgtgggcacacgagaagtacggaatcgaatcttttgcactgattcacgactccttcggtaccattccggctgac
gctgcgaacctgttcaaagcagtgcgcgaaactatggttgacacatatgagtcttgtgatgtactggctgatttctacgaccagttcgctgacc
agttgcacgagtctcaattggacaaaatgccagcacttccggctaaaggtaacttgaacctccgtgacatcttagagtcggacttcgcgttcg
cgtaactcgaggaattcgactcaattagttcagtcagtttcaggatattagtcatctctacattgattatgagtattcagaaattccttaaatattctg
acaaatgctctttccctaaactccccccataaaaaaacccgccgaagcgggtttttacgttatttgcggattaacgattactcgttatcagaacc
gcccagacctgcgttcagcagttctgccaggctggcagatgcgtcttccgaattgatccgtcgaccaaagcccgccgaaaggcgggctttt
ctgtgccggcatgataagctgtcaaacatgagaattacaacttatatcgtatggggctgacttcaggtgctacatttgaagagataaattgcact
gaaatctagaaatattttatctgattaataagatgatcttcttgagatcgttttggtctgcgcgtaatctcttgctctgaaaacgaaaaaaccgcctt
gcagggcggtttttcgaaggttctctgagctaccaactctttgaaccgaggtaactggcttggaggagcgcagtcaccaaaacttgtcctttca
gtttagccttaaccggcgcatgacttcaagactaactcctctaaatcaattaccagtggctgctgccagtggtgcttttgcatgtctttccgggtt
ggactcaagacgatagttaccggataaggcgcagcggtcggactgaacggggggttcgtgcatacagtccagcttggagcgaactgcct
acccggaactgagtgtcaggcgtggaatgagacaaacgcggccataacagcggaatgacaccggtaaaccgaaaggcaggaacagga
gagcgcacgagggagccgccaggggaaacgcctggtatctttatagtcctgtcgggtttcgccaccactgatttgagcgtcagatttcgtga
tgcttgtcaggggggcggagcctatggaaaaacggctttgccgcggccctctcacttccctgttaagtatcttcctggcatcttccaggaaatc
tccgccccgttcgtaagccatttccgctcgccgcagtcgaacgaccgagcgtagcgagtcagtgagcgaggaagcggaatatatcctgtat
cacatattctgctgacgcaccggtgcagcctttttctcctgccacatgaagcacttcactgacaccctcatcagtgccaacatagtaagccag
tatacactccgctagggtcatgagattatcaaaaaggatcttcacctagatccttttaaattaaaaatgaagttttaaatcaatctaaagtatatatg
agtaaacttggtctgacagttaccaatgcttaatcagtgaggcacctatctcagcgatctgtctatttcgttcatccatagttgcctgactccccgt
cgtgtagataactacgatacgggagggcttaccatctggccccagtgctgcaatgataccgcgagacccacgctcaccggctccagatttat
cagcaataaaccagccagccggaagggccgagcgcagaagtggtcctgcaactttatccgcctccatccagtctattaattgttgccggga
agctagagtaagtagttcgccagttaatagtttgcgcaacgttgttgccattgctacaggcatcgtggtgtcacgctcgtcgtttggtatggcttc
attcagctccggttcccaacgatcaaggcgagttacatgatcccccatgttgtgcaaaaaagcggttagctccttcggtcctccgatcgttgtc
agaagtaagttggccgcagtgttatcactcatggttatggcagcactgcataattctcttactgtcatgccatccgtaagatgcttttctgtgactg
gtgagtactcaaccaagtcattctgagaatagtgtatgcggcgaccgagttgctcttgcccggcgtcaatacgggataataccgcgccacat
agcagaactttaaaagtgctcatcattggaaaacgttcttcggggcgaaaactctcaaggatcttaccgctgttgagatccagttcgatgtaac
ccactcgtgcacccaactgatcttcagcatcttttactttcaccagcgtttctgggtgagcaaaaacaggaaggcaaaatgccgcaaaaaag
ggaataagggcgacacggaaatgttgaatactcatactcttcctttttcaatattattgaagcatttatcagggttattgtctcatgagcggataca
tatttgaatgtatttagaaaaataaacaaataggggttccgcgcacatttccccgaaaagtgccacctgacgtctaagaaaccattattatcatg
acattaacctataaaaataggcgtatcacgaggccctttcgtcttcaagaattctggcgaatcctctgaccagccagaaaacgacctttctgtg
```

Continued on next page.

3.5.2.1 pID108, continued: T7 RNA Polymerase

gtgaaaccggatgctgcaattcagagcggcagcaagtgggggacagcagaagacctgaccgccgcagagtggatgtttgacatggtgaa
gactatcgcaccatcagccagaaaaccgaattttgctgggtgggctaacgatatccgcctgatgcgtgaacgtgacggacgtaaccaccgc
gacatgtgtgtgctgttccgctgggcatgccaggacaacttctggtccggtaacgtgctgagctaacaccgtgcgtgttgacaattttacctct
ggcggtgataatggttgcagctagc

3.5.2.2 pID 345: SmBIT fused to IBP sensor #492 Ultraspiracle chain

gaattcgcatctagatggtagagccacaaacagccggtacaagcaacgatctccaggaccatctgaatcatgcgcggatgacacgaactc
acgacggcgatcacagacattaacccacagtacagacactgcgacaacgtggcaattcgtcgcaataccgtctcactgaactggccgataa
ttgcagacgaacgcgttgagcaccgccgccgcaaggaatggtgcatgcaaggagatggcgcccaacagtcccccggccacggggcct
gccaccatacccacgccgaaacaagcgctcatgagcccgaagtggcgagcccgatcttccccatcggtgatgtcggcgatataggcgcc
agcaaccgcacctgtggcgccggtgatgccggccacgatgcgtccggcgtagaggatcgagatctcgatcccgcgaaattaatacgactc
actataggggaattgtgagcggataacaattcccctctagaaataattttgtttaactttaagaaggagatatatatggtgaccggctaccggct
gttcgaggagattctgggtagcggcagcggcagcggtagcggcagcggcagggtagcggcttctggcacatcgaatttacaagcagaca
tgcctctggagaggataatcgaagcggagaaacgagtcgaatgcaacgatcccttggtggcattggtggtaaacgagaataataccactgt
gaacaatatctgtcaagcaacacacaagcaactgtttcaattggtccaatgggcgaagctcgtacctcatttcacatcattgccgttgacagat
caggtgcaattgttaagggcgggatggaatgaattgctcatagccgccttctcgcaccggtcgatgcaagcacaggatgctatagttctagc
gacgggattgacagtcaacaaatcgactgcacacgctgtcggcgtcggcaacatctacgaccgcgtcctctccgagctggtgaacaaaat
gaaagaaatgaaaatggacaaaacggaattgggttgtttgcgggcgataattctctacctgcctgcggttcgagggataaagtcggtgcaag
aagtgcgtatgttgctgcgtaaaatcatgggcgtcctcgaggagtacaccaggacgactcatccaaacgagcctggaaggtttgccaaatta
ttagcgcgtttgccggctttaaggtccattgggttgaaatgtctcgaacatctcttctttttcaaactgatcggtgatgtcccgatagatactttcct
aatggagatgttggagggcacaacggattcgtaaatccccaggcatcaaataaaacgaaaggctcagtcgaaagactgggcctttcgtttta
tctgttgtttgtcggtgaacgctctctactagagtcacactggctcaccttcgggtgggcctttctgcgtttatagctgccaatgagacgacggg
gtcatcacggctcatcatgcgcccaacaaatgtgtgccatacacgctcggatgactgcctgatgaccgcactgactggggacagccgatcc
acctaagcctgtgagagaagcagacacccgacagatcaaggcagttaactagtgcactgcagtacagcggccgcgattatcaaaaaggat
cttcacctagatccttttaaattaaaaatgaagttttaaatcaatctaaagtatatatgagtaaacttggtctgacagttaccaatgcttaatcagtg
aggcacctatctcagcgatctgtctatttcgttcatccatagttgcctgactccccgtcgtgtagataactacgatacgggagggcttaccatct
ggccccagtgctgcaatgataccgcgggacccacgctcaccggctccagatttatcagcaataaaccagccagccggaagggccgagc
gcagaagtggtcctgcaactttatccgcctccatccagtctattaattgttgccgggaagctagagtaagtagttcgccagttaatagtttgcgc
aacgttgttgccattgctacaggcatcgtggtgtcacgctcgtcgtttggtatggcttcattcagctccggttcccaacgatcaaggcgagttac
atgatcccccatgttgtgcaaaaaagcggttagctccttcggtcctccgatcgttgtcagaagtaagttggccgcagtgttatcactcatggtta
tggcagcactgcataattctcttactgtcatgccatccgtaagatgcttttctgtgactggtgagtactcaaccaagtcattctgagaatagtgtat
gcggcgaccgagttgctcttgcccggcgtcaatacgggataataccgcgccacatagcagaactttaaaagtgctcatcattggaaaacgtt
cttcggggcgaaaactctcaaggatcttaccgctgttgagatccagttcgatgtaacccactcgtgcacccaactgatcttcagcatcttttactt
tcaccagcgtttctgggtgagcaaaaacaggaaggcaaaatgccgcaaaaaagggaataagggcgacacggaaatgttgaatactcatac
tcttcctttttcaatattattgaagcatttatcagggttattgtctcatgagcggatacatatttgaatgtatttagaaaaataaacaaataggggttc
cgcgcacatttccccgaaaagtgccacctgtcatgaccaaaatcccttaacgtgagttttcgttccactgagcgtcagaccccgtagaaaaga
tcaaaggatcttcttgagatcctttttttctgcgcgtaatctgctgcttgcaaacaaaaaaaccaccgctaccagcggtggtttgtttgccggatc
aagagctaccaactctttttccgaaggtaactggcttcagcagagcgcagataccaaatactgttcttctagtgtagccgtagttaggccacca
cttcaagaactctgtagcaccgcctacatacctcgctctgctaatcctgttaccagtggctgctgccagtggcgataagtcgtgtcttaccggg
ttggactcaagacgatagttaccggataaggcgcagcggtcgggctgaacggggggttcgtgcacacagcccagcttggagcgaacgac
ctacaccgaactgagatacctacagcgtgagctatgagaaagcgccacgcttcccgaagggagaaaggcggacaggtatccggtaagc
ggcagggtcggaacaggagagcgcacgagggagcttccagggggaaacgcctggtatctttatagtcctgtcgggtttcgccacctctga
cttgagcgtcgatttttgtgatgctcgtcaggggggcggagcctatggaaaaacgccagcaacgcggcctttttacggttcctggccttttgct
ggccttttgctcacatgttctttcctgcgttatcccctgattctgtggataaccgtgcggccgcccct

3.5.2.3 pID379: LgBIT fused to IBP sensor #492 Ecdysone Receptor chain

gaattcgcatctagatggtagagccacaaacagccggtacaagcaacgatctccaggaccatctgaatcatgcgcggatgacacgaactc
acgacggcgatcacagacattaacccacagtacagacactgcgacaacgtggcaattcgtcgcaataccgtctcactgaactggccgataa
ttgcagacgaacgcgttgagcaccgccgccgcaaggaatggtgcatgcaaggagatggcgcccaacagtcccccggccacggggcct
gccaccatacccacgccgaaacaagcgctcatgagcccgaagtggcgagcccgatcttccccatcggtgatgtcggcgatataggcgcc
agcaaccgcacctgtggcgccggtgatgccggccacgatgcgtccggcgtagaggatcgagatctcgatcccgcgaaattaatacgactc
actataggggaattgtgagcggataacaattcccctctagaaataattttgtttaactttaagaaggagatatatatggtcttcacactcgaagatt
tcgttggggactgggaacagacagccgcctacaacctggaccaagtccttgaacagggaggtgtgtccagtttgctgcagaatctcgccgt
gtccgtaactccgatccaaaggattgtccggagcggtgaaaatgccctgaagatcgacatccatgtcatcatcccgtatgaaggtctgagcg
ccgaccaaatggcccagatcgaagaggtgtttaaggtggtgtaccctgtggatgatcatcactttaaggtgatcctgccctatggcacactgg
taatcgacggggttacgccgaacatgctgaactatttcggacggccgtatgaaggcatcgccgtgttcgacggcaaaaagatcactgtaac
agggaccctgtggaacggcaacaaaattatcgacgagcgcctgatcacccccgacggctccatgctgttccgagtaaccatcaacagcgg
tagcggcagcggcagttctggtaatggaagtaaaggaatttcgccggagcaagaggagctcatacatcgactggtttatttccagaatgagt
acgaacatccgtctgaggaagacgttaaacggatcattaaccagccgatggatggcgaagatcagtgtgatgttcggtttaggcatatcacg
gaaattaccatcttgacggtgcaacttatcgttgagtttgccaagcggttaccaggctttgacaaactcttaagggaagaccagatcgctctctt
gaaagcatgttccagcgaagtgatgatgttcaggatggcgcgccgttacgacgtacaaacggattccatcctcttcgtaaacaaccaaccgt
attcaagagacagctacaatttggctggcatgggggaaaccatcgaagatctcttgcgtttctgcagatggatgtattggatgcgtgtggaca
acgccgaatacgccttactcacagccatcgtaatattctcagagcgtccggcgctgatcgagggctggaaggtggagaagatccaggaga
tctacttggaggcgctgcgcgcgtacgtggacaaccggaggaagcccaagccgggcacgatattcgcggcgctcctcatgtggctagcg
gcgttggcgacgttaggcaaccaaaattccgagatgtgcttctcgctaaaactgaaaaacaagaaactgccgccgttcttagcggagatctg
ggacgtcgacctgaagacataaatccccaggcatcaaataaaacgaaaggctcagtcgaaagactgggcctttcgttttatctgttgtttgtcg
gtgaacgctctctactagagtcacactggctcaccttcgggtgggcctttctgcgtttatagctgccaatgagacgacggggtcatcacggct
catcatgcgcccaacaaatgtgtgccatacacgctcggatgactgcctgatgaccgcactgactggggacagccgatccacctaagcctgt
gagagaagcagacacccgacagatcaaggcagttaactagtgcactgcagtacagcggccgcgattatcaaaaaggatcttcacctagat
ccttttaaattaaaaatgaagttttaaatcaatctaaagtatatatgagtaaacttggtctgacagttaccaatgcttaatcagtgaggcacctatct
cagcgatctgtctatttcgttcatccatagttgcctgactccccgtcgtgtagataactacgatacgggagggcttaccatctggccccagtgct
gcaatgataccgcgggacccacgctcaccggctccagatttatcagcaataaaccagccagccggaagggccgagcgcagaagtggtc
ctgcaactttatccgcctccatccagtctattaattgttgccgggaagctagagtaagtagttcgccagttaatagtttgcgcaacgttgttgcca
ttgctacaggcatcgtggtgtcacgctcgtcgtttggtatggcttcattcagctccggttcccaacgatcaaggcgagttacatgatcccccat
gttgtgcaaaaaagcggttagctccttcggtcctccgatcgttgtcagaagtaagttggccgcagtgttatcactcatggttatggcagcactg
cataattctcttactgtcatgccatccgtaagatgcttttctgtgactggtgagtactcaaccaagtcattctgagaatagtgtatgcggcgaccg
agttgctcttgcccggcgtcaatacgggataataccgcgccacatagcagaactttaaaagtgctcatcattggaaaacgttcttcggggcga
aaactctcaaggatcttaccgctgttgagatccagttcgatgtaacccactcgtgcacccaactgatcttcagcatcttttactttcaccagcgttt
ctgggtgagcaaaaacaggaaggcaaaatgccgcaaaaaagggaataagggcgacacggaaatgttgaatactcatactcttcctttttca
atattattgaagcatttatcagggttattgtctcatgagcggatacatatttgaatgtatttagaaaaataaacaaataggggttccgcgcacattt
ccccgaaaagtgccacctgtcatgaccaaaatcccttaacgtgagttttcgttccactgagcgtcagaccccgtagaaaagatcaaaggatct
tcttgagatcctttttttctgcgcgtaatctgctgcttgcaaacaaaaaaaccaccgctaccagcggtggtttgtttgccggatcaagagctacc
aactctttttccgaaggtaactggcttcagcagagcgcagataccaaatactgttcttctagtgtagccgtagttaggccaccacttcaagaact
ctgtagcaccgcctacatacctcgctctgctaatcctgttaccagtggctgctgccagtggcgataagtcgtgtcttaccgggttggactcaag
acgatagttaccggataaggcgcagcggtcgggctgaacggggggttcgtgcacacagcccagcttggagcgaacgacctacaccgaa
ctgagatacctacagcgtgagctatgagaaagcgccacgcttcccgaagggagaaaggcggacaggtatccggtaagcggcagggtcg
gaacaggagagcgcacgagggagcttccagggggaaacgcctggtatctttatagtcctgtcgggtttcgccacctctgacttgagcgtcg
atttttgtgatgctcgtcaggggggcggagcctatggaaaaacgccagcaacgcggcctttttacggttcctggccttttgctggccttttgctc
acatgttctttcctgcgttatcccctgattctgtggataaccgtgcggccgcccct

3.5.2.4 pID604: SmBIT fused to IBP sensor #492 Ultraspiracle chain [R336A]

gaattcgcatctagatggtagagccacaaacagccggtacaagcaacgatctccaggaccatctgaatcatgcgcggatgacacgaactc
acgacggcgatcacagacattaacccacagtacagacactgcgacaacgtggcaattcgtcgcaataccgtctcactgaactggccgataa
ttgcagacgaacgcgttgagcaccgccgccgcaaggaatggtgcatgcaaggagatggcgcccaacagtcccccggccacggggcct
gccaccatacccacgccgaaacaagcgctcatgagcccgaagtggcgagcccgatcttccccatcggtgatgtcggcgatataggcgcc
agcaaccgcacctgtggcgccggtgatgccggccacgatgcgtccggcgtagaggatcgagatctcgatcccgcgaaattaatacgactc
actataggggaattgtgagcggataacaattcccctctagaaataattttgtttaactttaagaaggagatatatatggtgaccggctaccggct
gttcgaggagattctgggtagcggcagcggcagcggtagcggcagcggcagggtagcggcttctGGCACATCGAATTTA
CAAGCAGACATGCCTCTGGAGAGGATAATCGAAGCGGAGAAACGAGTCGAATGCA
ACGATCCCTTGGTGGCATTGGTGGTAAACGAGAATAATACCACTGTGAACAATATCT
GTCAAGCAACACACAAGCAACTGTTTCAATTGGTCCAATGGGCGAAGCTCGTACCTC
ATTTCACATCATTGCCGTTGACAGATCAGGTGCAATTGTTAAGGGCGGGATGGAATG
AATTGCTCATAGCCGCCTTCTCGCACCGGTCGATGCAAGCACAGGATGCTATAGTTC
TAGCGACGGGATTGACAGTCAACAAATCGACTGCACACGCTGTCGGCGTCGGCAAC
ATCTACGACCGCGTCCTCTCCGAGCTGGTGAACAAAATGAAAGAAATGAAAATGGA
CAAAACGGAATTGGGTTGTTTGCGGGCGATAATTCTCTACCTGCCTGCGGTTCGAGG
GATAAAGTCGGTGCAAGAAGTGCGTATGTTGCTGgcgAAAATCATGGGCGTCCTCGA
GGAGTACACCAGGACGACTCATCCAAACGAGCCTGGAAGGTTTGCCAAATTATTAG
CGCGTTTGCCGGCTTTAAGGTCCATTGGGTTGAAATGTCTCGAACATCTCTTCTTTTT
CAAACTGATCGGTGATGTCCCGATAGATACTTTCCTAATGGAGATGTTGGAGGGCAC
AACGGATTCGtaaatccccaggcatcaaataaaacgaaaggctcagtcgaaagactgggcctttcgttttatctgttgtttgtcggtg
aacgctctctactagagtcacactggctcaccttcgggtgggcctttctgcgtttatagctgccaatgagacgacggggtcatcacggctcat
catgcgcccaacaaatgtgtgccatacacgctcggatgactgcctgatgaccgcactgactggggacagccgatccacctaagcctgtga
gagaagcagacacccgacagatcaaggcagttaactagtgcactgcagtacagcggccgcgattatcaaaaaggatcttcacctagatcct
tttaaattaaaaatgaagttttaaatcaatctaaagtatatatgagtaaacttggtctgacagttaccaatgcttaatcagtgaggcacctatctca
gcgatctgtctatttcgttcatccatagttgcctgactccccgtcgtgtagataactacgatacgggagggcttaccatctggccccagtgctgc
aatgataccgcgggacccacgctcaccggctccagatttatcagcaataaaccagccagccggaagggccgagcgcagaagtggtcctg
caactttatccgcctccatccagtctattaattgttgccgggaagctagagtaagtagttcgccagttaatagtttgcgcaacgttgttgccattg
ctacaggcatcgtggtgtcacgctcgtcgtttggtatggcttcattcagctccggttcccaacgatcaaggcgagttacatgatcccccatgtt
gtgcaaaaaagcggttagctccttcggtcctccgatcgttgtcagaagtaagttggccgcagtgttatcactcatggttatggcagcactgcat
aattctcttactgtcatgccatccgtaagatgcttttctgtgactggtgagtactcaaccaagtcattctgagaatagtgtatgcggcgaccgagt
tgctcttgcccggcgtcaatacgggataataccgcgccacatagcagaactttaaaagtgctcatcattggaaaacgttcttcggggcgaaa
actctcaaggatcttaccgctgttgagatccagttcgatgtaacccactcgtgcacccaactgatcttcagcatcttttactttcaccagcgtttct
gggtgagcaaaaacaggaaggcaaaatgccgcaaaaaagggaataagggcgacacggaaatgttgaatactcatactcttcctttttcaat
attattgaagcatttatcagggttattgtctcatgagcggatacatatttgaatgtatttagaaaaataaacaaataggggttccgcgcacatttcc
ccgaaaagtgccacctgtcatgaccaaaatcccttaacgtgagttttcgttccactgagcgtcagaccccgtagaaaagatcaaaggatcttc
ttgagatcctttttttctgcgcgtaatctgctgcttgcaaacaaaaaaaccaccgctaccagcggtggtttgtttgccggatcaagagctaccaa
ctctttttccgaaggtaactggcttcagcagagcgcagataccaaatactgttcttctagtgtagccgtagttaggccaccacttcaagaactct
gtagcaccgcctacatacctcgctctgctaatcctgttaccagtggctgctgccagtggcgataagtcgtgtcttaccgggttggactcaaga
cgatagttaccggataaggcgcagcggtcgggctgaacggggggttcgtgcacacagcccagcttggagcgaacgacctacaccgaac
tgagatacctacagcgtgagctatgagaaagcgccacgcttcccgaagggagaaaggcggacaggtatccggtaagcggcagggtcgg
aacaggagagcgcacgagggagcttccagggggaaacgcctggtatctttatagtcctgtcgggtttcgccacctctgacttgagcgtcgat
ttttgtgatgctcgtcaggggggcggagcctatggaaaaacgccagcaacgcggcctttttacggttcctggccttttgctggccttttgctca
catgttctttcctgcgttatcccctgattctgtggataaccgtgcggccgcccct

3.5.2.5 pID606: SmBIT fused to IBP sensor #492 Ultraspiracle chain [M343]

gaattcgcatctagatggtagagccacaaacagccggtacaagcaacgatctccaggaccatctgaatcatgcgcggatgacacgaactc
acgacggcgatcacagacattaacccacagtacagacactgcgacaacgtggcaattcgtcgcaataccgtctcactgaactggccgataa
ttgcagacgaacgcgttgagcaccgccgccgcaaggaatggtgcatgcaaggagatggcgcccaacagtcccccggccacggggcct
gccaccatacccacgccgaaacaagcgctcatgagcccgaagtggcgagcccgatcttccccatcggtgatgtcggcgatataggcgcc
agcaaccgcacctgtggcgccggtgatgccggccacgatgcgtccggcgtagaggatcgagatctcgatcccgcgaaattaatacgactc
actataggggaattgtgagcggataacaattcccctctagaaataattttgtttaactttaagaaggagatatatatggtgaccggctaccggct
gttcgaggagattctgggtagcggcagcggcagcggtagcggcagcggcagggtagcggcttctGGCACATCGAATTTA
CAAGCAGACATGCCTCTGGAGAGGATAATCGAAGCGGAGAAACGAGTCGAATGCA
ACGATCCCTTGGTGGCATTGGTGGTAAACGAGAATAATACCACTGTGAACAATATCT
GTCAAGCAACACACAAGCAACTGTTTCAATTGGTCCAATGGGCGAAGCTCGTACCTC
ATTTCACATCATTGCCGTTGACAGATCAGGTGCAATTGTTAAGGGCGGGATGGAATG
AATTGCTCATAGCCGCCTTCTCGCACCGGTCGATGCAAGCACAGGATGCTATAGTTC
TAGCGACGGGATTGACAGTCAACAAATCGACTGCACACGCTGTCGGCGTCGGCAAC
ATCTACGACCGCGTCCTCTCCGAGCTGGTGAACAAAATGAAAGAAATGAAAATGGA
CAAAACGGAATTGGGTTGTTTGCGGGCGATAATTCTCTACCTGCCTGCGGTTCGAGG
GATAAAGTCGGTGCAAGAAGTGCGTATGTTGCTGCGTAAAATCgcgGGCGTCCTCGA
GGAGTACACCAGGACGACTCATCCAAACGAGCCTGGAAGGTTTGCCAAATTATTAG
CGCGTTTGCCGGCTTTAAGGTCCATTGGGTTGAAATGTCTCGAACATCTCTTCTTTTT
CAAACTGATCGGTGATGTCCCGATAGATACTTTCCTAATGGAGATGTTGGAGGGCAC
AACGGATTCGtaaatccccaggcatcaaataaaacgaaaggctcagtcgaaagactgggcctttcgttttatctgttgtttgtcggtg
aacgctctctactagagtcacactggctcaccttcgggtgggcctttctgcgtttatagctgccaatgagacgacggggtcatcacggctcat
catgcgcccaacaaatgtgtgccatacacgctcggatgactgcctgatgaccgcactgactggggacagccgatccacctaagcctgtga
gagaagcagacacccgacagatcaaggcagttaactagtgcactgcagtacagcggccgcgattatcaaaaaggatcttcacctagatcct
tttaaattaaaaatgaagttttaaatcaatctaaagtatatatgagtaaacttggtctgacagttaccaatgcttaatcagtgaggcacctatctca
gcgatctgtctatttcgttcatccatagttgcctgactccccgtcgtgtagataactacgatacgggagggcttaccatctggccccagtgctgc
aatgataccgcgggacccacgctcaccggctccagatttatcagcaataaaccagccagccggaagggccgagcgcagaagtggtcctg
caactttatccgcctccatccagtctattaattgttgccgggaagctagagtaagtagttcgccagttaatagtttgcgcaacgttgttgccattg
ctacaggcatcgtggtgtcacgctcgtcgtttggtatggcttcattcagctccggttcccaacgatcaaggcgagttacatgatcccccatgtt
gtgcaaaaaagcggttagctccttcggtcctccgatcgttgtcagaagtaagttggccgcagtgttatcactcatggttatggcagcactgcat
aattctcttactgtcatgccatccgtaagatgcttttctgtgactggtgagtactcaaccaagtcattctgagaatagtgtatgcggcgaccgagt
tgctcttgcccggcgtcaatacgggataataccgcgccacatagcagaactttaaaagtgctcatcattggaaaacgttcttcggggcgaaa
actctcaaggatcttaccgctgttgagatccagttcgatgtaacccactcgtgcacccaactgatcttcagcatcttttactttcaccagcgtttct
gggtgagcaaaaacaggaaggcaaaatgccgcaaaaaagggaataagggcgacacggaaatgttgaatactcatactcttcctttttcaat
attattgaagcatttatcagggttattgtctcatgagcggatacatatttgaatgtatttagaaaaataaacaaataggggttccgcgcacatttcc
ccgaaaagtgccacctgtcatgaccaaaatcccttaacgtgagttttcgttccactgagcgtcagaccccgtagaaaagatcaaaggatcttc
ttgagatcctttttttctgcgcgtaatctgctgcttgcaaacaaaaaaaccaccgctaccagcggtggtttgtttgccggatcaagagctaccaa
ctctttttccgaaggtaactggcttcagcagagcgcagataccaaatactgttcttctagtgtagccgtagttaggccaccacttcaagaactct
gtagcaccgcctacatacctcgctctgctaatcctgttaccagtggctgctgccagtggcgataagtcgtgtcttaccgggttggactcaaga
cgatagttaccggataaggcgcagcggtcgggctgaacggggggttcgtgcacacagcccagcttggagcgaacgacctacaccgaac
tgagatacctacagcgtgagctatgagaaagcgccacgcttcccgaagggagaaaggcggacaggtatccggtaagcggcagggtcgg
aacaggagagcgcacgagggagcttccagggggaaacgcctggtatctttatagtcctgtcgggtttcgccacctctgacttgagcgtcgat
ttttgtgatgctcgtcaggggggcggagcctatggaaaaacgccagcaacgcggcctttttacggttcctggccttttgctggccttttgctca
catgttctttcctgcgttatcccctgattctgtggataaccgtgcggccgcccct

3.5.2.6 pID608: LgBIT fused to IBP sensor #492 Ecdysone Receptor chain [M322A]

gaattcgcatctagatggtagagccacaaacagccggtacaagcaacgatctccaggaccatctgaatcatgcgcggatgacacgaactcacga
cggcgatcacagacattaacccacagtacagacactgcgacaacgtggcaattcgtcgcaataccgtctcactgaactggccgataattgcagac
gaacgcgttgagcaccgccgccgcaaggaatggtgcatgcaaggagatggcgcccaacagtcccccggccacggggcctgccaccataccc
acgccgaaacaagcgctcatgagcccgaagtggcgagcccgatcttccccatcggtgatgtcggcgatataggcgccagcaaccgcacctgtg
gcgccggtgatgccggccacgatgcgtccggcgtagaggatcgagatctcgatcccgcgaaattaatacgactcactataggggaattgtgagc
ggataacaattcccctctagaaataattttgtttaactttaagaaggagatatatatggtcttcacactcgaagatttcgttggggactgggaacagaca
gccgcctacaacctggaccaagtccttgaacagggaggtgtgtccagtttgctgcagaatctcgccgtgtccgtaactccgatccaaaggattgtc
cggagcggtgaaaatgccctgaagatcgacatccatgtcatcatcccgtatgaaggtctgagcgccgaccaaatggcccagatcgaagaggtgtt
taaggtggtgtaccctgtggatgatcatcactttaaggtgatcctgccctatggcacactggtaatcgacggggttacgccgaacatgctgaactatt
tcggacggccgtatgaaggcatcgccgtgttcgacggcaaaaagatcactgtaacagggaccctgtggaacggcaacaaaattatcgacgagc
gcctgatcacccccgacggctccatgctgttccgagtaaccatcaacagcggtagcggcagcggcagttctGGTAATGGAAGTAA
AGGAATTTCGCCGGAGCAAGAGGAGCTCATACATCGACTGGTTTATTTCCAGAATGAGT
ACGAACATCCGTCTGAGGAAGACGTTAAACGGATCATTAACCAGCCGATGGATGGCGA
AGATCAGTGTGATGTTCGGTTTAGGCATATCACGGAAATTACCATCTTGACGGTGCAAC
TTATCGTTGAGTTTGCCAAGCGGTTACCAGGCTTTGACAAACTCTTAAGGGAAGACCAG
ATCGCTCTCTTGAAAGCATGTTCCAGCGAAGTGATGATGTTCAGGATGGCGCGCCGTTA
CGACGTACAAACGGATTCCATCCTCTTCGTAAACAACCAACCGTATTCAAGAGACAGCT
ACAATTTGGCTGGCATGGGGGAAACCATCGAAGATCTCTTGCGTTTCTGCAGATGGATG
TATTGGATGCGTGTGGACAACGCCGAATACGCCTTACTCACAGCCATCGTAATATTCTCA
GAGCGTCCGGCGCTGATCGAGGGCTGGAAGGTGGAGAAGATCCAGGAGATCTACTTGG
AGGCGCTGCGCGCGTACGTGGACAACCGGAGGAAGCCCAAGCCGGGCACGATATTCGC
GGCGCTCCTCgcgTGGCTAGCGGCGTTGGCGACGTTAGGCAACCAAAATTCCGAGATGTG
CTTCTCGCTAAAACTGAAAAACAAGAAACTGCCGCCGTTCTTAGCGGAGATCTGGGACG
TCGACCTGAAGACAtaaatccccaggcatcaaataaaacgaaaggctcagtcgaaagactgggcctttcgttttatctgttgtttgtcg
gtgaacgctctctactagagtcacactggctcaccttcgggtgggcctttctgcgtttatagctgccaatgagacgacggggtcatcacggctcatc
atgcgcccaacaaatgtgtgccatacacgctcggatgactgcctgatgaccgcactgactggggacagccgatccacctaagcctgtgagagaa
gcagacacccgacagatcaaggcagttaactagtgcactgcagtacagcggccgcgattatcaaaaaggatcttcacctagatccttttaaattaaa
aatgaagttttaaatcaatctaaagtatatatgagtaaacttggtctgacagttaccaatgcttaatcagtgaggcacctatctcagcgatctgtctatttc
gttcatccatagttgcctgactccccgtcgtgtagataactacgatacgggagggcttaccatctggccccagtgctgcaatgataccgcgggacc
cacgctcaccggctccagatttatcagcaataaaccagccagccggaagggccgagcgcagaagtggtcctgcaactttatccgcctccatcca
gtctattaattgttgccgggaagctagagtaagtagttcgccagttaatagtttgcgcaacgttgttgccattgctacaggcatcgtggtgtcacgctc
gtcgtttggtatggcttcattcagctccggttcccaacgatcaaggcgagttacatgatcccccatgttgtgcaaaaaagcggttagctccttcggtcc
tccgatcgttgtcagaagtaagttggccgcagtgttatcactcatggttatggcagcactgcataattctcttactgtcatgccatccgtaagatgctttt
ctgtgactggtgagtactcaaccaagtcattctgagaatagtgtatgcggcgaccgagttgctcttgcccggcgtcaatacgggataataccgcgc
cacatagcagaactttaaaagtgctcatcattggaaaacgttcttcggggcgaaaactctcaaggatcttaccgctgttgagatccagttcgatgtaa
cccactcgtgcacccaactgatcttcagcatcttttactttcaccagcgtttctgggtgagcaaaaacaggaaggcaaaatgccgcaaaaaaggga
ataagggcgacacggaaatgttgaatactcatactcttcctttttcaatattattgaagcatttatcagggttattgtctcatgagcggatacatatttgaat
gtatttagaaaaataaacaaataggggttccgcgcacatttccccgaaaagtgccacctgtcatgaccaaaatcccttaacgtgagttttcgttccact
gagcgtcagaccccgtagaaaagatcaaaggatcttcttgagatcctttttttctgcgcgtaatctgctgcttgcaaacaaaaaaaccaccgctacca
gcggtggtttgtttgccggatcaagagctaccaactctttttccgaaggtaactggcttcagcagagcgcagataccaaatactgttcttctagtgtag
ccgtagttaggccaccacttcaagaactctgtagcaccgcctacatacctcgctctgctaatcctgttaccagtggctgctgccagtggcgataagtc
gtgtcttaccgggttggactcaagacgatagttaccggataaggcgcagcggtcgggctgaacggggggttcgtgcacacagcccagcttgga
gcgaacgacctacaccgaactgagatacctacagcgtgagctatgagaaagcgccacgcttcccgaagggagaaaggcggacaggtatccgg
taagcggcagggtcggaacaggagagcgcacgagggagcttccagggggaaacgcctggtatctttatagtcctgtcgggtttcgccacctctg
acttgagcgtcgatttttgtgatgctcgtcaggggggcggagcctatggaaaaacgccagcaacgcggcctttttacggttcctggccttttgctgg
ccttttgctcacatgttctttcctgcgttatcccctgattctgtggataaccgtgcggccgcccct

3.5.3 Appendix 3.3: IDs of design constructs

3.5.3.1 Designs targeting caffeine

Scaffold PDB 4DS7: 84, 92, 93, 187, 525, 526, 527, 528, 543, 544, 547, 548, 549, 550, 551, 553, 554, 555, 556, 545

3.5.3.2 Designs targeting *p*-coumaric acid

Scaffold PDB 2AIJ: 100, 104, 127, 160, 170, 462, 463, 464, 465, 470, 472, 504, 505, 506, 510, 511, 512, 513, 514, 515

3.5.3.3 Designs targeting ibuprofen

Scaffold PDB 2NXX: 13, 14, 68, 101, 468, 490, 491, 492, 493, 494, 501, 541

3.5.3.4 Designs targeting serotonin

Scaffold PDB 3EAB: 130, 475, 476, 477

Scaffold PDB 3IA3: 183, 518, 519, 520, 521, 522

Scaffold PDB 3NW0: 131, 145, 480, 481, 482, 483, 496, 497, 498, 499

3.5.3.5 Designs targeting theophylline

84, 92, 93, 187, 525, 526, 528, 543, 544, 547, 548, 549, 550, 551, 553, 554, 555, 556, 545

3.5.3.6 Designs targeting ergosterol

Scaffold PDB 3SFV: 566, 570, 754, 755, 760, 764, 765, 788, 856, 1058, 1061, 1063, 1064, 1065, 1067

3.5.3.7 Designs targeting homoserine lactone

Scaffold PDB 3LWN: 601, 620, 756, 757, 759, 761, 762, 766, 772, 782, 785, 790

3.5.3.8 Designs targeting naproxen

Scaffold PDB 2Z0D: 626, 633, 636, 637, 753, 778, 783, 786, 794, 797, 804, 805, 806, 808

3.5.3.9 Designs targeting thiacloprid

Scaffold PDB 1FQV (232): 652, 684, 770, 767, 774, 775, 789, 795,

Scaffold PDB 3MTN (1147): 667, 802

Scaffold PDB 3N3K (1216): 659, 784, 809

3.6 Figures

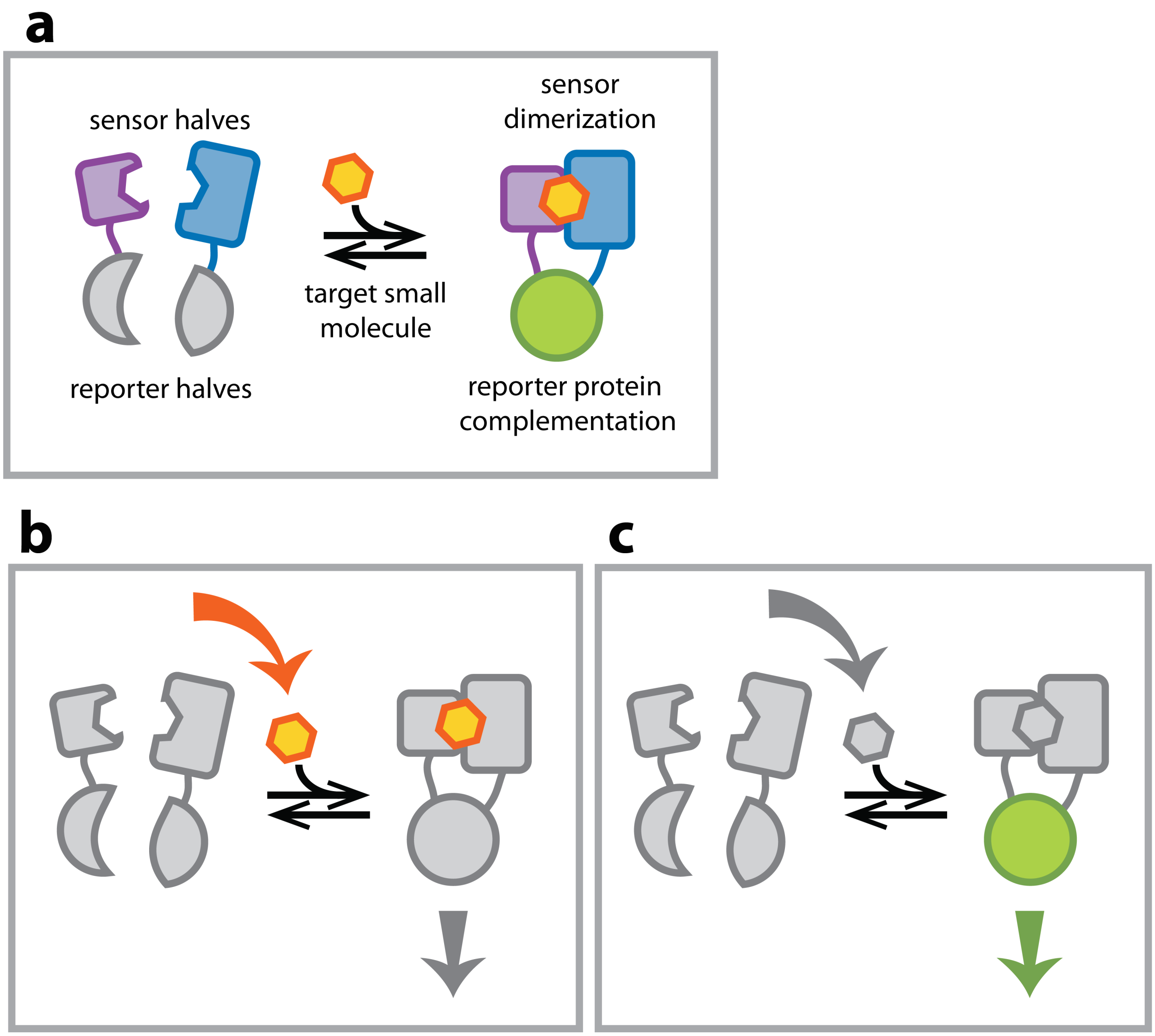


Figure 3.1: Biosensor design schema.
(a) General strategy, wherein ligand binding to a site at the interface of a heterodimer stabilizes dimerization, which is reported by a protein complementation system. (b) Schema in which the desired application is detection of a target small molecule of interest. Any appropriate reporter may be used to respond to presence of the small molecule. (c) Schema in which the desired application is activation of a response, which can be activated by any small molecule for which an inducible heterodimer exists.

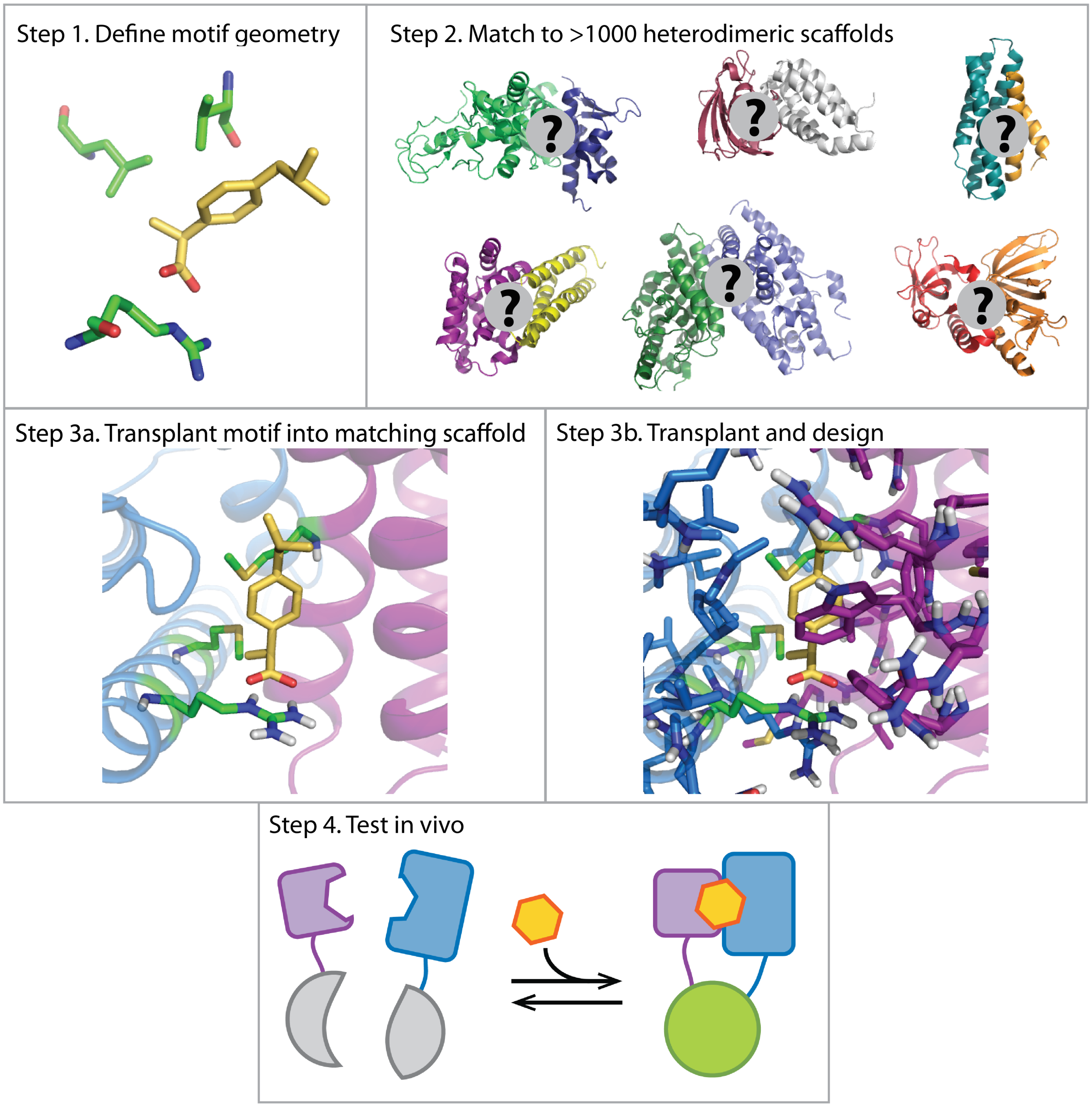

Step 1. Define motif geometry
Step 2. Match to >1000 heterodimeric scaffolds
?
?
?
?
?
?
Step 3a. Transplant motif into matching scaffold
Step 3b. Transplant and design
Step 4. Test in vivo

Figure 3.2: Computational biosensor design method.
In Step 1, selected motif residues of an extant ibuprofen (IBP) binding site in a crystal structure (PDB: 1EQG) of COX-1 are shown in green sticks, while the ligand IBP is shown in yellow sticks. In Step 2, six examples are shown of natural heterodimers from the larger heterodimer library to which the binding site was matched. In Step 3, the selected natural motif residues (green) were transplanted onto a heterodimer Ultraspiracle/Ecdysone Receptor (PDB: 2NXX) (blue and purple cartoon, respectively; PDB: 2NXX), and additional mutations to surrounding side chains (blue and purple sticks) are designed with the goal of accommodating and stabilizing the motif and ligand. In Step 4, designs are tested experimentally.

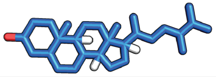
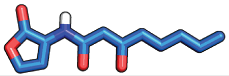
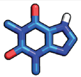
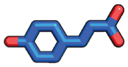
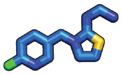
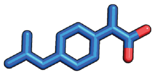
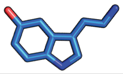
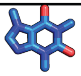
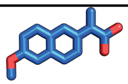
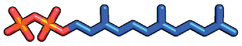
Target	# Designs	Scaffold PDBs	# Interface mutations
Ergosterol (ERG) *Fungal steroid* 	15	1FAP, 3KBT, 3SFV	12-21
Homoserine lactone (LAE) *Quorum sensing* 	12	2LWN, 3LW8	8-15
Naproxen (NPS) *Analgesic* 	14	2F4M, 2Z0D	11-14
Thiacloprid (TH4) *Insecticide* 	13	1FQV, 2QK7, 3MTN, 3N3K	10-22
Coumaric acid (HC4) *generally nontoxic* 	12	2AIJ	13-17
Ibuprofen (IBP) *generally nontoxic* 	12	2NXX	14-20
Serotonin (SRO) *cellular communication* 	15	3IA3, 3NW0	8-16
Caffeine (CFF) *generally nontoxic* 	19	4DS7	11-19
Theophylline (TEP) *generally nontoxic* 	6	2PI2	8-16
Farnesyl pyrophosphate (FPP) *metabolic intermediate* 	5	3FAP	10-19

Figure 3.3: Target ligands for which sensors were designed.

Target column shows chemical name, three letter abbreviation, and general description, along with a structural representation. Also shown are number of computational designs selected for experimental screening, the PDB codes of the scaffold proteins, and number of mutations.

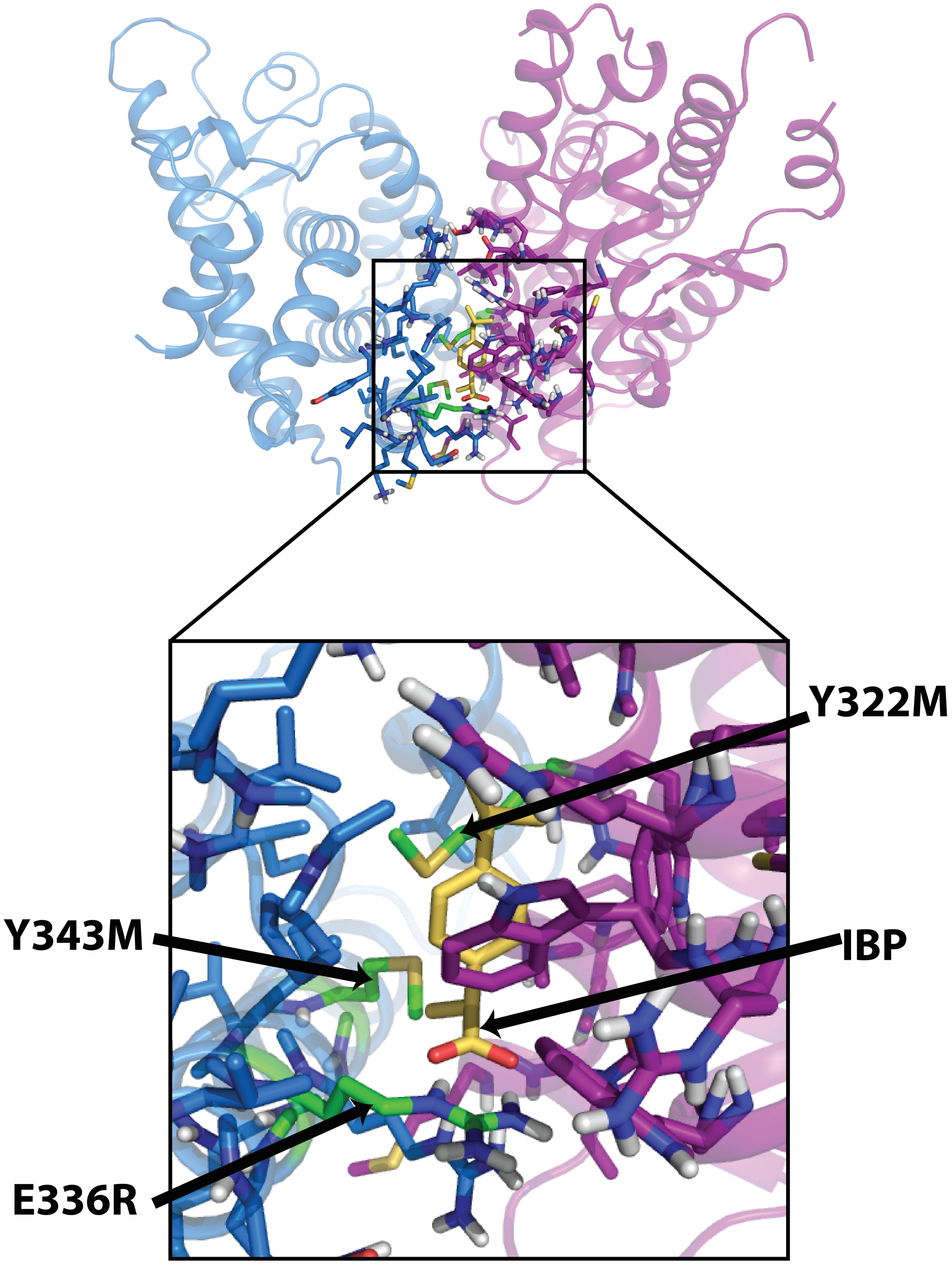


Figure 3.4: Ibuprofen design model.
Cartoon representation of the design model, and closer view of the grafted binding site in Ultraspiracle (blue) and Ecdysone Receptor (purple). The grafted binding site motif residues (green sticks) contains ligand ibuprofen (yellow sticks).

```
Chain A          ULTRASPIRACLE (USP, NR2B4) (Tribolium Castaneum)
Motif residues   R157, M164
Mutations        N144L, D146A, E157R, R160L, E161R, Y164M

Wildtype         DMPLERIIEAEKRVECNDPLVALVVNENNTTVNNICQATHKQLFQLVQWA
Wildtype         KLVPHFTSLPLTDQVQLLRAGWNELLIAAFSHRSMQAQDAIVLATGLTVN

Wildtype         KSTAHAVGVGNIYDRVLSELVNKMKEMKMDKTELGCLRAIILYNPDVRGI
Rosetta design   -------------------------------------------L-A----
Design #490      -------------------------------------------L-A----
Design #492      -------------------------------------------L-A----

Wildtype         KSVQEVEMLREKIYGVLEEYTRTTHPNEPGRFAKLLLRLPALRSIGLKCL
Rosetta design   ------R--LR--M------------------------------------
Design #490      ------R--LL--M------------------------------------
Design #492      ------R--LR--M----------------------A-------------

Wildtype         EHLFFFKLIGDVPIDTFLMEMLEG

Chain E          ECDYSONE RECEPTOR (ECR, NRH1) (Tribolium Castaneum)
Motif residues   M420
Mutations        H352R, T356W, S359W, M360N, T413M, K417A, S420M, V421W, T423A, E424A

Wildtype         ISPEQEELIHRLVYFQNEYEHPSEEDVKRIINDGEDQCDVRFRHITEITI
Wildtype         LTVQLIVEFAKRLPGFDKLLREDQIALLKACSSEVMMFRMARRYDVQTDS

Wildtype         ILFVNNQPYSRDSYNLAGMGETIEDLLHFCRTMYSMRVDNAEYALLTAIV
Rosetta design   ---------------------------R---W--WN--------------
Design #490      ---------------------------A---W--E---------------
Design #492      ---------------------------R---W--W---------------

Wildtype         IFSERPALIEGWKVEKIQEIYLEALRAYVDNRRKPKPGTIFAKLLSVLTE
Rosetta design   --------------------------------------M---A--MW-AA
Design #490      ------------------------------------------A--MW-AA
Design #492      ------------------------------------------A--MW-AA

Wildtype         LRTLGNQNSEMCFSLKLKNKKLPPFLAEIWDVDL
Rosetta design   ----------------------------------
Design #490      ----------------------------------
Design #492      -A--------------------------------
```

Figure 3.5: Sequence alignments.
Sequence alignment showing mutations between the original scaffold protein, a top-ranking design produced by Rosetta, and two designs #490 and 492 containing additional mutations from visual inspection. Motif residues are colored blue.

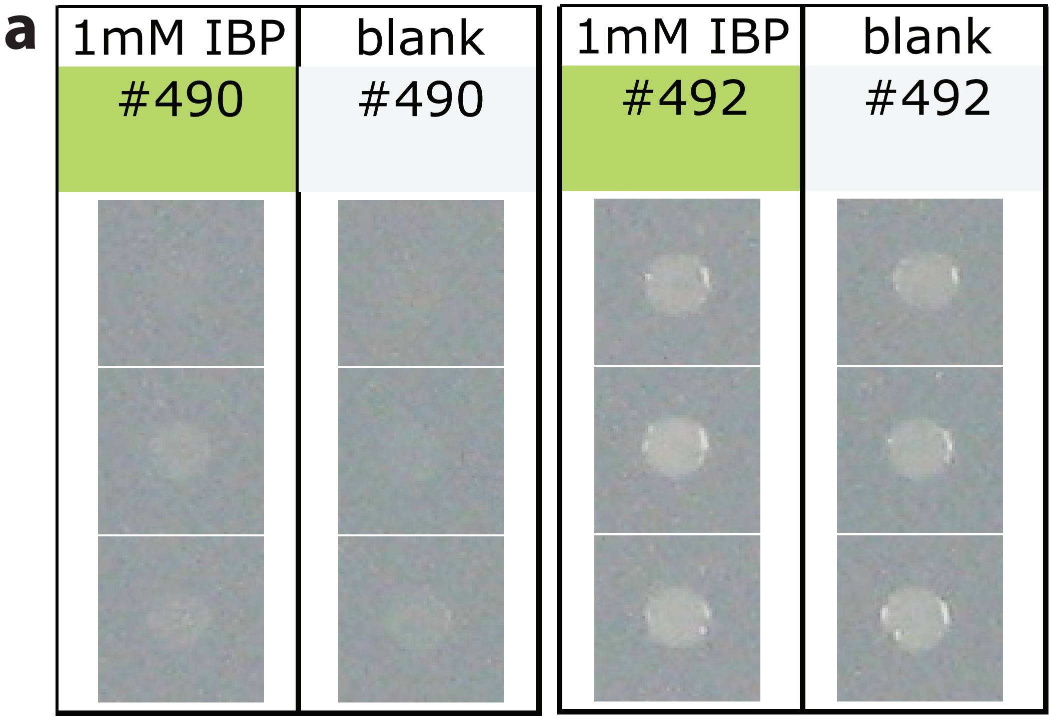


Figure 3.6: Ibuprofen sensor signal in *E. coli* with DHFR reporter on agar plates. Comparison of colony growth with 1mM ibuprofen (IBP) in ethanol, or blank, as described in section 3.4.2. (a) Side-by-side comparison of colony growth with 1mM ibuprofen (IBP) in ethanol, or solvent ethanol blank, after 72 hours for designs #490 (left panel) and #492 (right panel). (b, c) Colony prints for biological triplicates of additional ibuprofen sensor designs. Photographs are shown for 24, 48, and 72 hours (top, middle, and bottom panels, respectively) of growth and labeled by the date the photograph was taken. Plates with ibuprofen are shown in (b), and plates without ibuprofen are shown in (c).

Continued on next page.

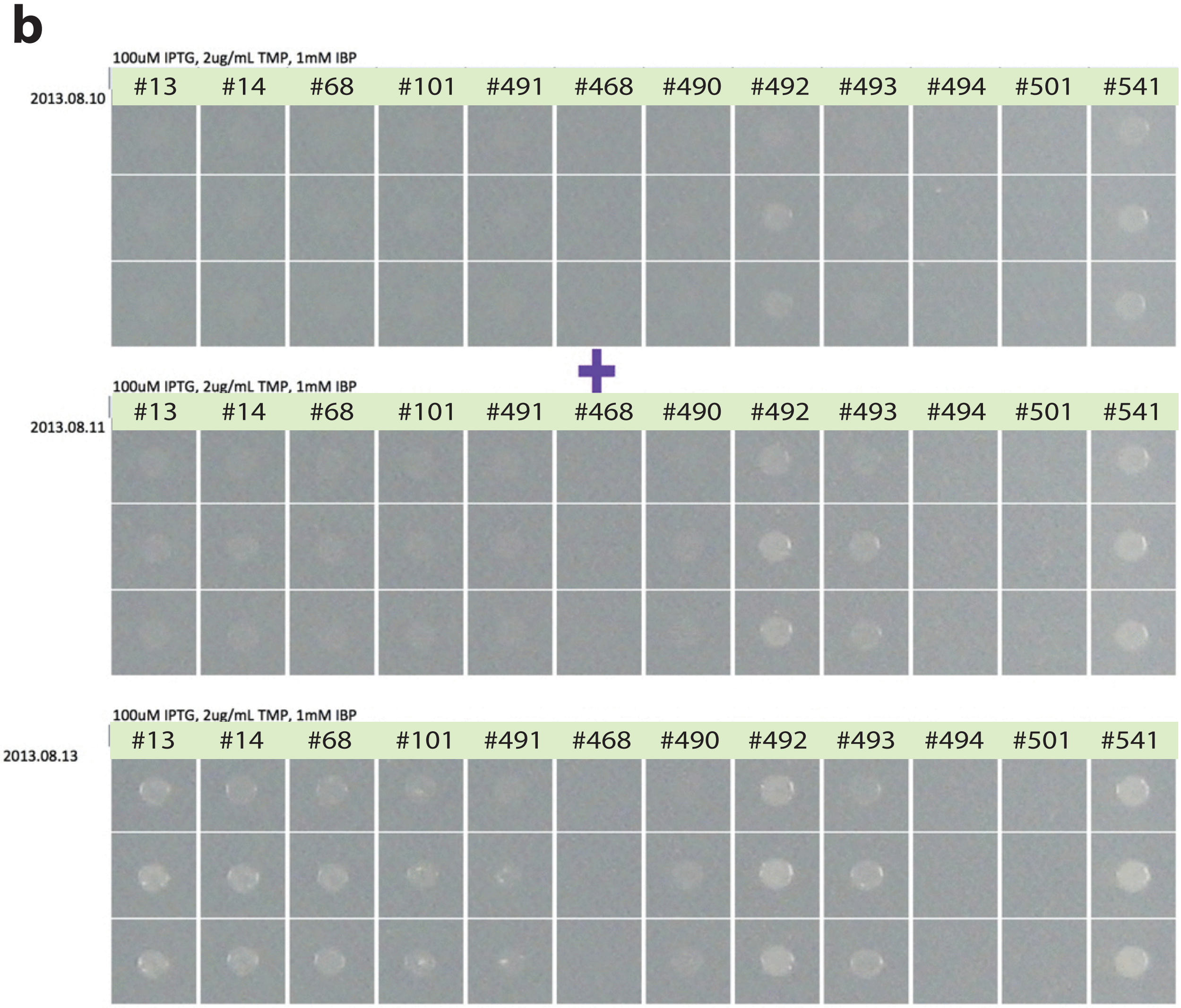


Figure 3.6, continued: Ibuprofen sensor signal in *E. coli* with DHFR reporter on agar plates.

Comparison of colony growth with 1mM ibuprofen (IBP) in ethanol, or blank, as described in section 3.4.2. (a) Side-by-side comparison of colony growth with 1mM ibuprofen (IBP) in ethanol, or solvent ethanol blank, after 72 hours for designs #490 (left panel) and #492 (right panel). (b, c) Colony prints for biological triplicates of additional ibuprofen sensor designs. Photographs are shown for 24, 48, and 72 hours (top, middle, and bottom panels, respectively) of growth and labeled by the date the photograph was taken. Plates with ibuprofen are shown in (b), and plates without ibuprofen are shown in (c).

Continued on next page.

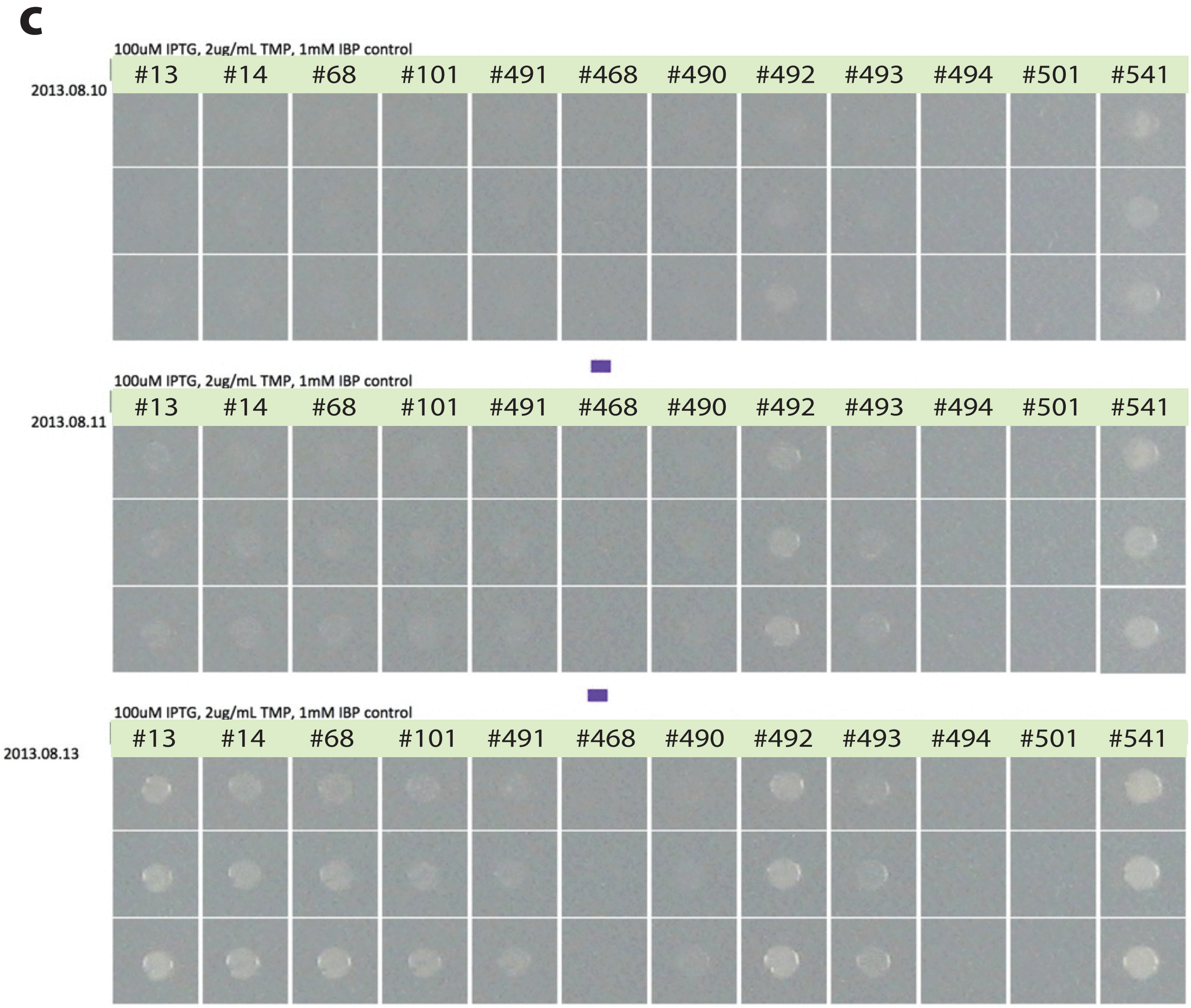
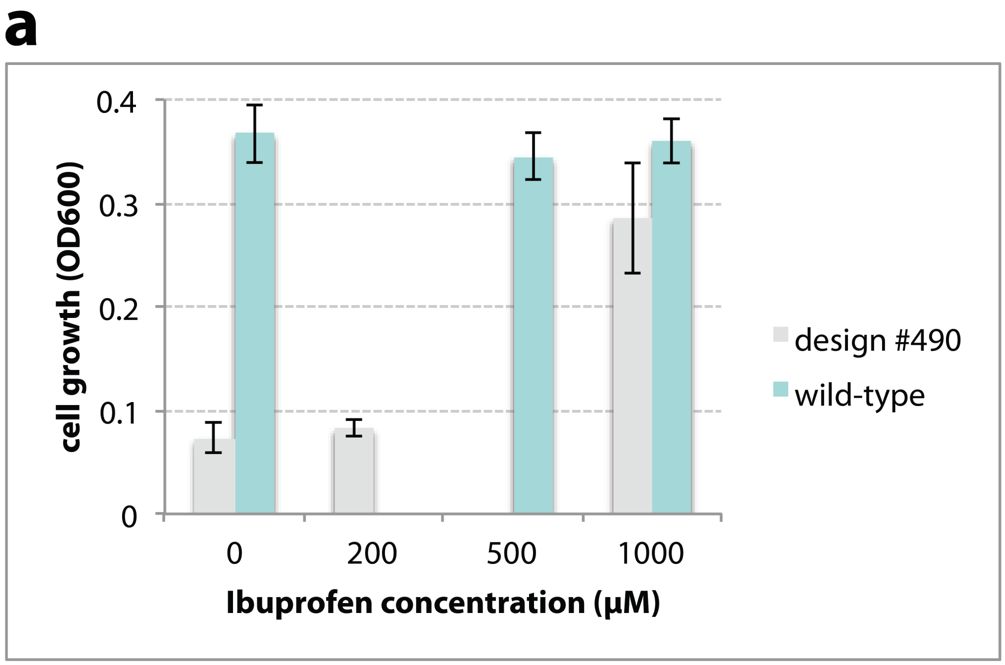


Figure 3.6, continued: Ibuprofen sensor signal in *E. coli* with DHFR reporter on agar plates.

Comparison of colony growth with 1mM ibuprofen (IBP) in ethanol, or blank, as described in section 3.4.2. (a) Side-by-side comparison of colony growth with 1mM ibuprofen (IBP) in ethanol, or solvent ethanol blank, after 72 hours for designs #490 (left panel) and #492 (right panel). (b, c) Colony prints for biological triplicates of additional ibuprofen sensor designs. Photographs are shown for 24, 48, and 72 hours (top, middle, and bottom panels, respectively) of growth and labeled by the date the photograph was taken. Plates with ibuprofen are shown in (b), and plates without ibuprofen are shown in (c).



biological replicate 1				biological replicate 2				biological replicate 3			
1	2	3	4	5	6	7	8	9	10	11	12
0	200	1000	0	200	1000	0	200	1000	0	200	1000
200	1000	0	200	1000	0	200	1000	0	200	1000	0
1000	0	200	1000	0	200	1000	0	200	1000	0	200
0	200	1000	0	200	1000	0	200	1000	0	200	1000
200	1000	0	200	1000	0	200	1000	0	200	1000	0
1000	0	200	1000	0	200	1000	0	200	1000	0	200
0	200	1000	0	200	1000	0	200	1000	0	200	1000
200	1000	0	200	1000	0	200	1000	0	200	1000	0

Figure 3.7: Ibuprofen sensor signal with in *E. coli* DHFR reporter in liquid culture.
(a) Comparison growth for cells expressing either ibuprofen sensor design #490 or wild-type scaffold. Constructs are linked to the essential metabolic enzyme DHFR, such that cell growth (OD_{600}) is dependent on complementation of the DHFR portion of the sensor/DHFR construct. Growth is shown for a titration of ibuprofen concentrations. Values represent the average and standard deviation across 32 wells of the plate for each ibuprofen concentration. A caveat for the data shown is different experiment dates; data for ibuprofen sensor design #490 were collected on 5/6/2015, while data for ibuprofen wild-type scaffold protein were collected on 6/13/2015. (b) Plate layout for design #490 data shown in (a). The number in each well of the 96-well plate indicates the ibuprofen concentration, while the headers indicate which columns correspond to each biological replicate.

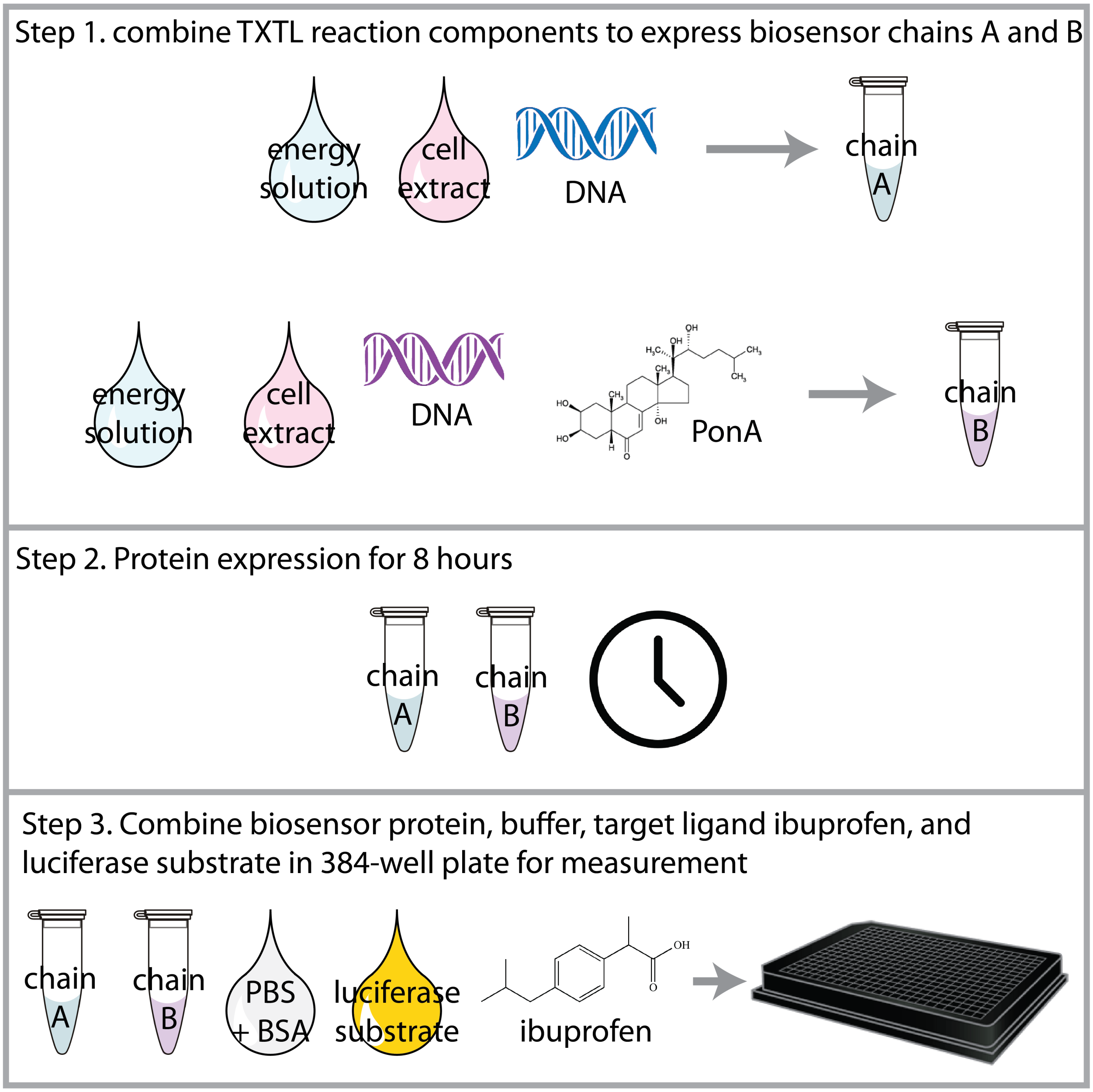

Step 1. combine TXTL reaction components to express biosensor chains A and B
energy solution
cell extract
DNA
chain A
energy solution
cell extract
DNA
PonA
chain B
Step 2. Protein expression for 8 hours
chain A
chain B
Step 3. Combine biosensor protein, buffer, target ligand ibuprofen, and luciferase substrate in 384-well plate for measurement
chain A
chain B
PBS + BSA
luciferase substrate
ibuprofen

Figure 3.8: TXTL method for biosensor characterization.
Components of TXTL reaction, as described in Methods. Energy buffer, cell extract, DNA, and cofactors are combined for the protein expression reaction, which takes 8 hours. The expressed protein is then diluted in PBS+BSA and mixed with the target ligand ibuprofen and the luciferase substrate in a 384-well plate, wherein signal is immediately measured.

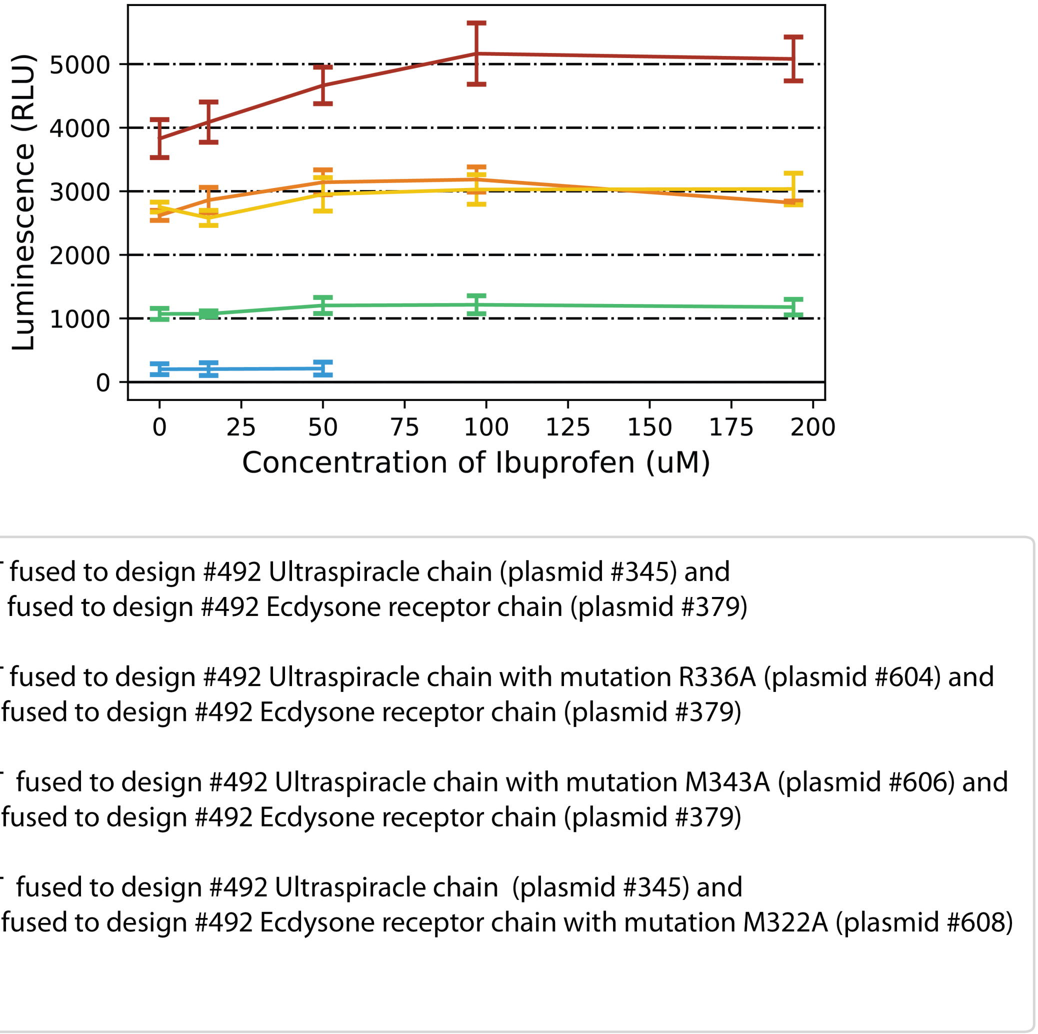


Figure 3.9: Ibuprofen sensor signal with NanoLuc reporter in TXTL.
Signal for ibuprofen sensor #492 compared to alanine mutations for each of the motif residues. Design #492 (red) is compared to alanine mutations of the two motif residues on the Ultraspiracle chain (orange and gold), and of the one motif residue on the Ecdysone receptor chain (green), and a blank sample containing no protein (blue). As described in Methods, SmBIT and LgBIT are the two halves of the NanoLuc protein complementation system. Values represent the average and standard deviation of four wells measured for each construct (or eight wells for each blank) and ligand concentration, the layout of which is defined in Table 3.2.

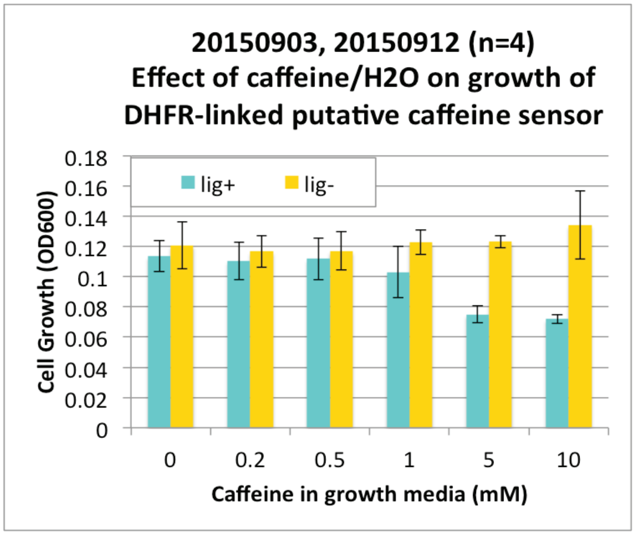


Figure 3.10: Effect of ligand caffeine on cell growth.
Caffeine decreases culture density, with effect increasing with caffeine concentration. The cells expressed control constructs composed of the wild-type scaffold used to design sensors for target ligand fused to the DHFR reporter. Averages and standard deviations are across data collected in biological duplicate on experiments on two different days, for a total of 4 data points per condition and construct. Biological duplicate here refers to separate colonies picked from an agar plate to create separate cell cultures, which were then subjected to identical growth conditions.

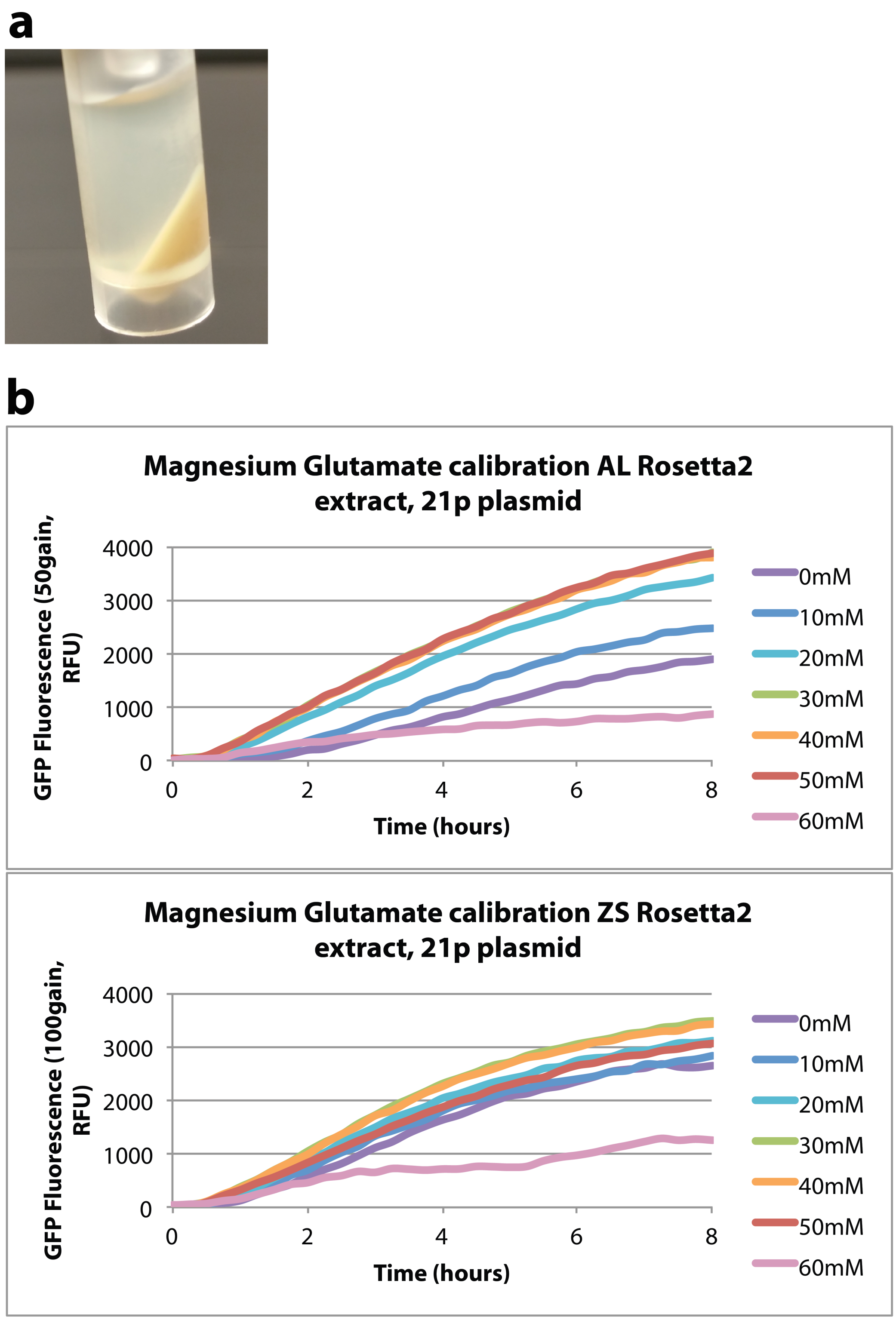


Figure 3.11: TXTL extract preparation.
(a) Photograph of TXTL cell extract prepared in our lab. The supernatant is the extract at the final stage of preparation; the pellet was discarded. (b) Magnesium glutamate calibration for TXTL extract prepared in our lab (top) and TXTL extract prepared by Sun et al [81] (bottom).

3.7 Tables

Table 3.1: DNA concentrations used for TXTL expression.
Shown are the concentrations of DNA used for sensor design protein expression in TXTL.

pID	DNA concentration (nM)
345	1
379	2
604	1
606	2
608	1

Table 3.2: NanoLuc experimental plate layout
Shown are columns of a 384-well plate. Each cell is labeled with a description of the design construct or blank it contained. The ligand concentrations, which are identical for all columns, are shown at the left.

Ligand (µM)		1	2	3	4	5	6	7	8
0	A	492	492	492 [R336A]	492 [R336A]	492 [M343A]	492 [M343A]	492 [M322A]	492 [M322A]
15	B	492	492	492 [R336A]	492 [R336A]	492 [M343A]	492 [M343A]	492 [M322A]	492 [M322A]
s50	C	492	492	492 [R336A]	492 [R336A]	492 [M343A]	492 [M343A]	492 [M322A]	492 [M322A]
97	D	492	492	492 [R336A]	492 [R336A]	492 [M343A]	492 [M343A]	492 [M322A]	492 [M322A]
194	E	492	492	492 [R336A]	492 [R336A]	492 [M343A]	492 [M343A]	492 [M322A]	492 [M322A]
0	F	blank	blank	blank	blank	blank	blank	blank	blank
15	G	blank	blank	blank	blank	blank	blank	blank	blank
50	H	blank	blank	blank	blank	blank	blank	blank	blank
0	I	492	492	492 [R336A]	492 [R336A]	492 [M343A]	492 [M343A]	492 [M322A]	492 [M322A]
15	J	492	492	492 [R336A]	492 [R336A]	492 [M343A]	492 [M343A]	492 [M322A]	492 [M322A]
50	K	492	492	492 [R336A]	492 [R336A]	492 [M343A]	492 [M343A]	492 [M322A]	492 [M322A]
97	L	492	492	492 [R336A]	492 [R336A]	492 [M343A]	492 [M343A]	492 [M322A]	492 [M322A]
194	M	492	492	492 [R336A]	492 [R336A]	492 [M343A]	492 [M343A]	492 [M322A]	492 [M322A]
0	N	blank	blank	blank	blank	blank	blank	blank	blank
15	O	blank	blank	blank	blank	blank	blank	blank	blank
50	P	blank	blank	blank	blank	blank	blank	blank	blank

Chapter 4: Conclusions

Design of protein sensor/actuators for molecules for which no sensors exist presents a number of unique challenges. Computational design of ligand 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. We developed a benchmark framework for comparing Rosetta design methods against each other by quantifying the ability of each design method to recapitulate known sequence profiles from experimental data from library screens, including deep sequencing enrichment/depletion data, and from sequence alignments of naturally evolved proteins.

In addition to the challenges surrounding *in silico* protein design, it remains laborious to screen individual proteins for dimerization in *E. coli* and *in vitro*. We present an experimental method for efficient screening of ligand-inducible heterodimers without protein purification. Cell-free protein expression requires only microliter volumes and a few hours, contributing efficiency in both material cost and time to generate results. The use of cell extract for protein expression eliminates the need to deliver components, such as ligand or reporter substrate, to cell interiors during screening. Direct addition of reporter substrate enables screens to take advantage of the modularity of the designed system (Figure 3.1) to use enzymatic reporters, which amplify signal. Direct ligand addition enables screening of proteins designed to target ligands, which may not be found in the interior of the cell strains typically used to express designed proteins, but which are of utility for synthetic biology applications.

However, difficulties remain even with the methods presented here. Experimental screens are often characterized by low signal to noise ratio, especially when testing initial computational designs that may exhibit low affinity or stability which contribute to low signal. Both protein stability and ligand-mediated affinity could be increased, and background signal decreased, by directed evolution with positive and negative selection.

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 transplanted rather than pre-existing binding sites. We used Rosetta design to create sensor proteins targeting ligands such as ibuprofen. Our experimental results demonstrate that these designs were far from optimal, highlighting the need for continued improvements in methods for sampling protein sequence and conformational space.

Bibliography

1. Glasgow A. A. , H.Y.-M., Mandell D. J., Thompson M., Ritterson R., Loshbaugh A. L., Pellegrino J., Krivacic C., Pache R. A., Barlow K. A., Ollikainen N., Kelly M. J. S., Fraser J. S., Kortemme T., *Computational design of a modular sense/response system.* In preparation, 2019.

2. Bick, M.J., et al., *Computational design of environmental sensors for the potent opioid fentanyl.* Elife, 2017. **6**.

3. Tinberg, C.E., et al., *Computational design of ligand-binding proteins with high affinity and selectivity.* Nature, 2013. **501**(7466): p. 212-6.

4. Procko, E., et al., *Computational design of a protein-based enzyme inhibitor.* J Mol Biol, 2013. **425**(18): p. 3563-75.

5. Rothlisberger, D., et al., *Kemp elimination catalysts by computational enzyme design.* Nature, 2008. **453**(7192): p. 190-5.

6. Hill, Z.B., et al., *Human antibody-based chemically induced dimerizers for cell therapeutic applications.* Nat Chem Biol, 2018. **14**(2): p. 112-117.

7. Arber, C., M. Young, and P. Barth, *Reprogramming cellular functions with engineered membrane proteins.* Curr Opin Biotechnol, 2017. **47**: p. 92-101.

8. Bi, S., et al., *Engineering Hybrid Chemotaxis Receptors in Bacteria.* ACS Synth Biol, 2016. **5**(9): p. 989-1001.

9. Dou, J., et al., *Sampling and energy evaluation challenges in ligand binding protein design.* Protein Sci, 2017. **26**(12): p. 2426-2437.

10. Leaver-Fay, A., et al., *ROSETTA3: an object-oriented software suite for the simulation and design of macromolecules.* Methods Enzymol, 2011. **487**: p. 545-74.

11. Alford, R.F., et al., *The Rosetta All-Atom Energy Function for Macromolecular Modeling and Design.* J Chem Theory Comput, 2017. **13**(6): p. 3031-3048.

12. Gordon, D.B., S.A. Marshall, and S.L. Mayo, *Energy functions for protein design.* Curr Opin Struct Biol, 1999. **9**(4): p. 509-13.

13. Ponder, J.W. and F.M. Richards, *Tertiary templates for proteins. Use of packing criteria in the enumeration of allowed sequences for different structural classes.* J Mol Biol, 1987. **193**(4): p. 775-91.

14. Dahiyat, B.I. and S.L. Mayo, *De novo protein design: fully automated sequence selection.* Science, 1997. **278**(5335): p. 82-7.

15. Eriksson, A.E., et al., *Response of a protein structure to cavity-creating mutations and its relation to the hydrophobic effect.* Science, 1992. **255**(5041): p. 178-83.

16. Baldwin, E.P., et al., *The role of backbone flexibility in the accommodation of variants that repack the core of T4 lysozyme.* Science, 1993. **262**(5140): p. 1715-8.

17. Davis, I.W., et al., *The backrub motion: how protein backbone shrugs when a sidechain dances.* Structure, 2006. **14**(2): p. 265-74.

18. Rocklin, G.J., et al., *Global analysis of protein folding using massively parallel design, synthesis, and testing.* Science, 2017. **357**(6347): p. 168-175.

19. Harbury, P.B., et al., *High-resolution protein design with backbone freedom.* Science, 1998. **282**(5393): p. 1462-7.

20. Koga, N., et al., *Principles for designing ideal protein structures.* Nature, 2012. **491**(7423): p. 222-7.

21. Huang, P.S., et al., *High thermodynamic stability of parametrically designed helical bundles.* Science, 2014. **346**(6208): p. 481-485.

22. Correia, B.E., et al., *Proof of principle for epitope-focused vaccine design.* Nature, 2014. **507**(7491): p. 201-6.

23. Procko, E., et al., *A computationally designed inhibitor of an Epstein-Barr viral Bcl-2 protein induces apoptosis in infected cells.* Cell, 2014. **157**(7): p. 1644-56.

24. Kries, H., R. Blomberg, and D. Hilvert, *De novo enzymes by computational design.* Curr Opin Chem Biol, 2013. **17**(2): p. 221-8.

25. Fleishman, S.J., et al., *Computational design of proteins targeting the conserved stem region of influenza hemagglutinin.* Science, 2011. **332**(6031): p. 816-21.

26. Kundert, K. and T. Kortemme, *Computational design of structured loops for new protein functions.* Biol Chem, 2019. **400**(3): p. 275-288.

27. Desjarlais, J.R. and T.M. Handel, *Side-chain and backbone flexibility in protein core design.* J Mol Biol, 1999. **290**(1): p. 305-18.

28. Larson, S.M., et al., *Thoroughly sampling sequence space: large-scale protein design of structural ensembles.* Protein Sci, 2002. **11**(12): p. 2804-13.

29. Davey, J.A. and R.A. Chica, *Improving the accuracy of protein stability predictions with multistate design using a variety of backbone ensembles.* Proteins, 2014. **82**(5): p. 771-84.

30. Davey, J.A. and R.A. Chica, *Multistate Computational Protein Design with Backbone Ensembles.* Methods Mol Biol, 2017. **1529**: p. 161-179.

31. Fu, X., J.R. Apgar, and A.E. Keating, *Modeling backbone flexibility to achieve sequence diversity: the design of novel alpha-helical ligands for Bcl-xL.* J Mol Biol, 2007. **371**(4): p. 1099-117.

32. Ding, F. and N.V. Dokholyan, *Emergence of protein fold families through rational design.* PLoS Comput Biol, 2006. **2**(7): p. e85.

33. Humphris, E.L. and T. Kortemme, *Prediction of protein-protein interface sequence diversity using flexible backbone computational protein design.* Structure, 2008. **16**(12): p. 1777-88.

34. Friedland, G.D., et al., *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): p. e1000393.

35. Simons, K.T., et al., *Assembly of protein tertiary structures from fragments with similar local sequences using simulated annealing and Bayesian scoring functions.* J Mol Biol, 1997. **268**(1): p. 209-25.

36. Simons, K.T., et al., *Ab initio protein structure prediction of CASP III targets using ROSETTA.* Proteins, 1999. **Suppl 3**: p. 171-6.

37. Hu, X., et al., *High-resolution design of a protein loop.* Proc Natl Acad Sci U S A, 2007. **104**(45): p. 17668-73.

38. Saunders, C.T. and D. Baker, *Recapitulation of protein family divergence using flexible backbone protein design.* J Mol Biol, 2005. **346**(2): p. 631-44.

39. Kuhlman, B., et al., *Design of a novel globular protein fold with atomic-level accuracy.* Science, 2003. **302**(5649): p. 1364-8.

40. Coutsias, E.A., et al., *A kinematic view of loop closure.* J Comput Chem, 2004. **25**(4): p. 510-28.

41. Mandell, D.J., E.A. Coutsias, and T. Kortemme, *Sub-angstrom accuracy in protein loop reconstruction by robotics-inspired conformational sampling.* Nat Methods, 2009. **6**(8): p. 551-2.

42. Ollikainen, N., R.M. de Jong, and T. Kortemme, *Coupling Protein Side-Chain and Backbone Flexibility Improves the Re-design of Protein-Ligand Specificity.* PLoS Comput Biol, 2015. **11**(9): p. e1004335.

43. Smith, C.A. and T. Kortemme, *Backrub-like backbone simulation recapitulates natural protein conformational variability and improves mutant side-chain prediction.* J Mol Biol, 2008. **380**(4): p. 742-56.

44. Friedland, G.D., et al., *A simple model of backbone flexibility improves modeling of side-chain conformational variability.* J Mol Biol, 2008. **380**(4): p. 757-74.

45. Kapp, G.T., et al., *Control of protein signaling using a computationally designed GTPase/GEF orthogonal pair.* Proc Natl Acad Sci U S A, 2012. **109**(14): p. 5277-82.

46. Dou, J., et al., *De novo design of a fluorescence-activating beta-barrel.* Nature, 2018. **561**(7724): p. 485-491.

47. Chevalier, A., et al., *Massively parallel de novo protein design for targeted therapeutics.* Nature, 2017. **550**(7674): p. 74-79.

48. Li, Q., et al., *Discovery of peptide inhibitors targeting human programmed death 1 (PD-1) receptor.* Oncotarget, 2016. **7**(40): p. 64967-64976.

49. Khatib, F., et al., *Algorithm discovery by protein folding game players.* Proc Natl Acad Sci U S A, 2011. **108**(47): p. 18949-53.

50. Tyka, M.D., et al., *Alternate states of proteins revealed by detailed energy landscape mapping.* J Mol Biol, 2011. **405**(2): p. 607-18.

51. Chen, Z., et al., *Programmable design of orthogonal protein heterodimers.* Nature, 2019. **565**(7737): p. 106-111.

52. Hosseinzadeh, P., et al., *Comprehensive computational design of ordered peptide macrocycles.* Science, 2017. **358**(6369): p. 1461-1466.

53. Silva, D.A., et al., *De novo design of potent and selective mimics of IL-2 and IL-15.* Nature, 2019. **565**(7738): p. 186-191.

54. Smith, C.A. and T. Kortemme, *Predicting the tolerated sequences for proteins and protein interfaces using RosettaBackrub flexible backbone design.* PLoS One, 2011. **6**(7): p. e20451.

55. Barlow, K.A., et al., *Flex ddG: Rosetta Ensemble-Based Estimation of Changes in Protein-Protein Binding Affinity upon Mutation.* J Phys Chem B, 2018. **122**(21): p. 5389-5399.

56. Ollikainen, N., et al., *Flexible backbone sampling methods to model and design protein alternative conformations.* Methods Enzymol, 2013. **523**: p. 61-85.

57. Smith, C.A. and T. Kortemme, *Structure-based prediction of the peptide sequence space recognized by natural and synthetic PDZ domains.* J Mol Biol, 2010. **402**(2): p. 460-74.

58. Gerstner, R.B., P. Carter, and H.B. Lowman, *Sequence plasticity in the antigen-binding site of a therapeutic anti-HER2 antibody.* J Mol Biol, 2002. **321**(5): p. 851-62.

59. Pal, G., et al., *Comprehensive and quantitative mapping of energy landscapes for protein-protein interactions by rapid combinatorial scanning.* J Biol Chem, 2006. **281**(31): p. 22378-85.

60. Babor, M., D.J. Mandell, and T. Kortemme, *Assessment of flexible backbone protein design methods for sequence library prediction in the therapeutic antibody Herceptin-HER2 interface.* Protein Sci, 2011. **20**(6): p. 1082-9.

61. Treynor, T.P., 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): p. 48-53.

62. Hayes, R.J., et al., *Combining computational and experimental screening for rapid optimization of protein properties.* Proc Natl Acad Sci U S A, 2002. **99**(25): p. 15926-31.

63. Voigt, C.A., et al., *Computational method to reduce the search space for directed protein evolution.* Proc Natl Acad Sci U S A, 2001. **98**(7): p. 3778-83.

64. Koehl, P. and M. Levitt, *Protein topology and stability define the space of allowed sequences.* Proc Natl Acad Sci U S A, 2002. **99**(3): p. 1280-5.

65. Larson, S.M., et al., *Increased detection of structural templates using alignments of designed sequences.* Proteins, 2003. **51**(3): p. 390-6.

66. Humphris-Narayanan, E., et al., *Prediction of mutational tolerance in HIV-1 protease and reverse transcriptase using flexible backbone protein design.* PLoS Comput Biol, 2012. **8**(8): p. e1002639.

67. Fowler, D.M., et al., *High-resolution mapping of protein sequence-function relationships.* Nat Methods, 2010. **7**(9): p. 741-6.

68. Cunningham, B.C. and J.A. Wells, *Comparison of a structural and a functional epitope.* J Mol Biol, 1993. **234**(3): p. 554-63.

69. Park, H., et al., *Simultaneous Optimization of Biomolecular Energy Functions on Features from Small Molecules and Macromolecules.* J Chem Theory Comput, 2016. **12**(12): p. 6201-6212.

70. Georgiev, I., et al., *Algorithm for backrub motions in protein design.* Bioinformatics, 2008. **24**(13): p. i196-204.

71. Go, N.S., H. A., *Ring Closure and Local Conformational Deformations of Chain Molecules.* Macromolecules, 1970. **3**(2): p. 178–187.

72. Ollikainen, N.d.J., R. M., Kortemme, T., *Coupling protein side-chain and backbone flexibility improves the re-design of protein-ligand specificity.* PLoS Comput Biol, 2015. **11**(9): p. e1004335.

73. Martin, A.C., *Accessing the Kabat antibody sequence database by computer.* Proteins, 1996. **25**(1): p. 130-3.

74. Peralta-Yahya, P.P., et al., *Identification and microbial production of a terpene-based advanced biofuel.* Nat Commun, 2011. **2**: p. 483.

75. Rollins, C.T., et al., *A ligand-reversible dimerization system for controlling protein-protein interactions.* Proc Natl Acad Sci U S A, 2000. **97**(13): p. 7096-101.

76. Nishimura, N., et al., *Structural mechanism of abscisic acid binding and signaling by dimeric PYR1.* Science, 2009. **326**(5958): p. 1373-9.

77. Zanghellini, A., et al., *New algorithms and an in silico benchmark for computational enzyme design.* Protein Sci, 2006. **15**(12): p. 2785-94.

78. Rainsford, K.D., *Ibuprofen: pharmacology, efficacy and safety.* Inflammopharmacology, 2009. **17**(6): p. 275-342.

79. Remy, I., F.X. Campbell-Valois, and S.W. Michnick, *Detection of protein-protein interactions using a simple survival protein-fragment complementation assay based on the enzyme dihydrofolate reductase.* Nat Protoc, 2007. **2**(9): p. 2120-5.

80. Dixon, A.S., et al., *NanoLuc Complementation Reporter Optimized for Accurate Measurement of Protein Interactions in Cells.* ACS Chem Biol, 2016. **11**(2): p. 400-8.

81. Sun, Z.Z., et al., *Protocols for implementing an Escherichia coli based TX-TL cell-free expression system for synthetic biology.* J Vis Exp, 2013(79): p. e50762.

Publishing Agreement

It is the policy of the University to encourage the distribution of all theses, dissertations, and manuscripts. Copies of all UCSF theses, dissertations, and manuscripts will be routed to the library via the Graduate Division. The library will make all theses, dissertations, and manuscripts accessible to the public and will preserve these to the best of their abilities, in perpetuity.

I hereby grant permission to the Graduate Division of the University of California, San Francisco to release copies of my thesis, dissertation, or manuscript to the Campus Library to provide access and preservation, in whole or in part, in perpetuity.

Author Signature ____________________ Date March 26, 2019