Computational Design of Protein-Ligand Interfaces Using RosettaLigand

By

Brittany Ann Allison

Dissertation
Submitted to the Faculty of the
Graduate School of Vanderbilt University
in partial fulfillment of the requirements
for the degree of

DOCTOR OF PHILOSOPHY
in

Chemistry
May, 2016
Nashville, Tennessee

Approved:

Jens Meiler, Ph.D.

Brian O. Bachmann, Ph.D.

John A. Capra, Ph.D.

Michael P. Stone, Ph.D.

To my parents, Warren and Talita Allison, and my sister Kelly

Thank you for always believing in me

ACKNOWLEDGEMENTS

To Vanderbilt University and especially the Department of Chemistry, thank you for the opportunity and supporting me in my time as a doctoral student. To earn a PhD in chemistry is no small feat, and I am thankful for all the resources and continued support from faculty and staff in the department who have helped me along the way. Vanderbilt is an outstanding research institution, and I am grateful to have completed my doctoral studies here.

To Jens Meiler, my PhD advisor, thank you for everything. Thank you for not only the opportunity to be a research student in your lab, but for actually supporting and guiding me for the past 6 years. This experience would not be the same without you. Since the day we met at the CSB gathering in November/December 2009, one of my favorite memories during my time here, thank you for always inspiring me and being committed to my success as a phD student. Thank you for pushing me outside of my comfort zone. Thank you for recognizing my talents and helping me to develop them. When I needed to push harder you always let me know, but it was in an energizing way, which left me excited to keep going. Thank you for developing my career interests and allowing me to pursue avenues to enhance these, such as teaching opportunities on campus and attending conferences. Traveling together to my undergrad institution for my research presentation, another one of my favorite memories, was a two-fold amazing experience as an alum and also as a PhD student. I am grateful to be in a lab where we share scientific knowledge as well as fun times, and I enjoyed the lab outings, retreats, holiday socials, and lab gatherings at your house. Thank

you for the fun stories you share and for the random chit-chats. Thank you for always reminding me to “push forward” towards the goal. Jens, your continued insight and encouragement have left a lasting imprint on me, and I will always appreciate having such a great PhD advisor.

To my committee members, Brian Bachmann, Tony Capra, and Mike Stone, thank you for guidance and input in my research as I matriculated through the years. Your help in scheduling meetings and presence at my exams has been greatly appreciated. Thank you for being on board to support my doctoral studies. I look forward to keeping you updated as I progress in my career. I would like to thank my past committee members Richard Armstrong (deceased) and Laura Mizoue for their investment in me as well. Thank you to the Director of Graduate Studies, Carmelo Rizzo, for readily proving information about the PhD journey.

To my parents, Warren and Talita Allison, and my sister Kelly, you are the reason I never gave up. I could not have done this without you and God in my corner on a daily basis. All my life you have been my number one supporters, and because of you I have been able to achieve every time. Thank you for instilling confidence in me from a young age. I always remember being told, from very young up to just a few hours ago, “You can do this Britt!”, and you have always being right. Your love, encouragement, support, insight, phone calls, guidance, and prayers have allowed me to overcome every challenge I encountered in my PhD journey. From the moment I was accepted to Vanderbilt, you continuously told me I would make it to the end, and I have. Along the way, you have provided unwavering support as I made it through difficult graduate classes, my pre-qualifying exam, my qualifying exam, stressful poster presentations and oral presentations in the department and at conferences, my independent research proposal exam, and now in defending my dissertation. And I know

this support will continue as I begin my journey as a post-doc, and then on to career positions. Thank you for always being invested in my dreams, and encouraging me to do whatever it takes to achieve them. I am who I am, because of who you are. Daddy, thank you for reminding me to keep "Faith, family, and focus" near to me, no matter what else is going on, and that with faith and God by my side, I have what it takes to press forward. Thank you reminding me to recognize all the family members who are invested in my success, especially my grandparents who send love and prayers my way daily. And thank you for always being a source of wisdom and kindness, whenever I need to refocus my balance and get back on track. Momma, thank you continuously telling me "You are closer than you think". Often times, the goal seemed so far away, but you always readjusted my thinking into positive thoughts so that I could see my little milestones along the way. Thank you for reminding me that our family background is strong, therefore I am strong too. And thank you for the long phone conversations about life, which I always needed as a good distraction. Kelly, my sister and best friend, thank you for always telling me to "Stop stressing you want to be the best, but KNOW you will do your best, and it will be great". This has often carried me as I strive to be the best graduate student I can be. Thank you for always loving and supporting me and listening to my random stories and offering thoughtful advice, even when you have a lot on your plate too. It means so much. And thank you for all the fun we have and inside jokes we share, whether at home or on the phone or via text message, these get me through on a daily basis. My family, I love you three very much and I am so thankful, and so blessed, to always have you in my corner and in my life. It makes me proud to know that you are proud of me.

On campus, I have received tremendous support from various people and institutes. Thank you to the Meiler Lab, my lab home since March 2010, for amazing science, valuable

research insight, and awkward science humor. A special thanks to Amanda Duran and Steffen Lindert for being supportive lab members as well as friends, and for the walks outside. Thank you to Heather Darling, Caitlin Harrison, and Xuan Zhang for being a great help in the wet lab. Thank you to the Vanderbilt Institute of Chemical Biology (VICB) for awarding me my first year fellowship and the training grant for my second and third years. I participated in many VICB events over the years, which were beneficial and enjoyable. Amanda Renick and Michelle Sulikowski provided much encouragement, help, and mentorship during these years. Thank you to the Center for Structural Biology (CSB) for providing many CSB associated resources that I use for my research and the administrative support team, especially Karen Davis. Thanks to the Biomolecular NMR facility and Markus Voehler for assistance in my research. Thank you to Sandra Ford in the chemistry department, for answering my questions and for always being a friendly face. Thank you to Dean Ruth Schemmer for the amazing career development workshops, which I often attend. Thank you to Dean Elizabeth Rapisarda for being the point of contact for my NSF-GRFP fellowship. Thank you to Dean Don Brunson for connecting me with many great minority graduate students on campus and for your work in recruiting minority students to pursue a PhD at Vanderbilt. Thank you to the Center for Teaching for hosting teaching workshops and for the college teaching course. Thank you to the Programs for Talented Youth for allowing me the opportunity to teach and share my research with 7th and 8th graders. Thank you to the Graduate Student Council for the fun social events. Thank you to the Office of Black Graduate and Professional Students (OBGAPS), Alliance for Cultural Diversity in Research, and Black Cultural Center, for all providing support and friendships in various ways as a black woman

pursuing a PhD. I am thankful to be on a campus where I could connect with people who share experiences similar to myself.

Off campus, I have received great support as well. Thank you to the National Science Foundation Graduate Research Fellowship Program for awarding me their prestigious fellowship, which supported me in my last three years and allowed me the opportunity to present my research at many conferences. Thank you to the Rosetta Commons for being a community of scientists to engage with and gain valuable insight on the computational side of my project. Thank you to the National Organization of Black Chemists and Chemical Engineers (NOBCChE), for being not only a scientific community but a community to share experiences. I am very thankful for all the connections and guidance I receive from being a part of NOBCChE.

Thank you to my undergraduate institution, Xavier University of Louisiana, especially the chemistry department, for showing me continued encouragement in my time here at Vanderbilt. To Galina Goloverda, my undergrad research advisor, thank you for being the first person to tell me “Brittany, you should be getting a phD” and explaining to me what it meant to pursue a phD in chemistry, and of course for many letters of recommendation. To Michelle Carter, thank you for suggesting Vanderbilt as an institution to pursue doctoral studies.

To my friends and extended family, thank you for always being in my corner throughout this entire process. My friends, thank you for the laughs and for filling my time as a PhD student with fun memories. You all have helped me make it through in so many ways, and I am appreciative to have you in my life. My Xavier friends, especially Courtney Jason Givens, Ashley McNeil, Dominique Gabriel, Kasey Robinson, Keisha Mitchell, Geannette

Green, and Alycia Boutte, your love and amazing encouragement via various means of technology means so much to me. To Nik Richard, thank you for being my favorite person to gchat with and for the random yet meaningful conversations. My Vanderbilt/Nashville friends, especially Darla Washington Adams, Melissa Harrison Fortuna, Brittany Taylor, Brielle Habrin, and Holly Carrell Rachel, your presence here has gotten me through many hard days and I am thankful to have found genuine friendships during my graduate student career. Ashley, Courtney, and Darla, thank you for all the long phone conversations where you continuously told me remain optimistic, and thanks for the lunch dates and mall trips and random mini vacations to help keep me balanced. To all my friends, near and far, who have provided support and kind words through the years, I am very thankful. To my Aunty Tammy and Uncle David, thank you for inspiring me to overcome adversity. To my mentor Bennetta Horne, thank you for the messages which always make me smile. To my aunts, uncles, cousins, godmother, mentors, and extended family, thank you for the continued encouragement and telling me to hang in there. A very special thanks to my home church in New Orleans, St Peter Claver Catholic Church, for keeping me lifted in prayer as I embarked on my PhD journey.

To any and every person who has offered a smile, encouraging words, and fun along the way as I pursued my doctorate, thank you very much. I am deeply appreciative.

TABLE OF CONTENTS

Page

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1: Introduction

Background

Proteins that bind small molecules can act as therapeutics by sequestering ligands, stimulating signaling pathways, delivering other molecules to sites of action, and serving as *in vivo* diagnostics[1]. Improving biotechnological applications in these fields requires understanding the chemical basis of protein-small molecule interaction. Computational modeling serves as a test of how well these interactions are understood. Expansive protein structural information combined with powerful algorithms have allowed computational methods to grow significantly. Although the computational design of proteins that can bind to any ligand is not yet possible[2], recent successes small molecule-macromolecule design suggest that it is within reach (highlighted below). Protein-ligand interactions are difficult to model, and current methods still fail to predict optimal amino acid identities even in the first shell around the ligand[3]. Correct side chain identity and positions play just as an important role as correct ligand positions when modeling protein-small molecule interactions. I hypothesize that an iterative protocol that extensively searches the protein sequence space, side chain conformation, and ligand positions, along with an updated scoring function, will overcome this challenge. Creating new interfaces or even modifying existing ones requires computational tools that sample and select native-like interactions. Even with current challenges, rational protein optimization and *de novo* design hold much potential for the future of protein bioengineering.

The objective of this thesis proposal was to develop a computational protocol to design a ligand binding pocket within $(\beta\alpha)_8$ barrel proteins[4] for a target molecule. A library of 10,000 small molecules is available to us through the Vanderbilt High Throughput Screening Facility (http://www.vanderbilt.edu/hts/). Imidazole glycerol phosphate synthase (HisF, PDB 1THF)[5] provides the $(\beta\alpha)_8$ barrel protein scaffold that will be screened against the small molecule library. The computational design program RosettaLigand[6,7] of the Rosetta[8,9] modeling suite will be used to generate models, and the best scoring models are chosen for successive rounds of dock and design. This iterative approach builds upon the best models round after round, and optimizes the protein-ligand interface by searching and selecting the best residues for tight binding. Previous studies have shown the importance of experimental and structural validation for computational design methodologies[2]. This serves to not only verify predicted models, but to also test the current scoring function and sampling efficiency during computational calculations.

Accuracy of the computational models will be assessed through experimental characterization of protein variants, each optimized to bind one small molecule out of the library of screened compounds. Nuclear magnetic resonance (NMR)- based screening experiments allow detection of strong to weak binding of the target, determination of binding affinities, and verification of the binding site at atomic-level detail[10,11]. This approach creates a detailed map of the designed interfaces and captures how binding is affected by the chemical environment. The experimentally-determined binding affinities will be compared to those predicted by RosettaLigand, providing feedback on the accuracy of the energy function and sampling efficiency. Caveats in the programs are then addressed. Once established, this protocol can be applied to strategies in creating novel therapeutic

proteins and/or proteins with biotechnological capabilities. Applications include: developing recombinant proteins against disease mechanisms that proceed by binding or not binding a small molecule, design of biocatalysts, design of biosensors, modulating disease pathways, and development of proteins that inhibit enzymes.

Thesis Focus

The focus of my thesis has been to design proteins that bind small molecules. I approached this goal via three specific aims: (**Aim 1**) Identify small molecule scaffolds with intrinsic HisF Affinity (1a) Create HisF cysteine-less variant protein, and screen for scaffolds with intrinsic HisF affinity (1b) Structural verification and thermodynamic characterization of protein-ligand interaction as starting point for Aim IIb (**Aim 2**) Computational Design of Protein-Small Molecule Interfaces (2a) Develop and benchmark a protocol to simultaneously dock ligands and design the protein binding interface (2b) Design novel protein-ligand interfaces from a library of 10,000 small molecules (**Aim 3**) Experimentally Verify Subset of Designed Protein-Ligand Interfaces (3a) Express and purify variant proteins, ^{15}N-labeled for HMQC NMR titration (3b) Structural verification and thermodynamic characterization of protein-ligand interaction. Aims 1 and 2 have been completed, and Aim 3 has one successful design, with others in progress. Research presented encompasses significance and holds merit for many reasons, which have been outlined below.

Significance

Protein-based therapeutics play an important part in today's medical society: They can serve to alter enzymatic or regulatory activity, to target a special activity, as a vaccine, or even in diagnostics[1]. As of 2008, over 130 therapeutic proteins had been approved for

use in humans for treatment of more than 30 different diseases[1]. The market for clinical protein therapeutics reached about $94 billion in 2010[12]. US biologic sales reached $63 billion in 2012[13]. Therapeutic enzyme sales reached $1.4 billion in sales in 2012[13]. As of 2008, biopharmaceuticals in the USA account for one in four submissions for Food and Drug Administration (FDA) approval[14]. Protein-small molecule interactions are central to many biological processes including enzymatic catalysis, receptor-small molecule signaling, transporter selectivity, modulating ligand activity, and regulating homeostasis[14]. The ability to manipulate these functions would be a biotechnological advance. As of 2009, more than 50 engineered protein scaffolds have been described[15]. Although most approved biologics are antibodies, as of 2013, over 50 non-antibody 'alternative scaffolds' have been proposed as possible drug candidates[16]. Computationally engineering novel interfaces between proteins and small molecules holds great value in biotechnology and serves as a stringent test of understanding of the chemical basis of molecular recognition. Thus, computational design of protein-ligand interfaces represents an important step towards the development of novel therapeutics.

Successes in computational design highlight the power and potential of computational methods applied to biotechnological applications: Even with much to achieve, many computational milestones have been reached in recent years. Here I present just a few examples, highlighting the breadth of these achievements. In 2016, Huang *et al de novo* designed a four-fold symmetric TIM-barrel protein[17]. In 2015, Huang *et al* designed a green fluorescent protein to be a self-reporting biosensor[18]. In 2014, Penchovsky described the methods for designing small molecule-sensing ribozymes[19]. In 2014, Piece *et al* identified multiple mutations that improved the affinity and specificity of a therapeutic T-cell

receptor[20]. In 2013 Procko *et al* describe the design of a protein that binds the active site of a lysozyme protein and inhibits the enzyme[21]. In 2013, Bjelic *et al* designed proteins to catalyze the Morita–Baylis–Hillman reaction, which forms a carbon–carbon bond between the α-carbon of a conjugated carbonyl compound and a carbon electrophile[22]. In 2013, Mills *et al* designed an amino acid dependent metalloprotein[23]. In 2011 Fleishman *et al* designed a protein to bind a conserved region on the influenza hemagglutinin stem[24]. In 2010, Ashworth *et al* used optimized protein-DNA interactions to rival that of the wild-type by altering the cleavage specificity for three contiguous base pair substitutions[25]. In 2010 Baker *et al* created an enzyme catalyst for a biomolecular Diels-Alder reaction, which produces a cyclohexene[26]. In 2009, Chen *et al* redesigned an enzyme to bind substrates that previously had no specificity for the wild type enzyme[27]. In 2009, Keating *et al* created a computational framework for designing protein-interaction specificity, and used this framework to design bZIP-binding peptides[28]. In 2008, Wand *et al* designed a di-metal metalloprotein to bind zinc[29]. In 2008, Baker *et al* designed a biocatalyst to carry out the Kemp elimination, deprotonation of a carbon by a base[30]. In 2008, Jiang *et al* described the *de novo* design of retro-aldol enzymes[31]. In 2006 Sood and Baker developed a computational approach to increase the affinity of peptides that bind to proteins[32]. In 2007 Humphris and Kortemme published a 'multi-constraint' design protocol to optimize protein-protein interactions[33]. In 2003, Kortemme and Morozov describe a hydrogen binding potential to improve specificity in protein-protein complexes[34]. Computational design accomplishments demonstrate the potential of computational modeling and contribute to the growing body of understanding how molecules recognize one another.

Computational design of protein-ligand interfaces is not a solved problem: The ultimate goal of automated design of binding to any target is still out of reach[2]. The problem is complicated by the many degrees of freedom by the ligand and its placement in the binding pocket[35]. The problem is further complicated by the program's ability (or inability) to recognize strong as well as weak binding interactions[35]. Reported successes of designed receptors and their novel binding ligands could not be experimentally reproduced[2]. Other studies have investigated whether protein-ligand binding and enzyme active sites could be predicted by ligand-binding affinity rather than structural stability[36]. Two successfully designed proteins do bind a rigid steroid hormone, but also details the other 15 designs which were computationally favorable but showed no experimental indication of binding[37]. Computational enzyme design, which tackles a similar problem, has seen some successes (a few highlighted above), but turnover rates are minimal compared to wild type[35]. Among the many unsuccessful attempts, the computational successes, especially those involving small molecule – macromolecule recognition, demonstrate that protein-ligand interface design is an achievable goal. My research conducted over the past six years contributes to the evolving body of knowledge surrounding interface design, and will be applied to future projects dedicated to making the goal a reality.

Ligand flexibility as well as side chain rotamers must be sampled to create an optimal binding interface: Computational protein design seeks to identify amino acid sequences that are compatible with a given three dimensional (3D) protein structure. Specifically for interface design, structure coordinates of the protein scaffold and ligand are input, and the design algorithm proceeds through iterative rounds of sequence-conformational searching, followed by evaluation of the resulting designed sequences. This requires a search

algorithm that can rapidly sample the vast sequence and conformational space, and a scoring function that can identify low energy designs. To reduce the complexity of the search space, discrete conformations of the side chains (rotamers) are sampled during design[6,7]. Knowledge-based potentials that rely on statistical parameters derived from databases of known protein properties are used to increase the accuracy of scoring functions[6]. A large number of docking algorithms have been developed based on a variety of search algorithms, all which seek to identify the lowest free energy pose of the ligand in the protein binding site[38]. Successful applications have been described with various algorithms, but frequently the protein flexibility is not taken into account[6]. Ligand flexibility plays a crucial part in protein-ligand interactions and must be included to accurately model the interface at atomic level detail[6,7]. Ligand flexibility is treated similarly to side chain conformations, where pre-generated discrete ligand conformations are all docked into the protein. Allowing side chain and ligand flexibility more closely mimics a native-like protein-ligand complex and enhances the chance of approaching the lowest energy conformation possible.

RosettaLigand predicts protein-small molecule interactions: Rosetta, a protein modeling software suite for protein structure prediction and design[9], has been successfully used to tackle a number of computational projects involving macromolecule-small molecule interactions. Since its development, Rosetta has been expanded to other macromolecule modeling systems, available as a versatile, rapidly developing set of tools applicable to many challenges[39]. Many of the successes highlighted above are Rosetta achievements. Rosetta seeks to find the lowest energy conformation of a model by combining discrete side chain conformation (rotamer) optimization with Monte Carlo

minimization[9]. This includes sampling random perturbations of the backbone torsion angles, rigid body degrees of freedom, and rotamer conformations, followed by an all-over local minimization to resolve clashes[9]. The energy function that Rosetta uses to discriminate between native-like and non-native-like atom arrangements includes a van der Waals-like attractive and repulsive potential, solvation term, hydrogen bonding potential, electrostatics potential, rotamer probability, and protein backbone angle probabilities[6]. RosettaLigand is an application within Rosetta that was originally developed to dock small molecules into a protein with full protein and ligand flexibility[6,7]. RosettaLigand does indeed recover the native pose of the experimentally determined crystal structure; the best scoring models are within 2 Å root mean square deviation (RMSD) compared to the crystal structure[6,7]. This is significant because for this methodology to be valid, there must be confidence that the lowest energy models are sampled, accepted, and correctly scored. Using the 3D coordinates of the protein and ligand as input files, the RosettaLigand protocol involves optimized placement of the ligand, optimized positioning of the surrounding residue side chains, and a minimization to resolve clashes. Protein sequence optimization can be included in the protocol if needed[40]. RosettaLigand allows for protein backbone flexibility, side chain rotamer searching, and full ligand flexibility, all of which are necessary for accurately modeling the interface[6,7]. For each model, RosettaLigand calculates an 'interface energy' as the total score of the protein-ligand complex minus the total score of the apo-protein[7]. Published literature describes the RosettaLigand method, protocol options, and tips for model analysis[41].

(βα)$_8$ barrel proteins provide a good scaffold for design: (βα)$_8$ barrels, also known as TIM-barrels, are frequently observed among soluble enzymes in metabolic pathways[42]. The

tertiary structure traditionally consists of 8 repeating (βα) units folded into a barrel structure[43]. TIM-barrels are widely seen in the Protein Data Bank (PDB); a ScopTree search for 'TIM beta-alpha barrel' in the PDB returns more than 2000 crystal structures. These are often crystallized with ligands, noting their diverse sequences, range of reaction mechanisms, and affinities for different ligands, despite all maintaining the same fold[4]. Their functions are widespread, and there are instances of TIM-barrels used and/or investigated as protein therapeutics. GlcCerase has been used to treat Gaucher disease[44]. Heparanase is being investigated as a therapeutic target for antitumor therapy[45]. On the agricultural side, design of chitinases are being investigated as an approach to bio-pesticides[46]. The TIM-barrel chosen for this proposal, imidazole glycerol phosphate synthase (HisF), catalyzes a ring closure in the histidine biosynthesis pathway[42]. Being highly characterized, HisF provides an advantageous scaffold for computational and experimental studies. There are multiple structures determined by x-ray crystallography with high resolution, providing various starting points as the input structure for computational studies. I utilized PDB 1THF (1.45 Å)[5]. HisF is native to a thermophile, therefore highly stable and tolerant to mutation, and has been previously used for design, providing a stable protein for experimental studies[47,48]. Additionally, using a protein scaffold with an assigned 2D spectrum is advantageous because of the difficulty that comes with resonance assignment and/or crystallization. The literature HisF 2D spectrum provides the starting point for the experimental validation[49]. HisF has been crystallized in complex with its native substrate, prfar (PDB 1OX5), by inactivating the protein's cyclase signaling residue[50]. This provides insight into the binding motif of the native substrate and possibly about other naively binding ligands. The prfar active site sits at the top of the $(\beta\alpha)_8$

barrel (as most TIM-barrels), with prfar binding in a deep cleft stretching across the top of the barrel[50]. There is a cysteine (Cys9) in the HisF binding pocket, therefore a HisF cysteine-less variant protein will be created, expressed, and purified. Replacing the cysteine with serine (C9S_HisF) gives confidence that small molecule binding is not due to covalent interactions with the cysteine. The interactions for binding should be driven by non-covalent interactions, including hydrogen bonding, dipole dipole interactions, electrostatics, van der Waals, hydrophobic effects, and geometric/shape complementarity. In our lab, C9S_HisF purifications yield almost 1 g of pure protein from a 12-L growth. Selecting HisF as the protein scaffold for this proposal provides the computational reliability and experimental stability needed for this research project.

^{15}N-HMQC NMR allows for detection of small conformational changes induced by ligand binding: Small molecule binding, predicted to be in the milli to sub-micromolar range, must be detectable to verify that the ligand is indeed bound in the protein-ligand binding pocket. Nuclear magnetic resonance (NMR) experiments provide an alternative for protein-ligand characterization when crystallization (the traditional method) is difficult. NMR spectroscopy has emerged as a reliable tool in identifying binding, elucidating the interacting residues, and calculating the binding affinity[51,52]. Binding can be detected by two-dimensional ^{15}N-heteronuclear multiple quantum correlation NMR (2-D ^{15}N-HMQC NMR)[10]. Observation of ^{15}N- or ^{1}H-amide chemical shift peak changes in the protein-ligand complex spectrum when compared to the apo protein spectrum, indicates a change in the environment of that ^{15}N- or ^{1}H-amide pair, likely due to ligand binding. An advantage of ^{15}N-HMQC NMR is that quality spectra can be rapidly obtained[10,11], in ~30 minutes. As noted above, the assigned 2D spectrum allows one to correlate the peak shifts to the actual

residue in the protein. The ability to verify even weak binding is crucial for validation of interface design studies.

Design of proteins to bind biologically relevant small molecules can be applied to disease treatment strategies: Receptor-small molecule signaling regulates biological pathways, and disease may occur when this signaling is disrupted. The literature contains reviews and examples of computational methods as a tool in combatting disease[53]. In diseases that progress by proliferation of damaged cells, such as cancer, an approach to treatment could be to stop a signaling pathway in the damaged cells. Knudsen et al describe small molecule depletion as a strategy in prostate cancer treatment[54]. Hao et al used experimental methods combined with computational methods to identify a high-affinity ligand for CRIP1 (cysteine-rich intestinal protein 1 has been identified as a novel marker for early detection of cancers)[55]. Isitivan et al describe the successful application of their methods to design a bioactive peptide analogue with cytotoxic effects on tumor cells only[56]. Bellows et al describe a framework for the discovery of entry inhibitors for HIV-1[57]. Sievers et al show that computer-aided, structure-based design can yield specific peptide inhibitors of amyloid formation, a strategy against Alzheimer's disease[58]. Development of proteins as therapeutics is still only in its infancy, yet more than 100 different proteins/peptides have been approved for clinical use by the FDA, with many more in the development stage[1]. Based on the several thousand different genes in the human genome, bioengineered recombinant proteins represent a class with the potential to grow into one of the leading areas of disease treatment[1]. Even with current challenges, recombinant proteins comprise the majority of approved FDA biotechnology treatments[1]. Reliable computational methods

would allow therapeutic development to expand, due combinations that could eliminated and/or pushed forward for experimental testing.

Innovation

Research presented contains various novel aspects. On the computational side, RosettaLigand was tested many ways in its ability to recapture sidechains necessary for protein-ligand interaction. My RosettaLigand manuscript was the first to benchmark RosettaLigand with the inclusion of sequence design[40]. On the experimental side, NMR-obtained data was used as the basis for a computational benchmark, and HisF was used as a scaffold in identifying naïve binders. I am the first to conduct a RosettaLigand benchmark using only NMR experimental data. Because interface design presents such a complex challenge, my project functions as a 'stepping stone' in elucidating where RosettaLigand can be improved. It is commonly accepted in the directed evolution field that it is easier to begin engineering when some desired activity is present[59], which I have incorporated into my approach. The experimentally determined naïve binders serve as the starting point for design, therefore both design successes and failures expand knowledge about the program. Hopefully, these methods and results will be applied in subsequent experiments which then contribute to the evolving nature of computational work. In various ways, my projects ask RosettaLigand to distinguish a binding ligand from non-binding ligands, and to recognize binding interactions. In order to understand its capabilities and limitations, I have systematically assessed the program's ability to identify interactions favorable for binding. Below I have expanded on the novel themes of my project, in a broad and narrow sense.

Engineered protein scaffolds are emerging as next-generation therapeutics:

Biotechnological advances such as recombinant proteins and protein engineering have paved the way for a field of study which focuses on introducing novel binding properties into naturally occurring proteins, termed 'engineered protein scaffolds'[60]. Ideal candidates for scaffold engineering include proteins that are soluble and thermostable[15]. Once the protein is chosen and a scaffold is identified, chemical diversity is introduced into the binding site by means of *in-vitro* mutagenesis techniques, creating a library of variant proteins to be tested for binding[61]. Applications for medical therapies include serving as agonists, antagonists, enzyme inhibitors, and antidotes[62]. Although only a few engineered protein scaffolds have advanced to clinical development, over 50 have been described, showing their promise as a future medical therapy[63].

Computational methods emerging as a tool to aid in high throughput screening: Drug discovery relies on high-throughput screening as one of the established methods in identifying lead compounds to pursue[64]. The strong increase in both the number of available compounds as well as molecular targets has created a massive database to screen, in some cases exceeding one million compounds[64]. The need for optimized, accelerated methods in screening has been pointed out in the literature[64]. With the growing database of experimental structures of ligands bound to proteins, *in silico* screening methods have emerged as a possible 'stepping stone' tool in the drug discovery process. Review articles highlight successes, such as the ability to recapture the native binding pose, but also point out caveats, such as scoring methods and the need for robust benchmark experiments[65]. An in-depth review provides insight into the challenges we still face with virtual screening, such as incorporation of protein flexibility/conformational changes, treatment of active site

water molecules, ligand flexibility/conformers, scoring functions that accurately assess interactions, and post-processing analysis[38]. Since RosettaLigand is a docking program, assessing the ability to distinguish interacting ligands from non-interacting ligands is of particular interest.

Computational benchmarks uncover the strengths/weaknesses of computational methods: Computational benchmarks allow one to assess the strengths/weaknesses of a computational method. A closer look into current virtual screening/docking methods reveals that although the end goal is the same, the approach varies among different programs[66]. Benchmark studies for small molecule docking programs in self-assessment and comparative assessments have shown the importance of such computational experiments[67]. Metrics to compare different programs reveal that no one computational program handles all complexes well[67]. Some programs handle certain ligand/protein systems better, some programs are better at docking ligands versus protein-ligand interface design, etc. Benchmarks enable the computational community to grow by exposing and addressing persistent problems/caveats within similar programs. Constant changes and updates to the program necessitate the need for new benchmarks, or even a repeat of previous ones. In 2015, Conchuir et al dedicated an entire manuscript to standardized benchmarks for Rosetta protocols[68]. The original RosettaLigand papers highlight that the program can recover the native ligand position and side chain residues of protein-ligand crystal structures[6,7]. Subsequent benchmarks expand upon these and test the capabilities of RosettaLigand in more 'real-world' scenario cases, such competing in challenges that assess many docking programs at once[69] and applying the program as a strategic tool to combat disease[70]. A blind evaluation of RosettaLigand, using private data

from a real drug discovery company, performed on average comparable with that of the best commercially available current small molecule docking programs[71]. Experiments have tested the program's ability in virtual high throughput screening, after optimizing the ligand placement step[72]. Specifically, I have tested RosettaLigand in recovering the protein-ligand interface residues, recovering the crucial interacting residues if mutated to alanine, and identifying the interacting residues based on experimental data. Specific to the research of my second manuscript, NMR-obtained data can expand the breadth of benchmarks, by including data that cannot or has not yet been obtained by crystallography.

CHAPTER 2: Computational Design of Protein-Small Molecule Interfaces

Allison, Brittany; Combs, Steven; DeLuca, Sam; Lemmon, Gordon; Mizoue, Laura; Meiler, Jens, "Computational Design of Protein- Small Molecule Interfaces". *Journal of Structural Biology* **2014,** *185 (2)***,** 193 – 202.

Contribution

I am the first author of this manuscript. I contributed the bulk of the: abstract, introduction, RosettaLigand computational setup and analysis, figures 1 – 5, tables 1 and 2, conclusion, and supplemental information. I also reviewed the parts that the other authors contributed.

Abstract

The computational design of proteins that bind small molecule ligands is one of the unsolved challenges in protein engineering. It is complicated by the relatively small size of the ligand which limits the number of intermolecular interactions. Furthermore, near-perfect geometries between interacting partners are required to achieve high binding affinities. For apolar, rigid small molecules the interactions are dominated by short-range van der Waals forces. As the number of polar groups in the ligand increases, hydrogen bonds, salt bridges, cation-π, and π-π interactions gain importance. These partial covalent interactions are longer ranged, and additionally, their strength depends on the environment (e.g. solvent exposure). To assess the current state of protein-small molecule interface design, we benchmark the popular computer algorithm Rosetta on a diverse set of

43 protein-ligand complexes. On average, we achieve sequence recoveries in the binding site of 59% when the ligand is allowed limited reorientation, and 48% when the ligand is allowed full reorientation. When simulating the redesign of a protein binding site, sequence recovery among residues that contribute most to binding was 52% when slight ligand reorientation was allowed, and 27% when full ligand reorientation was allowed. As expected, sequence recovery correlates with ligand displacement.

Introduction

Engineering protein-small molecule interactions is key for advancement of several grand challenges in computational biology. Protein-small molecule interactions are the basis for enzymatic catalysis, receptor-small molecule signaling, and transporter selectivity and are thus essential for carrying out biological processes and maintaining overall homeostasis in the body. Designed proteins that bind small molecule targets can act as therapeutics by sequestering ligands, stimulating or extinguishing signaling pathways, delivering other molecules to sites of action, and serving as *in vivo* diagnostics[1]. For example, small molecule depletion has been suggested as a strategy for treatment of prostate cancer[54], cocaine abuse[73], and bacterial infection[74] Proteins that bind small molecules also have applications in environmental chemistry and food chemistry as biosensors[75]. Thus, the ability to engineer highly precise and specific interactions at protein interfaces can serve in many capacities.

Computational design of protein-small molecule interfaces continues to present challenges. Although the creation of new enzymes is a landmark achievement in protein design[31],[30], [26],[76], the success rate is low and the designed proteins are poor catalysts compared to naturally-occurring enzymes. To help pinpoint the causes, a systematic study

was conducted introducing mutations into the active site of three designed retro-aldolases (RA34, RA45, and RA95) derived from the TIM-barrel scaffold IGPS. In RA34 and RA95, mutations that increase substrate binding affinity and thereby enzymatic activity involve increases in side chain volume and hydrophobicity, including G233F/I/V/Y in RA34[77] and T51Y, T83K, S110H, M180F and R182M in RA95[78]. In contrast, many improvements to the RA45 design arose from large-to-small mutations including W8A/T/V, F133L, V159C, and R182V/I[78]. In all cases, key functional groups that engage the ligand are introduced or removed. These observations indicate that neither the hydrophobic packing nor the positioning of substrate within the binding pocket were optimal in the initial designs. Similarly, a previously reported successful computational design of a protein-small molecule interface[79] did not withstand close examination[80].

Rosetta, a protein modeling software suite for protein structure prediction and design[9], has been successfully used to tackle a number of interface design problems. Some of these successes include creating novel enzymes[31], [30], [26], altering the specificity of protein-peptide[32], protein-DNA[25], and protein-protein interfaces[34]. Rosetta seeks to find the lowest energy conformation for a design by combining discrete side chain conformation (rotamer) optimization with Monte Carlo minimization[9]. This includes sampling random perturbations of the backbone torsion angles, rigid body degrees of freedom, and rotamer conformations, followed by an all-over local minimization to resolve clashes[9]. These methods enable much faster and larger exploration of sequence and conformational space compared to experimental methods such as phage display[81].
The energy function that Rosetta uses to discriminate between native-like and non-native-like atom arrangements includes a van der Waals-like attractive and repulsive potential,

solvation term, hydrogen bonding potential, electrostatics potential, rotamer probability, and (φ, ψ) angle probabilities in the protein backbone[6]. The total energy of the system is computed as a weighted sum of all interactions with weights optimized through a series of benchmarks. All energy functions are pairwise decomposable (i.e. they depend on no more than two interacting partners). This design of the energy function maximizes algorithm speed since interaction energies can be pre-computed and stored. However, it also limits the accuracy of the energy function, particularly electrostatic and partial covalent interactions which vary greatly in strength depending on the environment of the interacting partners. Experimental characterization of some of the best scoring designs is used to validate and improve the computational protocols. In this way, both design successes and failures help test and expand our understanding of the fundamental forces involved in molecular recognition.

RosettaLigand is an application within Rosetta that was originally developed to dock small molecules into a protein with full protein and ligand flexibility[6,7]. [82]. In these studies, we expand RosettaLigand to include amino acid optimization (design) at the protein-small molecule interface. Using the full-atom energy function and Monte Carlo minimization procedure, RosettaLigand optimizes the small molecule and protein side chain rotamers simultaneously[6]. RosettaLigand allows for protein backbone flexibility, side chain rotamer searching, and full ligand flexibility, all of which are necessary for accurately modeling the interface[6,7]. Figure 1 details each step of the ligand docking protocol. For each model, RosettaLigand calculates an 'interface energy' as the total score of the protein-ligand

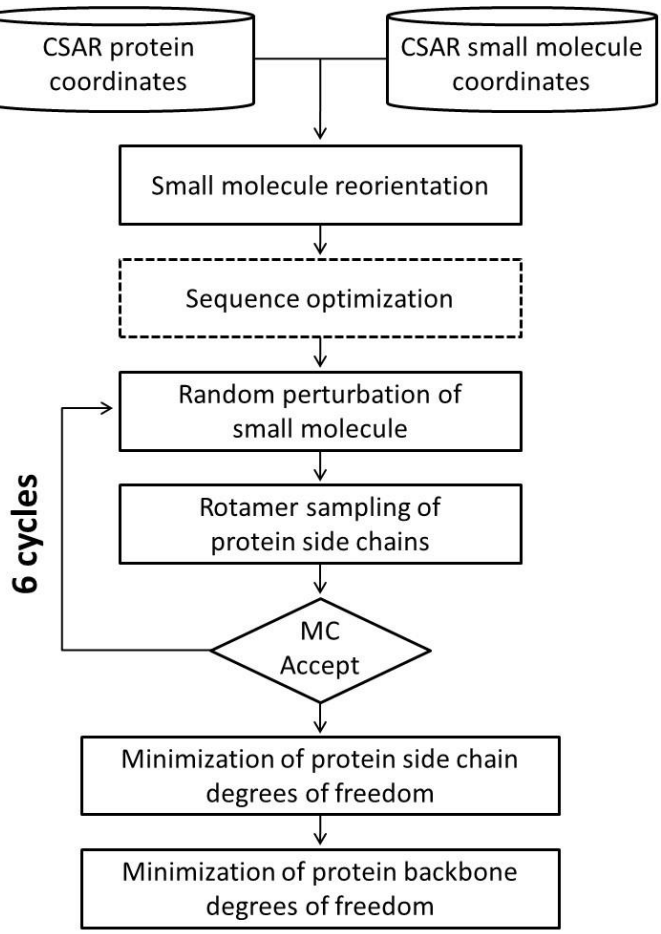


Figure 1: Flowchart of small molecule docking with design. The RosettaLigand protocol was modified to include interface design (dotted line box). From the input coordinates, the small molecule is allowed to rotate and translate before sequence optimization of nearby residues. After 6 cycles of small molecule perturbation, side chain rotamer sampling, and Monte Carlo (MC) minimization, a final gradient-based minimization of the protein is performed to resolve any clashes. (Reprinted from Journal of Structural Biology.)

complex minus the total score of the apo-protein [83]. The accuracy of models in terms of ligand placement is determined by computing the root-mean-square distance (RMSD) over all ligand atoms between model and co-crystal structure. RosettaLigand is the foundation[76] of a number of the successfully design enzymes[30,31,77], with the before-mentioned caveat that the computationally predicted residues are often sub-optimal even in the first shell surrounding the ligand. In order to understand its capabilities and limitations, the present

work systematically assesses RosettaLigand's ability to design protein-small molecule interfaces. This analysis is an important, and so far omitted, benchmark to identify design challenges that can currently be solved and to work towards improvements needed to achieve consistent success.

Recovering native protein-small molecule interfaces in sequence and conformation is a benchmark for designing novel interfaces. Creating new interfaces or even modifying existing ones requires computational tools that sample and select native-like interactions. In this study, we examine how RosettaLigand performs in sequence recovery within protein-small molecule interfaces while allowing for small molecule reorientation and side chain conformational changes. The benchmark consists of two parts. Part one tests overall sequence recovery when all residues within the protein-small molecule interface are allowed to change identity. Part two simulates a protein-small molecule design more closely by mutating up to five residues that contribute most to the interaction with the small molecule to Alanine. This effectively removes the binding site's memory of the native ligand. In the design experiment a scoring bonus is given to the starting sequence. These experiments test RosettaLigand's ability to distinguish between native and non-native binding interaction and whether RosettaLigand can identify key mutations needed to bind the small molecule while limiting the total number of mutations. The results illustrate the types of ligands that Rosetta handles best and provide insights into weaknesses where continued method development is required.

Results and Discussion

The setup of the experiments allows us to determine overall protein-ligand interface sequence recovery as well as an optimal strategy for re-designing proteins to recognize

different small molecules using a minimal set of mutations. For this purpose separate measures for sequence recovery among the residues critical for ligand binding are determined. We investigate how sequence recovery varies with ligand size, binding affinity, and RosettaLigand interface energy. We appreciate that sequence recovery measures have one critical limitation: they assume that the native protein-small molecule interface is optimal for tight small molecule binding, which is certainly incorrect. In result, if a position fails to recover to the native amino acid it can be because of an actual failure of the design algorithm or because the alternate amino acid is tolerated or even favorable in an actual protein-small molecule interface – a distinction that only the experiment can make. Therefore, we do not expect a 100% success rate for sequence recovery. This poses a dilemma for the development of protein design algorithms: at what point can we stop optimizing for increased sequence recovery? Ideally, one wants to capture native-like designs and interactions that would be seen in nature, but not to a point where the algorithm over-fits the designs. To circumvent part of this problem, we developed a Position-Specific Scoring Matrix (PSSM) recovery measure[84] which computes the fraction of residues that revert to an amino acid observed in evolution. PSSM recovery is a more robust measure of design success as it tolerates mutations that have been observed in evolution.

Experimental Setup: A set of 43 high resolution protein-small molecule crystal structures were selected from the Community Structure-Activity Resource (CSAR) database and used directly in testing. In practice, however, the task is often to redesign a binding site to recognize a (different) specific small molecule. In this setting, one is interested in identifying the minimal number of mutations needed to achieve the desired

functionality and avoiding additional mutations that provide little or no benefit. This can be achieved by including a 'favor native' residue bonus (FNRB) energy that must be overcome before a mutation is accepted.

The starting sequences in the CSAR benchmark set are already (close to) optimal for binding the target small molecule. Therefore, we created mutant proteins that are expected to have reduced or no binding to the target small molecule. First, the five residues that contribute most to small molecule binding according to the RosettaLigand energy function were determined and then sequentially mutated to Alanine. Next, these artificial mutants were employed to test RosettaLigand's ability to recognize sub-optimal interactions and replace them with those that are optimal for binding.

A total of four experiments were conducted: (1) re-designing the protein-small molecule interface in the native protein without reorienting the ligand (Design Native), (2) re-designing the protein-small molecule interface in the native protein with ligand reorientation (Dock/Design Native), (3) re-designing the protein-small molecule interface in the Alanine mutants using a FNRB without reorienting the ligand (Design Alanine Mutants), and (4) re-designing the protein-small molecule interface in the Alanine mutants using a FNRB and ligand reorientation (Dock/Design Alanine Mutants). The latter experiment tests if RosettaLigand can identify critical mutations and distinguish them from arbitrary sequence changes. For each experiment, 1000 models were generated, filtered by RosettaLigand interface energy, and the top 50 were selected for analysis.

Trends for sequence recovery across all four experiments: The ligand RMSD, sequence recovery, number of mutations, and interface energy from each experiment were averaged to identify trends across experiments (Table 1). As expected, comparison of the

Table 1

Table 1
Varying parameters of experiments and sequence recovery results.

	Design Native (1)	Dock/Design Native (2)	Design Alanine Mutants (3)	Dock/Design Alanine Mutants (4)
Ligand Translation (Å)[a]	0.1	2.0	0.1	2.0
Ligand Rotation (deg)[a]	2	360	2	360
Ligand RMSD (Å)	0.5 ± 0.4	2.2 ± 1.0	0.7 ± 0.6	2.4 ± 1.0
Sequence Recovery (%)	64.0 ± 12.4	36.6 ± 11.9	59.8 ± 14.2	48.7 ± 11.9
Number of Mutations	6.5 ± 2.5	11.5 ± 3.4	7.2 ± 2.6	9.3 ± 2.7
Interface Energy (REU)[b]	-19.0 ± 5.4	-15.1 ± 3.5	-17.5 ± 5.2	-14.1 ± 3.5

[a] Reorientation allowed from initial pose during docking
[b] REU – Rosetta Energy Units

(Reprinted from Journal of Structural Biology.)

Design (1, 3) vs. Dock/Design (2, 4) experiments shows that lower RMSDs and better interface scores are observed when an optimal ligand pose is inputted and allowed to move only slightly (0.1 Å translation, 2° rotation) versus when the ligand pose needs to be identified (2 Å translation, 360° rotation). The conformational space increases if the ligand is allowed to reorient and other binding poses with different sequences achieve favorable interface scores. Large changes in ligand position alter the interactions with the protein and encourage mutations. Unfortunately, other than extensive experimental studies, there is no way to test the plausibility of these alternative protein-small molecule interfaces. Applying a FNRB that favors retention of the native amino acid (3, 4) yields fewer mutations and higher sequence recovery only when the ligand is allowed full reorientation. The highest sequence recovery, lowest number of mutations, and best interface energy is observed in experiment (1) where the ligand is held in its approximate initial pose and no

Table 2

Table 2
Recovery of wild type (WT) residue from the Alanine mutants experiments

	Design Alanine Mutants (3)	Dock/Design Alanine Mutants (4)
Ligand RMSD (Å)	0.7 ± 0.6[a]	2.5 ± 1.0[a]
Sequence Recovery (%)	58.9 ± 14.4[a]	47.6 ± 11.8[a]
Alanine Mutations Recovered to WT (%)	51.7 ± 32.8	26.8 ± 20.5
Alanine Mutations Designed to Other Amino Acids (%)	26.5 ± 25.1	36.8 ± 17.1
Alanine Mutations Remaining Alanine (%)	21.8 ± 25.2	36.3 ± 22.4

[a] The slight differences in values compared to Table 1 are due to exclusion of data from the native complexes.

(Reprinted from Journal of Structural Biology.)

alanine mutations are introduced into the sequence. However, in the design of novel protein-small molecule interfaces, full ligand reorientation must be allowed in order for the ligand to search the entire binding pocket for optimal placement (4).

Trends in recovery of wild type (WT) residues across Alanine mutants experiments: Similar results occur when introducing Alanine mutations in the binding site. To maximize sequence recovery, these had to be converted to the native amino acid overcoming the FNRB. For each experiment, the ligand RMSD, sequence recovery, recovery of WT residue, retention of Alanine, and mutation to another residue were determined for each model and averaged (Table 2). Since both of these experiments include a FNRB, the only variable is whether or not the ligand was allowed full reorientation. The penalty for allowing full ligand movement is larger in recovering the specific WT residue (51.7% vs. 26.8%) than the penalty in overall sequence recovery (58.9% vs. 47.6%). This result was not surprising because in cases where the ligand position is not recovered there is a minimal chance that the correct residue will be selected. Considering that the chance of randomly selecting the

correct residue is 5% (1 out of 20), RosettaLigand's ability to recover 26.8% of WT residues under the most stringent conditions represents a significant improvement. Some designs have been selected to show the diversity of ligands that have high sequence and wild type recovery versus ones that have low recoveries (Figure 2).

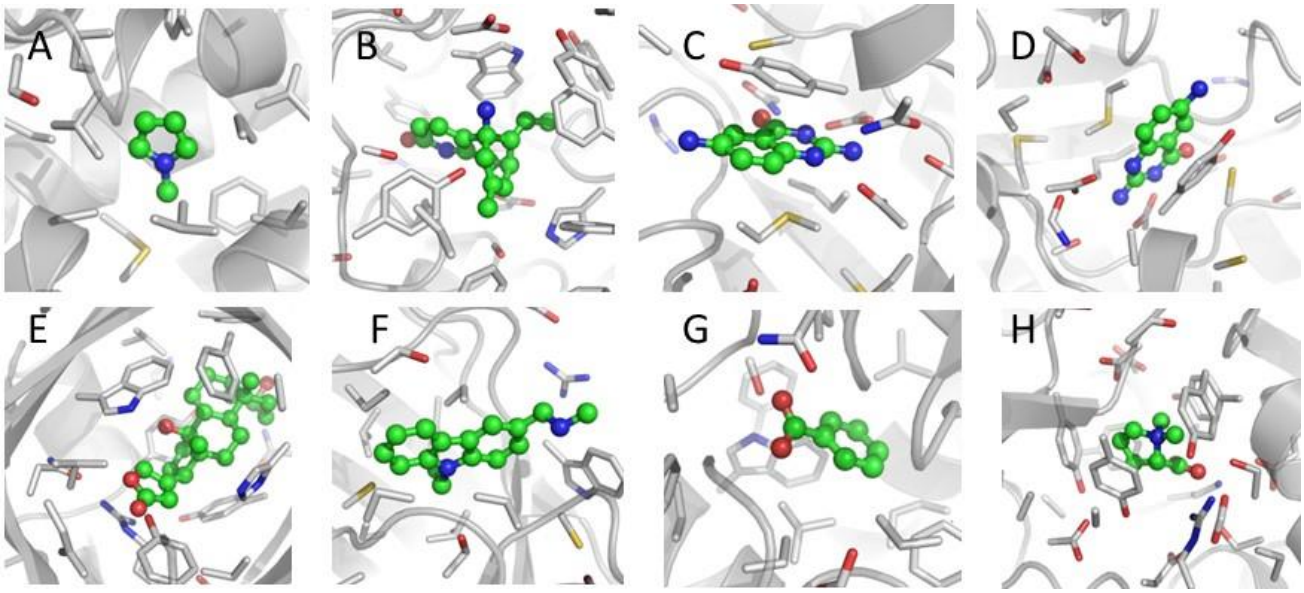


Figure 2: Examples of the best and worst designs from each experiment. Best designs from each experiment (A, B, C, D) are shown in contrast to the worst designs from each experiment (Panels E, F, G, H). For experiment 1 Design Native, a best design model had a sequence recovery of 90% (A), while a worst design model had a sequence recovery of 58% (E). For experiment 2 Dock/Design Native, a best design model had a sequence recovery of 80% (B), while a worst design model had a sequence recovery of 51% (F). For experiment 3 Design Alanine Mutants, a best design model had a wild type recovery of 94% (C), while a worst design model had a wild type recovery 8% (G). For experiment 4 Dock/Design Alanine Mutants, a best design model had a wild type recovery of 55% (D), while a worst design model had a wild type recovery 10% (H). (Reprinted from Journal of Structural Biology.)

Detailed analysis of the Dock/Design Alanine mutants experiment (4): Since the ultimate goal is to use RosettaLigand to design novel protein-small molecule interfaces, we took an in-depth look at the results from the experiment that most closely resembles this scenario. Sequence recovery was plotted against a number of variables to see if RosettaLigand performs better with different types of ligand and/or protein properties (Figure 3, Figure 18 (SF1)). As seen in previous results, sequence recovery decreases with

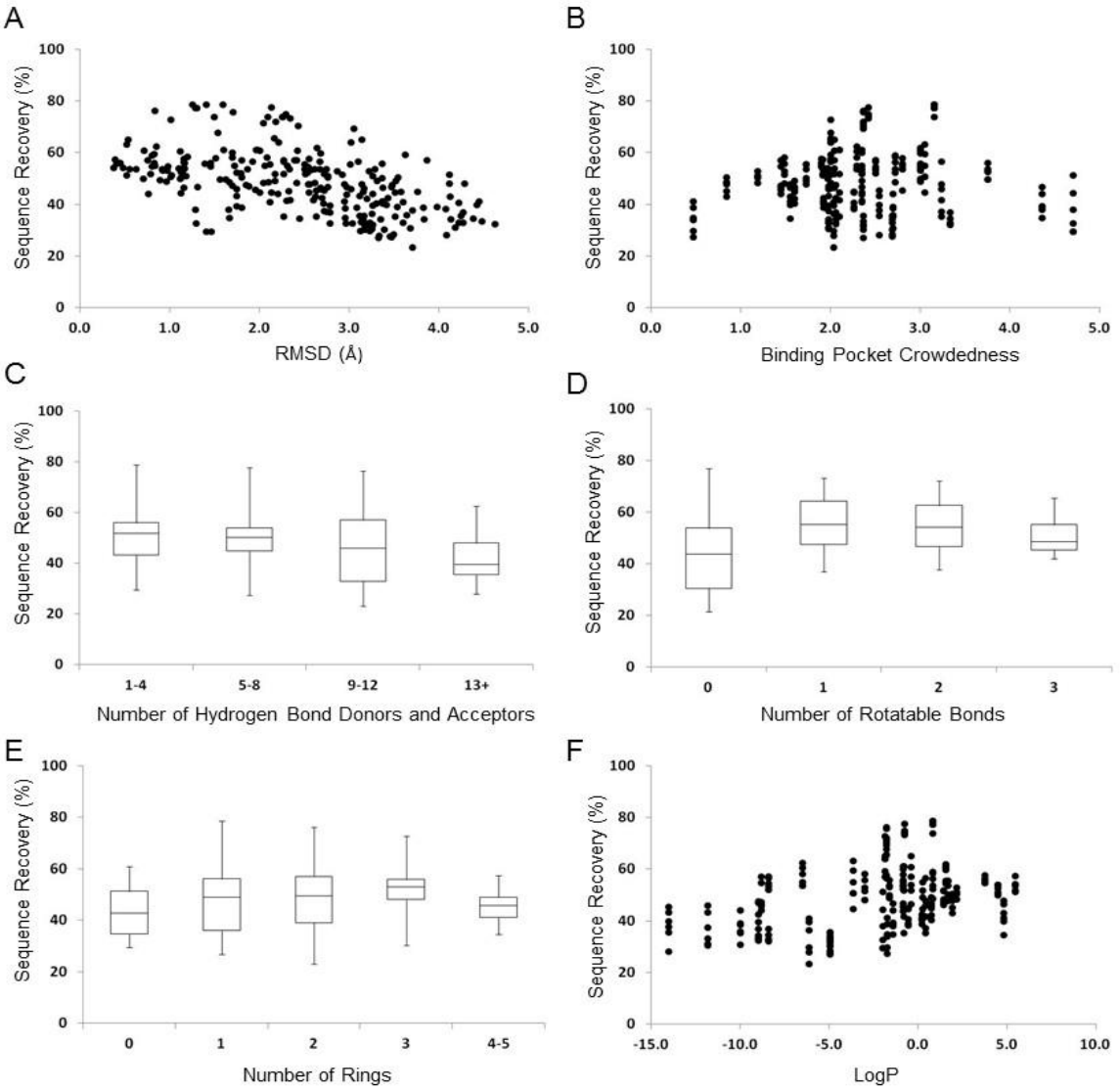


Figure 3: Sequence recovery from Dock/Design Alanine Mutants experiment. Increasing RMSD decreases the sequence recovery (A). Ligand atoms in contact with ~2-3 protein atoms show the best recoveries (B). Ligands having very many hydrogen bond donors and acceptors show a decrease in recovery (C). Ligands containing 1-3 rotatable bonds achieve the best recoveries (D). The number of rings in a ligand has slight correlation with recovery; having many rings decreases the range of recoveries (E). Amphipathic ligands have better recoveries than more hydrophilic and hydrophobic ligands (F). These results imply that RosettaLigand best recovers the protein-ligand interface when the ligand has a moderate number of hydrogen bond donors and acceptors, is amphipathic, and when the binding pocket is not too loose or too crowded. (Reprinted from Journal of Structural Biology.)

increasing RMSD (Figure 3A). Binding pocket crowdedness measures how tightly packed the ligand is in binding pocket, calculated as the number protein/ligand atom pairs within

3 Angstroms of each other, divided by the total number of ligand atoms. A high crowdedness indicates a ligand surrounded by protein contacts, whereas low crowdedness indicates that only a portion of the ligand is in contact with the protein. Sequence recovery was best when the ligand had 2-3 protein contacts per atom (Figure 3B). As crowdedness deviated from this range, recovery decreased, implying that RosettaLigand has difficulties in tightly packed as well as under-packed protein/small molecule interfaces. For the number of ligand hydrogen bond donors and acceptors, WT recovery remained consistent until there is a drop at 13+ donors/acceptors (Figure 3C). A complex hydrogen bonding network would be more difficult to recover than a simple network, so this was expected. For number of ligand rotatable bonds, ligands with 1, 2, or 3 rotatable bonds surprisingly have the best recoveries on average (Figure 3D). Ligands with rigid ring systems have a decreased range of recoveries, not the highest but also not the lowest values (Figure 3E). One would expect that since rings provide the ligand a more defined shape, it would be easier for RosettaLigand to identify the correct binding pose. LogP, a measure of lipophilicity comparing the concentration of ligand in octanol vs water, shows the best recoveries for ligands with a logP around zero (between ~-2.5 and 2.5, Figure 3F). As the ligand becomes more hydrophilic, sequence recovery decreases. Aside from a few outliers, as the number of ligand atoms increases, recovery decreases (Figure 18A (SF1) A). Our interpretation is that larger ligands have fewer well-defined contacts that are more difficult to recover. Surprisingly, the number of residues considered for design has little impact on sequence recovery; one may expect that more residues in the binding pocket would decrease recovery, but this was not the case (Figure 18 (SF1) B). Binding affinity showed little effect on recovering the interface (Figure 18 (SF1) C). There appears to be a drop in

recovery for very tight binders, however there are few of these complexes to begin with. Binding affinity normalized by ligand molecular weight does not influence recovery (Figure 18 (SF1) D). Topological polar surface area (Figure 18 (SF1) E) and van der Waals surface area (Figure 18 (SF1) F) both show the same trend; as surface area increases, maximum recovery decreases. This was not surprising, considering that surface area and number of ligand atoms correlate with each other. Ligand interface energy correlates little with sequence recovery (Figure 18 (SF1) G) even if the interface energies were normalized by small molecule molecular weight (Figure 18 (SF1) H). Taken together, these results suggest that RosettaLigand is biased towards non-polar or slightly polar ligands for achieving maximum recovery. For moderately sized ligands, the interface was recovered better than larger ligands. Interfaces with many hydrogen bond donors and acceptors were difficult to recover. Overall, there seems to be a preference for ligands that are moderately sized and not too polar, because interfaces containing ligands with these properties are recovered the best. However, correlations were generally weak with many outliers confirming that there is no single parameter that identifies an interface that is easier to design.

The same parameters were analyzed for recovery of Alanine mutants to the wild type residue (Figure 4 and Figure 19 (SF2)). Trends are more difficult to discern because in sequence recovery varies in a larger range. Not surprisingly, most of the complexes with high RMSDs had low recovery rates, while those with low RMSDs displayed a wider range of recovery rates, from very high to very low (Figure 4A). Similar to sequence recovery, ligands with 2-3 protein contacts per atom showed the best WT recoveries (Figure 4B). The number of ligand hydrogen bond donors and acceptors shows a trend, where recovery drops with 9+ donors/acceptors (Figure 4C). It is expected that under the more stringent

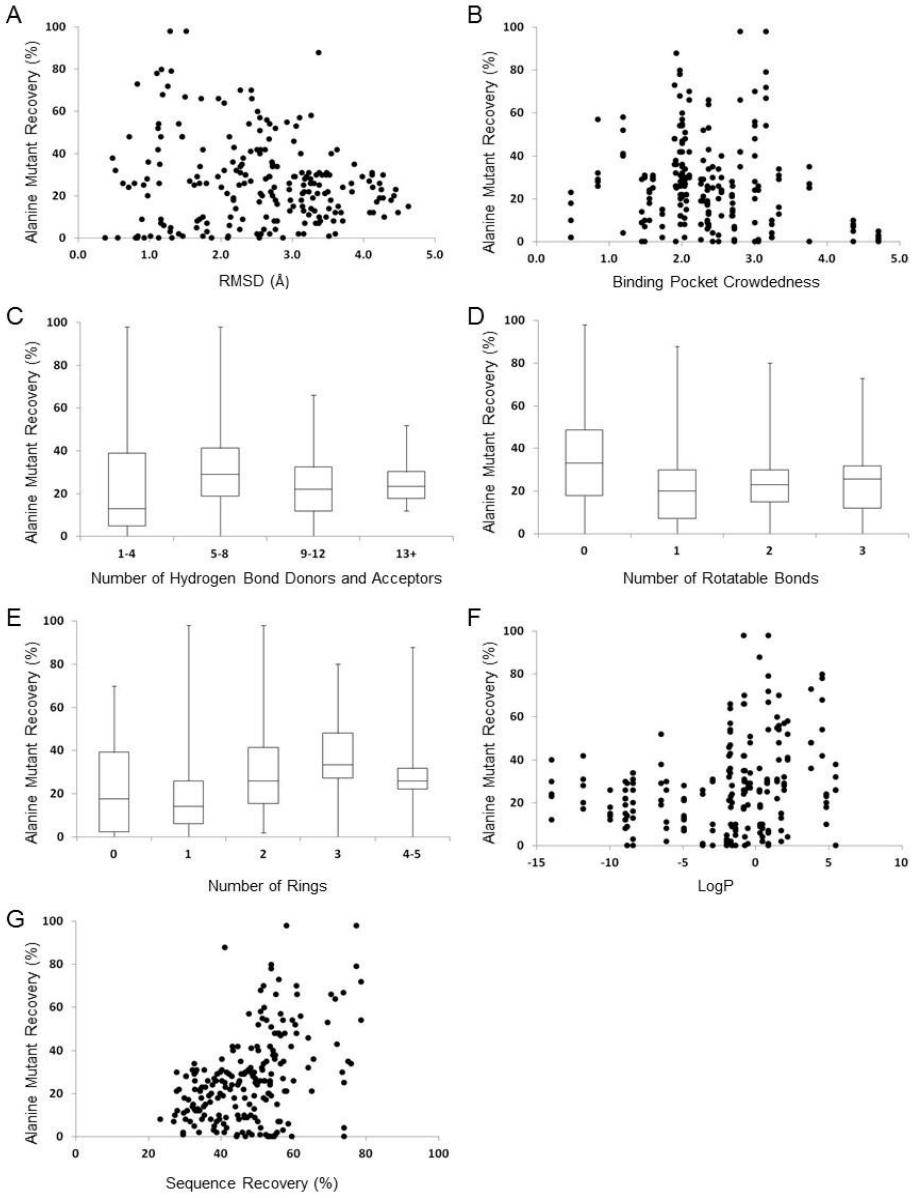


Figure 4: Alanine mutant to Wild Type recovery from Dock/Design Alanine Mutants experiment. Most complexes with high RMSDs had low Alanine to WT recovery, whereas complexes with low RMSDs had a range of WT recovery (A). Ligand atoms in contact with ~2-3 protein atoms show the best recoveries (B). WT recovery drops when the ligand has 9 or more hydrogen bond donors and acceptors (C). As the number of ligand rotatable bonds increases, the maximum WT recovery decreases (D). Ligands with zero rings have the lowest recoveries; ligands with 3 rings, although they do not reach maximum recovery, the average is the highest (E). Amphipathic ligands have better recoveries than more hydrophilic ligands (F). Positive correlation seen between sequence recovery and the recovery of the alanine mutants to WT (G). These results imply that RosettaLigand best recovers the WT residues when the ligand has less than 8 hydrogen bond donors and acceptors, contains fewer rotatable bonds, contains more rings, is amphipathic, and when the binding pocket is not too loose or too crowded. (Reprinted from Journal of Structural Biology.)

condition of only measuring alanine to WT recovery, a simpler hydrogen bonding network

is easier to recover, also compared with overall sequence recovery which dropped after 13+ donors/acceptors. As the number of ligand rotatable bond increases, the maximum recovery decreases (Figure 4D). Ligands containing at least one ring have better recovery than ligands without rings (Figure 4E). Ligands with 1 or 2 rings reach the highest maximum, while ligands with 3 rings have the best average. This confirms that protein/ligand interfaces for rigid ligands are easier to design for RosettaLigand. As seen with sequence recovery, amphipathic ligands with a logP between -2.5 and 2.5 have the highest WT recoveries (Figure 4F). Hydrophobic ligands perform well, and hydrophilic ligands worst. Comparing overall sequence recovery to WT sequence recovery shows a positive correlation (Figure 4G), which demonstrates as expected that the complexes that recovered the most of the interface had the best chance of recovering the specific WT residues; instances where interface sequence recovery was very low resulted in very low WT recovery as well. This implies that Rosetta is able to discern which protein residues are most critical for ligand binding when most of the non-critical residues are correct as well. Other than a few outliers, increasing the number of ligand atoms decreases the maximum recovery (Figure 19 (SF2) A). The number of residues considered for design (Figure 19 (SF2) B) and the binding affinity (Figure 19 (SF2) C) had little impact on WT recovery. Lower normalized binding affinities had a better chance of recovering the WT residues (Figure 19 (SF2) D). As seen with sequence recovery, ligands with a high topological polar surface area have decreased recoveries (Figure 19 (SF2) E). Van der Waals surface area (Figure 19 (SF2) F), Rosetta interface energy (Figure 19 (SF2) G), and normalized Rosetta interface energy (Figure 19 (SF2) H) show no effect on WT recovery. Overall, the trends for WT recovery correlate with sequence recovery, which is expected. Rosetta performed best

when the ligand contains a small/moderate number of hydrogen bond donors and acceptors, at least one ring, low number of rotatable bonds, moderately sized, and amphipathic.

PSSM recovery results: The Position-Specific Scoring Matrix (PSSM) identifies amino acid mutations that are tolerated in homologous proteins. Thus, evaluating protein designs based on PSSM score provides a more robust assessment of favorable mutations than sequence recovery alone[84]. By tolerating amino acids seen in evolution a more robust judgment of RosettaLigand's ability to capture biological sequences is made. Evolutionarily advantageous mutations may contribute to the interface in ways other than stability or low energy, and it is important to consider these mutations as well. In addition to designing in residues that contribute to binding and promote strong interactions, we also want to design an interface that is native-like. Is the most optimal protein-ligand interface, one that could be seen in nature, the lowest in energy? This is a fair question to consider, and one that can be assessed with PSSM recovery. PSSM recovery is expected to be higher than sequence recovery. However, it also has its limitations: 1) not all amino acids tolerated or beneficial will have been sampled in evolution, 2) the space of known related protein sequences might be incomplete, and 3) some mutation seen in other proteins might alter specificity and are not tolerated for the particular small molecule in the benchmark. The average sequence recovery and average PSSM recovery of every structure in each dataset are plotted in Figure 5. As noted earlier, applying a FNRB improves the sequence recovery, which is helpful because recovery is more difficult in the experiments that include mutated alanine residues. It was expected that allowing full ligand movement would decrease the sequence recovery compared to allowing slight ligand movement. Because PSSM views a

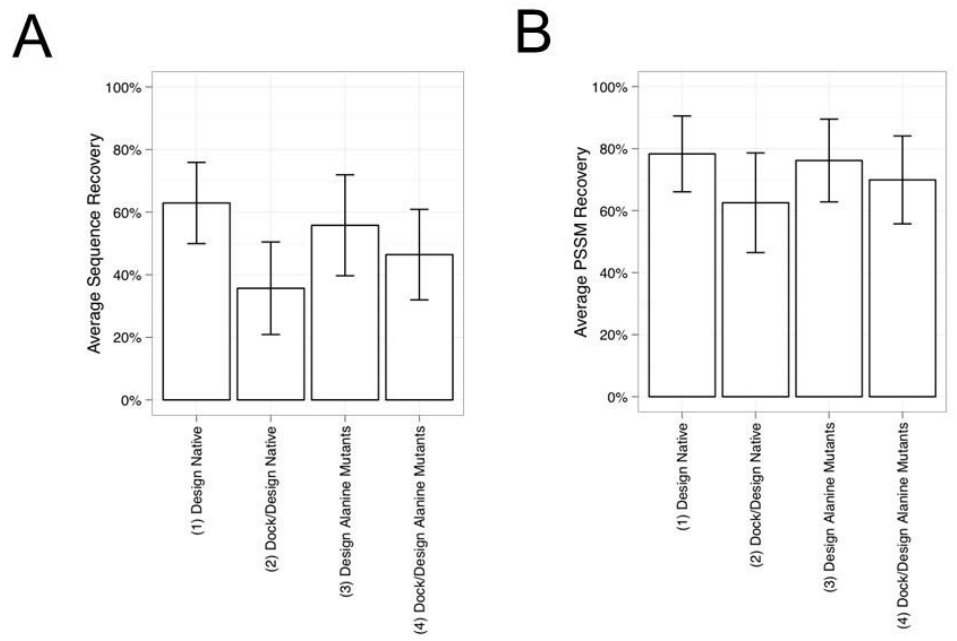


Figure 5: Sequence and PSSM recovery of the experiments. The sequence recovery for each experiment was calculated (A). The PSSM recovery for experiment was calculated (B). For both plots, error bars are 1 standard deviation from the mean. Sequence recovery, although reported earlier, is included in this form for a side-by-side comparison to PSSM recovery. Applying a bonus to the native sequence improves the sequence when the ligand is allowed full reorientation. Allowing the ligand full reorientation decreases the sequence recovery when compared to its similar experiment that only allows slight ligand reorientation. (Reprinted from Journal of Structural Biology.)

mutation favorably if the new amino acid is frequently seen at that position, PSSM recovery will always be higher than sequence recovery, where all mutations are counted as incorrect. The percentage of PSSM recovery was also computed on a per residue basis (Figure 6A). Glycine, Alanine, Leucine, Valine, and Threonine were frequently recovered, while Tryptophan and Glutamine were often mutated. Cysteine was omitted as it was not included during design. All four data sets exhibit similar biases in terms of PSSM recovery. The change in sequence composition provides information about overall biases in sequence design. Sequence composition change is calculated as (design_count-native_count)/(design_count). The sequence composition difference per residue is plotted

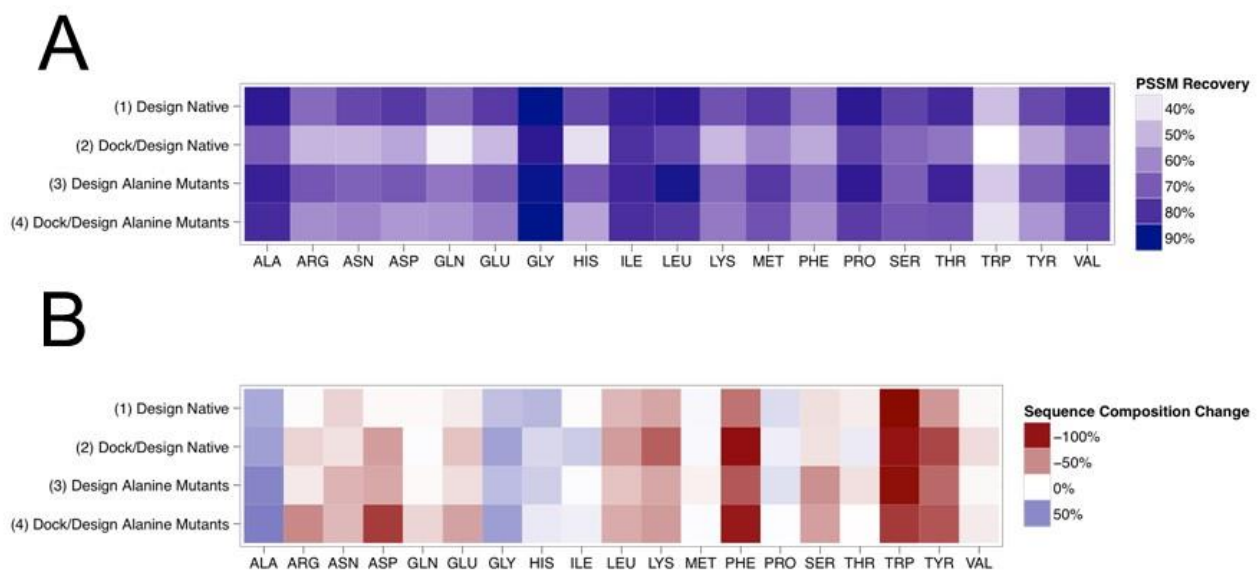


Figure 6: Heatmap of PSSM recovery per residue for each experiment (A). Dark blue indicates that these residues are mutated to residues with good PSSM scores, while light blue indicates a mutation to a residue seldom seen at that particular position. Heatmap of change in sequence composition for each experiment (B). Sequence composition change is calculated as (design_count-native_count)/(design_count). White indicates a residue that remains consistent in composition, red indicates a residue that is designed out of the sequences, blue indicates a residue that is designed into the sequences. Cys was omitted as it was not involved in design. (Reprinted from Journal of Structural Biology.)

in Figure 6B and Table 3 (ST1). Sequence composition remains consistent for Glutamine, Isoleucine, Methionine, Proline, Threonine, and Valine. There are large negative biases to design out Phenylalanine and Tryptophan. While some degree of unfavorable mutation can be tolerated or even desired at certain positions in a ligand binding pocket, one would not expect to see such a significant loss of aromatic residues. Poor recovery of aromatic amino acids may reflect the absence of π-π and cation-π interaction scoring terms in the Rosetta energy function.

Sequence recovery, alanine to wild-type recovery, and PSSM recovery provide feedback to evaluate RosettaLigand's performance in designing protein-small molecule interfaces. The results demonstrate that by recovering native-like interactions, RosettaLigand shows promise as a tool for designing novel protein-small molecule

interfaces. *In silico*, the best assessment of accuracy is to compare designs to the sequences of the protein-ligand complexes in the benchmark set. Other algorithms that seek to computationally design protein-small molecule interfaces include OSPREY[85] and PocketOptimizer[86]. In a number of studies computationally designed mutations in protein-small molecule interfaces were experimentally verified. For example when redesigning an enzyme for target substrates[27], design of a peptide inhibitor which rescues regulatory activity[87], and to predict mutations that arise from drug resistance[88]. Some methodological improvements that can be considered to improve RosettaLigand performance further include: continuous flexibility of rotamers[85], continuous backbone flexibility[89], local backrub motions[90], and computing partition functions over molecular ensembles[91]. However, the first critical step is to perform an experimental verification of RosettaLigand designed protein-small molecule interfaces.

Conclusions

RosettaLigand has been used previously to dock small molecules into proteins, allowing full ligand and protein flexibility and recovering small molecule position and most interface side chain conformations within 2 Å of the experimental structure. The results described here have expanded the methods further to include sequence optimization and performed stringent tests on them following the protocol typically used when designing novel protein-small molecule interfaces. We designed experiments to test RosettaLigand's ability to recover the sequence and ligand position while reorienting the small molecule and applying a native sequence bonus. In addition to sequence recovery, we tested RosettaLigand's ability to recover WT residues from those that were intentionally mutated. Most of the trends we saw were expected, such as lower sequence recovery with higher

ligand RMSD and higher sequence recovery with fewer rotatable bonds. As overall sequence recovery increased, recovery of WT residues increased as well. This implies that RosettaLigand can recognize residues necessary for binding and not over-design the interface. Recognizing ligand properties that maximize the recovery of native-like interactions and also recognizing the ligand properties that Rosetta struggled with is two-fold: (1) It gives us crucial feedback for improving the algorithm, and (2) it gives us an advantage in designing novel interfaces, by starting out with designs for ligands that have shown good results.

Many factors contribute to the difficulty in computationally designing protein-small molecule interfaces. The design algorithm must sample the correct ligand and side chain identity and conformation and also have a comprehensive energy function that can distinguish between interactions that promote binding and those that abolish it. One may naively assume that because the binding pocket is significantly smaller than the entire protein, interface design is less challenging than complete protein design. However, there are several arguments why this is not the case: 1) As the protein-small molecule interface is small compared to the core of a protein, there is less tolerance for error, 2) varying 10 positions with all 20 amino acids yields $20^{10} = 10^{13}$ sequences which is near the limit of the sequence space that can be screened experimentally[92,93], and 3) designing protein-small molecule interfaces requires often precise positioning of interacting functional groups which is more challenging than optimizing apolar van der Waals interactions. RosettaLigand can more successfully design sites for apolar small molecules whose binding is dominated by van der Waals interactions. This was seen in many of the sequence recovery plots, where recoveries were the worst for ligands that were very hydrophilic,

contained many hydrogen bond donors and acceptors, and had high topological polar surface areas. An in-depth analysis of recovery by polar vs apolar amino acids reveals that for all experiments, sequence recovery and PSSM recovery for apolar residues was higher than for polar residues (Table 4 (ST2)). For example, in the Dock/Design Alanine Mutants experiment, apolar residues at the protein-ligand interface had sequence recovery of 62.4% while, polar residues were recovered 32.8%. Also, PSSM recovery shows that apolar residues were recovered 78.7%, while polar residues recovered 63.1%. Additional energy terms will likely be needed for accurate design of interfaces that rely primarily on partial covalent interactions.

Materials and Methods

Compilation of a benchmark of 43 protein-ligand complexes: The Community Structure-Activity Resource (CSAR) database[94] contains a diverse set of protein-small molecule crystal structures and includes information on binding affinities. The full dataset of 343 complexes was filtered to obtain a suitable subset for the present study. First, complexes where metal ions or water molecules were deemed critical for the interaction were excluded since design of interfaces that contain more than two interaction partners requires further modification of the design algorithms. From the remaining set, proteins containing more than 800 amino acids were excluded to limit the time needed in the protein minimization step, leaving 102 complexes. Half of these contained ligands with more than 3 rotatable bonds, and these were excluded to limit the degrees of freedom of the ligand. Lastly, complexes where the ligand was at the interface of two protein chains were excluded. The final benchmark set contained 43 protein-ligand crystal structures,

with resolutions better than 2.50 Å and ligand molecular weights varying between 70 – 400 g/mol.

Preparing the benchmark set: Files from the Community Structure-Activity Resource (CSAR) dataset[94] were prepared as described previously[95]. The ligand atom coordinates were extracted from the input files, and the script 'mol_file_to_params.py' was used to create .params files that describe chemical properties of each ligand and assign each ligand a Rosetta atom type. BioPython was used to align residue names, and convert non-canonical residues to their canonical base residues. Neutralizing caps were removed from the N- and C-termini of the protein chain. Protein chains were relabeled alphabetically, and the ligand was given the chain identifier 'X' and residue code 'INH' in each file. This dataset was filtered to exclude protein chains longer than 800 amino acids, ligands with more than 3 rotatable bonds, metal ions and water molecules tightly bound at the interface (within 3.0 Å of protein or ligand), leaving 43 complexes that were used for the native complexes datasets (Table 5 (ST3)).

Determining critical residues in the protein-ligand interface: For each of the 43 protein-small molecule complexes, RosettaLigand was used to generate 100 'relaxed' models, employing a Rosetta protocol that relies on gradient minimization and side chain repacking. The contribution of each residue to the interface energy was determined as the difference in per-residue energy in the free and bound forms of the protein, and averaged among the 100 models generated. The residues with the highest contributions to ligand binding were mutated sequentially to Alanine to create five new complexes (e.g. in the first complex only the residue contributing most to the stability of the interface was mutated, in

the second complex the two highest contributors were mutated, etc.). The final Alanine-modified benchmark set contained 5x43=215 complexes.

Determining residues in the design sphere: Interface residues were selected for design and repacking based on four distances measured between the Cα of a protein residue and the closest non-hydrogen atom of the ligand. Residues within 6 Å were designed (i.e. side chains can change to any other amino acid, excluding cysteine). Residues within 6-8 Å were considered for design only if the residue was pointing towards the ligand (i.e. the distance between Cβ and any non-hydrogen ligand atom was less than the distance between Cα and the same ligand atom). Residues within 8-10 Å were repacked (i.e. side chain rotamers were sampled but residues were not mutated). Residues within 10-12 Å were repacked only if the residue was pointing towards the ligand. The cutoff values were chosen to ensure that the design sphere was small enough to allow for mutations close to the ligand, yet large enough to include the longest residue, Arginine, in the sphere as well.

Determining the optimal favor native residue bonus (FNRB): In interface design, only the protein residues within the known or putative ligand binding site are allowed to mutate. Designs with the lowest number of mutations are preferred to minimize perturbation of the protein fold. In order to achieve this, a small energy bonus is added to keep the original residue unless introducing an alternate amino acid results in a significant energetic gain. The 'favor native' residue bonus (FNRB) is typically chosen in the range of the per residue standard deviation of the RosettaLigand score. The optimal FNRB cannot be determined by simply redesigning protein-ligand complexes with their native sequence since an increased bonus will always result in increased sequence recovery. Instead, we first determine the five most critical binding residues and mutate these amino acids to

Alanine. These mutants are then redesigned using RosettaLigand to test whether the Alanine reverts to the correct residue.

Of the 43 complexes in the benchmark, a subset was randomly selected to establish the optimal FNRB. This subset contained no duplicate proteins or ligands. For each experiment, the ligand was allowed full reorientation (2 Å translation, 360˚ rotation) with FNRB values between 0.5 and 1.5. One thousand designs were generated for each Alanine mutant with each bonus and then the models with the top 50 interface scores were selected for analysis. As expected, when the FNRB is increased, the sequence recovery increases (number of mutations decreases). However, the percent reversion of Alanine to WT residue increases until FNRB = 1.0 and then decreases as FNRB is increased further. Based on these results, FNRB = 1.0 was applied for the subsequent experiments.

Description of each experiment: Four sequence recovery experiments (percentage of designed residues that are identical to native residues) were conducted on the full dataset of 43 complexes. Design Native (1) probed sequence recovery of the native complexes when there was no FNRB and the ligand was allowed limited reorientation (0.1 Å translation, 2˚ rotation). Dock/Design Native (2) probed sequence recovery of the native complexes when there was no FNRB and the ligand was allowed full reorientation (2 Å translation, 360˚ rotation). Design with Alanine Mutants (3) probed sequence recovery and Alanine-to-WT recovery of the native and Alanine mutant complexes when there was a FNRB and the ligand was allowed slight reorientation. Lastly, Dock/Design with Alanine Mutants (4) probed sequence recovery and alanine-to-WT recovery of the native and Alanine mutant complexes when there was a FNRB and the ligand was allowed full reorientation.

Determining the number of residues considered for design to compute sequence recovery: In the experiments that allowed full ligand movement, the total number of residues that interact with the ligand is a moving target. To report overall sequence recovery as a percentage, the number of residues at the interface was chosen as the average number of residues considered for design. This is still somewhat problematic as not all residues were necessarily allowed to change in each of the docking/design trajectories. We counterbalance this limitation by also reporting the absolute number of mutations (Table 7 (ST5)). However, these numbers are not comparable from complex to complex because the number of residues at the interface varies depending on the ligand size and the shape of the binding pocket.

Individual ligand parameters determined by the BCL The BioChemistryLibrary (BCL) is a software suite tailored for small molecule modeling, and contains a variety of small molecule descriptors[96]. The ligand parameters calculated by the BCL include: number of hydrogen bond donors and acceptors, number of ligand rings, number of atoms, topological polar surface area[97], van der Waals surface area (computed using the BCL's algorithm, which considers the overlapping spheres of neighboring atoms), and logP[98].

Evaluating RosettaLigand design performance using Position Specific Scoring Matrices (PSSMs) PSSM data provide a more quantitative insight into specific residues that are successes/failures when subjected to design. PSSMs were generated from the native protein sequences using BLAST[99]. The designed residues were then scored using the PSSM. Thus, a residue of a type frequently seen at that position would have a positive PSSM score, while a residue seldom seen at that position would have a negative PSSM score. The

percent PSSM recovery is a measure of the percentage of residues with favorable mutations.

Acknowledgments

Work in the Meiler laboratory is supported through NIH (R01 GM099842). B.A. is supported through the National Science Foundation Graduate Research Fellowship Program, grant number DGE-0909667.

CHAPTER 3: Experimental and Computational Identification of Naïve Binders to a TIM-Barrel Protein Scaffold

Brittany Allison, Alex Geanes, Brian Bender, Jens Meiler

Contribution

Once completed, I will be the first author of this manuscript. I contributed the bulk of the: introduction, NMR experimental setup and analysis, RosettaLigand analysis, figures 7 – 15, conclusion, and supplemental information. I also reviewed the parts that the other authors contributed.

Abstract

Proteins evolved to recognize a wide variety of small molecules. Given the large number of such proteins, this process must have happened many times independently. A low, baseline affinity for the small molecule in question must have existed by chance in the naïve protein that provided a benefit for the organism for evolution to kick it into gear. For such a low affinity in the μM range it would be sufficient if the protein binds to a 150-250 g/mol fragment of a larger small molecule. We hypothesize that this baseline affinity exists as soon as a small molecule binding pocket in a protein is present, i.e. in a gene duplication event a second copy of the protein emerges that is free to evolve to recognize additional ligands. In this study we focus on $(\beta\alpha)_8$ barrels, also known as TIM-barrels, that are frequently observed among soluble enzymes in metabolic pathways. For imidazole glycerol phosphate synthase (HisF), we experimentally determine small molecule fragments with

intrinsic binding affinity. Two mutations introduced into HisF, D130V and D176V, allow rCdRP (MW 348 g/mol) to bind at Kd = 0.2 μM[100]. Of around 3500 small molecule fragments screened, 28 displayed intrinsic binding affinity for HisF with a MW range of 174 – 258 g/mol, dissociation constants ranging between 338 – 1112 μM. These molecules cluster into 7groups. In a second experiment we use RosettaLigand to test the ability of computational methods to mimic this process *in silico* and identify the binding pocket for these fragments. This is an important question for engineering novel proteins or enzymes *in silico* that would leverage such built-in binding pockets. Results indicate that interacting residues were most successfully recovered when there are strong interactions, such as hydrogen bonds.

Introduction

Evolutionary pathways to small molecule binding pockets in proteins provide insight about the similarity of binding in similar proteins. Protein-small molecule recognition is highly specific, and the ability to recapture this functionality would be a great asset. Literature suggests the importance for some intrinsic affinity to a small molecule to begin with[59]. This initial intrinsic affinity is needed to have some benefit for the organism that evolution needs to 'kick into gear', i.e. the signal to optimize upon. The possibility that a binding pocket on the surface of a protein is likely to emerge by chance given the fold. For example $(\beta\alpha)_8$ barrels have a fold that almost automatically creates a pocket between the eight βα loops on one side of the barrel[101,102]. In cases of enzyme pathways, where the product of one enzyme becomes the substrate for the next enzyme, the binding pocket does not need to be 'reinvented', but rather only modified[101]. We hypothesize that once functional groups such as charges, hydrogen bond donors, acceptors are positioned by

evolution to bind one small molecule, they are likely promiscuous in recognizing parts/fragments of other small molecules by chance. This would be important as it would spare nature from re-inventing the wheel over and over again, and would also be applicable in cases of small molecule binding. Studies have re-established enzymatic activity and as well as binding events of TIM-barrels onto similar TIM-barrels with only a few mutations, namely HisF and similar proteins[48]. A gene duplication event creates a second copy of the protein, and the second copy evolves to tightly bind a new small molecule for which an intrinsic affinity to a fragment has been existing[102]. The present experiment is designed to test this, by identifying fragments with intrinsic binding affinity to HisF. Then the 'gene duplication event' would be introducing mutations suggested by follow-up computational or experimental studies.

Computational engineering of small molecule binding pockets has been difficult, and even successes have been met with limitations. Morin *et al* attempted to design a protein to bind a peptide, and although the protein crystal structure superimposed with the computationally predicted structure, the peptide did not bind[3]. Tinberg *et al* described two successfully designed proteins which do bind a rigid steroid hormone, but also details the other 15 designs which were computationally favorable but showed no experimental indication of binding[37]. Reported successes of designed receptors and their novel binding ligands could not be experimentally reproduced[2]. Computational enzyme design, which tackles a similar problem, has seen some successes (Kemp elimination reaction[30], Diels-Alder reaction[26], retro-aldol enzymes[31]), but turnover rates are minimal compared to wild type[35]. Allison *et al* described a computational benchmark, which achieved 27% success in recovering residues necessary for protein-ligand interaction after being manually changed

to alanine[40]. As computational methods continue to evolve, successes as well as limitations highlight where improvement is needed.

Leveraging evolutionary principles aids the process for computational design of small molecule binding pockets. We argue that the computational re-engineering of proteins to recognize small molecules can be accelerated by mimicking nature's approach of repurposing existing ligand binding pockets with an intrinsic ability to recognize a small molecule of interest or a fragment thereof. This approach has several advantages: (1) Optimization of an existing binding pocket with an intrinsic affinity is an easier task than *de novo* design of a new binding pocket; (2) an intrinsic affinity for the small molecule of interest or a fragment thereof indicates the general ability of the protein to bind molecules of this structure; (3) as computational design is usually conducted in an iterative loop with experimental verification, an intrinsic affinity for the ligand of interest provides a baseline signal to improve upon. This avoids a situation where a protein that does not bind a certain small molecule is computationally re-engineered and still does not recognize the target protein. In such cases it is difficult to assess where the computational approach failed.

Results of experiments allow for further improvement of computational design algorithms. The iterative nature of developing and improving computational algorithms requires systematic experiments to test their accuracy and reliability. For this purpose, the ability to predict native-like protein-ligand interactions needs to be systematically benchmarked. Since the mid-1990's, successful application of x-ray crystallography and NMR structure information have greatly influenced the growth of computational programs[38]. Specifically to small molecule docking, the interaction is dependent on highly specific ligand and side chain positioning. As discussed above, computational algorithms

fail to achieve 100% accuracy in such predictions. To further improve computational algorithms, we suggest to test their ability to predict intrinsic affinity of a small molecule fragment to a protein and position these fragments accurately. This approach has several advantages: (1) accurate prediction of binding for a small molecule fragment is a stringent test of the computational algorithm, as only few interactions are needed to confer the low affinity, little room for error is left; (2) the low-medium affinity of the small molecule allows usage of HMQC-NMR experiments for detecting ligand binding, experiments that provide feedback on the location of the binding pocket at the level of amino acid resolution, give direct access to dissociation constants to determine binding affinity, and can be conducted in medium throughput so that many proteins/small molecule pairs can be tested.

HisF provides a scaffold for design of small molecule binding pockets. $(\beta\alpha)_8$ barrels, also known as TIM-barrels, consist of 8 repeating (βα) units and are frequently observed among soluble enzymes in metabolic pathways[42]. It is estimated that 10% of enzymes have adopted this fold, despite sequence diversity and catalyzing a wide variety of reactions[101]. TIM-barrels comprise a superfold widely seen in the Protein Data Bank (PDB), often containing bound ligands[4]. The TIM-barrel chosen for this proposal, HisF, is advantageous because there are multiple structures determined by x-ray crystallography with high resolution, providing a good starting point for computational studies. HisF is native to a thermophile, therefore highly stable and tolerant to mutation, and has been previously used for design[47],[102]. HisF from *Thermotoga maritima* (bacterial) has been crystallized with a resolution of 1.45 Å[5], which serves as a good starting point for computational designs/analysis (Figure 7). Using a protein scaffold with a high quality resolution crystal

structure is important for follow-up computational studies. Using a protein scaffold with a residue resonance assigned 2D spectrum is advantageous because of the difficulty that comes with resonance assignment; the HisF literature 2D spectrum provides a crucial starting point for the experimental studies[49]. HisF's native role as an enzyme catalyzes a ring closure (cyclase reaction) in the histidine biosynthesis pathway[50].

The HisF native binding pocket contains functional groups to bind aromatic and negatively charged moieties. HisF from yeast has been crystallized in complex with its native substrate, Prfar[50] (Figure 7A). The prfar cyclase reaction of HisF is tightly coupled to an ammonia-producing glutaminase reaction of HisH[50]. The active site is deactivated by

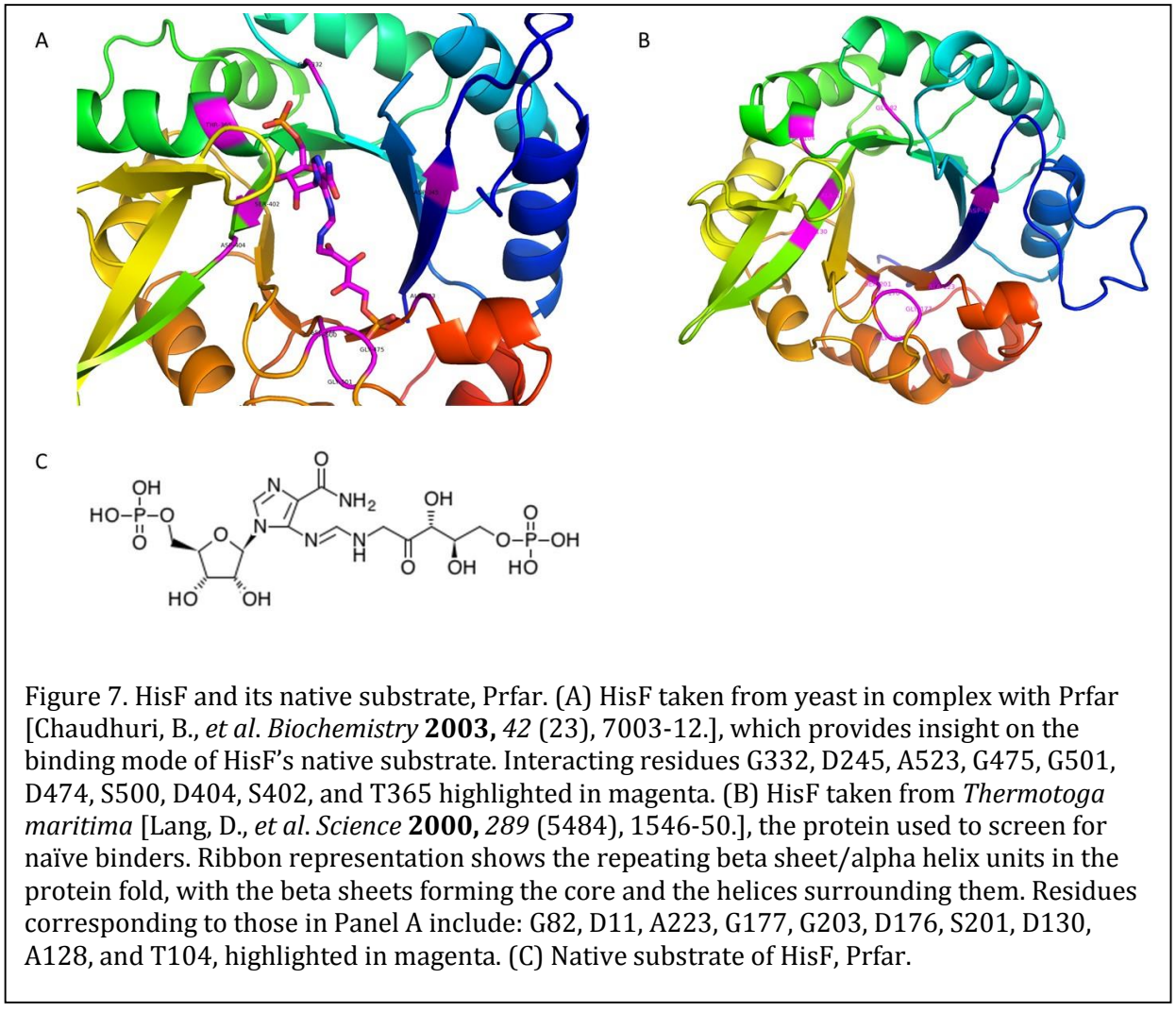


Figure 7. HisF and its native substrate, Prfar. (A) HisF taken from yeast in complex with Prfar [Chaudhuri, B., *et al*. *Biochemistry* **2003,** *42* (23), 7003-12.], which provides insight on the binding mode of HisF's native substrate. Interacting residues G332, D245, A523, G475, G501, D474, S500, D404, S402, and T365 highlighted in magenta. (B) HisF taken from *Thermotoga maritima* [Lang, D., *et al*. *Science* **2000,** *289* (5484), 1546-50.], the protein used to screen for naïve binders. Ribbon representation shows the repeating beta sheet/alpha helix units in the protein fold, with the beta sheets forming the core and the helices surrounding them. Residues corresponding to those in Panel A include: G82, D11, A223, G177, G203, D176, S201, D130, A128, and T104, highlighted in magenta. (C) Native substrate of HisF, Prfar.

pre-soaking with acivicin which inactivates the protein[50]. This offers insight into the binding motif of the native substrate and possibly about other fragments as well. The prfar active site sits at the top of the $(\beta\alpha)_8$ barrel, with prfar binding in a deep cleft stretching across the top of the barrel[50]. The phosphate groups of prfar bind to specific ends of the barrel, and each phosphate forms 4 hydrogen bonds with protein groups[50]. The prfar glycerol phosphate group interacts with Gly524, Ala523 (*corresponds to 1THF Ala223*), Gly475 (*Gly177*), and Gly501 (*Gly203*)[50]. The glycerol hydroxyl groups interact with Asp245 (*Asp11*), Lys258, Asp474 (*Asp176*), and Ser500 (*Ser200*)[50]. Asp245 (*Asp11*) and Asp404 (*Asp130*) shown to have essential roles in the cyclase reaction mechanism[42,103]. There is a cysteine (C9) in the HisF binding pocket, therefore a HisF cysteine-less variant protein will be created, expressed, and purified for screening. Replacing the cysteine with serine (C9S_HisF) gives confidence that small molecule binding is not due to covalent interactions with the cysteine. The interactions we want to see for binding are the non-covalent interactions, including hydrogen bonding, dipole dipole interactions, electrostatics, van der Waals, hydrophobic effects, and geometric/shape complementarity.

Evolution and studies of the HisF small molecule binding pocket serve as examples of repurposing. Gene duplication, one of the major factors in evolution, allows for the adjustment and reuse of functional proteins[102]. Enzymes with the TIM-barrel fold comprise 10% of all enzymes with a known structure, indicating how diverse functionality can be within the same protein fold[101]. Along the histidine synthesis pathway, the enzyme precursor to HisF is HisA (HisH), also a TIM-barrel, and both of these contain phosphate binding sites[100]. Another TIM-barrel protein often compared to HisF, phosphoribosylanthranilate isomerase (TrpF), catalyzes a chemically equivalent reaction

and also contains a phosphate binding site[100]. Via only one mutation, the TrpF reaction was able to be (weakly) established onto HisA (D127V), HisF (D130V), as well as a HisA/HisF (D127V) chimera by a mechanism different than the TrpF mechanism[100]. This indicates that a new functionality can indeed be introduced with a single mutation[100]. Studies probing the stability of fusing two halves of HisF have produced a stable sequence and structure symmetric protein, by directed evolution[104] as well as computational methods[47]. One half of HisF, along with a flavodoxin-like protein, were fused together to yield a stable chimera which also bound a phosphorylated compound due to the HisF binding site remaining intact[105]. Examples of protein design led by evolution have been reviewed[102].

Results

Hits identified in first round of screening were identified by screening a large subset of fragments. The literature residue assignments for the 2D ^{1}H-^{15}N NMR ^{15}N-HisF spectrum[49] were transferred in a stepwise fashion to our ^{15}N-C9S_HisF spectrum. Of the assignments that could be confidentially transferred, 63 residues were determined to be

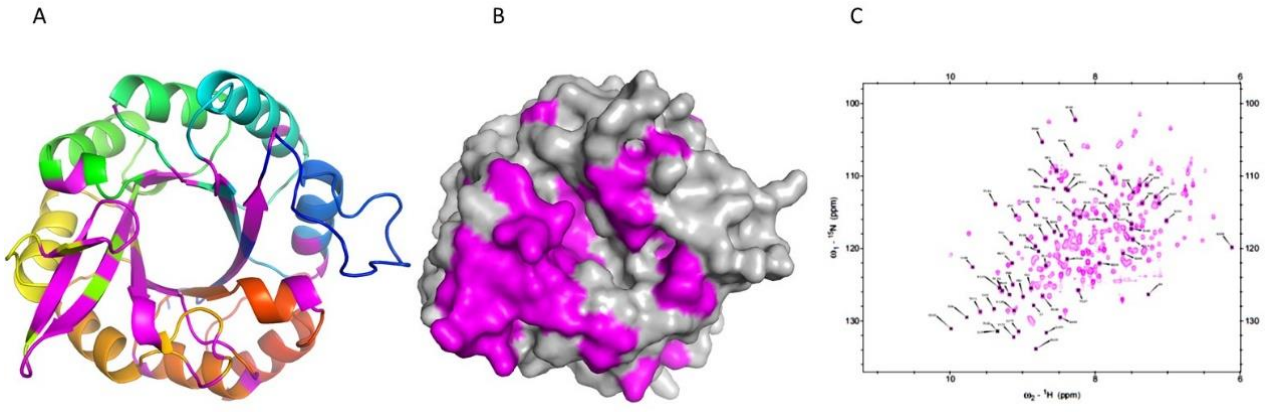


Figure 8. HisF putative binding site determined. Via a series of NMR experiments and assignment transferal, 63 residues were identified as the putative binding site (magenta), as residues that may interact with naively binding ligands in the binding pocket. These residues are identified in magenta on ribbon representation (A) and surface representation (B). (C) HMQC-NMR spectrum of C9S_HisF, putative binding site residues labelled.

the putative binding site, based on residues that were within 8 Å of the center of the binding pocket (Figure 8). Residues pointed out specifically by the literature during other structure/catalysis studies were included in this set as well. The focus of this study is to identify ligands that bind in the binding pocket, therefore attention was focused to this putative binding site. The Vanderbilt fragment library was used to screen a subset of the fragment library in order to identify ligands with native binding affinity for the protein, C9S_HisF (Figure 9). Due to the volume of ligands to be screened, it was most fitting to perform NMR experiments within a 96 well plate setup[10, 11]. The ^{15}N-heteronuclear multiple quantum correlation (^{15}N-HMQC) NMR experiment allows for rapid detection of small conformational changes induced by ligand binding[11]. A change in the environment of the measured atoms causes the peaks to shift, an indication of ligand binding. Mapping peak shifts to the corresponding residues in the protein allows for detailed binding analysis. Of 3,456 compounds screened, identified 25 as hits. Of these 25, only 13 fit a true single binding event curve; 13 hits out of 3456 screened = 0.3% hit rate. The 12 excluded hits all exhibited evidence of binding, but data suggested a very weak or multiple binding event. These pseudo-hits are excluded for future analysis purposes, but are included for identifying matches (Figure 20 (SF3)).

The identified hits were used to search for similar fragments. Used chemcart (a web based tool to query and retrieve chemical structures, reactions, data) to search for ligands similar to the 25 hits, to possibly identify more naïve binders without having to screen the entire library [http://www.deltasoftinc.com/products-overview.html]. For matching ligands, the search was filtered by atomic charge, bond type, and match primary fragment. Search was limited to ligands with 0 – 3 rotatable bonds because this is the range of the

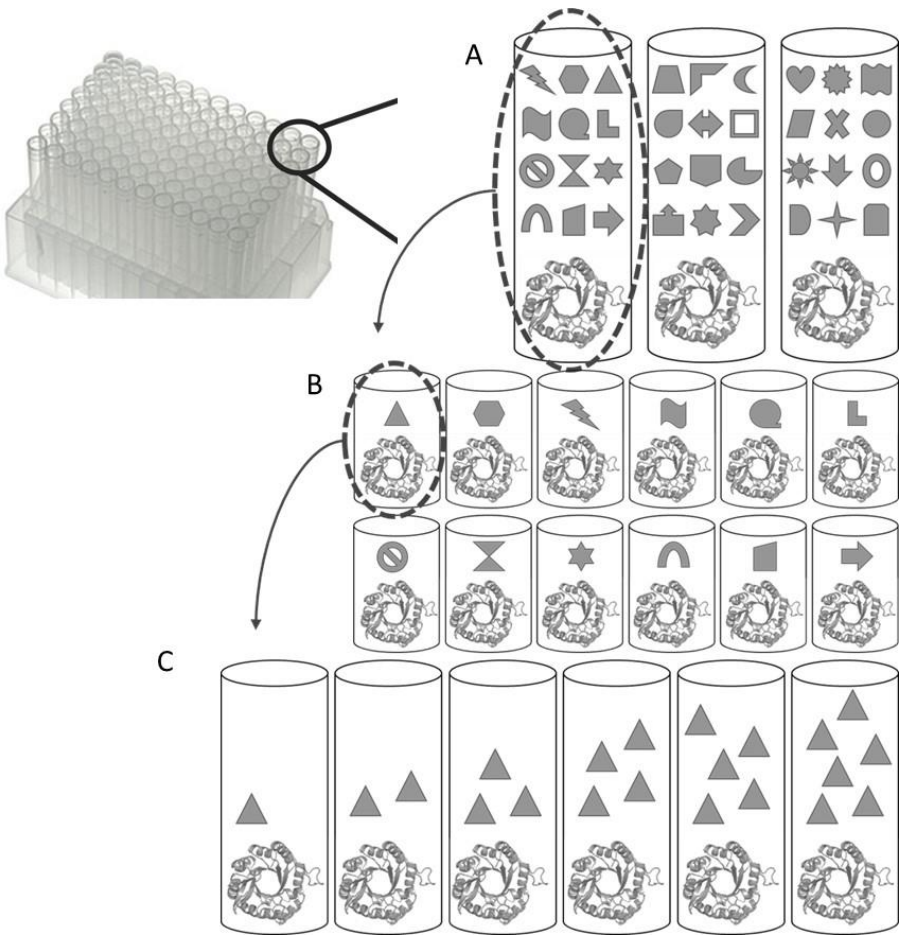


Figure 9. Schematic representation of the process of identifying the naïve binders from a 96-well plate setup using NMR experiments. (A) On a 96-well tube rack, each NMR tube contains 12 compounds at ~600 μM each, already prepared from the Vanderbilt Small Molecule Library facility. 100 μM 15N-C9S-HisF is added day of NMR screening. NMR data is analyzed, and wells displaying peaks shifts (dotted line) are then ordered from the facility with one compound per well. (B) Compounds are screened with only one compound per well, with 100 μM C9S-HisF. NMR data is analyzed, and wells displaying peak shifts (dotted line) move on to the next step. (C) Compounds are screened titration-style, with 40, 80, 160, 400, 800, and 1200 μM compound with 100 μM HisF. Binding curves and Kd are calculated, and these compounds are termed the 'naïve binders'. See text for full details of the method.

original binders. 86 'matching ligands' were screened, and of these 15 were determined to be hits; 15 hits out of 86 screened = 17% hit rate in selecting matches. This gives a total of 28 hits with intrinsic binding affinity for C9S_HisF, the 'naïve binders'.

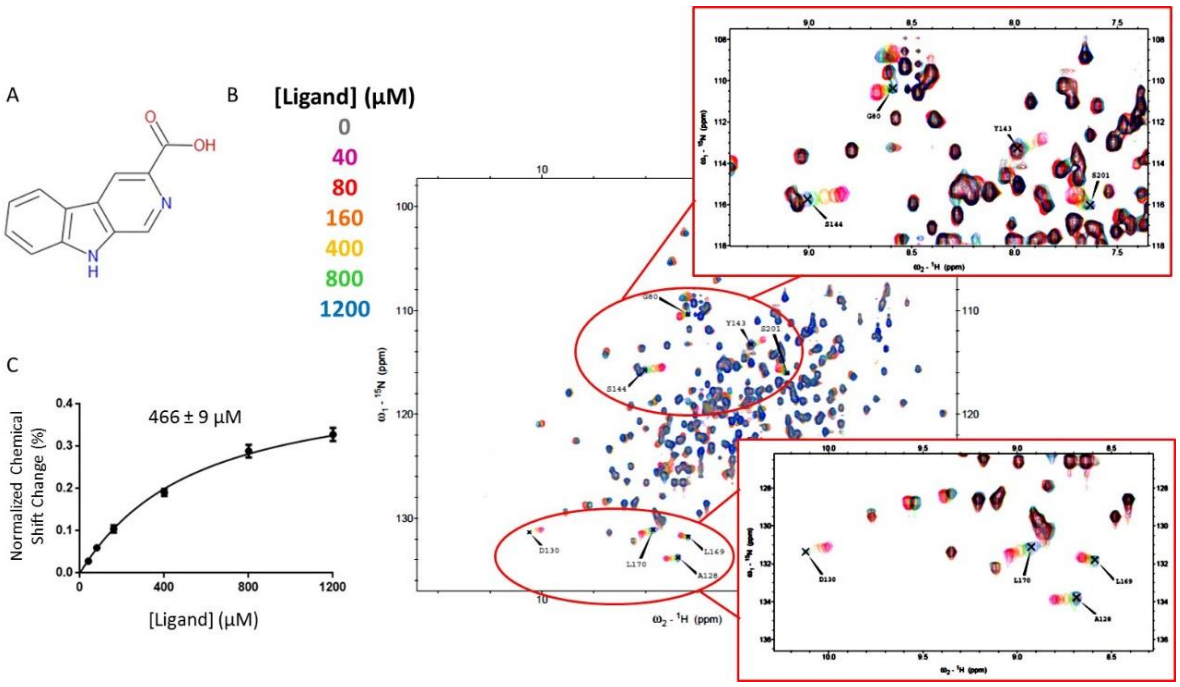


Figure 10. An example of the dataset collected for every naïve binder. (A) 2D image of compound VU0139210. (B) Overlay of HMQC-NMR spectra of compound at 0, 40, 80, 160, 400, 800, and 1200 μM with 100 μM HisF reveals peaks that shift with increasing ligand concentration. Most significantly shifting peaks have been identified on zoomed in inserts. (C) The plotted binding curve of normalized chemical shift change (%) vs ligand concentration (μM) gives a Kd of 290 ± 10 μM.

Binding curves and dissociation constants were calculated for each ligand. Peak assignments were collected for each C9S-HisF-ligand complex with [ligand] at 0, 40, 80, 160, 400, 800, and 1200 μM, imported these to Excel. The ^{1}H and ^{15}N resonances are combined into one term, and then measured from the [ligand] at 0 μM. Those residues whose combined shift at 1200 μM is one standard deviation higher than the average of all 1200 μM combined shifts[106] are identified as the 'significantly shifting residues' for that protein-ligand complex. Dissociation constants were determined by measuring the chemical shift changes as a function of ligand concentration, along with binding curves, created by GraphPad Prism [GraphPad Prism version 6.04 for Windows, GraphPad Software, La Jolla California USA, www.graphpad.com]. The significant shift residues within

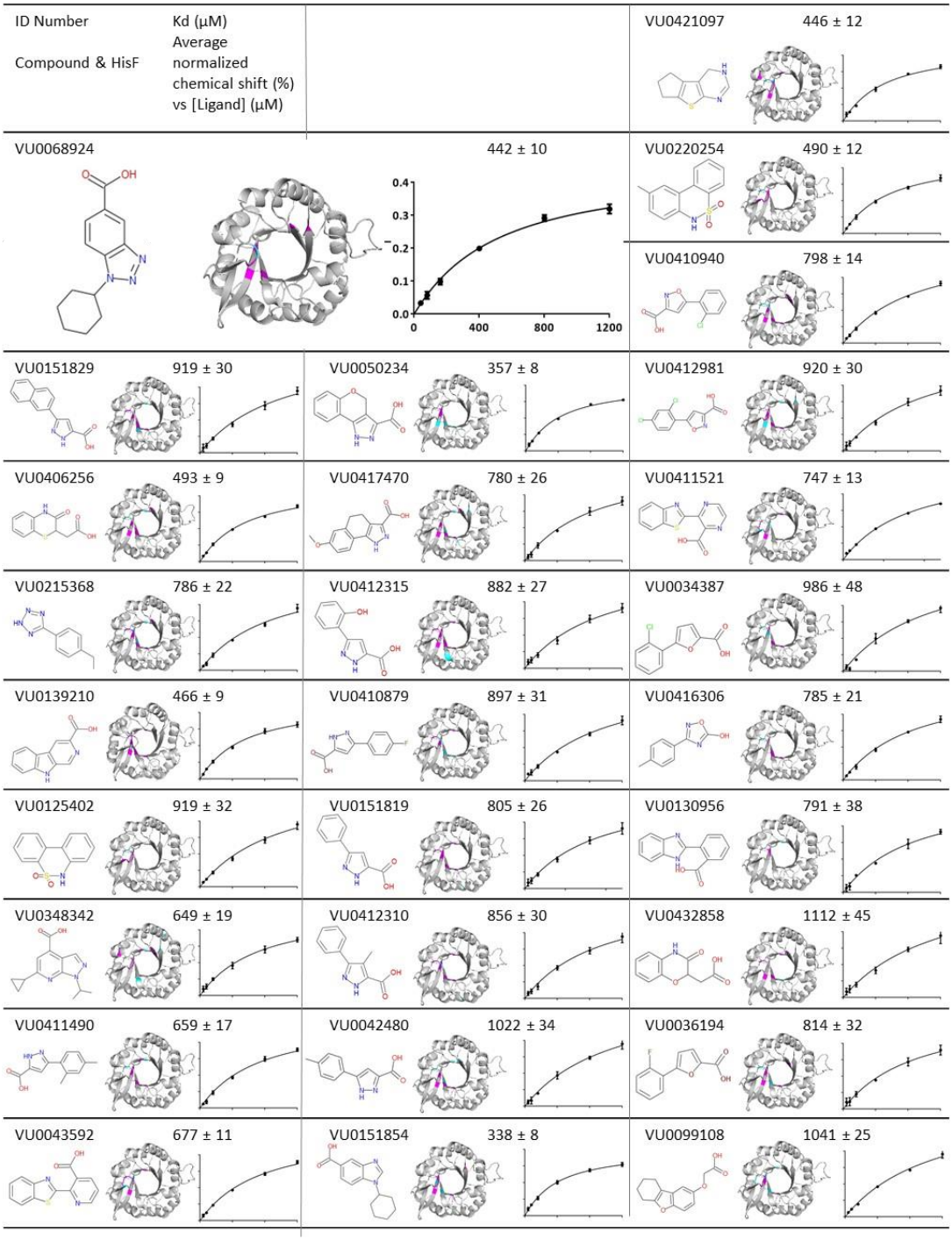


Figure 11. Naïve binding ligands and binding data. VU compound number, 2D structure, pymol image of C9S_HisF highlighting significantly shifting residues (magenta) and significantly shifting residues not included in binding data (cyan), binding curve of normalized chemical shift change (%) vs ligand concentration (μM).

each set that did not follow the binding shifting pattern were quantitatively identified as 'outlier residues' and excluded from Kd calculation. Figure 10 highlights a naïve binding

ligand, with its accompanying titration spectra and binding curve.

Discussion

The 28 naïve binders bind weakly, dissociation constants range between 338 – 1112 μM (-19.80 - -16.85 kJ/mol) (Figure 11). They range in molecular weight between 174 – 258 g/mol, and vary between 21 – 34 atoms. They contain 3 – 8 hydrogen bond donors+acceptors, and contain 1 – 3 aromatic rings. Flexibility ranges between 0 – 3 rotatable bonds. LogP ranges between 1.66 – 5.06. Within the group of 28 ligands, 23 contain a carboxylic acid group, 6 contain a sulfur, and 5 contain a halogen.

As a group, the 28 naïve binding ligands contain similar moieties. Of the 28, all contain at least one aromatic ring, 23 contain a carboxylic acid group, 6 contain a sulfur, and 5 contain halogens. Many of the ligands are a carboxylic acid attached to a nitrogen ring group. This is not surprising, considering prfar's phosphate group and nitrogen-

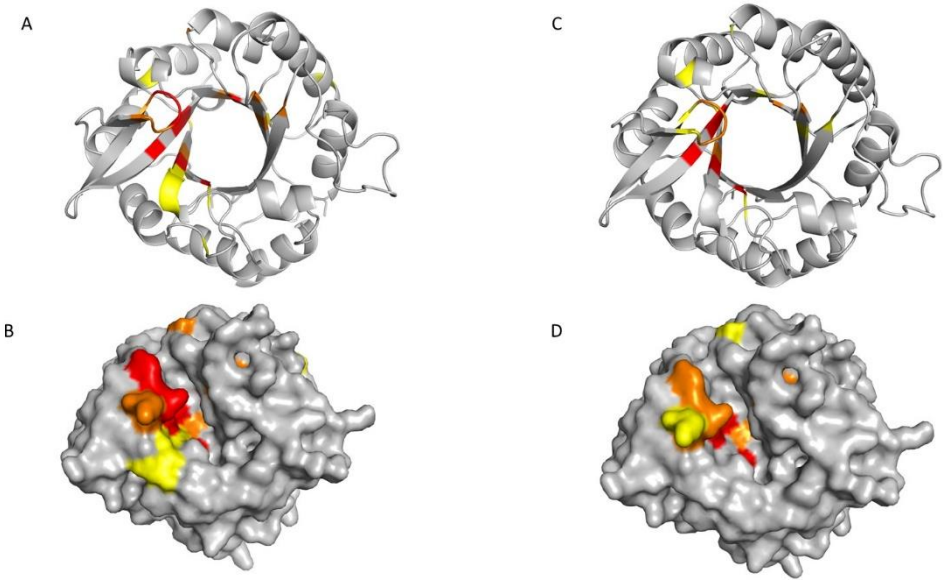


Figure 12: Analysis of significantly shifting residues for all naïve binders reveals a preferred binding spot. Of all 28 naïve binders including the outlier residues (A, B), residues that shift most frequently (red) shift 15 - 21 times, residues shift 3 - 10 times (orange), and residues that shift least frequently (yellow) 1 - 2 times. These numbers include the outlier residues (see text for explanation). Excluding the outlier residues (C, D), does not change the overall binding hotspot, but the loop drops from the red/orange category to the orange/yellow category, with the core of the barrel displaying the most peak shifts.

containing ring group. Crystal structures of HisF show phosphate or sulfate ions in complex with HisF where the prfar phosphate groups bind[42,5].

A C9S_HisF binding pocket hotspot appeared after analyzing the compiled data of the significantly shifting residues from the 28 naïve binders. Within the putative binding site, a preferred binding mode emerged (Figure 12). Among the 63 residues selected as the putative binding site, 36 never displayed any peak shifts, therefore believed to not interact with the binders. The majority of these residues are located on the beta sheet extension residues (residues 132 – 141), the tightly packed area within the beta barrel (residues 198 – 200), or the tightly packed area within the beta barrel which then extends to the surface (residues 7 – 11). The remaining 27 interacting residues have been grouped by frequency as a significantly shifting residue: shift 15 – 21 times (Asp130, Ser144, Val127, Gly80, Ala128, Tyr143, S201, Leu170), 3 – 10 times (Phe49, Val100, Thr142, Asp85, Lys146, Val12, Asp51, Leu169, Gly145), or 1 – 2 times (Glu71, Phe77, Gly166, Thr171, Ile173, Asp174, Ile198, Ala106, Gly181, Gly202). Hot spot for interaction includes the catalytic residue Asp130 and many surrounding residues, as well as the nearby loop (residues 142 – 144). Interestingly, catalytic residue Asp11 displayed no peak shifts. Although both Asp11 and Asp130 facilitate catalysis, only Asp130 shows binding activity. Another highly interacting residue, Ala128, interacts with prfar[50]. The term 'barrel' in $(\beta\alpha)_8$ barrel may deceivingly convey that a ligand may bind to any of the beta barrel residues within this large open space. However, the HisF surface representation shows that the hot spot residues are indeed the ones that are most accessible (Figure 12B). Residues on the Asp11 side are tightly packed and would not provide the shape complementarity/support for a sustainable interaction. Looking at prfar in complex with HisF, the hotspot residues are

also near to where prfar's more cyclic end is bound. This provides insight to why many of the naïve binders are acids attached to rings, mimicking the prfar binding mode. Not all residues in the binding hot spot contributed to the binding interaction. Thr171, near hotspot residues L170 and D130, never displayed peak shifts.

Some residues that shift may not contribute to the binding interaction. Of the shifting residues during the titration, some of these exhibit significant shifts yet were quantitatively identified as outliers. These 'outlier' residues were excluded from Kd calculations but still included on the figures. These outliers occurred at most 10 times (Gly80 and Tyr143), 3 – 5 times (6 residues), once or twice (15 residues), or not at all (40 residues, including the 36 that did not shift at all). Many of the reoccurring outliers appear in the binding hotspot. Tyr143 is located on the highly flexible loop and points out into solvent, away from the binding pocket. Gly80 sits at the top of the barrel and near/above the cyclic-end phosphate binding region of prfar, therefore possibly easily perturbed by solvent and/or binding events. Outlier residues that occur 3 – 5 times (Val12, Leu169, Ser201, Val127, Ser144, Leu170) are all located in the hot spot binding site, with the exception of V12 located at the top of the barrel. Excluding the outliers does not change the hot-spot binding landscape. Val127, Ala128, Asp130, Leu170, and Ser201 remain as the highly shifting residues even with the outliers excluded. These beta sheet residues all reside in the HisF binding region and are all near the catalytic residue Asp130, as described above. Excluding the outliers, the loop (residues Thr142, Tyr143, Ser144, Gly145, Lys146) seems less involved in binding, but moreso a mode for structural stability/ geometric complementarity for the ligand interacting with the hot-spot residues.

Similar naïve binders induce similar significant shifts. Many of the naïve binders can be grouped by functional groups and similar moieties. Analysis of Kd and significant shifters within these groups provides insight. VU0220254 (490 ± 12 μM) and VU0125402 (919 ± 32 μM) are both a 3-ring system containing a sulfate group; the two molecules are identical with the exception that VU0220254 has a methyl group extending from one of the rings. These 2 binders perturb the same significant shifters in only the loop region, an indication of weaker binders. But VU0220254 also perturbs Val127 within the binding pocket, therefore not surprising that VU0220254 has a tighter interaction; perhaps the extended bulkiness of the methyl group allows for added shape/structural complementarity. VU0406256 (493 ± 9 μM) and VU0432858 (1112 ± 45 μM) are both acids connected to a 2 ring system containing nitrogen; the molecules are identical except VU0406256 contains a sulfur and VU0432858 contains an oxygen. The 2 ligands both perturb the same residues in the hotspot binding region, and in the lesser binding region (Asp51, Gly80, Asp130, Leu170). Both also perturb Tyr143 (loop residue) but it is an outlier for VU0406256. These similar peak shifts, especially for ligands that contain prfar-like moieties, may indicate similar Kd. However, VU0406256 also perturbs Ala128 and Leu169, two residues deeper in the binding pocket. These additional interactions allow for tighter binding, and VU0406256 is indeed the tighter binder. The sulfur perhaps makes contacts that the oxygen cannot, due to larger size and larger electron cloud density. VU0050234 (357 ± 8 μM), VU0139210 (466 ± 9 μM), and VU0417470 (780 ± 26 μM) are all acids connected to a 3-ring system containing nitrogen and/or oxygen. Unlike other cases of similar ligands, VU0050234 perturbs less significant shifters than the other two, yet is the tightest binder. The interacting residues, Val127, Ala128, and Ser144 allow for a tight

interaction structurally supported by the loop. VU0139210, in a close second, perturbs many of the hopspot residues, therefore not surprising that is it on the tighter end of the spectrum within this set of binders. VU0417470, the bulkiest of these 3 due to an extended group containing oxygen and methyl, interacts with many of the same residues as the other two, yet is a mid range binder, perhaps due to the added bulkiness (which does deviate from the prfar likeness). VU0151854 (338 ± 8 μM), VU0068924 (442 ± 10 μM), and VU0348342 (649 ± 19 μM), are all acids attached to a nitrogen-containing 2-ring system which is then attached to another ring. VU0151854 and VU0068924 are identical except that VU0068924 has an additional nitrogen; VU0348342 was included due to the same core group. VU0151854 and VU0068924 both perturb the same hotspot residues (Val127, Asp130, Gly145) as well as Val12; Ser144 and Leu170 were perturbed but outlier residues. VU0151854 additionally interacts with Ser201 and Gly202 at the bottom of the hotspot region, contributing to the slightly tighter interaction. VU0348342, although a similar molecule, interacted with different residues spanning all around the putative binding site, therefore a mid-range binder. VU0043592 (677 ± 11 μM), VU0411521 (747 ± 13 μM), and VU0130956 (791 ± 38 μM) are all acids attached to a hexene ring, then attached to a 2-ring system. The three molecules are identical with varying placements of nitrogen and sulfur. The binding range of all 3 molecules is very similar, considering that two molecules contain a sulfur and one does not (unlike examples noted above, where the inclusion/exclusion of sulfur influenced the binding). The sulfur-containing molecules (VU0043592 and VU0411521) do however perturb most of the same residues (Gly80, Asp130, Thr142) and the loop residues which were outliers, whereas VU0130956 perturbs Ser144 and Val127, a different binding mode all together. The sulfur containing molecules highly perturb the

loop region, but this may not decrease/increase the binding activity, which is why all three molecules have close a close Kd range. VU0410940 (798 ± 14 μM), VU0036194 (814 ± 32 μM), VU0034387 (986 ± 48 μM), and VU0412981 (920 ± 30 μM) are the halogen-containing molecules; they are all an acid attached to a nitrogen/oxygen 5-member ring that is then attached to a 6-member ring, with either chlorine or fluorine attached to the 6-member ring. The molecules all perturb Val127, Leu170, and Ser201 (and Ala128 including an outlier), all residues within the hotspot binding pocket. This set of four molecules all similarly bind in the ~800 – 900 μM range, yet none of these bind as tightly as other molecules noted above, that interacted with the same set of residues. The halogens possibly interact in a way that prohibits tighter binding. The last group of similar ligands, VU0151829 (919 ± 30 μM), VU0411490 (659 ± 17 μM), VU0412315 (882 ± 27 μM), VU0410879 (897 ± 31 μM), VU0151819 (805 ± 26 μM), VU0412310 (856 ± 30 μM), VU0042480 (1022 ± 34 μM), are all acids attached to a 2,3-nitrogen-containing ring attached to another ring system. Being similar to the prfar structure, an acid attached to a nitrogen-containing ring, it is not surprising that this is the largest group. Within this group, it is interesting to note the role that methyl groups play in binding. For example, VU0151819 (805 ± 26 μM) has only a benzene ring. Adding one methyl group to the benzene weakens binding to 1022 ± 34 μM, but adding two methyl groups tightens binding to 659 ± 17 μM. However, adding the methyl group to the nitrogen ring seems to not impact binding, binding remains in the similar range 856 ± 30 μM.

In addition to comparing the ligand binding positions among similar fragments, we also want to assess how well these experimental results could be reproduced

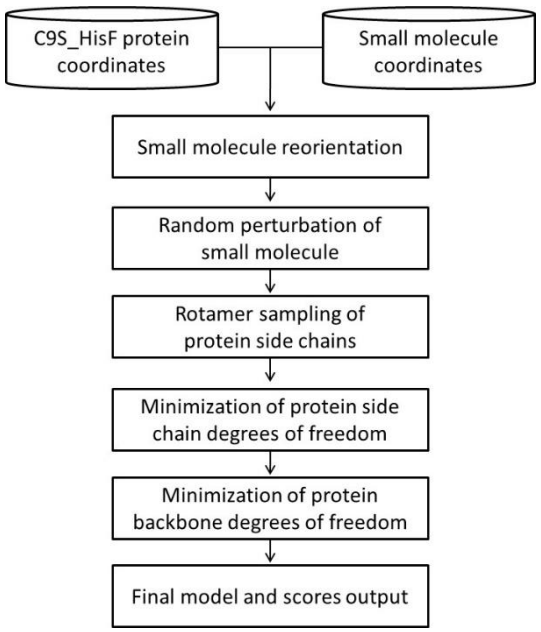


Figure 13: Flowchart of RosettaLigand small molecule docking. From the input coordinates, the small molecule is allowed to rotate and translate (transform mover) to an optimal placement. Side chain rotamer sampling allows for optimal side chain positioning. A final gradient-based minimization of the protein is performed to resolve any clashes, and the final model and accompanying scores are output. See text for detailed description.

computationally. Computational methods in protein-ligand design seek to find the lowest energy model of a protein-ligand complex. Ideally, this model emulates native-like positions and properties of protein-ligand complexes found in nature. Rosetta, a protein modeling software suite for protein structure prediction and design[9], seeks to find the lowest energy conformation of a model, which would ideally be a native-like conformation. Some of these successes include creating novel enzymes[31],[77],[30], altering the specificity of protein-peptide[32], protein-DNA[25], and protein-protein interfaces[34], and designing proteins that bind a selected surface of a virus[24]. RosettaLigand is an application within Rosetta, developed to dock small molecules into a protein with full protein and ligand flexibility[6],[7]. Using the 3D coordinates of the protein and ligand as input files, the RosettaLigand protocol involves optimized placement of the ligand, optimized positioning of the

surrounding residue side chains, and a minimization to resolve clashes[7]. In this study, we ask RosettaLigand to recapture the correct ligand binding position by identifying the residues that contribute to the protein ligand interaction. The small molecules were docked into C9S-HisF using RosettaLigand as described previously[41,107] (Figure 13). In addition to generating the protein-ligand interface score[83], scores are calculated for each residue within a radius of the ligand molecule (6 Å for sidechain atoms, 7 Å for backbone atoms). This ddg calculation measures the strength of contribution of each side chain in its predicted interaction with the ligand. Through the subsequent rounds of models created, the final top 10 by interface score were selected for analysis. In computational studies, analysis is usually done by a percentage or number of the top models, rather than only the one top model. Of the top 10 models selected, the ddg interactions of all these were pooled together, then filtered by the top 200 interacting residues. These computationally determined interacting residues were then compared to the experimentally determined significant shift residues (excluding the outliers). The number of significant shift residues contained in the list of ddg interacting residues, divided by the total number of experimental significant shifts, gives the RosettaLigand percent recovery. This ddg calculation analysis serves as a useful tool in determining whether or not the computational program recognizes when a residue interacts with a ligand.

Of the 28 naïve binding ligands, Rosetta achieved an average significant shift recovery of 53%, ranging from 0% - 100%. Rosetta achieved a 100% recovery for 2 of the complexes, VU0410879 and VU0411521 (Figure 14A, B). VU0410879 (Kd 897 μM) and VU0411521 (Kd 747 μM) both contain an acid group, nitrogen in their rings, and 6 H-bond donors + acceptors. In contrast, Rosetta recovered 0% of the significant shifts for one of the

complexes, VU0034387 (Figure 14C). VU0034387 (Kd 986 μM) contains an acid group, a halogen, no nitrogen, and only 4 H-bond donors + acceptors. More examples in each category would be needed to elucidate a trend.

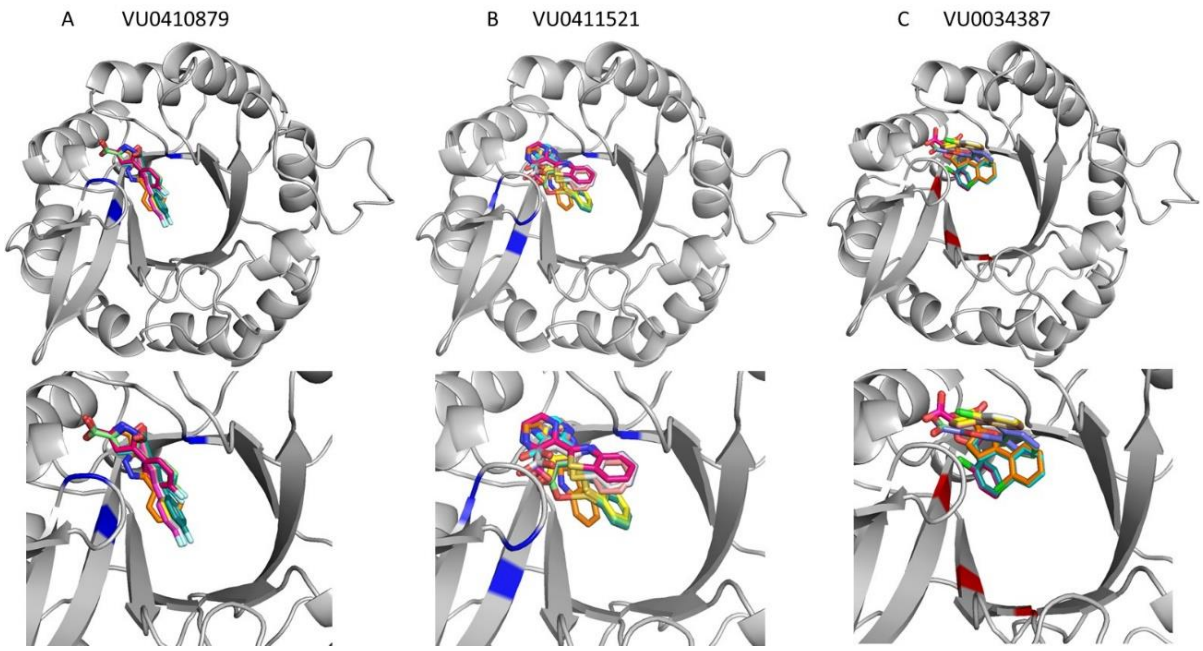


Figure 14. Highlighted examples of Rosetta significant shift recovery. Rosetta computationally identifies all of the experimentally determined interacting residues for two naïve binders, VU0410879 (A) and VU0411521 (B), seen in blue. Rosetta does not recognize any of the experimentally determined interacting residues, VU0034387 (C), seen in red.

Trends in recovery indicate the reliability and accuracy of computational methods. Comparing the RosettaLigand significant shift recovery against a number of metrics reveal a trend (Figure 15). Comparing significant shift recovery vs ligand hydrogen bond donors + acceptors, recovery improved with more hydrogen donors + acceptors. Ligands with less than 4 H-bond donors + acceptors achieved less than 40% recovery, while ligands with more than 7 donors + acceptors achieved more than 60% recovery. This implies that RosettaLigand has a better chance at recognizing a residue interacting with a ligand when

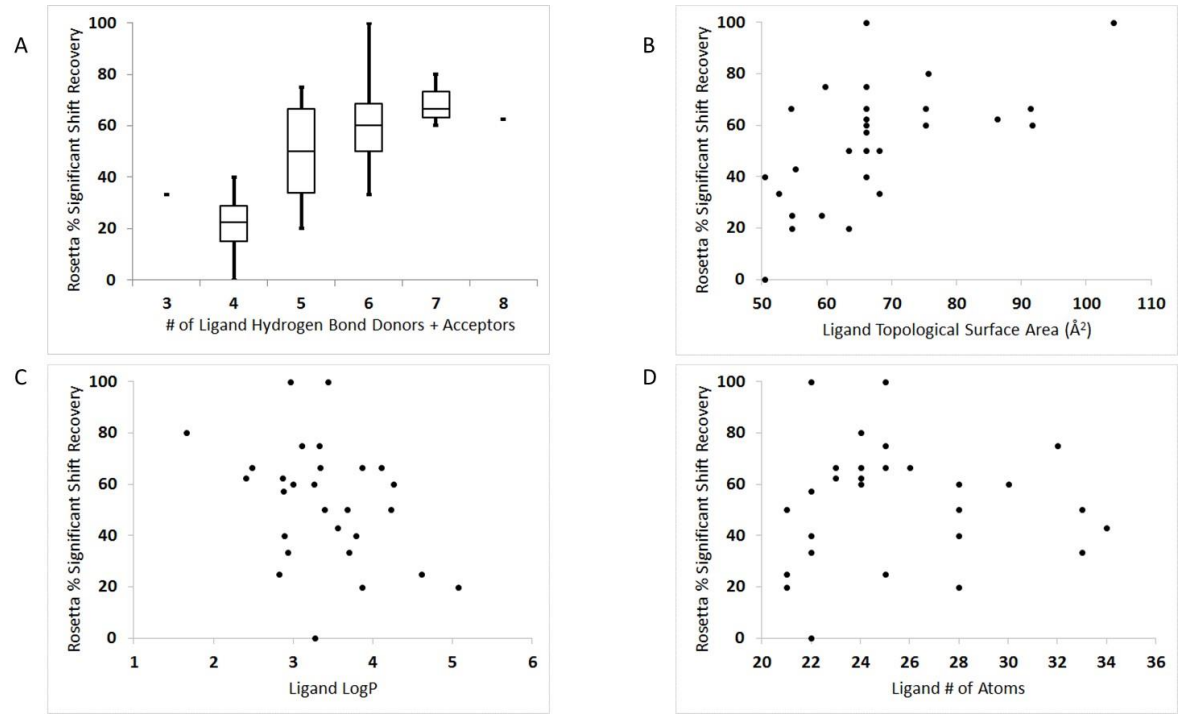


Figure 15. Significant shift recovery by Rosetta. (A) Recovery improved with more hydrogen donors + acceptors. (B) Ligands with a higher TPSA had a higher chance of recoveries, indicating the program recognizes interactions when the ligand contains more polar atoms. (C) Ligands with a very low logP (1.5 – 2.5) had a 60%+ recovery, where ligands with a very high logP (4.5 – 5.5) had lower than 30% recovery. Ligands in the middle range showed no trend. This implies that in extreme cases of lipophilicity, Rosetta may have an easier or more difficult time identifying protein-ligand interactions. (D) Number of ligand atoms showed no significant trend, however of the 11 ligands with 23 – 26 atoms, all but one produced a 60%+ higher recovery.

it is a strong interaction, such as a hydrogen bond. We do expect that there is a cap to this number, where ligands with many H bond donors and acceptors (10+) may see a decrease in recovery, due to a very complex hydrogen bonding network[40]. Another trend was seen in comparing recovery to ligand topological surface area (TPSA), which is a measure of how much of the surface area of a molecule is taken up by polar atoms and their attached hydrogens, with a lower TPSA indicating less polar molecules, and a larger TPSA indicating more polar molecules. Ligands with a higher TPSA had a higher chance of high recoveries, indicating the program recognizes interactions when the ligand contains more polar atoms.

This is consistent with more hydrogen bond donors + acceptors have higher recovery. Another slight trend was seen with ligand logP, which is a measure of lipophilicity comparing the concentration of ligand in octanol vs water, with a low logP indicating less lipophilic (more polar; less likely to dissolve in a fatty substance like octanol) and a higher logP indicating more lipophilic (less polar, more fatty). Ligands with a very low logP (1.5 – 2.5) had a 60%+ recovery, where ligands with a very high logP (4.5 – 5.5) had lower than 30% recovery. Ligands in the middle range (logP 2.5 – 4.5) showed no trend. This implies that in extreme cases of lipophilicity, RosettaLigand may have an easier or more difficult time identifying protein-ligand interactions. These three trends are all consistent with one another. The number of ligand atoms showed no significant trend, however of the 11 ligands with 23 – 26 atoms, all but one produced a 60%+ recovery. Identifying where computational methods do recognize native-like trends shows reliability and accuracy of the scoring function of these programs.

Comparing the RosettaLigand significant shift recovery against a number of metrics showed no trend. These include: Kd, ligand weight, ligand flexibility (number of rotatable bonds), ligand aromaticity, and RosettaLigand interface score. It was surprising that significant shift recovery vs RosettaLigand interface score didn't show a trend, we did expect to see that complexes with a lower interface energy would give a better chance at recovering the significant shift residues. One may have also expected to see a correlation between the Kd and recovery, where RosettaLigand would have recovered more significant shift residues for the more tightly binding ligands. Identifying where computational methods do not recognize trends where there should be a trend shows how these programs can be improved.

Another point of interest was to assess whether similar ligands captured the same recovery by Rosetta, since these ligands probed similar significant shift residues. VU034842, VU0151854, and VU068924, which all contain an acid attached to a 2-ring nitrogen (containing) system, received 33 – 50% recovery from Rosetta. VU0050234, VU0417470, and VU0139210, which all contain an acid group attached to a 3-ring nitrogen (containing) system, received 60 – 67% recovery from Rosetta. VU0410940, VU0036194, VU0034387, VU0412981, which all contain an acid attached to groups of halogen, oxygen, and nitrogen-containing rings, received 0 – 50% recovery from Rosetta. VU0220254 and VU0125402, both a 3-ring system containing a sulfate group, received 20% and 25% recovery from Rosetta, respectively. Overall the highest recoveries were seen for ligands that are acids attached to a nitrogen-containing ring (although not always the case).

We also chose to assess how well Rosetta recovered the residues that are a part of the 'binding hotspot', on a residue-by-residue basis. Within the mid to highest range of residues that exhibit peak shifts (orange and red regions on Figure 12), Rosetta performance in recovering these varied but was overall successful. In order of increasing times as a significant shift, the Rosetta recovered: Leu169 80%; Asp51 100%; Gly80 100%; Tyr143 100%; Gly145 44%; Ser144 92%; Val127 0%; Asp130 100%; Ala128 100%; Leu170 0%; Ser201 19% (Table 8 (ST6)). Analysis about Asp130 and Leu170 never being recovered is included in the supplemental information. Computational programs gain reliability by consistently reproducing experimentally determined data. To see the majority of this set perform above 90% recovery shows where the program does indeed capture native-like interactions.

There are a number of residues that Rosetta often identified as interacting with the ligand, but are not on the significant shifts list. Residues that Rosetta often identifies as contributing to the protein-ligand interaction include 82 – 84 and 103 – 105, which are not surprising because these residues lie in a glycine loop region near the binding pocket and a loop region near the binding hotspot loop, respectively. One residue of interest is Ser101, which never experimentally occurred as a significantly shifting residue, yet Rosetta often identified as contributing to the interaction.

Conclusion

Studying evolutionary pathways of binding and similar protein binding pockets provides insight on the nature of protein-small molecule recognition, as well as the types of ligands that may interact with a protein. The ability to accurately predict ligand docking and even design in binding interactions would be a great asset. Of ~3500 fragments screened, 28 displayed intrinsic binding affinity for C9S_HisF, Kd ranging between 338 – 1112 μM. Within the group of 28 ligands, they contain 3 – 8 hydrogen bond donors+acceptors, and 23 contain a carboxylic acid group. Many ligands contained similar moieties, and could be grouped into one of seven groups by similarity. NMR experiments provide atomic-detail insight into protein-ligand interactions by tracking the chemical shift peaks, allowing for analysis of the preferring binding mode. Val127, Ala128, Asp130, Leu170, and Ser201 are the most frequently interacting residues. This was not surprising, because these residues are near where the ribose moiety of prfar interacts with HisF in its wild type state[50]. This ribose moiety is also connected to an acid group as well as an imidazole ring, which explains why many of the binding fragments contain an acid and a nitrogen-containing ring. Just as HisF and similar proteins catalyze similar substrates[48],

ligands with intrinsic binding affinity for HisF are similar as well. The experimental results compared to a computational benchmark revealed that RosettaLigand achieved an average significant shift recovery of 53%, ranging from 0% - 100%. RosettaLigand better recovered the residues necessary for interaction when binding is driven by strong interactions, such as hydrogen bonds. Ligands with a higher TPSA had a higher chance of high recoveries, indicating the program recognizes interactions when the ligand contains more polar atoms. Computational benchmarks, such as this one, provide necessary insight to improve the algorithms. The computational algorithm methods could be enhanced by the ability to recognize binding even when the interactions are weak. They could also be enhanced by a metric to gauge when backbone atoms interact with a ligand, rather than side chain atoms only.

Methods

Assignment transferal from literature spectrum and identification of putative binding site. The literature assigned 2D 1H-^{15}N NMR ^{15}N-HisF spectrum was performed with buffer of 10 mM MES pH 6.8, 50 mM KCl, 1 mM EDTA, 5% $2H_2O$, on a 600 MHz spectrometer TROSEY experiment at 30° C, and 96.8% of the backbone resonances (239 residues) had been assigned[49]. Due to screening setup 96-well plate style and the nature of the experiment, the final experiment conditions needed for these studies were a SoFast HMQC[108] of ^{15}N-C9S_HisF at 25° C. The following experiments were performed and assignments were transferred from the original literature conditions through a step-wise fashion: ^{15}N-HisF TROSEY at 30° C → ^{15}N-C9S_HisF TROSEY at 30° C → ^{15}N-C9S_HisF SoFastHMQC at 30° C → ^{15}N-C9S_HisF SoFastHMQC at 25° C. All other conditions (buffer, pH, NMR 600 MHz spectrometer) remained constant. Assignments were transferred step

by step, only carrying over assignments that could be transferred with confidence. Reasonably so, many assignments in the binding pocket, could not be transferred in the HisF to C9S_HisF step. Of the 239 original assignments, 150 of these were confidently transferred. Of these 150, a set of 63 residues were determined to be the putative binding site, based on residues that were within 8 Å of the center of the binding pocket. Residues pointed out specifically by the literature during other structure/catalysis studies were included in this set as well, (Ile42, Glu71, Lys99, Glu167, Thr178)[109,42]. The focus of this study is to identify ligands that bind in the binding pocket, therefore we focused our attention to this specific set of residues.

Screening process. The strategy was to use the Vanderbilt fragment library to screen (using NMR experiments on a 96 well plate[11]) a subset of the fragment library in order to identify ligands with native binding affinity for the protein, C9S_HisF. Screening takes place using a 600 MHz magnet, at 298 K. NMR processing is done using Topsin for crude analysis and Sparky for in-depth analysis. Excel is used for data processing. GraphPad Prism is used to create binding curves and calculate dissociation constants. The detailed methods follow. Express and purify large amounts of protein, ^{15}N-C9S_HisF (to screen one plate requires ~250 mg of protein). Store at -30° C, let thaw at 4° C a few days before screen; spin down (3700 rpm, 4° C, ~15 min) to remove precipitate before measuring A280 for concentration. Three plates were screened from the fragment library, termed 'representative plates' because they are a random representation of the fragment library, containing many different ligand scaffolds, containing 12 compounds per well = 1,152 compounds per plate. 3 plates = 3,456 ligands screened total. Well setup: ligands are 2 µL of 200 mM each (so, 24 µL total) = ~17 mM each ligand in 24 µL total. Previous screening benchmarks have shown

that 100 μM C9S-HisF in a 600 μL NMR sample provide good quality spectra. Based on the conditions of the assigned 2D NMR HisF spectrum[49], each sample should be 100 μM protein, 10% D2O, 4% DMSO, filled to 600 μL with MES (NMR) buffer (MES buffer: 10 mM MES, 50 mM KCl, 1 mM EDTA, ph 6.8, stored at room temperature). Add protein/buffer solution to the 96 well tray of ligands, then transfer these to the corresponding NMR tubes in the rack. The final concentrations are ~600 μM ligand with ~100 μM protein. The NMR rack is ready to be screened; each SoFast NMR experiment takes ~35 minutes x 96 experiments = 56 hours = 2.5 days. NMR spectra are processed and analyzed one by one in Topspin software [Version 3.2, Bruker] and Sparky [Goddard and Kneller, SPARKY 3, UCSF], to identify spectra displaying peak shifts compared to the reference C9S_HisF spectrum. Of the 3 representative plates screened, 32 spectra were selected to move on for further screening in sets of 4 (2 μL @ 200 mM of each ligand + 12 μL d6-DMSO, for a total of 20 μL @ 20 mM). Of this set, 23 spectra were selected to move on for further screening one-by-one (2 μL @ 200 mM of ligand + 18 μL d6-DMSO, for a total of 20 μL @ 20 mM). Of these 92 ligands screened and spectra analyzed, 25 spectra displayed significant peak shifts, therefore these corresponding ligands were identified as hits. Once identified as hits, these ligands went through a final titration step, to observe how the peaks shift with ligand concentration. The ligands are ordered as 8 μL @ 200 mM of ligand + 32 μL d6-DMSO, for a total of 40 uL @ 40 mM, then distributed to the 96 well rack to give varying ligand concentrations. Six different [ligand] per well: 40, 80, 160, 400, 800, and 1200 μM, with 100 μM C9S_HisF. SoFast HMQC NMR experiments run, and all spectra analyzed in Sparky. Peak assignments transferred to the reference spectra ([ligand] = 0 μM), then step by step

transferred to the spectra corresponding to 40, 80, 160, 400, 800, and 1200 μM ligand. This process is repeated for each of the ligands.

Processing raw NMR titration data. Each protein-ligand complex has a set of peak assignments, for [ligand] at 0, 40, 80, 160, 400, 800, and 1200 μM, import these to Excel. All residue assignments have an assignment in the ^{1}H and the ^{15}N dimension, which are combined into one term and then the change being compared to the original position of that peak at [ligand] 0 μM. The 'combined shift' seen in the ^{1}H and ^{15}N is combined into one term by the equation:

(((NewHydrogenAssign-StandardHydrogenAssign)^2)+(((NewNitrogenAssign-StandardNitrogenAssign)/6.5)^2))^(1/2) (Assign = assignment; Standard = assignment at [ligand] 0 μM)[110,111].

$$\partial_{csd} = \sqrt{(H_x - H_0)^2 + ((N_x - N_0)/6.5)^2}$$

Complete 'combined shift' for [ligand] at 40, 80, 160, 400, 800, and 1200 μM. Calculate average of all combined shifts at [ligand] 1200 μM, then calculate standard deviation. Identify residues where the combined shift change at 1200 μM is at least one standard deviation higher than average at 1200 μM[106], these are the 'significant shifter' residues. Normalize these shifts to each other, so the combined shift change remains comparable from residue to residue. The normalization equation adds the combined shifts at 40, 80, 160, 400, 800, and 1200 μM together, then divides the individual combined shift by that total sum; the result is a decimal representing how much each [ligand] contributes to the total shift within 1.00. Some data points within the significant shifters do not follow the same pattern as the others. These were quantitatively identified as 'outliers' and excluded from Kd calculation. Identify data points (ligand concentration points) within a

residue set that are more than one standard deviation above or below the average normalized shifts within that [ligand]. Calculate the average and standard deviation for each [ligand]. Establish an upper (avg + stdev) and lower (avg – stdev) limit for each [ligand]. Use the "if then" function in excel to find the outliers. Based on observations in the data, excluded [ligand]s 40 μM and 80 μM due to possible human error in pipetting. The criteria to be an outlier in this case: two data points outside of the standard deviation of the average at that [ligand]. These residues were determined to be 'outliers' and therefore excluded from Kd calculations (but still included in images highlighting the significant shifters). Copy data over to Graphpad Prism to create binding curve and calculate Kd based on a non-linear regression curve equation[112]. GraphPad Prism is a comprehensive curve fitting software used to analyze and graph scientific data [GraphPad Prism version 6.04 for Windows, GraphPad Software, La Jolla California USA, www.graphpad.com].

Computational software: Rosetta and RosettaLigand. Rosetta is a protein modeling software suite for protein structure prediction and design[9]. Rosetta seeks to find the lowest energy conformation of a model by combining discrete side chain conformation (rotamer) optimization with Monte Carlo minimization[9]. This consists of sampling random perturbations of the backbone torsion angles, rigid body degrees of freedom, and rotamer conformations, followed by an all-over local minimization to resolve clashes[9]. The energy function that Rosetta uses to discriminate between native-like and non-native-like atom arrangements includes a van der Waals-like attractive and repulsive potential, solvation term, hydrogen bonding potential, electrostatics potential, rotamer probability, and protein backbone angle probabilities[6].

RosettaLigand, an application within Rosetta, docks small molecules into a protein with full protein and ligand flexibility[6,7]. Using the 3D coordinates of the protein and ligand as input files, the RosettaLigand protocol involves optimized placement of the ligand, optimized positioning of the surrounding residue side chains, and a minimization to resolve clashes. Protein sequence optimization can be included in the protocol if needed[40]. RosettaLigand allows for protein backbone flexibility, side chain rotamer searching, and full ligand flexibility, all of which are necessary for accurately modeling the interface[6,7]. For each model, RosettaLigand calculates an 'interface energy' as the total score of the protein-ligand complex minus the total score of the apo-protein[83]. Rosetta can also calculate the ddg, a prediction of how much every residue contributes to the protein-ligand interaction. Published literature describes the RosettaLigand method, protocol options, and tips for model analysis[41].

Computational screening process. The small molecules were docked into a model of C9S_HisF using RosettaLigand as described previously[41,107]. Mol files of the ligands were used to generate ligand conformers using confab. The mol files were then converted into Rosetta-readable params and pdb file. Since the binding site was unknown, ligands were placed in the center of the large convex face of C9S_HisF using the `StartFrom` mover in RosettaScripts[113]. The entire width of this pocket (10 Å) was used as the potential binding region and defined the box size (radius) as 6 Å such that the scoring grid would cover a space just larger than the potential binding pocket. In the first round of docking, the ligand was allowed freedoms of 1 Å step sizes and full 360° rotation in the `Transform` mover[72]. 5000 models were generated in the first round of docking. The models were sorted by overall Rosetta energy and then the top 10% were sorted by interface energy. The top 50

models by interface energy were used as input in the second round of docking. In the second round of docking, ligand movement was reduced to 0.2 Å step sizes and 45° rotation in the `Transform` mover. Again 5000 models were generated (100 from each parent model), and the same sorting scheme was used to identify the top 50 models. To ensure model diversity, no more than 4 models from a single parent model (or less than 10% of the 50 models) were allowed in the final set of 50 models. These 50 models were then used in a last round of docking in which the ligand freedom was reduced further to 0.04 Å step sizes and 5° rotation in the `Transform` mover. 5000 models were generated. After the top 10% by Rosetta energy score and sorting by interface score, the top 10 models were selected for analysis. In every stage of docking the InterfaceScoreCalculator[83] was used to determine the interface score of the ligand-protein complex. This is calculated by scoring the model of the complex and then moving the ligand 1000 Å away from the protein and rescoring. In addition to generating the total interface score, interface scores are calculated for each residue within a radius of the ligand molecule (6 Å for sidechain atoms, 7 Å for backbone atoms). This measures the strength of contribution of each side chain in its predicted interaction with the ligand. Through the subsequent rounds of models created, the final top 10 by interface score were selected for analysis.

Computationally determined 'interacting residues' and percent recovery by RosettaLigand. Of the top 10 models selected, the ddg interactions of all these were pooled together, then filtered by the top 200 interacting residues. This gives about 20 interacting residues per protein-ligand complex model. The interactions range from strongest ~-2 Rosetta energy units (REU) to weakest -0.1 REU (used -0.1 as a cutoff because anything smaller than that was deemed too weak of an interaction to count). These computationally

determined interacting residues were then compared to the experimentally determined significant shift residues (excluding the outliers). The number of significant shift residues contained in the list of interacting residues, divided by the total number of significant shifts, gives the percent recovered by RosettaLigand.

Acknowledgements

Work in the Meiler laboratory is supported through NIH (R01 GM099842). B.A. is supported through the National Science Foundation Graduate Research Fellowship Program, grant number DGE-0909667 and DGE 1445197. Supported in part by grants for NMR instrumentation from the NSF (0922862), NIH (S10 RR025677) and Vanderbilt University matching funds. This work was conducted in part using the resources of the Advanced Computing Center for Research and Education at Vanderbilt University.

CHAPTER 4: Rosetta and Design of Ligand Binding Sites

Moretti, Rocco; Bender, Brian; Allison, Brittany; Meiler, Jens, **2016**, Rosetta and Design of Ligand Binding Sites, Stoddard, B., ed., *Computational Design of Ligand Binding Proteins*, Humana Press, New York City. *Accepted*.

Contribution

I am a secondary author of this book chapter. I contributed figure 16, and material for the introduction. I also reviewed the text for clarity before submission.

Summary

Proteins that bind small molecules (ligands) can be used as biosensors, signal modulators, and sequestering agents. When naturally occurring proteins for a particular target ligand are not available, artificial proteins can be computationally designed. We present a protocol based off of RosettaLigand to redesign an existing protein pocket to bind a target ligand. Starting with a protein structure and the structure of the ligand, Rosetta can optimize both the placement of the ligand in the pocket and the identity and conformation of the surrounding sidechains, yielding proteins that bind the target compound.

Introduction

Proteins which bind to small molecules (i.e. ligands) are involved in many biological processes such as enzyme catalysis, receptor signaling, and metabolite transport. Designing these interactions can produce reagents which can serve as biosensors, *in vivo* diagnostics, signal modulators, molecular delivery devices, and sequestering agents[1,54,75,3,114].

Additionally, the computational design of proteins which bind small molecules serves as a critical test of our understanding of the principles that drive protein/ligand interactions. While *in vitro* techniques for the optimization of protein/ligand interactions have shown success[115], these are limited in the number of sequence variants which can be screened, and often require at least a modest starting affinity which to further optimize[59]. Computational techniques allow searching larger regions of sequence space and permit design in protein scaffolds with no detectable intrinsic affinity for the target ligand. Computational and *in vitro* techniques are often complementary and starting activity achieved via computational design can often be improved via *in vitro* techniques[37]. Although challenges remain, computational design of small molecule interactions have yielded success on a number of occasions[35],[114] and further attempts will refine our predictive ability to generate novel ligand binders.

The Rosetta macromolecular modeling software suite[9],[116] has proven to be a robust platform for protein design, having produced novel protein folds[8],[117], protein/DNA interactions[25], protein/peptide interactions[118], protein/protein interactions[24], and novel enzymes[31],[77],[30]. Technologies for designing protein/ligand interactions have also been developed and applied[3],[40],[37]. Design of ligand binding proteins using Rosetta approaches the problem in one of two way. One method derives from enzyme design, where predefined key interactions to the ligand are emplaced onto a protein scaffold and the surrounding context is subsequently optimized around them[37]. The other derives from ligand docking, in which the interactions with a movable ligand is optimized comprehensively[40],[3]. Both approaches have proven successful in protein redesign, and features from both can be

combined using the RosettaScripts system[113], tailoring the design protocol to particular design needs.

Here we present a protocol derived from RosettaLigand ligand docking[6,7,41], which designs a protein binding site around a given small molecule ligand (Figure 16). After preparing the protein and ligand structures, the placement of the ligand in the binding pocket is optimized, followed by optimization of sidechain identity and conformation. This process is repeated iteratively, and the proposed designs are sorted and filtered by a number of relevant structural metrics, such as predicted affinity and hydrogen bonding. This design process should be considered as part of integrated program of computational and experimental work, where proteins designed computationally are tested experimentally and the experimental results are used to inform subsequent rounds of computational design.

Materials

(1) A computer running a Unix-like operating system such as Linux or MacOS. Use of a multi-processor computational cluster is recommended for productions runs, although test runs and small production runs can be performed on conventional laptop and desktop systems.

(2) Rosetta: The Rosetta modeling package can be obtained from the RosettaCommons website (https://www.rosettacommons.org/software/license-and-download). Rosetta licenses are available free to academic users. Rosetta is provided as source code and must be compiled before use. See the Rosetta Documentation (https://www.rosettacommons.org/docs/latest/) for instructions on how to compile

Rosetta. The protocol in this paper has been tested with Rosetta weekly release version 2015.12.57698.

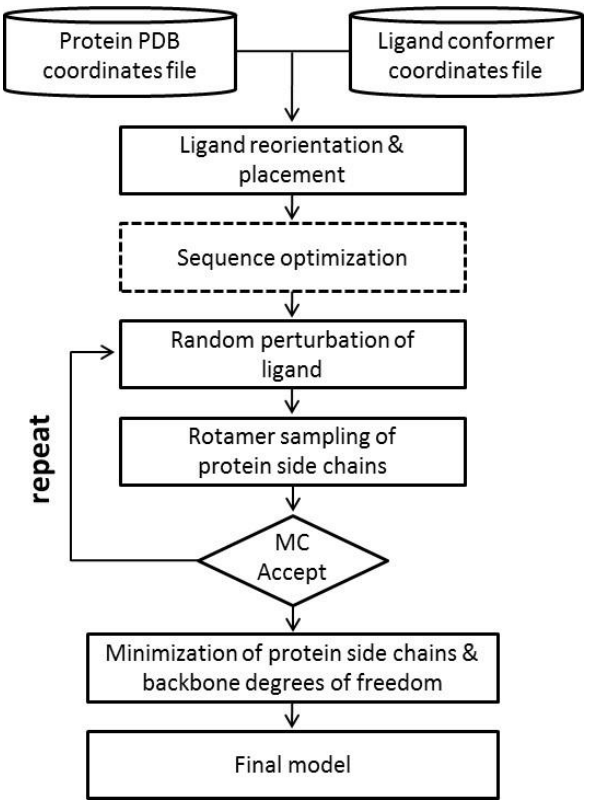


Figure 16. Flowchart of ligand design protocol. This protocol based on RosettaLigand includes interface design (dotted line box). From the input coordinates, the ligand is allowed to rotate and translate before sequence optimization of nearby residues. After several cycles of small molecule perturbation, side chain rotamer sampling, Monte Carlo (MC) minimization, and a final gradient-based minimization of the protein to resolve any clashes, the final model is output. Most variables in this protocol are user-defined, and the user will determine which variables best fit the protein-ligand complex in the study.

(3) A program to manipulate small molecules: OpenBabel[119] is a free software package which allows manipulation of many small molecule file formats. See http://openbabel.org/ for download an installation information. The protocol in this paper has been tested with OpenBabel version 2.3.1. Other small molecule manipulation programs can also be used.

(4) A ligand conformer generation program: We recommend the BCL[120] which is freely available from http://meilerlab.org/index.php/bclcommons for academic use but does require an additional license to the Cambridge Structural Database[121] for conformer generation. The protocol in this paper has been tested with BCL version 3.2. Other conformer generation programs such as Omega[122], MOE[123], or RDKit[124] can also be used.

(5) The structure of the target small molecule in a standard format such as SDF or SMILES (*see* **Note 3**).

(6) The structure of the protein to be redesigned, in PDB format (*see* **Note 1&2**).

Methods

Throughout the protocol $(ROSETTA) represents the directory in which Rosetta has been installed. File contents and commands to be run in the terminal are in *italics*. The use of a bash shell is assumed – users of other shells may need to modify the syntax of command lines.

(1) Pre-relax the protein structure into the Rosetta scoring function[125]. Structure from non-Rosetta sources or structures from other Rosetta protocols can have minor structural variations resulting in energetic penalties which adversely affect the design process (*see* **Note 4&5**).

$(ROSETTA)/main/source/bin/relax.linuxgccrelease -ignore_unrecognized_res -ignore_zero_occupancy_false -use_input_sc -flip_HNQ -no_optH false -relax:constrain_relax_to_start_coords -relax:coord_constrain_sidechains -relax:ramp_constraints false -s PDB.pdb

For convenience, rename the output structure.

mv PDB_0001.pdb PDB_relaxed.pdb

(2) Prepare the ligand

(2.1) Convert the small molecule to SDF format, including adding hydrogens as needed (*see* **Note 6**).

obabel LIG.smi --gen3D -O LIG_3D.sdf

obabel LIG_3D.sdf -p 7.4 -O LIG.sdf

(2.2) (Generate a library of ligand conformers (*see* **Note 7&8**).

bcl.exe molecule:ConformerGenerator -top_models 100 -ensemble_filenames LIG.sdf -conformers_single_file LIG_conf.sdf

(2.3) Convert the conformer library into a Rosetta-formatted "params file" (*see* **Note 9**).

$(ROSETTA)/main/source/src/python/apps/public/molfile_to_params.py -n LIG -p LIG --conformers-in-one-file LIG_conf.sdf

This will produce three files: LIG.params, a Rosetta-readable description of the ligand; LIG.pdb, a selected ligand conformer; LIG_conformers.pdb, the set of all conformers (*see* **Note 11**).

(3) Place the ligand into the protein (*see* **Note 12&13**).

(3.1) Identify the location of desired interaction pockets. Visual inspection using programs like PyMol or Chimera[126] is normally the easiest method (*see* **Note 14**). Use the structure editing mode of PyMol to move the LIG.pdb file from step 2.3 into the starting conformation. Save the repositioned molecule with its new coordinates as a new file (LIG_positioned.pdb) (*see* **Note 15**).

(3.2) If necessary, use a text editor to make the ligand to be residue 1 on chain X (*see* **Note 16**).

(3.3) Using a structure viewing program, inspect and validate the placement of the ligand (LIG_positioned.pdb) in the binding pocket of the protein (PDB_relaxed.pdb) (*see* **Note 17**).

(4) Run Rosetta design

(4.1) Prepare a residue specification file. A Rosetta resfile allows specification of which residues should be designed and which shouldn't. A good default is a resfile which permits design at all residues at the auto-detected interface (*see* **Note 18**).

ALLAA

AUTO

start

1 X NATAA

(4.2) Prepare a docking and design script(design.xml) This particular protocol is based off of RosettaLigand docking using the RosettaScripts framework[6,7,41]. It will optimize the location of ligand in the binding pocket (low_res_dock), redesign the surrounding sidechains (design_interface), and refine the interactions in the designed context (high_res_dock). To avoid spurious mutations, a slight energetic bonus is given to the input residue at each position (favor_native).

```
<ROSETTASCRIPTS>
  <SCOREFXNS>
    <ligand_soft_rep weights=ligand_soft_rep />
    <hard_rep weights=ligandprime />
  </SCOREFXNS>
  <TASKOPERATIONS>
```

<DetectProteinLigandInterface name=design_interface cut1=6.0 cut2=8.0 cut3=10.0 cut4=12.0 design=1 resfile="PDB.resfile"/> # see ***Note 19***

</TASKOPERATIONS>

<LIGAND_AREAS>

<docking_sidechain chain=X cutoff=6.0 add_nbr_radius=true all_atom_mode=true minimize_ligand=10/>

<final_sidechain chain=X cutoff=6.0 add_nbr_radius=true all_atom_mode=true/>

<final_backbone chain=X cutoff=7.0 add_nbr_radius=false all_atom_mode=true Calpha_restraints=0.3/>

</LIGAND_AREAS>

<INTERFACE_BUILDERS>

<side_chain_for_docking ligand_areas=docking_sidechain/>

<side_chain_for_final ligand_areas=final_sidechain/>

<backbone ligand_areas=final_backbone extension_window=3/>

</INTERFACE_BUILDERS>

<MOVEMAP_BUILDERS>

<docking sc_interface=side_chain_for_docking minimize_water=true/>

<final sc_interface=side_chain_for_final bb_interface=backbone minimize_water=true/>

</MOVEMAP_BUILDERS>

<SCORINGGRIDS ligand_chain=X width=15> # see ***Note 20***

<vdw grid_type=ClassicGrid weight=1.0/>

</SCORINGGRIDS>

<MOVERS>

<FavorNativeResidue name=favor_native bonus=1.00 /> # see ***Note 21&22***

<Transform name=transform chain=X box_size=5.0 move_distance=0.1 angle=5 cycles=500 repeats=1 temperature=5 rmsd=4.0 /> # see ***Note 23***

<HighResDocker name=high_res_docker cycles=6 repack_every_Nth=3 scorefxn=ligand_soft_rep movemap_builder=docking/>

<PackRotamersMover name=designinterface scorefxn=hard_rep task_operations=design_interface/>

<FinalMinimizer name=final scorefxn=hard_rep movemap_builder=final/>

<InterfaceScoreCalculator name=add_scores chains=X scorefxn=hard_rep />

<ParsedProtocol name=low_res_dock>

<Add mover_name=transform/>

</ParsedProtocol>

<ParsedProtocol name=high_res_dock>

<Add mover_name=high_res_docker/>

<Add mover_name=final/>

</ParsedProtocol>

</MOVERS>

<PROTOCOLS>

<Add mover_name=favor_native/>

<Add mover_name=low_res_dock/>

<Add mover_name=design_interface/> # see ***Note 24***

<Add mover_name=high_res_dock/>

<Add mover_name=add_scores/>

</PROTOCOLS>

</ROSETTASCRIPTS>

(4.3) Prepare an options file ("design.options"). Rosetta options can be specified either on the command line or in a file. It is convenient to put options which do not change run-to-run (such as those controlling packing/sidechain placement and scoring) in an options file rather than the command line.

-packing

-ex1

-ex2

-linmem_ig 10

-restore_pre_talaris_2013_behavior # see ***Note 25***

(4.4) Run the design application (*see* **Note 26&27**). This will produce a number of output PDB files (named according to the input file names), and a summary score file ("design_results.sc").

$(ROSETTA)/main/source/bin/rosetta_scripts.linuxgccrelease @design.options -parser:protocol design.xml -extra_res_fa LIG.params -s "PDB_relaxed.pdb LIG_positioned.pdb" -nstruct <number of output models> -out:file:scorefile design_results.sc

(5) Selection of the Designs

A rule of thumb is that filtering should remove unlikely solutions, rather than selecting the single "best" result. Successful designs are typically good across a range of relevant metrics, rather than being the best structure on a single metric (*see* **Note 28**).

The metrics to use can vary based on the desired properties of the final design. Good standard metrics include the predicted interaction energy of the ligand, the stability score of the complex as a whole, the presence of any clashes[127], shape complementarity of the protein/ligand interface[128], the interface area, the energy density of the interface (binding energy per unit of interface area), and the number of unsatisfied hydrogen bonds formed on binding.

(5.1) Prepare a file (metric_thresholds.txt) specifying thresholds at which to filter runs. IMPORTANT: The exact values of the thresholds need to be tuned for your particular system (*see* **Note 29**).

```
req total_score value < -1010          # measure of protein stability
req if_X_fa_rep value < 1.0            # measure of ligand clashes
req ligand_is_touching_X value > 0.5  # 1.0 if ligand is in pocket
output sortmin interface_delta_X       # binding energy
```

(5.2) Filter on initial metrics from the docking run.

```
perl $(ROSETTA)/main/source/src/apps/public/enzdes/DesignSelect.pl -d <(grep SCORE
design_results.sc) -c metric_thresholds.txt -tag_column last > filtered_designs.sc

awk '{print $NF ".pdb"}' filtered_designs.sc  > filtered_pdbs.txt
```

(5.3) Calculate additional metrics (*see* **Note 30**). Rosetta's InterfaceAnalyzer[129] calculates a number of additional metrics. These can take time to evaluate, though, so are best run on only a pre-filtered set of structures. After the metrics are generated, the structures can be filtered as in steps 5.1-5.2.

$(ROSETTA)/main/source/bin/InterfaceAnalyzer.linuxgccrelease -interface A_X -compute_packstat -pack_separated -score:weights ligandprime -no_nstruct_label -out:file:score_only design_interfaces.sc -l filtered_pdbs.txt -extra_res_fa LIG.params

Contents of metric_thresholds2.txt, an example filtering file used with design_interfaces.sc, the output scorefile from the InterfaceAnalyzer.

req packstat value > 0.55 # packing metric; 0-1 higher better

req sc_value value > 0.45 # shape complementarity; 0-1 higher better

req delta_unsatHbonds value < 1.5 # unsatisfied hydrogen bonds on binding

req dG_separated/dSASAx100 value < -0.5 # binding energy per contact area

output sortmin dG_separated # binding energy

(6) Manual inspection of selected sequences. While automated procedures are continually improving and can substitute to a limited extent[130], there is still no substitute for expert human knowledge in evaluating designs. Visual inspection of interfaces can capture system-specific requirements that are difficult to encode into an automated filter. (*see* **Note 31**).

(7) Extract protein sequences from the final selected designs into FASTA format.

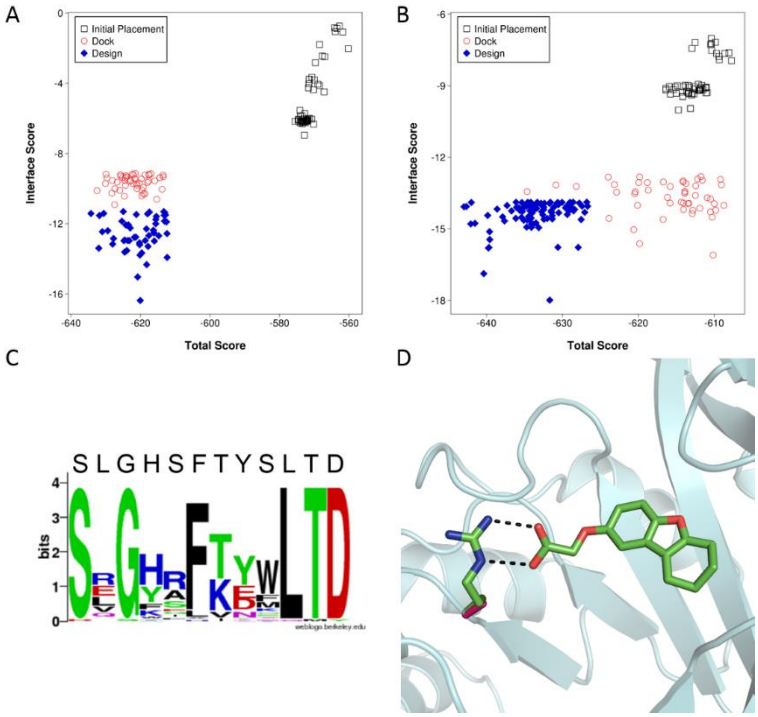


Figure 17**.** Interface Design with RosettaLigand. (A,B) Comparison in improvements in Interface Score and Total Score for top models from an initial placement, docking without sequence design, and docking with design for a hydrophilic and hydrophobic ligand, respectively. (C) Sequence logo of mutation sites among the top models from a round of interface design [Crooks *et al*. *Genome Research* **2004** *14*, 1188-1190.]. For most positions, the consensus sequence is the native sequence. Amino acids with side chains that directly interact with the ligand show a high prevalence to mutation as seen in the positions with decreased consensus. (D) Example of a typical mutation introduced by RosettaLigand. The protein structure is represented in cartoon (cyan). The native alanine (pink) is mutated to a lysine residue (green) to match ionic interactions with the negatively charged ligand (green). Image generated in PyMol.

$(ROSETTA)/main/source/src/python/apps/public/pdb2fasta.py $(cat final_filtered_pdbs.txt) > selected_sequences.fasta

(8) Iteration of design. Only rarely will the initial design from a computational protocol give exactly the desired results. Often it is necessary to perform iterative cycles of design and experiment, using information learned from experiment to alter the design process (Figure 17)**.**

Notes

(1) High resolution experimental structures determined in complex with a closely related ligand are most desirable, but not required. Experimental structures of the unliganded protein and even homology models can be used[107,131].

(2) While Rosetta can ignore chainbreaks and missing loops far from the binding site, the structure of the protein should be complete in the region of ligand binding. If the binding pocket is missing residues, remodel these with a comparative modeling protocol, using the starting structure as a template.

(3) Acceptable formats depend on the capabilities of your small molecule handling program. OpenBabel can be used to convert most small molecule representations, including SMILES and InChI, into the sdf format needed by Rosetta.

(4) The option "-relax:coord_constrain_sidechains" should be omitted if the starting conformation of the sidechains are from modeling rather than experimental results.

(5) Rosetta applications encode the compilation conditions in their filename. Applications may have names which end with *.linuxgccrelease, *.macosclangrelease, *.linuxiccrelease, etc. Use whichever ending is produced on for your system. Applications ending in "debug" have additional error checking which slows down production runs.

(6) It is important to add hydrogens for the physiological conditions under which you wish to design. At neutral pH, for example, amines should be protonated and carboxylates deprotonated. The "-p" option of OpenBabel uses heuristic rules to reprotonate molecules for a given pH value. Apolar hydrogens should also be present.

(7) Visually examine the produced conformers and manually remove any which are folded back on themselves or are otherwise unsuitable for being the target design conformation.

(8) It is unnecessary to sample hydrogen position during rotamer generation, although any ring flip or relevant heavy atom isomeric changes should be sampled.

(9) molfile_to_params.py can take a number of options – run with the "-h" option for details. The most important ones are: "-n", which allows you to specify a three letter code to use with the PDB file reading and writing, permitting you to mix multiple ligands; "-p", which specifies output file naming; "--recharge", which is used to specify the net charge on the ligand if not correctly autodetected; "--nbr_atom", which allows you to specify a neighbor atom (*see* **Note 10**)

(10) Specifying the neighbor atom is important for ligands with offset "cores". The neighbor atom is the atom which is superimposed when conformers are exchanged. By default the neighbor atom is the "most central" atom. If you have a ligand with a core that should be stable when changing conformers, you should specify an atom in that core as the neighbor atom.

(11) LIG.params expects LIG_conformers.pdb to be in the same directory, so keep them together when moving files to a new directory. If you change the name of the files, you will need to adjust the value of the PDB_ROTAMERS line in the LIG.params file.

(12) Rosetta expects the atom names to match those generated in the molfile_to_params.py step. Even if you have a starting structure with the ligand correctly placed, you should align the molfile_to_params.py generated structure into the pocket so that atom naming is correct.

(13) Other methods of placing the ligand in the pocket are also possible. Notably, Tinberg et al.[37] used RosettaMatch[76] both to place the ligand in an appropriate scaffold and to place key interactions in the scaffold.

(14) Other pocket detection algorithms can also be used[132]

(15) If you have a particularly large pocket, or multiple potential pockets, save separate ligand structures at different positions and perform multiple design runs. For a large number of locations, the StartFrom mover in RosettaScripts can be used to randomly place the ligand at multiple specified locations in a single run.

(16) Being chain X residue 1 should be the default for molfile_to_params.py produced structures. Chain identity is important as the protocol can be used to design for ligand binding in the presence of cofactors or multiple ligands. For fixed-location cofactors, simply change the PDB chain of the cofactor to something other than X, add the cofactor to the input protein structure, and add the cofactors' params file to the -extra_res_fa commandline option. For designing to multiple movable ligands, including explicit waters, see Lemmon, et al.[95].

(17) To refine the initial starting position of the ligand in the protein, you can do a few "design" runs as in step 4, but with design turned off. Change the value of the design option in the DetectProteinLigandInterface tag to zero. A good starting structure will likely have good total scores and good interface energy from these runs, but will unlikely to result in ideal interactions. Pay more attention to the position and orientation of the ligand than to the energetics of this initial placement docking run (*see* **Note 18**).

(18) The exact resfile to use will depend on system-specific knowledge of the protein structure and desired interactions. Relevant commands are ALLAA (allow design to all amino acids) PIKAA (allow design to only specified amino acids) NATAA (disallow design but permit sidechain movement) and NATRO (disallow sidechain movement). The AUTO

specification allows the DetectProteinLigandInterface task operation to remove design and sidechain movement from residues which are “too far” from the ligand.

(19) Change the name of the resfile in the XML script to match the full path and filename of the resfile you’re using. The cut values decide which residues with the AUTO specification to design. All residues with a C-beta atom within cut1 Angstroms of the ligand will be designed, as will all residues within cut2 which are pointing toward the ligand. The logic in selecting sidechains is similar for cut3 and cut4, respectively, but with sidechain flexibility rather than design. Anything outside of the cut shells will be ignored during the design phase, but may be moved during other phases.

(20) The grid width must be large enough to accommodate the ligand. For longer ligands, increase the value to at least the maximum extended length of the ligand plus twice the value of box_size in the Transform mover.

(21) Allison et al. found that a value of 1.0 for the FavorNativeSequence bonus worked best over their benchmark set[40]. Depending on your particular requirements, though, you may wish to adjust this value. Do a few test runs with different values of the bonus and examine the number of mutations which result. If there are more mutations than desired, increase the bonus. If fewer than expected, decrease the bonus.

(22) More complicated native favoring schemes can be devised by using FavorSequenceProfile instead of FavorNativeSequence. For example, you can add weights according to BLOSUM62 relatedness scores, or even use a BLAST-formatted position specific scoring matrix (PSSM) to weight the bonus based on the distribution of sequences seen in homologous proteins.

(23) The value of box_size sets the maximum rigid body displacement of the ligand from the starting position. The value of rmsd sets the maximum rmsd from the starting position. Set these to smaller values if you wish to keep the designed ligand closer to the starting conformation, and to large values if you want to permit more movement. These are limits for the active sampling stage of the protocol only. Additional movement may occur during other stages of the protocol.

(24) The provided protocol only does one round of design and minimization. Additional rounds may be desired for further refinement. Simply replicate the low_res_dock, design_interface, and high_res_dock lines in the PROTOCOLS section to add additional rounds of design and optimization. Alternatively, the EnzRepackMinimize mover may be used for finer control of cycles of design and minimization (although it does not incorporate any rigid body sampling).

(25) Refinement of the Rosetta scorefunction for design of protein/ligand interfaces is an area of current active research. The provided protocol uses the standard ligand docking scorefunction which was optimized prior to the scorefunction changes which occurred in 2013 and thus requires an option to revert the behavior. Decent design performance has also been seen with the “enzdes” scorfunction (which also requires the -restore_pre_talaris_2013 option) and the standard “talaris2013” scorefunction.

(26) Use of a computational cluster is recommended for large production runs. Talk to your local cluster administrator for instructions on how to launch jobs on your particular cluster system. The design runs are “trivially parallel” and can either be manually split or run with an MPI-compiled version. If splitting manually, change the value of the -nstruct option to reduce the number of structures produced by each job, and use the options -out:file:prefix

or -out:file:suffix to uniquely label each run. The MPI version of rosetta_scripts can automatically handle distributing structures to multiple CPUs, but requires Rosetta to be compiled and launched in cluster-specific ways. See the Rosetta documentation for details.

(27) The number of output models needed (the value passed to -nstruct) will depend on the size of the protein pocket and the extent of remodeling needed. Normally, 1000-5000 models is a good sized run for a single starting structure and protocols. At a certain point, you will reach "convergence" and the additional models will not show appreciable metric improvement or sequence differences. If you have additional computational resources, it's often better to run multiple smaller runs (100-1000 models) with slightly varying protocols (different starting location, number of rounds, extent of optimization, native bonus, etc.), rather than have a larger number of structures from the identical protocol.

(28) Relevant metrics can be determined by using "positive controls". That is, run the design protocol on known protein-ligand interactions which resemble your desired interactions. By examining how the known ligand-protein complexes behave under the Rosetta protocol, you can identify features which are useful for distinguishing native-like interactions from non-native interactions. Likewise, "negative controls", where the design protocol is run without design (*see* **Note 17**) can be useful for establishing baseline metric values.

(29) The thresholds to use are system-specific. A good rule of thumb is to discard at least a tenth to a quarter by each relevant metric. More important metrics can receive stricter thresholds. You may wish to plot the distribution of scores to see if there is a natural threshold to set the cut at. You will likely need to do several test runs to adjust the thresholds to levels which give the reasonable numbers of output sequences.

(30) Other system-specific metric values are available through the RosettaScripts interface as “Filters”. Adding “confidence=0” in the filter definition tag will turn off the filtering behavior and will instead just report the calculated metric for the final structure in the final score file. Many custom metrics, such as specific atom-atom distances, can be constructed in this fashion.

(31) Certain automated protocol can ease this post-analysis. For example, Rosetta can sometimes produce mutations with minor influence on binding energy. While the native bonus (*see* **Notes 21&22**) mitigates this somewhat, explicitly considering mutation-by-mutation reversions, as with the protocol of Nivon *et al.*[130] can further reduce the number of such “spurious” mutations seen.

Acknowledgments

This work is supported through NIH (R01 GM099842, R01 DK097376, R01 GM073151) and NSF (CHE 1305874). RM is further partially supported by grant from the RosettaCommons.

CHAPTER 5: Conclusion and Future Directions

Macromolecule-small molecule interactions drive much of the world around us. Out of an infinite number of possibilities, nature has created receptor-based interfaces that proceed with accuracy and precision every time (and when these are disrupted, illness/disease occur). If we could understand and then recapture such accurate and precise interactions, the biotechnology field would burst in numerous directions. By carrying out our experiments and sharing the results with the scientific community, we collectively contribute to making this a reality. Successes, as well as failures, provide insight into our understanding and what questions to ask next. During my time as a PhD student, my research and results contribute to the rapidly growing knowledge set we have about protein-ligand binding, and especially how this knowledge can be applied to computational modeling. I have established benchmark datasets, protocols, and data analysis pipelines that have already been used by other lab members, and available for anyone who would need to repeat or update any of my experiments in the future. Looking towards the future, I see many ways that my project can spin into subsequent projects of a similar nature.

Summarization of key findings and future directions: Computational Design of Protein Small Molecule Interfaces (manuscript). This experiment allowed us to determine overall protein-ligand interface sequence recovery as well as an optimal strategy for re-designing proteins to recognize different small molecules using a minimal set of mutations. The benchmark consisted of two parts. Part one tested overall sequence recovery when all

residues within the protein-small molecule interface are allowed to change identity. Part two simulated a protein-small molecule design more closely by mutating up to five residues that contribute most to the interaction with the small molecule to alanine (and a scoring bonus was applied to the starting sequence in this step, therefore mutations had to be better than the starting residue for the mutation to occur). We used RosettaLigand on a diverse set of 43 protein-ligand complexes (from the CSAR dataset, filtered to our needs). On average, we achieve sequence recoveries in the binding site of 59% when the ligand is allowed limited reorientation (ie starts in the correct binding position), and 48% when the ligand is allowed full reorientation. When simulating the redesign of a protein binding site, sequence recovery among residues that contribute most to binding was 52% when slight ligand reorientation was allowed, and 27% when full ligand reorientation was allowed. One key finding, which was expected, is that recovery of the correct ligand position is a pre-requisite for recovering the residues necessary for binding. This makes a strong argument for computational programs solely dedicated to optimizing algorithms for ligand docking (virtual high throughput screening). A question about measuring sequence recovery however, is that when the ligand position is not recovered, do the subsequent mutations actually support binding interactions? This would be a follow-up study of interest. Another point of interest is how we define 'sequence recovery'. In my manuscript, I focus on recovery as recapturing the wild type residue, but there could be tolerable mutations that would produce the same result, for example a non-polar residue swapped out for a similarly sized non-polar residue (ie Val to Leu), or a hydrogen bond donor swapped out for another (ie Ser to Thr). We account for this phenomenon by including Position-Specific Scoring Matrix (PSSM) analysis, which identifies amino acid mutations that are tolerated in

homologous proteins. While PSSM recovery has its limitations (described in the text), this allows for a more robust judgment of the program's ability to capture residues favorable for interaction. The PSSM recovery was calculated over overall sequence recovery, but if a similar benchmark is conducted in the future, it would be of interest to also include PSSM recovery specifically of the alanine to wild-type mutations. I would also look deeper for alternative ways to account for 'similar residue' mutations, which would require an extensive search of docking/design programs and their measure of recoveries. Also for future studies, reporting the recovery on a residue by residue basis (as was done for the PSSM recovery) would provide valuable insight to which residues RosettaLigand recovers often or seldom. Recovery could be unfairly biased by certain residues that are never recovered (as what happened in my second manuscript, discussed below), and we could investigate underlying causes. Identifying caveats in the program sets the stage for future projects. For example, it is known that Rosetta struggles with π-π interactions, which was reflected in this paper by phenylalanine and tryptophan often designed out. An entire manuscript was dedicated to experiments investigating and hopefully improving this problem (Combs et al, 'Partial Covalent Interactions in Rosetta', results published soon). Also based on my results, future Rosetta project could investigate: design and recovery among complex hydrogen-bond networks, design/recovery among highly flexible ligands, and design/recovery among binding guided by weak interactions. Another area to investigate would be if there should be some type of stringency or allowance (an applied weight, like FNRB) based on binding pocket crowdedness, calculated as the number protein/ligand atom pairs within 3 Å of each other, divided by the total number of ligand atoms. Highest recoveries were seen between 2 – 3 contacts; why does the program

struggle when the pocket is very tight or very loose, is a valuable question to consider. In the computational community, we often discuss under what condition the program is 'most successful', compared to where the program struggles. But within any range of 'most successful', there are just as many data points that indicate less successful recoveries. For example, we note that alanine to wild type recovery is best when the correct ligand position is recovered (within 2 Å is accepted), but within 2 Å there are many data points with recovery below 25%. Also, recovery is best when number of ligand hydrogen bond donors + acceptors is less than 8, but within that range there are many with recoveries under 25%. I think it would be interesting to look at the data points that, even under the most optimal conditions, still struggle with recovery. RosettaLigand, like many docking programs, saw no trend in recovery vs binding affinity. It would be a huge accomplishment for a docking program to make such a correlation. Because ligand binding is governed by numerous factors, all of which are not well incorporated into the algorithms (or maybe even known yet), the programs fail to make reliable and consistent binding affinity correlations. In addition to new ideas and projects to pursue, it is also interesting to repeat benchmarks with updated versions of the program, to make direct comparisons of progress. The literature has pointed out the value of reliable and consistent benchmarks, to truly assess how the program is improving and how programs compare to one another[133],[67]. Since I published this manuscript, DeLuca et al combined two steps in the docking protocol into one step (rotation and translation combined into transform), and saw improved ligand position recoveries of 10 – 15% [72]. (Using the very same CSAR dataset of 43 protein-ligand complexes that I created.) Since the 'transform' update has been incorporated into the program, we should repeat my experiment with this new addition.

We should also repeat the benchmark to optimize the 'favor native residue bonus', as new algorithm incorporations may change the bonus needed in allowing mutations. As I mentioned earlier, available structural information guides computational studies. The CSAR database contains a diverse set of protein-small molecule crystal structures and includes information on binding affinities, and at the time of my computational experiments, contained 343 complexes[94]. As of 2013, they have added more complexes to the database, possibly including complexes that meet our filtering standards[134]. Every benchmark provides more insight and guides new ideas to be explored.

Summarization of key findings and future directions: Experimental and Computational Identification of Naïve Binders to a TIM-Barrel Protein Scaffold (manuscript). In this study, we experimentally identified ligands with intrinsic binding affinity for a protein, and then used computational methods to re-create the experiment. Of ~3500 small molecules screened, 28 displayed intrinsic binding affinity for C9S_HisF, dissociation constants ranging between 338 – 1112 μM. The computational docking program RosettaLigand was assessed in its ability to recapture the correct ligand binding position by identifying the residues that contribute to the protein-ligand interaction. One key finding indicated that interacting residues were most successfully recovered when there are strong bonds contributing to the interaction, such as hydrogen bonds. On the experimental side, it should be noted that HisF from *Thermotoga maritima* provides a very stable and easy protein to work with, therefore allowing us to produce enough protein for the screening experiments. Working with a protein that is difficult to express and purify would significantly decrease the chance to carry out the experiments needed. It should also be noted that the ^{15}N-HMQC-NMR experiment was a reliable tool in identifying the naïve binders. For follow up studies

and continuing forward, I propose some thoughts. C9S_HisF was screened at 100 μM, but experiments carried out afterwards (Designed C9S_HisF binds VU0068924 more tightly), high quality spectra were produced at 50 μM. If someone were to repeat this experiment or conduct a similar one, I would suggest screening at 50 μM, even if only at the titration stage, to reduce the amount of protein that needs to be produced. Also, considering the size of the binding pocket, it would be of interest to expand the Chemcart search for 'similar ligands' to include ligands with more flexibility (4+ rotatable bonds) and increased molecular weight. These were excluded to filter the search to ligands that would be included in the small molecule library, but if there are larger ligands available to us, these could be screened also. It would also be advantageous to have more of the binding pocket residues resonance assigned on the 2D spectrum. Many assignments were lost transferring from HisF to C9S_HisF. Even if a full assignment cannot be done, one could seek out alternative assignment transfer routes. For example, would it have been more advantageous to do the TROSY to HMQC and 30° C to 25° C transfers first, and the HisF to C9S_HisF transfer last? This could be something to investigate, especially if follow-up studies continue to use the C9S_HisF 2D assigned spectrum. A C9S_HisF crystal structure would also be advantageous, as well as C9S_HisF crystallized with some of the naïve binders. Crystal structure validation of the binding position and interacting side chains would add great depth to the computational analysis. This project could also be expanded by conducting a similar screening experiment with a different TIM-barrel. The limiting factor, however, is the necessity of a resonance assigned 2D spectrum. A quick search in the journal of Biomolecular NMR Assignments gave no results for indole-3-glycerol phosphate synthase and phosphoribosyl anthranilate isomerase, two TIM-barrels often compared to HisF[135],[136].

These proteins do have crystal structures in complex with bound ligands, so perhaps NMR studies could be used as a 'crude' way to identify ligands that cause peak shifts, and then repeat crystallization conditions in hopes of elucidating the complex structure and confirming the bound ligand in that way. Another approach would be to use a different protein altogether, one that is thermostable, has a resonance assigned 2D spectrum, and has an apo and ligand-bound crystal structure. Results on the computational side provide insight and ideas in moving forward as well. As reflected in my manuscript above, RosettaLigand recovery was highest when binding is dominated by strong polar interactions. Recovery is highest at 6 – 8 hydrogen bond donors + acceptors; recovery increases as topological surface area approaches 100, and decreases as LogP approaches 5. Also as noted above, digging deeper into our measure of recovery provided valuable insight. Val127 and L170, pointing away from the ligand, were never computationally recovered as 'interacting residues', yet experimental results indicated that they do. These results could guide a Rosetta project that investigates how the ddg is calculated for protein-ligand interactions, to not only include side chain measurements but backbone measurements as well. In addition to the computational analysis included in this manuscript, identified naïve binders will also be used as the starting point for designing tighter interactions, to be published in a later manuscript.

As I review the body of work presented, I am proud of my results and what I have accomplished over the past six years. I look forward to seeing how the field of protein-ligand continues to grow, and how my research contributes to future computational and experimental projects.

APPENDIX

Appendix A: Computational Design of Protein-Small Molecule Interfaces

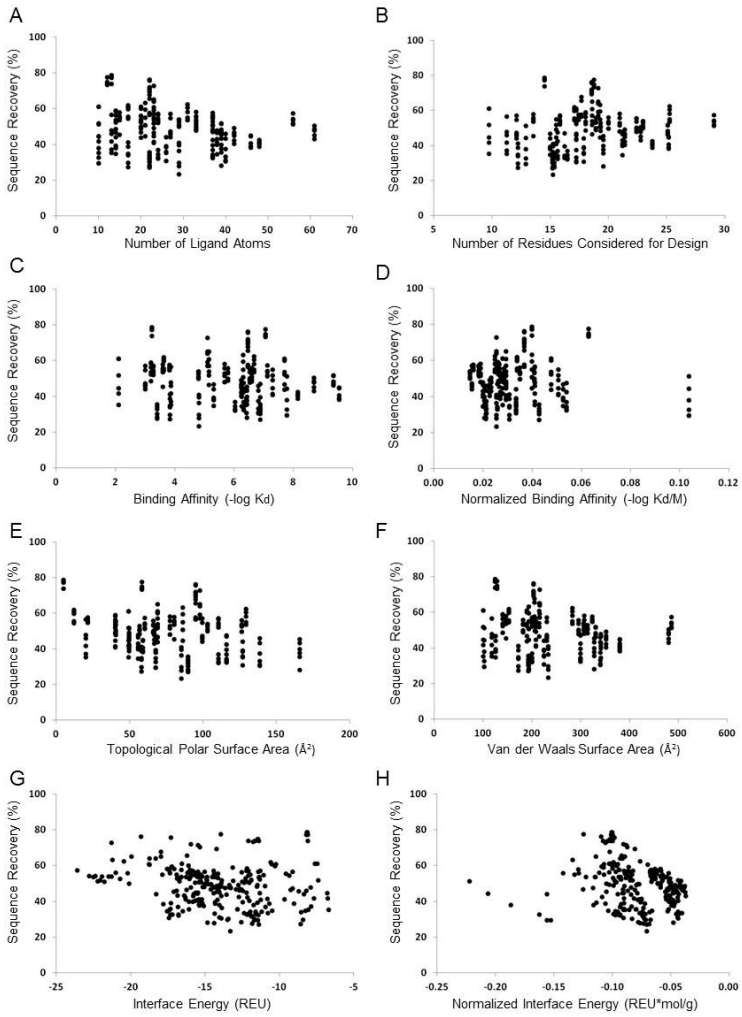


Figure 18 (Supplement Figure 1): Sequence recovery from Dock/Design Alanine Mutants experiment. Aside from a few outliers, increasing the number of ligand atoms decreases the recovery (A). Number of residues considered for design has little effect on recovery (B). Weak/moderate binders have the maximum recoveries, but overall no trend (C). The binding affinity normalized by ligand molecular weight shows no effect on recovery (D). Increasing the topological polar surface area (E) and increasing the van der Waals surface area (F) decreases the chance to achieve maximum recoveries. The Rosetta interface energy (G) and the Rosetta interface energy normalized by ligand molecular weight (H) have little effect on recovery. These results imply that for parameters correlated with ligand size, RosettaLigand recovers the interface better for moderately sized ligands over larger ligands. (Reprinted from Journal of Structural Biology.)

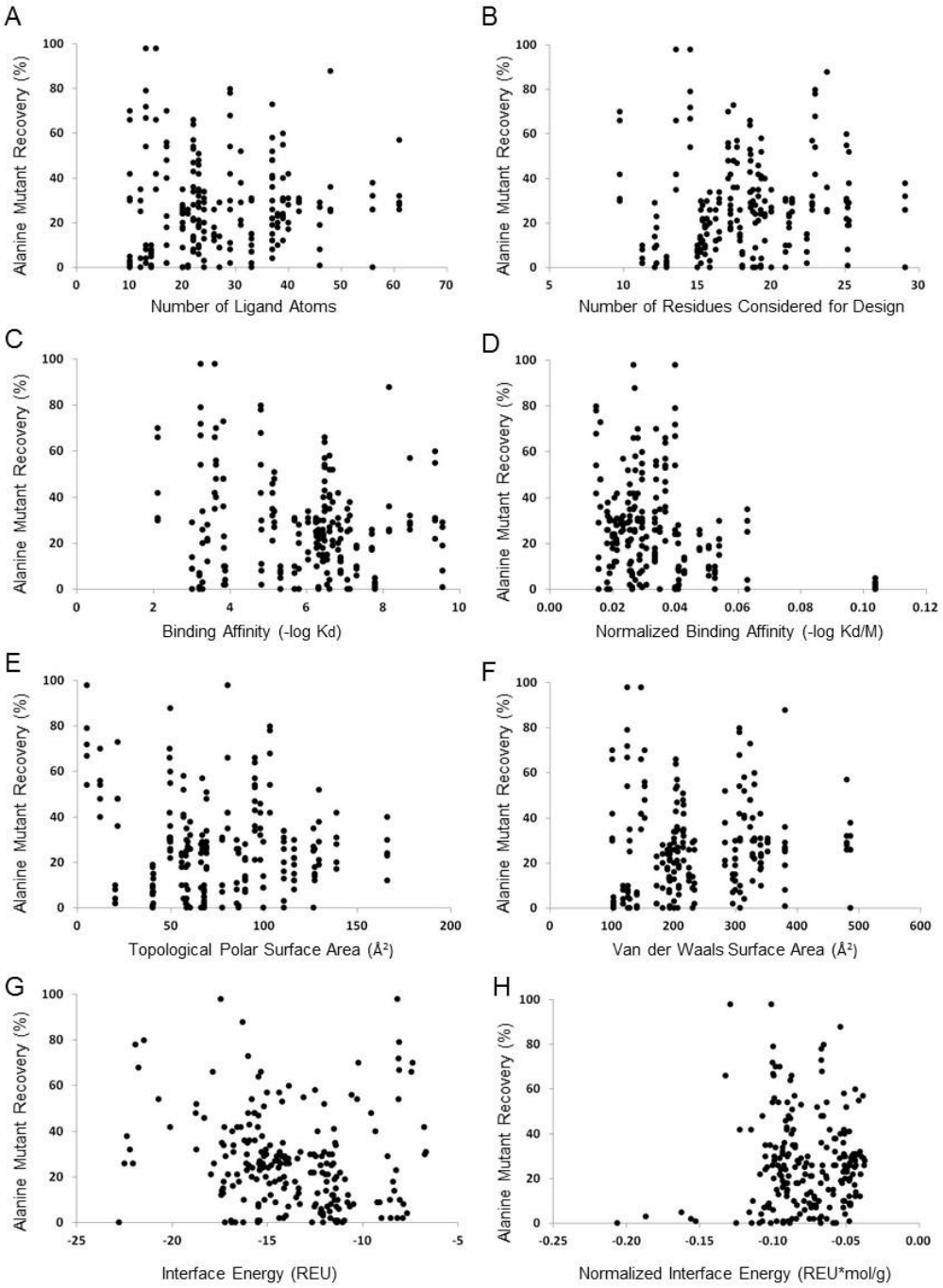


Figure 19 (SF2): Alanine Mutant to Wild Type recovery from Dock/Design Alanine Mutants experiment. Aside from a few outliers, increasing the number of ligand atoms decreases the maximum recovery of WT residues (A). Number of residues considered for design (B) and binding affinity (C) have little effect on recovery. Binding has little effect on recovery (C). The binding affinity normalized by ligand molecular weight shows a drop in recovery as the ratio increases (D). Ligands with the highest topological polar surface areas do not achieve the maximum recoveries (E). Van der Waals surface area (F), Rosetta interface energy (G), and Rosetta interface energy normalized by ligand molecular weight (H) have little effect on recovery. (Reprinted from Journal of Structural Biology.)

	ALA	ARG	ASN	ASP	CYS	GLN	GLU	GLY	HIS	ILE	LEU	LYS	MET	PHE	PRO	SER	THR	TRP	TYR	VAL
(1) Design Native	0.36824	-0.01478	-0.18745	-0.02381	-0.06576	-0.02348	-0.08669	0.26838	0.30111	-0.01164	-0.30614	-0.38359	0.02857	-0.60178	0.15170	-0.13882	-0.07955	-1.09807	-0.44246	-0.02977
(2) Dock/Design Native	0.41441	-0.18329	-0.12613	-0.41978	-0.35566	0.01174	-0.25679	0.40575	0.17001	0.22061	-0.42797	-0.68378	0.02264	-1.02664	0.06907	-0.12871	0.08871	-1.01001	-0.79039	-0.15147
(3) Design Alanine Mutants	0.52250	-0.09167	-0.32387	-0.37287	-0.05962	-0.02183	-0.14534	0.28225	0.20882	0.01425	-0.25519	-0.37687	-0.06230	-0.71176	0.13442	-0.47421	-0.13385	-1.05198	-0.64547	-0.03036
(4) Dock/Design Alanine Mutants	0.55024	-0.50398	-0.30192	-0.83662	-0.18891	-0.18163	-0.40048	0.41567	0.09237	0.06732	-0.35915	-0.42942	0.00713	-0.98253	0.00436	-0.41406	-0.00408	-0.84192	-0.72510	-0.08917

Table 3 (Supplement Table 1): Sequence composition difference per residue from PSSM recovery. (Reprinted from Journal of Structural Biology.)

Supplement Table 2
Sequence recovery and PSSM recovery results by polar vs nonpolar amino acids (AA).

	Design Native (1)	Dock/Design Native (2)	Design Alanine Mutants (3)	Dock/Design Alanine Mutants (4)
Ligand Translation (Å)[a]	0.1	2.0	0.1	2.0
Ligand Rotation (deg)[a]	2	360	2	360
Nonpolar AA Sequence Recovery (%)	70.4 ± 19.9	48.8 ± 22.1	68.5 ± 20.2	62.4 ± 19.3
Polar AA Sequence Recovery (%)	58.1 ± 18.4	24.7 ± 19.1	44.9 ± 21.5	32.8 ± 19.3
Nonpolar AA PSSM Recovery (%)	81.2 ± 17.9	71.1 ± 20.8	81.0 ± 17.6	78.7 ± 17.1
Polar AA PSSM Recovery (%)	77.0 ± 16.4	55.0 ± 21.6	72.3 ± 17.9	63.1 ± 19.8

[a] Reorientation allowed from initial pose during docking

Table 4 (ST2): Sequence recovery and PSSM recovery results by polar vs nonpolar amino acids (AA). (Reprinted from Journal of Structural Biology.)

PDB ID	CSAR Code	PDB Classification	EC	Ligand	-logK	Rotatable Bonds	Chiral	logS	SlogP	TPSA	MW	pH	Resolution
1bky	2202	MRNA PROCESSING	2.7.7.19	1MC	3.84	0	0	-0.50782	-0.0773	58.69	125.131	5.8	2.00
1fcx	2023	GENE REGULATION	NA	184	7.12	3	1	-8.96865	4.7295	60.36	387.499	7.9	1.47
1fh8	2026	HYDROLASE	3.2.1.91	XYS-XIF	6.89	2	6	0.81161	-3.8614	115.99	250.271	7	1.95
1fh9	2027	HYDROLASE	3.2.1.91	XYS-LOX	6.43	2	7	0.97068	-4.0767	164.23	294.26	7	1.72
1fhd	2028	HYDROLASE	3.2.1.91	XYS-XIM	6.82	2	7	0.86052	-2.5229	137.43	302.283	7	1.90
1lhw	2266	TRANSPORT PROTEIN	NA	ESM	8.16±0.143	1	5	-4.26877	3.61777	49.69	302.414	7.4	1.75
1lnm	2062	LIGAND BINDING PROTEIN	NA	DTX	8.7±0.101	1	8	-4.46464	3.6043	66.76	374.521	7.4	1.90
1nli	2071	HYDROLASE	3.2.2.22	ADE	3.59±0.077	0	0	-1.50408	-0.0649	80.48	135.13	6.48	1.93
1ow4	2077	TRANSPORT PROTEIN	NA	2AN	5.68	3	0	-5.00467	3.4875	63.24	298.342	8	1.60
1q4w	2223	TRANSFERASE	2.4.2.29	DQU	6.46±0.059	0	0	-1.85954	-0.0416	93.5	176.179	7.3	1.93
1r5y	2087	TRANSFERASE	2.4.2.29	DQU	6.46±0.059	0	0	-1.85954	-0.0416	93.5	176.179	7.3	1.20
1s38	2088	TRANSFERASE	2.4.2.29	MAQ	5.15	0	0	-2.29913	0.68462	67.48	175.191	7.3	1.81
1s39	2089	TRANSFERASE	2.4.2.29	AQO	7.7	0	0	-2.13866	0.3762	67.48	161.164	7.3	1.95
1s9t	2272	MEMBRANE PROTEIN	NA	QUS	6.6±0.054	3	1	-0.14649	-3.5764	126.41	189.127	7.2	1.80
1sw1	2092	PROTEIN BINDING	NA	PBE	7.3	1	1	-0.16672	-1.0249	40.13	143.186		1.90
1ui0	2104	HYDROLASE	3.2.2.-	URA	7.06	0	0	-0.41305	-0.6605	58.2	112.088	7.5	1.50
1ukb	1123	HYDROLASE	3.7.1.9	BEZ	3.19	1	0	-1.61473	0.0501	40.13	121.115	7.5	1.80
1uz1	2110	HYDROLASE	3.2.1.21	IFL	6.89	1	3	0.61934	-2.5535	89.79	161.157	5.8	2.00
1uz4	2111	HYDROLASE	NA	IFL	3.4	1	3	0.61934	-2.5535	89.79	161.157	7	1.71
1v0l	2113	HYDROLASE	3.2.1.8	XIF-XYP	6.32	2	6	0.81161	-3.8614	115.99	250.271	5.8	0.98
1vot	1127	HYDROLASE	3.1.1.7	HUP	6.6±0.017	0	2	-2.15614	1.2234	56.74	243.33	8	2.50
1ws4	2119	SUGAR BINDING PROTEIN	NA	GYP	3.0±0.0040	2	5	0.90477	-2.5673	99.38	194.183	7.4	1.90
1y20	2129	LIGAND BINDING PROTEIN	NA	1AC	5.32±0.053	1	0	-0.06618	-2.4892	67.77	101.105	6	1.40
1y93	2261	HYDROLASE	3.4.24.65	HAE	2.1	1	0	0.32015	-0.4883	49.33	75.067	7	1.03
2are	1003	SUGAR BINDING PROTEIN	NA	MAN	3.28	1	5	1.24995	-3.2214	110.38	180.156	7.2	1.80
2b3f	1078	SUGAR BINDING PROTEIN	NA	GAL	6.03±0.018	1	5	1.24995	-3.2214	110.38	180.156	7	1.56
2bbf	1079	TRANSFERASE	2.4.2.29	344	5.1±0.098	0	0	-2.55836	0.2525	96.16	201.189	7.3	1.70
2fai	2167	HORMONE/GROWTH FACTOR RECEPTOR	NA	459	6.24	2	4	-2.37677	3.14	49.69	274.36	8	2.10
2fqw	2172	TRANSPORT PROTEIN	NA	NOS	6.68±0.04	2	4	-0.61473	-2.0067	129.2	268.229	7.5	1.71
2j78	2196	HYDROLASE	3.2.1.21	GOX	6.42	1	4	1.08299	-3.1791	125.54	192.171	5.8	1.65
2jj3	1043	TRANSCRIPTION	NA	JJ3	9.55±0.03	3	3	-4.11127	4.6235	58.92	326.392	7.5	2.28
2nta	1023	HYDROLASE	3.1.3.48	521	4.8	2	0	-4.51499	2.2495	66.48	328.8	7	2.10
2otz	1093	HYDROLASE	3.2.1.17	1MR	3.63	1	0	-1.18134	1.7283	12.03	107.156	6.8	2.07
2ou0	1094	HYDROLASE	3.2.1.17	MR3	3.23	0	0	0.27789	1.3843	4.93	81.118	6.8	1.94
2p7g	1144	HORMONE RECEPTOR	NA	2OH	6.53	2	0	-3.8078	3.4237	40.46	228.291	7.5	2.10
2pwg	1008	ISOMERASE	5.4.99.11	CTS	4.82	0	5	0.84667	-3.8992	85.36	190.219		2.20
2pzv	1097	ISOMERASE	5.3.3.1	IPH	3.87±0.054	0	0	-1.02293	1.3922	20.23	94.113	8	1.25
2q88	1099	TRANSPORT PROTEIN	NA	4CS	5.8±0.075	1	1	-0.47214	-1.4834	64.52	141.15	7	1.90
2q89	1100	TRANSPORT PROTEIN	NA	6CS	6.3±0.079	1	2	-0.06783	-2.5126	84.75	157.149	7	2.30
2rca	1173	MEMBRANE PROTEIN	NA	GLY	7.79±0.087	1	0	0.46181	-2.305	66.15	74.059	7.2	1.58
2vuk	1194	TRANSCRIPTION	NA	P83	3.82±0.028	3	0	-3.45879	3.0404	21.54	239.342	7.2	1.50
2z4b	1042	TRANSCRIPTION	NA	DC8	9.36±0.203	1	3	-3.86974	4.4559	49.69	318.319		2.34
3bgz	1110	TRANSFERASE	2.7.11.1	VX3	6.26	3	0	-6.72619	3.8654	55.92	312.348	7.6	2.40

Table 5 (ST3): Dataset of complexes in benchmark. (Reprinted from Journal of Structural Biology.)

Table S4
Results from favor native residue bonus benchmark

Favor Native Residue Bonus (REU)	Alanine Mutants Recovery (%)	Sequence Recovery (%)	Number of Mutations
0.50	31.0 ± 18.8	33.8 ± 15.2	12.8 ± 4.7
0.75	32.3 ± 19.6	39.3 ± 14.4	11.8 ± 4.4
1.00	**33.1 ± 21.0**[a]	**46.0 ± 13.6**	**10.5 ± 4.1**
1.25	31.1 ± 21.0	54.1 ± 13.0	8.9 ± 3.6
1.50	30.4 ± 22.3	59.9 ± 13.1	7.8 ± 3.3

[a] For subsequent experiments, the favor native residue bonus was chosen to be 1.0 because at this value, the most wild type residues were recovered from the incorrect alanines.

Table 6 (ST4): Results from favor native residue bonus benchmark. (Reprinted from Journal of Structural Biology.)

Table 7 (ST5): Raw data from Design Native, Dock/Design Native, Design Alanine Mutants, and Dock/Design Alanine Mutants experiments. (See below; PDFs will be provided in thesis directory). (Reprinted from Journal of Structural Biology.)

	# Ala mutations (manual)	# of residues considered for design	avg # mutations	Rosetta interface score	RMSD	% mut (# mut/# resi sphere*100)	sequence recovery	affinity (log Kd or Ki))	ligand weight	interface E / ligand weight	binding affinity / ligand weight	Rotatable Bonds
1003-0	0	15.8	5.4	-16.297	0.39	34.1	65.9	3.28	180.190	-0.09044	0.01820	1
1008-0	0	15.2	5.3	-18.91	0.67	34.8	65.2	4.82	189.210	-0.09994	0.02547	0
1023-0	0	23.0	9.1	-22.872	0.55	39.6	60.4	4.80	328.800	-0.06956	0.01460	2
1042-0	0	25.1	9.3	-14.601	0.49	37.1	62.9	9.36	318.315	-0.04587	0.02940	1
1043-0	0	25.2	10.6	-16.619	0.52	42.1	57.9	9.55	326.390	-0.05092	0.02926	3
1078-0	0	16.5	8.2	-24.897	0.35	49.8	50.2	6.03	180.156	-0.13820	0.03347	1
1079-0	0	19.1	2.5	-28.848	0.26	13.1	86.9	5.10	201.189	-0.14339	0.02535	0
1092-0	0	17.1	2.8	-8.392	0.37	16.4	83.6	3.63	107.153	-0.07832	0.03388	1
1093-0	0	14.5	6.4	-10.775	2.43	44.1	55.9	3.23	81.116	-0.13283	0.03982	0
1097-0	0	11.3	3.1	-9.979	0.85	27.5	72.5	3.87	94.111	-0.10603	0.04112	0
1099-0	0	18.9	5.5	-23.623	0.2	29.0	71.0	5.80	142.156	-0.16618	0.04080	1
1100-0	0	19.3	4.4	-24.298	0.39	22.8	77.2	6.30	158.155	-0.15363	0.03983	1
1110-0	0	21.2	9.4	-18.92	0.25	44.3	55.7	6.26	313.349	-0.06038	0.01998	3
1123-0	0	18.1	5.4	-13.206	0.4	29.9	70.1	3.19	122.121	-0.10814	0.02612	1
1127-0	0	19.3	9.3	-11.915	0.54	48.1	51.9	6.60	242.316	-0.04917	0.02724	0
1144-0	0	22.4	6.1	-17.603	0.27	27.2	72.8	6.53	228.291	-0.07711	0.02860	2
1173-0	0	12.9	6.6	-21.78	0.21	51.2	48.8	7.79	75.067	-0.29014	0.10377	1
1194-0	0	17.4	4.7	-17.111	0.27	26.9	73.1	3.82	239.342	-0.07149	0.01596	3
2023-0	0	29.1	10	-25.807	0.28	34.4	65.6	7.12	387.499	-0.06660	0.01837	3
2026-0	0	16.2	7.9	-20.068	0.45	48.7	51.3	6.89	267.110	-0.07513	0.02579	2
2027-0	0	19.6	8.5	-21.143	0.75	43.5	56.5	6.43	312.120	-0.06774	0.02060	2
2028-0	0	17.2	5.3	-20.755	0.32	30.8	69.2	6.82	320.140	-0.06483	0.02130	2
2062-0	0	22.8	9.6	-18.802	0.66	42.2	57.8	8.70	374.521	-0.05020	0.02323	1
2071-0	0	13.6	5	-19.588	0.62	36.9	63.1	3.59	135.130	-0.14496	0.02657	0
2077-0	0	21.0	9.6	-20.775	0.6	45.8	54.2	5.68	298.342	-0.06963	0.01904	3
2087-0	0	18.6	4	-24.875	0.19	21.5	78.5	6.46	176.179	-0.14119	0.03667	0
2088-0	0	18.6	4.9	-25.834	0.2	26.3	73.7	5.15	175.191	-0.14746	0.02940	0
2089-0	0	17.2	4.2	-26.278	0.29	24.4	75.6	7.70	161.164	-0.16305	0.04778	0
2092-0	0	15.5	7.4	-13.221	1.12	47.7	52.3	7.30	144.192	-0.09169	0.05063	1
2104-0	0	18.8	3.5	-18.29	0.28	18.7	81.3	7.06	112.087	-0.16318	0.06299	0
2110-0	0	15.2	7.9	-15.916	0.46	52.0	48.0	6.89	161.157	-0.09876	0.04275	1
2111-0	0	15.3	9.6	-13.434	1.84	62.6	37.4	3.40	161.157	-0.08336	0.02110	1
2113-0	0	15.6	6.1	-20.361	0.4	39.0	61.0	6.32	117.150	-0.17380	0.05395	2
2119-0	0	12.1	2.1	-14.906	0.31	17.3	82.7	3.00	194.183	-0.07676	0.01545	2
2129-0	0	15.0	5.2	-20.558	0.34	34.7	65.3	5.32	101.105	-0.20333	0.05262	1
2167-0	0	21.4	7.7	-15.25	0.65	36.0	64.0	6.24	276.371	-0.05518	0.02258	2
2172-0	0	25.3	6.8	-23.93	0.21	26.9	73.1	6.68	268.229	-0.08921	0.02490	2
2196-0	0	17.9	8.9	-21.249	0.52	49.8	50.2	6.42	192.171	-0.11057	0.03341	1
2202-0	0	12.2	8.3	-11.992	0.79	67.8	32.2	3.84	125.129	-0.09584	0.03069	0
2223-0	0	17.7	3	-27.749	0.24	17.0	83.0	6.46	176.179	-0.15750	0.03667	0
2261-0	0	9.7	3	-8.293	0.57	30.8	69.2	2.10	75.067	-0.11047	0.02798	1
2266-0	0	23.8	10.4	-18.827	0.46	43.8	56.2	8.16	302.414	-0.06226	0.02698	1
2272-0	0	20.0	6.6	-27.42	0.13	33.0	67.0	6.60	189.126	-0.14498	0.03490	3

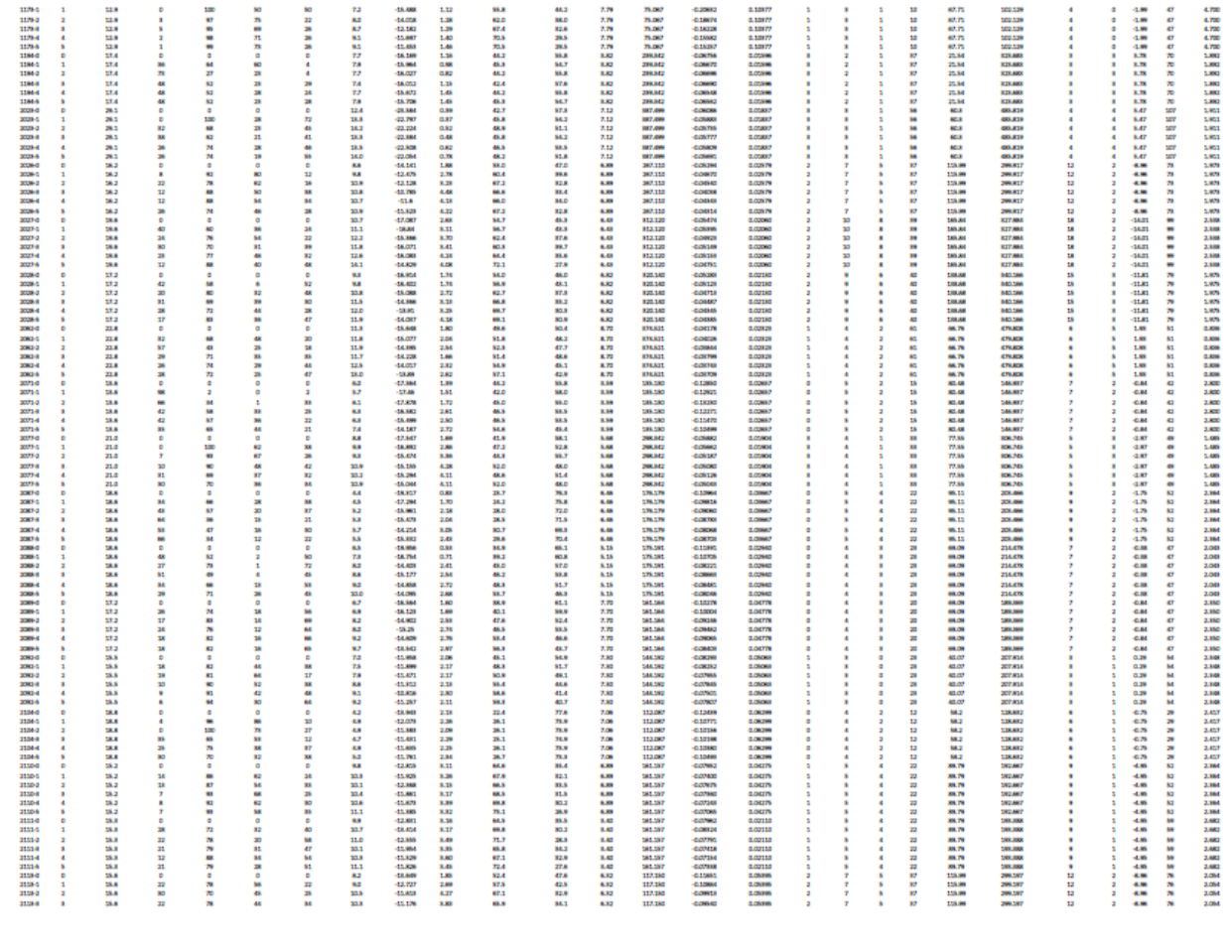
	# Ala mut (manual)	# resi considered for design	avg # mutations	Rosetta interface score	RMSD	% mut (# mut/# resi sphere*100)	sequence recovery	affinity (log Kd or Ki))	ligand weight	interface E / ligand weight	binding affinity / ligand weight	Rotatable Bonds
1003-0	0	15.8	8.7	-12.152	1.89	55.0	45.0	3.28	180.190	-0.06744	0.01820	1
1008-0	0	15.2	11.3	-14.467	3.30	74.2	25.8	4.82	189.210	-0.07646	0.02547	0
1023-0	0	23.0	14.1	-20.854	1.13	61.4	38.6	4.80	328.800	-0.06342	0.01460	2
1042-0	0	25.1	16.7	-13.911	2.92	66.5	33.5	9.36	318.315	-0.04370	0.02940	1
1043-0	0	25.2	19.1	-15.972	4.24	75.8	24.2	9.55	326.390	-0.04894	0.02926	3
1078-0	0	16.5	11.9	-17.541	2.90	72.2	27.8	6.03	180.156	-0.09737	0.03347	1
1079-0	0	19.1	8.8	-18.837	2.22	46.0	54.0	5.10	201.189	-0.09363	0.02535	0
1093-0	0	17.1	8.9	-10.7	2.23	52.1	47.9	3.63	107.153	-0.09986	0.03388	1
1094-0	0	14.5	4.7	-8.058	1.35	32.4	67.6	3.23	81.116	-0.09934	0.03982	0
1097-0	0	11.3	6.7	-9.387	2.00	59.4	40.6	3.87	94.111	-0.09974	0.04112	0
1099-0	0	18.9	11.5	-19.035	0.68	60.7	39.3	5.80	142.156	-0.13390	0.04080	1
1100-0	0	19.3	11.4	-20.182	0.79	59.1	40.9	6.30	158.155	-0.12761	0.03983	1
1110-0	0	21.2	13.7	-15.458	3.27	64.6	35.4	6.26	313.349	-0.04933	0.01998	3
1123-0	0	18.1	10.9	-13.095	0.91	60.4	39.6	3.19	122.121	-0.10723	0.02612	1
1127-0	0	19.3	11.1	-12.322	3.16	57.5	42.5	6.60	242.316	-0.05085	0.02724	0
1144-0	0	22.4	14.4	-14.886	2.25	64.3	35.7	6.53	228.291	-0.06521	0.02860	2
1173-0	0	12.9	8.1	-15.327	1.16	62.8	37.2	7.79	75.067	-0.20418	0.10377	1
1194-0	0	17.4	9.2	-16.121	0.87	52.8	47.2	3.82	239.342	-0.06736	0.01596	3
2023-0	0	29.1	18.1	-22.855	0.79	62.3	37.7	7.12	387.499	-0.05898	0.01837	3
2026-0	0	16.2	12.5	-12.821	2.81	77.1	22.9	6.89	267.110	-0.04800	0.02579	2
2027-0	0	19.6	13.6	-16.663	2.86	69.5	30.5	6.43	312.120	-0.05339	0.02060	2
2028-0	0	17.2	13.3	-15.224	3.40	77.2	22.8	6.82	320.140	-0.04755	0.02130	2
2062-0	0	22.8	17.4	-15.41	1.85	76.4	23.6	8.70	374.521	-0.04115	0.02323	1
2071-0	0	13.6	8.4	-17.314	1.66	61.9	38.1	3.59	135.130	-0.12813	0.02657	0
2077-0	0	21.0	14.3	-17.633	2.86	68.2	31.8	5.68	298.342	-0.05910	0.01904	3
2087-0	0	18.6	7.9	-17.243	1.63	42.5	57.5	6.46	176.179	-0.09787	0.03667	0
2088-0	0	18.6	9.9	-16.145	1.51	53.2	46.8	5.15	175.191	-0.09216	0.02940	0
2089-0	0	17.2	10.2	-16.269	1.84	59.2	40.8	7.70	161.164	-0.10095	0.04778	0
2092-0	0	15.5	9.9	-12.035	2.06	63.8	36.2	7.30	144.192	-0.08347	0.05063	1
2104-0	0	18.8	6.7	-13.8	2.58	35.7	64.3	7.06	112.087	-0.12312	0.06299	0
2110-0	0	15.2	13	-12.372	3.26	85.6	14.4	6.89	161.157	-0.07677	0.04275	1
2111-0	0	15.3	11.6	-13.03	3.19	75.6	24.4	3.40	161.157	-0.08085	0.02110	1
2113-0	0	15.6	11.4	-13.187	2.76	72.9	27.1	6.32	117.150	-0.11257	0.05395	2
2119-0	0	12.1	6.8	-11.036	0.78	56.1	43.9	3.00	194.183	-0.05683	0.01545	2
2129-0	0	15.0	10.7	-16.065	0.64	71.3	28.7	5.32	101.105	-0.15889	0.05262	1
2167-0	0	21.4	15.5	-12.307	3.39	72.4	27.6	6.24	276.371	-0.04453	0.02258	2
2172-0	0	25.3	14.3	-18.219	2.02	56.6	43.4	6.68	268.229	-0.06792	0.02490	2
2196-0	0	17.9	13.7	-18.036	3.50	76.6	23.4	6.42	192.171	-0.09385	0.03341	1
2202-0	0	12.2	10.8	-9.393	3.01	88.2	11.8	3.84	125.129	-0.07507	0.03069	0
2223-0	0	17.7	9.9	-17.666	1.82	56.0	44.0	6.46	176.179	-0.10027	0.03667	0
2261-0	0	9.7	5.8	-7.48	2.17	59.5	40.5	2.10	75.067	-0.09964	0.02798	1
2266-0	0	23.8	17.4	-16.109	3.71	73.2	26.8	8.16	302.414	-0.05327	0.02698	1
2272-0	0	20.0	11.5	-20.824	0.39	57.5	42.5	6.60	189.126	-0.11011	0.03490	3

	# Ala mut (manual)	# resi considered for design	% Ala --> WT	% Ala --> not WT	% Ala --> Ala	% Ala --> other AA	avg # mutations	Rosetta interface score	RMSD	% mut (# mut/# resi sphere*100)	sequence recovery	affinity (log Kd or Ki)	ligand weight	interface E / lig weight	binding affinity / lig weight
1003-1	1	15.8	0	100	4	96	5.7	-16.12	0.40	36.0	64.0	3.28	180.190	-0.08946	0.01820
1003-2	2	15.8	0	100	50	50	6.5	-13.369	0.37	41.1	58.9	3.28	180.190	-0.07419	0.01820
1003-3	3	15.8	24	76	42	34	7	-13.544	0.44	44.2	55.8	3.28	180.190	-0.07517	0.01820
1003-4	4	15.8	20	80	38	42	8.1	-13.054	0.43	51.2	48.8	3.28	180.190	-0.07245	0.01820
1003-5	5	15.8	17	83	28	54	7.8	-12.982	0.45	49.3	50.7	3.28	180.190	-0.07205	0.01820
1008-1	1	15.2	100	0	0	0	5.8	-16.482	0.65	38.1	61.9	4.82	189.210	-0.08711	0.02547
1008-2	2	15.2	85	15	13	2	7.2	-15.335	1.02	47.3	52.7	4.82	189.210	-0.08105	0.02547
1008-3	3	15.2	79	21	12	9	7.1	-15.357	0.89	46.6	53.4	4.82	189.210	-0.08116	0.02547
1008-4	4	15.2	34	66	58	8	10	-12.436	1.83	65.7	34.3	4.82	189.210	-0.06573	0.02547
1008-5	5	15.2	28	72	57	14	10	-12.4	2.27	65.7	34.3	4.82	189.210	-0.06554	0.02547
1023-1	1	23.0	66	34	0	34	9.6	-22.404	0.56	41.8	58.2	4.80	328.800	-0.06814	0.01460
1023-2	2	23.0	84	16	0	16	9.5	-22.71	0.58	41.3	58.7	4.80	328.800	-0.06907	0.01460
1023-3	3	23.0	61	39	12	27	10	-22.363	0.59	43.5	56.5	4.80	328.800	-0.06801	0.01460
1023-4	4	23.0	44	56	32	24	9.9	-22.542	0.76	43.1	56.9	4.80	328.800	-0.06856	0.01460
1023-5	5	23.0	51	49	25	24	10.4	-22.241	0.80	45.3	54.7	4.80	328.800	-0.06764	0.01460
1042-1	1	25.1	98	2	2	0	9.3	-14.445	0.52	37.1	62.9	9.36	318.315	-0.04538	0.02940
1042-2	2	25.1	97	3	2	1	9.5	-14.598	0.50	37.8	62.2	9.36	318.315	-0.04586	0.02940
1042-3	3	25.1	65	35	7	29	9.1	-14.517	0.52	36.3	63.7	9.36	318.315	-0.04561	0.02940
1042-4	4	25.1	45	55	28	28	10.7	-13.965	0.51	42.6	57.4	9.36	318.315	-0.04387	0.02940
1042-5	5	25.1	40	60	39	21	11.1	-13.374	0.56	44.2	55.8	9.36	318.315	-0.04201	0.02940
1043-1	1	25.2	2	98	20	78	10.9	-15.812	0.71	43.3	56.7	9.55	326.390	-0.04845	0.02926
1043-2	2	25.2	5	95	57	38	10.5	-16.139	0.59	41.7	58.3	9.55	326.390	-0.04945	0.02926
1043-3	3	25.2	35	65	38	27	10.3	-16.407	0.57	40.9	59.1	9.55	326.390	-0.05027	0.02926
1043-4	4	25.2	45	55	33	22	10.6	-16.166	0.62	42.1	57.9	9.55	326.390	-0.04953	0.02926
1043-5	5	25.2	53	47	26	21	11.1	-15.957	0.63	44.0	56.0	9.55	326.390	-0.04889	0.02926
1078-1	1	16.5	2	98	0	98	8.3	-25.148	0.33	50.4	49.6	6.03	180.156	-0.13959	0.03347
1078-2	2	16.5	52	48	0	48	8.4	-24.436	0.32	51.0	49.0	6.03	180.156	-0.13564	0.03347
1078-3	3	16.5	47	53	0	53	8.8	-23.689	0.33	53.4	46.6	6.03	180.156	-0.13149	0.03347
1078-4	4	16.5	56	44	0	44	9	-23.911	0.31	54.6	45.4	6.03	180.156	-0.13272	0.03347
1078-5	5	16.5	47	53	0	53	8.8	-23.85	0.31	53.4	46.6	6.03	180.156	-0.13239	0.03347
1079-1	1	19.1	100	0	0	0	2.5	-28.573	0.26	13.1	86.9	5.10	201.189	-0.14202	0.02535
1079-2	2	19.1	100	0	0	0	2.2	-28.322	0.26	11.5	88.5	5.10	201.189	-0.14077	0.02535
1079-3	3	19.1	100	0	0	0	2.5	-28.33	0.27	13.1	86.9	5.10	201.189	-0.14081	0.02535
1079-4	4	19.1	85	15	15	0	2.3	-28.212	0.25	12.0	88.0	5.10	201.189	-0.14023	0.02535
1079-5	5	19.1	84	16	13	3	2.6	-28.322	0.26	13.6	86.4	5.10	201.189	-0.14077	0.02535
1093-1	1	17.1	96	4	2	2	6.7	-10.799	2.46	39.2	60.8	3.63	107.153	-0.10078	0.03388
1093-2	2	17.1	100	0	0	0	6.3	-10.86	2.43	36.9	63.1	3.63	107.153	-0.10135	0.03388
1093-3	3	17.1	95	5	2	3	6.4	-10.709	2.43	37.5	62.5	3.63	107.153	-0.09994	0.03388
1093-4	4	17.1	74	26	22	4	7.6	-9.923	2.50	44.5	55.5	3.63	107.153	-0.09261	0.03388
1093-5	5	17.1	59	41	36	5	7.2	-9.912	2.40	42.2	57.8	3.63	107.153	-0.09250	0.03388
1094-1	1	14.5	100	0	0	0	2.8	-8.327	0.41	19.3	80.7	3.23	81.116	-0.10266	0.03982
1094-2	2	14.5	74	26	0	26	2.9	-8.34	0.45	20.0	80.0	3.23	81.116	-0.10282	0.03982
1094-3	3	14.5	78	22	0	22	3	-8.279	0.53	20.7	79.3	3.23	81.116	-0.10206	0.03982
1094-4	4	14.5	88	12	0	12	2.9	-8.317	0.46	20.0	80.0	3.23	81.116	-0.10253	0.03982
1094-5	5	14.5	92	8	0	8	2.7	-8.285	0.47	18.6	81.4	3.23	81.116	-0.10214	0.03982
1097-1	1	11.3	16	84	84	0	4.1	-8.151	1.20	36.3	63.7	3.87	94.111	-0.08661	0.04112
1097-2	2	11.3	16	84	84	0	4.5	-7.608	1.57	39.9	60.1	3.87	94.111	-0.08084	0.04112
1097-3	3	11.3	10	90	65	25	4.9	-6.9	1.57	43.4	56.6	3.87	94.111	-0.07332	0.04112
1097-4	4	11.3	16	84	61	24	5.8	-6.664	1.82	51.4	48.6	3.87	94.111	-0.07081	0.04112
1097-5	5	11.3	33	67	38	28	5.4	-6.792	1.30	47.9	52.1	3.87	94.111	-0.07217	0.04112
1099-1	1	18.9	0	100	100	0	5.4	-19.199	0.32	28.5	71.5	5.80	142.156	-0.13506	0.04080
1099-2	2	18.9	37	63	57	6	5.4	-19.06	0.31	28.5	71.5	5.80	142.156	-0.13408	0.04080
1099-3	3	18.9	45	55	45	10	6.2	-17.942	0.31	32.7	67.3	5.80	142.156	-0.12621	0.04080
1099-4	4	18.9	59	41	38	4	6.3	-18.032	0.32	33.3	66.7	5.80	142.156	-0.12685	0.04080
1099-5	5	18.9	44	56	30	26	6.2	-17.961	0.31	32.7	67.3	5.80	142.156	-0.12635	0.04080
1100-1	1	19.3	0	100	100	0	5.7	-19.654	0.46	29.6	70.4	6.30	158.155	-0.12427	0.03983
1100-2	2	19.3	9	91	87	4	6	-19.098	0.41	31.1	68.9	6.30	158.155	-0.12075	0.03983
1100-3	3	19.3	6	94	64	30	6	-19.007	0.39	31.1	68.9	6.30	158.155	-0.12018	0.03983
1100-4	4	19.3	29	71	48	23	5.9	-19.035	0.42	30.6	69.4	6.30	158.155	-0.12036	0.03983
1100-5	5	19.3	43	57	39	18	6.1	-18.941	0.43	31.6	68.4	6.30	158.155	-0.11976	0.03983
1110-1	1	21.2	12	88	0	88	9.3	-18.785	0.23	43.8	56.2	6.26	313.349	-0.05995	0.01998
1110-2	2	21.2	52	48	0	48	9.7	-18.836	0.23	45.7	54.3	6.26	313.349	-0.06011	0.01998
1110-3	3	21.2	69	31	0	31	9.4	-18.865	0.23	44.3	55.7	6.26	313.349	-0.06020	0.01998
1110-4	4	21.2	74	26	0	26	9.3	-18.78	0.23	43.8	56.2	6.26	313.349	-0.05993	0.01998
1110-5	5	21.2	42	58	24	34	10.8	-17.862	0.58	50.9	49.1	6.26	313.349	-0.05700	0.01998
1123-1	1	18.1	46	54	0	54	5.6	-13.109	0.41	31.0	69.0	3.19	122.121	-0.10734	0.02612
1123-2	2	18.1	21	79	0	79	5.6	-13.071	0.42	31.0	69.0	3.19	122.121	-0.10703	0.02612
1123-3	3	18.1	7	93	33	60	5.9	-12.244	0.40	32.7	67.3	3.19	122.121	-0.10026	0.02612
1123-4	4	18.1	3	97	25	72	5.9	-12.202	0.40	32.7	67.3	3.19	122.121	-0.09992	0.02612
1123-5	5	18.1	5	95	41	54	6.8	-11.986	0.46	37.7	62.3	3.19	122.121	-0.09815	0.02612
1127-1	1	19.3	0	100	0	100	9.1	-11.814	0.53	47.1	52.9	6.60	242.316	-0.04875	0.02724
1127-2	2	19.3	43	57	50	7	9.1	-11.957	0.80	47.1	52.9	6.60	242.316	-0.04934	0.02724
1127-3	3	19.3	63	37	33	3	8.9	-11.76	0.83	46.1	53.9	6.60	242.316	-0.04853	0.02724
1127-4	4	19.3	40	60	42	18	10.3	-13.568	0.96	53.3	46.7	6.60	242.316	-0.05599	0.02724
1127-5	5	19.3	39	61	52	9	10.1	-13.634	0.90	52.3	47.7	6.60	242.316	-0.05627	0.02724
1144-1	1	22.4	0	100	10	90	6.2	-16.926	0.28	27.7	72.3	6.53	228.291	-0.07414	0.02860
1144-2	2	22.4	0	100	9	91	6.4	-16.558	0.29	28.6	71.4	6.53	228.291	-0.07253	0.02860
1144-3	3	22.4	33	67	14	53	6.4	-16.018	0.30	28.6	71.4	6.53	228.291	-0.07016	0.02860
1144-4	4	22.4	38	62	18	44	6.9	-16.257	0.29	30.8	69.2	6.53	228.291	-0.07121	0.02860

1164-5	5	22.4	49	51	16	35	7.2	-16.323	0.32	32.1	67.9	6.53	228.291	-0.07150	0.02860
1173-1	1	12.9	6	94	94	0	7.2	-17.307	0.16	55.8	44.2	7.79	75.067	-0.23055	0.10377
1173-2	2	12.9	4	96	94	2	8.5	-15.104	0.47	65.9	34.1	7.79	75.067	-0.20121	0.10377
1173-3	3	12.9	3	97	83	14	9.5	-14.711	0.50	73.6	26.4	7.79	75.067	-0.19597	0.10377
1173-4	4	12.9	0	100	60	40	9.9	-12.604	1.14	76.7	23.3	7.79	75.067	-0.16790	0.10377
1173-5	5	12.9	0	100	62	38	9.9	-12.249	1.15	76.7	23.3	7.79	75.067	-0.16317	0.10377
1194-1	1	17.4	100	0	0	0	4.8	-17.092	0.27	27.5	72.5	3.82	239.342	-0.07141	0.01596
1194-2	2	17.4	100	0	0	0	4.5	-17.093	0.25	25.8	74.2	3.82	239.342	-0.07142	0.01596
1194-3	3	17.4	67	33	0	33	4.6	-17.101	0.27	26.4	73.6	3.82	239.342	-0.07145	0.01596
1194-4	4	17.4	75	25	0	25	4.8	-17.101	0.28	27.5	72.5	3.82	239.342	-0.07145	0.01596
1194-5	5	17.4	80	20	0	20	4.6	-17.057	0.28	26.4	73.6	3.82	239.342	-0.07127	0.01596
1223-1	1	29.1	0	100	56	44	10.9	-24.455	0.29	37.5	62.5	7.12	387.499	-0.06311	0.01837
1223-2	2	29.1	50	50	34	16	10.9	-24.458	0.27	37.5	62.5	7.12	387.499	-0.06312	0.01837
1223-3	3	29.1	38	62	33	29	10.9	-24.599	0.30	37.5	62.5	7.12	387.499	-0.06348	0.01837
1223-4	4	29.1	28	72	22	50	11	-24.667	0.29	37.9	62.1	7.12	387.499	-0.06366	0.01837
1223-5	5	29.1	38	62	17	44	11.2	-24.524	0.29	38.5	61.5	7.12	387.499	-0.06329	0.01837
1226-1	1	16.2	98	2	0	2	8.1	-19.406	0.43	49.9	50.1	6.89	267.110	-0.07265	0.02579
1226-2	2	16.2	91	9	8	1	8.2	-18.444	0.48	50.6	49.4	6.89	267.110	-0.06905	0.02579
1226-3	3	16.2	77	23	3	20	7.7	-18.191	0.52	47.5	52.5	6.89	267.110	-0.06810	0.02579
1226-4	4	16.2	61	39	18	21	7.6	-18.893	0.45	46.9	53.1	6.89	267.110	-0.07073	0.02579
1226-5	5	16.2	60	40	14	26	7.9	-19.134	0.45	48.7	51.3	6.89	267.110	-0.07163	0.02579
1227-1	1	19.6	98	2	2	0	8.9	-21.144	0.71	45.5	54.5	6.43	312.120	-0.06774	0.02060
1227-2	2	19.6	99	1	1	0	8.5	-20.87	0.67	43.5	56.5	6.43	312.120	-0.06687	0.02060
1227-3	3	19.6	77	23	0	23	9.4	-21.882	0.71	48.1	51.9	6.43	312.120	-0.07011	0.02060
1227-4	4	19.6	68	32	8	24	9.9	-20.676	0.76	50.6	49.4	6.43	312.120	-0.06624	0.02060
1227-5	5	19.6	55	45	8	36	9.8	-20.444	0.73	50.1	49.9	6.43	312.120	-0.06550	0.02060
1228-1	1	17.2	70	30	0	30	6.8	-20.314	0.41	39.5	60.5	6.82	320.140	-0.06345	0.02130
1228-2	2	17.2	93	7	3	4	5.7	-20.036	0.34	33.1	66.9	6.82	320.140	-0.06259	0.02130
1228-3	3	17.2	93	7	3	4	5.9	-19.937	0.35	34.3	65.7	6.82	320.140	-0.06228	0.02130
1228-4	4	17.2	92	7	0	7	6	-19.517	0.34	34.8	65.2	6.82	320.140	-0.06096	0.02130
1228-5	5	17.2	66	34	2	32	8.1	-17.877	0.41	47.0	53.0	6.82	320.140	-0.05584	0.02130
1262-1	1	22.8	82	18	8	10	9.4	-18.681	0.59	41.3	58.7	8.70	374.521	-0.04988	0.02323
1262-2	2	22.8	74	26	3	23	9.8	-18.528	0.61	43.1	56.9	8.70	374.521	-0.04950	0.02323
1262-3	3	22.8	41	59	19	39	10	-17.421	0.66	43.9	56.1	8.70	374.521	-0.04652	0.02323
1262-4	4	22.8	28	72	13	59	12	-16.723	0.98	52.7	47.3	8.70	374.521	-0.04465	0.02323
1262-5	5	22.8	27	73	14	59	12	-17.435	0.87	52.7	47.3	8.70	374.521	-0.04655	0.02323
1271-1	1	13.6	100	0	0	0	4.5	-19.78	0.62	33.2	66.8	3.59	135.130	-0.14638	0.02657
1271-2	2	13.6	64	36	1	35	4.5	-19.751	0.60	33.2	66.8	3.59	135.130	-0.14616	0.02657
1271-3	3	13.6	33	67	35	31	5.5	-15.862	0.53	40.6	59.4	3.59	135.130	-0.11738	0.02657
1271-4	4	13.6	50	50	32	18	6.1	-15.712	0.60	45.0	55.0	3.59	135.130	-0.11627	0.02657
1271-5	5	13.6	40	60	53	7	7.2	-13.302	0.72	53.1	46.9	3.59	135.130	-0.09844	0.02657
1277-1	1	21.0	8	92	66	26	10.7	-19.571	0.95	51.0	49.0	5.68	298.342	-0.06560	0.01904
1277-2	2	21.0	7	93	43	50	10.2	-18.937	0.97	48.6	51.4	5.68	298.342	-0.06347	0.01904
1277-3	3	21.0	26	74	33	41	11.7	-18.482	1.16	55.8	44.2	5.68	298.342	-0.06195	0.01904
1277-4	4	21.0	46	54	24	30	11.4	-18.248	1.21	54.3	45.7	5.68	298.342	-0.06116	0.01904
1277-5	5	21.0	36	64	22	42	11	-18.311	1.17	52.4	47.6	5.68	298.342	-0.06138	0.01904
1287-1	1	18.6	100	0	0	0	3.8	-24.641	0.20	20.5	79.5	6.46	176.179	-0.13986	0.03667
1287-2	2	18.6	100	0	0	0	3.8	-23.786	0.27	20.5	79.5	6.46	176.179	-0.13501	0.03667
1287-3	3	18.6	100	0	0	0	3.7	-24.118	0.24	19.9	80.1	6.46	176.179	-0.13689	0.03667
1287-4	4	18.6	100	0	0	0	3.8	-23.834	0.26	20.5	79.5	6.46	176.179	-0.13528	0.03667
1287-5	5	18.6	100	0	0	0	3.8	-23.273	0.28	20.5	79.5	6.46	176.179	-0.13210	0.03667
1288-1	1	18.6	100	0	0	0	5	-25.736	0.20	26.9	73.1	5.15	175.191	-0.14690	0.02940
1288-2	2	18.6	87	13	0	13	5.6	-23.743	0.28	30.1	69.9	5.15	175.191	-0.13553	0.02940
1288-3	3	18.6	87	13	0	13	5.3	-22.941	0.29	28.5	71.5	5.15	175.191	-0.13095	0.02940
1288-4	4	18.6	69	31	8	23	7.1	-20.664	0.95	38.1	61.9	5.15	175.191	-0.11795	0.02940
1288-5	5	18.6	60	40	22	18	7.7	-21.92	0.65	41.4	58.6	5.15	175.191	-0.12512	0.02940
1289-1	1	17.2	100	0	0	0	4	-26.148	0.30	23.2	76.8	7.70	161.164	-0.16224	0.04778
1289-2	2	17.2	69	31	2	29	5.3	-21.692	0.44	30.8	69.2	7.70	161.164	-0.13460	0.04778
1289-3	3	17.2	37	63	0	63	6.7	-19.891	0.58	38.9	61.1	7.70	161.164	-0.12342	0.04778
1289-4	4	17.2	30	70	24	46	7.2	-19.021	0.74	41.8	58.2	7.70	161.164	-0.11802	0.04778
1289-5	5	17.2	45	55	19	36	7	-18.144	0.71	40.7	59.3	7.70	161.164	-0.11258	0.04778
1292-1	1	15.5	0	100	0	100	7.1	-13.18	1.09	45.7	54.3	7.30	144.192	-0.09141	0.05063
1292-2	2	15.5	15	85	32	53	8.7	-12.755	1.29	56.1	43.9	7.30	144.192	-0.08846	0.05063
1292-3	3	15.5	6	94	39	55	9.1	-12.476	1.54	58.6	41.4	7.30	144.192	-0.08652	0.05063
1292-4	4	15.5	4	96	25	70	8.9	-12.671	1.67	57.3	42.7	7.30	144.192	-0.08788	0.05063
1292-5	5	15.5	2	98	27	71	8.6	-11.941	1.98	55.4	44.6	7.30	144.192	-0.08281	0.05063
1304-1	1	18.8	100	0	0	0	3.1	-17.945	0.26	16.5	83.5	7.06	112.087	-0.16010	0.06299
1304-2	2	18.8	49	51	9	42	2.8	-17.19	0.23	14.9	85.1	7.06	112.087	-0.15336	0.06299
1304-3	3	18.8	67	33	5	29	2.9	-17.148	0.23	15.5	84.5	7.06	112.087	-0.15299	0.06299
1304-4	4	18.8	50	50	2	48	2.9	-17.158	0.23	15.5	84.5	7.06	112.087	-0.15308	0.06299
1304-5	5	18.8	59	41	4	37	2.9	-16.904	0.23	15.5	84.5	7.06	112.087	-0.15081	0.06299
1310-1	1	15.2	58	42	0	42	8.3	-15.213	0.53	54.7	45.3	6.89	161.157	-0.09440	0.04275
1310-2	2	15.2	77	23	2	21	8.2	-15.478	0.48	54.0	46.0	6.89	161.157	-0.09604	0.04275
1310-3	3	15.2	42	58	26	32	8.4	-14.236	0.45	55.3	44.7	6.89	161.157	-0.08834	0.04275
1310-4	4	15.2	24	76	41	35	9.7	-13.385	1.22	63.9	36.1	6.89	161.157	-0.08306	0.04275
1310-5	5	15.2	14	86	48	37	11.1	-12.85	2.07	73.1	26.9	6.89	161.157	-0.07974	0.04275
1311-1	1	15.3	44	56	0	56	10	-13.684	2.00	65.2	34.8	3.40	161.157	-0.08491	0.02110
1311-2	2	15.3	20	80	11	69	11.4	-12.402	3.35	74.3	25.7	3.40	161.157	-0.07696	0.02110
1311-3	3	15.3	8	92	5	87	11.8	-12.352	3.43	76.9	23.1	3.40	161.157	-0.07665	0.02110
1311-4	4	15.3	9	91	6	86	11.8	-12.319	3.00	76.9	23.1	3.40	161.157	-0.07644	0.02110

1311-5	5	15.3	28	72	6	66	11.9	-12.373	3.04	77.6	22.4	3.40	161.157	-0.07678	0.02110
1313-1	1	15.6	100	0	0	0	6.2	-20.361	0.34	39.6	60.4	6.32	117.150	-0.17380	0.05395
1313-2	2	15.6	98	2	0	2	6.4	-20.132	0.37	40.9	59.1	6.32	117.150	-0.17185	0.05395
1313-3	3	15.6	91	9	1	9	6.4	-19.578	0.34	40.9	59.1	6.32	117.150	-0.16712	0.05395
1313-4	4	15.6	65	35	17	18	6.6	-19.464	0.43	42.2	57.8	6.32	117.150	-0.16615	0.05395
1313-5	5	15.6	61	39	18	20	7.6	-18.726	0.40	48.6	51.4	6.32	117.150	-0.15985	0.05395
1319-1	1	12.1	0	100	96	4	4.4	-14.351	0.44	36.3	63.7	3.00	194.183	-0.07390	0.01545
1319-2	2	12.1	50	50	49	1	2.8	-13.586	0.42	23.1	76.9	3.00	194.183	-0.06996	0.01545
1319-3	3	12.1	33	67	33	34	3	-13.665	0.41	24.8	75.2	3.00	194.183	-0.07037	0.01545
1319-4	4	12.1	16	84	48	36	4.4	-11.28	0.44	36.3	63.7	3.00	194.183	-0.05809	0.01545
1319-5	5	12.1	33	67	38	28	4.2	-11.531	0.41	34.7	65.3	3.00	194.183	-0.05938	0.01545
1329-1	1	15.0	0	100	100	0	4.7	-15.161	0.46	31.3	68.7	5.32	101.105	-0.14995	0.05262
1329-2	2	15.0	38	62	50	12	5	-15.074	0.48	33.3	66.7	5.32	101.105	-0.14909	0.05262
1329-3	3	15.0	25	75	69	6	6	-14.403	0.49	40.0	60.0	5.32	101.105	-0.14246	0.05262
1329-4	4	15.0	20	80	58	23	5.6	-14.563	0.49	37.3	62.7	5.32	101.105	-0.14404	0.05262
1329-5	5	15.0	7	93	45	48	8.3	-14.233	0.63	55.3	44.7	5.32	101.105	-0.14077	0.05262
1367-1	1	21.4	96	4	0	4	7.9	-15.077	0.86	36.9	63.1	6.24	276.371	-0.05455	0.02258
1367-2	2	21.4	49	51	28	23	9	-14.594	1.14	42.1	57.9	6.24	276.371	-0.05281	0.02258
1367-3	3	21.4	59	41	23	17	8.9	-14.384	0.97	41.6	58.4	6.24	276.371	-0.05205	0.02258
1367-4	4	21.4	67	33	18	14	9.2	-14.438	1.03	43.0	57.0	6.24	276.371	-0.05224	0.02258
1367-5	5	21.4	72	28	16	12	9.1	-14.367	1.01	42.5	57.5	6.24	276.371	-0.05198	0.02258
2172-1	1	25.3	92	8	0	8	7.2	-23.838	0.20	28.5	71.5	6.68	268.229	-0.08887	0.02490
2172-2	2	25.3	97	3	0	3	7.4	-23.737	0.20	29.3	70.7	6.68	268.229	-0.08850	0.02490
2172-3	3	25.3	97	3	0	3	7.3	-23.725	0.20	28.9	71.1	6.68	268.229	-0.08845	0.02490
2172-4	4	25.3	99	1	0	1	7.4	-23.636	0.20	29.3	70.7	6.68	268.229	-0.08812	0.02490
2172-5	5	25.3	80	20	0	20	7.4	-23.607	0.21	29.3	70.7	6.68	268.229	-0.08801	0.02490
2196-1	1	17.9	62	38	12	26	9.3	-20.053	0.55	52.0	48.0	6.42	192.171	-0.10435	0.03341
2196-2	2	17.9	39	61	2	59	9.3	-19.724	0.51	52.0	48.0	6.42	192.171	-0.10264	0.03341
2196-3	3	17.9	21	79	4	75	10.4	-19.629	0.55	58.2	41.8	6.42	192.171	-0.10214	0.03341
2196-4	4	17.9	35	65	12	54	10.1	-19.655	0.54	56.5	43.5	6.42	192.171	-0.10228	0.03341
2196-5	5	17.9	25	75	27	48	10.6	-18.78	0.56	59.3	40.7	6.42	192.171	-0.09773	0.03341
2202-1	1	12.2	100	0	0	0	8.1	-11.735	0.82	66.2	33.8	3.84	125.129	-0.09378	0.03069
2202-2	2	12.2	50	50	39	11	7.9	-9.331	0.61	64.5	35.5	3.84	125.129	-0.07457	0.03069
2202-3	3	12.2	40	60	22	38	8.2	-9.844	0.60	67.0	33.0	3.84	125.129	-0.07867	0.03069
2202-4	4	12.2	29	71	14	58	8.5	-10.638	0.60	69.4	30.6	3.84	125.129	-0.08502	0.03069
2202-5	5	12.2	20	80	33	47	8.5	-10.124	0.75	69.4	30.6	3.84	125.129	-0.08091	0.03069
2223-1	1	17.7	100	0	0	0	3	-27.561	0.24	17.0	83.0	6.46	176.179	-0.15644	0.03667
2223-2	2	17.7	100	0	0	0	2.9	-27.024	0.23	16.4	83.6	6.46	176.179	-0.15339	0.03667
2223-3	3	17.7	100	0	0	0	3.1	-27.381	0.24	17.5	82.5	6.46	176.179	-0.15542	0.03667
2223-4	4	17.7	100	0	0	0	2.9	-26.715	0.23	16.4	83.6	6.46	176.179	-0.15164	0.03667
2223-5	5	17.7	100	0	0	0	3.2	-26.495	0.23	18.1	81.9	6.46	176.179	-0.15039	0.03667
2261-1	1	9.7	0	100	0	100	4.6	-10.005	0.46	47.2	52.8	2.10	75.067	-0.13328	0.02798
2261-2	2	9.7	50	50	0	50	4.5	-9.83	0.46	46.2	53.8	2.10	75.067	-0.13095	0.02798
2261-3	3	9.7	62	38	5	33	4.8	-9.69	0.45	49.3	50.7	2.10	75.067	-0.12908	0.02798
2261-4	4	9.7	46	54	24	30	4.8	-6.243	1.09	49.3	50.7	2.10	75.067	-0.08317	0.02798
2261-5	5	9.7	33	67	20	46	6.1	-6.842	0.88	62.6	37.4	2.10	75.067	-0.09115	0.02798
2266-1	1	23.8	100	0	0	0	10.2	-18.761	0.46	42.9	57.1	8.16	302.414	-0.06204	0.02698
2266-2	2	23.8	55	45	43	2	10.9	-16.873	0.46	45.9	54.1	8.16	302.414	-0.05579	0.02698
2266-3	3	23.8	48	52	47	5	11	-16.851	0.45	46.3	53.7	8.16	302.414	-0.05572	0.02698
2266-4	4	23.8	54	46	38	8	11.1	-16.403	0.49	46.7	53.3	8.16	302.414	-0.05424	0.02698
2266-5	5	23.8	61	39	32	8	11.8	-16.462	0.45	49.7	50.3	8.16	302.414	-0.05444	0.02698
2272-1	1	20.0	0	100	100	0	6.8	-22.628	0.20	34.0	66.0	6.60	189.126	-0.11965	0.03490
2272-2	2	20.0	50	50	50	0	6.7	-22.579	0.19	33.5	66.5	6.60	189.126	-0.11939	0.03490
2272-3	3	20.0	67	33	33	0	6.9	-22.693	0.20	34.5	65.5	6.60	189.126	-0.11893	0.03490
2272-4	4	20.0	50	50	25	25	6.8	-22.329	0.20	34.0	66.0	6.60	189.126	-0.11806	0.03490
2272-5	5	20.0	60	40	20	20	6.8	-22.298	0.20	34.0	66.0	6.60	189.126	-0.11790	0.03490

	# Ala mut (manual)	# res considered for design	% Ala -> WT	% Ala -> not WT	% Ala -> Ala	% Ala -> other AA	# mutations	Rosetta interface score	RMSD	% mutations (# mut/(# res*symm*100)	seq recovery	affinity (log10d or IC)	ligand weight	interface score / ligandgbt	binding affinity / ligandgbt	# Rotatable Bonds	# Hbond Acceptors	# Hbond Donors	# ligand atoms	TorsionAngle/Polar SurfaceArea	WaterSurfaceArea	# H bond donors + acceptors	# rings	LogP	# Contacts	Crossedness
1009-0	0	59.8	0	0	0	0	6.8	-52.597	2.27	49.0	97.0	9.28	180.190	-0.06799	0.018520	1	6	5	24	110.58	200.695	11	1	-6.42	60	2.500
1009-1	1	59.8	0	100	72	28	7.2	-52.062	2.56	49.9	94.9	9.28	180.190	-0.06705	0.018520	1	6	5	24	110.58	200.695	11	1	-6.42	60	2.500
1009-2	2	59.8	3	67	64	33	6.8	-52.075	1.82	49.0	97.0	9.28	180.190	-0.06699	0.018520	1	6	5	24	110.58	200.695	11	1	-6.42	60	2.500
1009-3	3	59.8	34	66	66	22	6.8	-51.411	2.27	49.6	96.4	9.28	180.190	-0.06333	0.018520	1	6	5	24	110.58	200.695	11	1	-6.42	60	2.500
1009-4	4	59.8	28	72	69	31	7.6	-51.861	1.79	49.0	92.0	9.28	180.190	-0.06546	0.018520	1	6	5	24	110.58	200.695	11	1	-6.42	60	2.500
1009-5	5	59.8	20	80	29	51	7.4	-51.38	2.97	49.8	93.2	9.28	180.190	-0.06516	0.018520	1	6	5	24	110.58	200.695	11	1	-6.42	60	2.500
1009-0	0	59.2	0	0	0	0	9.2	-54.91	3.26	60.4	99.6	4.82	189.210	-0.07560	0.029247	0	5	5	29	85.36	284.947	10	2	-6.19	59	2.094
1009-1	1	59.2	2	98	93	68	9.2	-54.149	3.82	59.1	93.9	4.82	189.210	-0.07476	0.029247	0	5	5	29	85.36	284.947	10	2	-6.19	59	2.094
1009-2	2	59.2	26	74	26	50	9.7	-54.877	3.26	60.7	96.3	4.82	189.210	-0.07934	0.029247	0	5	5	29	85.36	284.947	10	2	-6.19	59	2.094
1009-3	3	59.2	30	70	31	39	11.0	-54.991	3.93	72.8	97.7	4.82	189.210	-0.07926	0.029247	0	5	5	29	85.36	284.947	10	2	-6.19	59	2.094
1009-4	4	59.2	11	89	82	28	10.7	-54.48	3.26	70.9	99.7	4.82	189.210	-0.07124	0.029247	0	5	5	29	85.36	284.947	10	2	-6.19	59	2.094
1009-5	5	59.2	8	92	69	43	11.7	-54.91	5.70	76.9	99.1	4.82	189.210	-0.07055	0.029247	0	5	5	29	85.36	284.947	10	2	-6.19	59	2.094
1028-0	0	29.0	0	0	0	0	10.6	-25.926	1.16	66.1	93.9	4.82	826.800	-0.06462	0.010462	2	5	1	29	109.1	836.529	6	3	4.92	57	1.966
1028-1	1	29.0	76	22	0	22	10.6	-25.962	1.32	66.1	93.9	4.82	826.800	-0.06679	0.010462	2	5	1	29	109.1	836.529	6	3	4.92	57	1.966
1028-2	2	29.0	80	20	0	20	10.6	-25.667	1.16	66.1	93.9	4.82	826.800	-0.06529	0.010462	2	5	1	29	109.1	836.529	6	3	4.92	57	1.966
1028-3	3	29.0	66	32	7	25	11.9	-25.765	1.16	69.2	93.8	4.82	826.800	-0.06622	0.010462	2	5	1	29	109.1	836.529	6	3	4.92	57	1.966
1028-4	4	29.0	54	46	30	16	10.9	-25.711	1.12	67.4	92.6	4.82	826.800	-0.06299	0.010462	2	5	1	29	109.1	836.529	6	3	4.92	57	1.966
1028-5	5	29.0	42	58	28	30	11.9	-25.122	1.11	90.0	90.0	4.82	826.800	-0.06120	0.010462	2	5	1	29	109.1	836.529	6	3	4.92	57	1.966
1042-0	0	29.1	0	0	0	0	10.0	-16.909	2.81	51.8	98.2	9.56	818.916	-0.02658	0.029242	1	9	2	39	69.69	863.997	5	4	1.43	78	2.000
1042-1	1	29.1	60	40	18	22	12.1	-16.899	2.93	68.2	91.8	9.56	818.916	-0.02654	0.029242	1	9	2	39	69.69	863.997	5	4	1.43	78	2.000
1042-2	2	29.1	55	45	16	27	12.3	-16.099	2.92	68.6	91.4	9.56	818.916	-0.02118	0.029242	1	9	2	39	69.69	863.997	5	4	1.43	78	2.000
1042-3	3	29.1	91	99	21	47	13.2	-16.271	3.48	92.8	87.4	9.56	818.916	-0.02169	0.029242	1	9	2	39	69.69	863.997	5	4	1.43	78	2.000
1042-4	4	29.1	82	70	39	34	13.4	-12.964	3.47	93.4	86.6	9.56	818.916	-0.01947	0.029242	1	9	2	39	69.69	863.997	5	4	1.43	78	2.000
1042-5	5	29.1	22	78	38	62	13.5	-12.699	3.36	93.8	86.2	9.56	818.916	-0.01909	0.029242	1	9	2	39	69.69	863.997	5	4	1.43	78	2.000
1049-0	0	29.2	0	0	0	0	13.9	-16.621	3.76	93.2	84.8	9.55	826.940	-0.04786	0.029226	9	6	2	66	58.92	860.061	6	4	-0.56	104	2.261
1049-1	1	29.2	8	92	42	50	13.9	-16.685	3.97	93.2	84.8	9.55	826.940	-0.04764	0.029226	9	6	2	66	58.92	860.061	6	4	-0.56	104	2.261
1049-2	2	29.2	1	99	51	48	14.0	-16.921	3.91	93.8	84.4	9.55	826.940	-0.04749	0.029226	9	6	2	66	58.92	860.061	6	4	-0.56	104	2.261
1049-3	3	29.2	19	81	31	69	14.0	-16.931	4.22	99.5	83.5	9.55	826.940	-0.04765	0.029226	9	6	2	66	58.92	860.061	6	4	-0.56	104	2.261
1049-4	4	29.2	27	73	36	40	13.6	-16.421	4.07	91.9	86.1	9.55	826.940	-0.04725	0.029226	9	6	2	66	58.92	860.061	6	4	-0.56	104	2.261
1049-5	5	29.2	29	71	22	68	13.4	-16.188	3.98	91.1	86.9	9.55	826.940	-0.04663	0.029226	9	6	2	66	58.92	860.061	6	4	-0.56	104	2.261
1078-0	0	58.6	0	0	0	0	10.8	-57.96	2.44	66.9	94.5	9.09	180.156	-0.09747	0.039147	1	6	5	24	110.58	200.069	11	1	-6.42	80	8.333
1078-1	1	58.6	36	64	0	64	10.4	-56.828	2.76	68.1	96.9	9.09	180.156	-0.09645	0.039147	1	6	5	24	110.58	200.069	11	1	-6.42	80	8.333
1078-2	2	58.6	19	87	9	82	11.1	-57.936	2.96	67.4	92.6	9.09	180.156	-0.09812	0.039147	1	6	5	24	110.58	200.069	11	1	-6.42	80	8.333
1078-3	3	58.6	29	71	5	66	11.1	-57.599	3.09	67.4	92.6	9.09	180.156	-0.09648	0.039147	1	6	5	24	110.58	200.069	11	1	-6.42	80	8.333
1078-4	4	58.6	34	66	16	50	11.1	-56.776	2.76	67.4	92.6	9.09	180.156	-0.09812	0.039147	1	6	5	24	110.58	200.069	11	1	-6.42	80	8.333
1078-5	5	58.6	91	99	14	94	11.2	-56.795	3.28	68.0	92.0	9.09	180.156	-0.09812	0.039147	1	6	5	24	110.58	200.069	11	1	-6.42	80	8.333
1079-0	0	59.1	0	0	0	0	9.2	-51.809	1.05	27.2	72.8	5.53	201.589	-0.12649	0.029265	0	6	4	29	97.77	216.289	10	3	-1.64	66	2.000
1079-1	1	59.1	92	68	16	50	8.8	-56.706	2.21	66.1	93.9	5.53	201.589	-0.09818	0.029265	0	6	4	29	97.77	216.289	10	3	-1.64	66	2.000
1079-2	2	59.1	21	79	19	60	8.7	-57.861	3.16	65.0	93.0	5.53	201.589	-0.09822	0.029265	0	6	4	29	97.77	216.289	10	3	-1.64	66	2.000
1079-3	3	59.1	86	14	11	63	8.8	-58.03	3.02	66.1	93.9	5.53	201.589	-0.09111	0.029265	0	6	4	29	97.77	216.289	10	3	-1.64	66	2.000
1079-4	4	59.1	42	57	16	26	7.8	-57.259	3.62	63.8	99.2	5.53	201.589	-0.09579	0.029265	0	6	4	29	97.77	216.289	10	3	-1.64	66	2.000
1079-5	5	59.1	89	89	22	68	8.2	-57.981	3.86	62.8	97.2	5.53	201.589	-0.09609	0.029265	0	6	4	29	97.77	216.289	10	3	-1.64	66	2.000
1093-0	0	57.1	0	0	0	0	8.7	-53.504	2.00	59.2	90.8	9.60	107.150	-0.09825	0.039368	1	1	1	17	12.03	192.68	2	1	1.59	51	8.000
1093-1	1	57.1	56	44	6	38	8.5	-53.568	2.64	56.1	91.9	9.60	107.150	-0.09685	0.039368	1	1	1	17	12.03	192.68	2	1	1.59	51	8.000
1093-2	2	57.1	54	46	4	42	8.8	-53.291	2.68	60.4	99.6	9.60	107.150	-0.09618	0.039368	1	1	1	17	12.03	192.68	2	1	1.59	51	8.000
1093-3	3	57.1	70	30	8	22	8.7	-52.29	2.42	59.2	90.8	9.60	107.150	-0.09547	0.039368	1	1	1	17	12.03	192.68	2	1	1.59	51	8.000
1093-4	4	57.1	68	52	29	23	7.8	-9.960	2.11	66.5	99.5	9.60	107.150	-0.09825	0.039368	1	1	1	17	12.03	192.68	2	1	1.59	51	8.000
1093-5	5	57.1	62	60	56	29	7.8	-9.929	2.94	65.7	94.3	9.60	107.150	-0.09736	0.039368	1	1	1	17	12.03	192.68	2	1	1.59	51	8.000
1094-0	0	58.5	0	0	0	0	9.1	-6.163	1.99	21.4	78.6	9.29	81.116	-0.03059	0.039862	0	1	0	10	4.80	124.365	1	1	0.84	41	8.164
1094-1	1	58.5	98	2	2	0	9.0	-6.183	1.28	22.8	77.2	9.29	81.116	-0.03098	0.039862	0	1	0	10	4.80	124.365	1	1	0.84	41	8.164
1094-2	2	58.5	94	66	1	65	9.1	-6.329	1.42	21.4	78.6	9.29	81.116	-0.09997	0.039862	0	1	0	10	4.80	124.365	1	1	0.84	41	8.164
1094-3	3	58.5	67	33	1	31	9.1	-6.286	1.49	26.2	73.8	9.29	81.116	-0.09968	0.039862	0	1	0	10	4.80	124.365	1	1	0.84	41	8.164
1094-4	4	58.5	72	28	6	23	9.1	-6.129	1.26	21.4	78.6	9.29	81.116	-0.03021	0.039862	0	1	0	10	4.80	124.365	1	1	0.84	41	8.164
1094-5	5	58.5	79	21	9	12	9.0	-6.260	1.92	22.8	77.2	9.29	81.116	-0.09963	0.039862	0	1	0	10	4.80	124.365	1	1	0.84	41	8.164
1097-0	0	51.0	0	0	0	0	8.8	-6.326	1.92	49.4	96.6	9.47	96.111	-0.09676	0.040112	0	1	1	10	20.23	117.186	2	1	0.45	42	8.291
1097-1	1	51.0	2	98	92	6	9.1	-6.182	2.68	54.1	95.9	9.47	96.111	-0.09894	0.040112	0	1	1	10	20.23	117.186	2	1	0.45	42	8.291
1097-2	2	51.0	2	98	87	11	9.8	-7.661	2.94	52.9	97.7	9.47	96.111	-0.09574	0.040112	0	1	1	10	20.23	117.186	2	1	0.45	42	8.291
1097-3	3	51.0	4	96	75	21	9.8	-7.668	2.60	56.5	91.5	9.47	96.111	-0.09148	0.040112	0	1	1	10	20.23	117.186	2	1	0.45	42	8.291
1097-4	4	51.0	8	92	59	32	7.1	-7.858	2.76	62.9	97.1	9.47	96.111	-0.09343	0.040112	0	1	1	10	20.23	117.186	2	1	0.45	42	8.291
1097-5	5	51.0	10	90	55	35	7.8	-8.044	2.60	64.7	95.3	9.47	96.111	-0.09547	0.040112	0	1	1	10	20.23	117.186	2	1	0.45	42	8.291
1099-0	0	58.9	0	0	0	0	8.4	-20.209	0.40	66.4	99.6	5.82	142.596	-0.14216	0.040362	1	6	2	20	66.07	181.95	6	1	-1.62	60	8.000
1099-1	1	58.9	0	100	100	0	8.8	-16.259	0.99	66.9	99.9	5.82	142.596	-0.13455	0.040362	1	6	2	20	66.07	181.95	6	1	-1.62	60	8.000
1099-2	2	58.9	20	80	56	24	8.9	-14.126	0.97	67.0	93.0	5.82	142.596	-0.09857	0.040362	1	6	2	20	66.07	181.95	6	1	-1.62	60	8.000
1099-3	3	58.9	9	91	58	33	9.6	-14.582	0.89	50.7	49.3	5.82	142.596	-0.12216	0.040362	1	6	2	20	66.07	181.95	6	1	-1.62	60	8.000
1099-4	4	58.9	28	72	49	23	9.7	-14.061	0.98	51.2	48.8	5.82	142.596	-0.09895	0.040362	1	6	2	20	66.07	181.95	6	1	-1.62	60	8.000
1099-5	5	58.9	16	76	42	33	9.5	-14.957	0.70	50.2	49.8	5.82	142.596	-0.13522	0.040362	1	6	2	20	66.07	181.95	6	1	-1.62	60	8.000
1110-0	0	58.0	0	0	0	0	7.1	-21.21	0.92	66.8	93.2	5.82	198.155	-0.13411	0.039903	1	5	3	25	88.3	199.693	8	1	-0.64	64	8.048
1110-1	1	58.0	0	100	100	0	7.8	-17.264	0.82	63.9	99.5	5.82	198.155	-0.13997	0.039903	1	5	3	25	88.3	199.693	8	1	-0.64	64	8.048
1110-2	2	58.0	1	99	80	19	8.7	-16.911	0.89	65.1	94.9	5.82	198.155	-0.13995	0.039903	1	5	3	25	88.3	199.693	8	1	-0.64	64	8.048
1110-3	3	58.0	1	99	89	10	9.5	-16.994	1.05	69.3	90.7	5.82	198.155	-0.09703	0.039903	1	5	3	25	88.3	199.693	8	1	-0.64	64	8.048
1110-4	4	58.0	28	72	50	22	9.5	-16.255	1.19	69.3	90.7	5.82	198.155	-0.09652	0.039903	1	5	3	25	88.3	199.693	8	1	-0.64	64	8.048
1110-5	5	58.0	16	76	47	29	10.7	-16.429	2.00	55.5	44.5	5.82	198.155	-0.09128	0.039903	1	5	3	25	88.3	199.693	8	1	-0.64	64	8.048
1110-0	0	21.2	0	0	0	0	11.1	-16.969	2.96	52.3	47.7	6.26	918.949	-0.09296	0.029998	9	9	1	58	55.86	841.597	4	4	4.82	59	1.593
1110-1	1	21.2	10	90	4	86	11.4	-16.293	3.97	53.7	46.3	6.26	918.949	-0.09187	0.029998	9	9	1	58	55.86	841.597	4	4	4.82	59	1.593
1110-2	2	21.2	16	76	4	72	12.1	-16.787	4.12	57.0	43.0	6.26	918.949	-0.09268	0.029998	9	9	1	58	55.86	841.597	4	4	4.82	59	1.593
1110-3	3	21.2	29	77	16	63	12.9	-16.997	4.46	58.9	41.1	6.26	918.949	-0.09914	0.029998	9	9	1	58	55.86	841.597	4	4	4.82	59	1.593
1110-4	4	21.2	20	80	12	68	12.8	-15.829	4.42	60.9	39.7	6.26	918.949	-0.09249	0.029998	9	9	1	58	55.86	841.597	4	4	4.82	59	1.593
1110-5	5	21.2	18	82	19	63	13.9	-16.689	4.96	65.5	34.5	6.26	918.949	-0.09282	0.029998	9	9	1	58	55.86	841.597	4	4	4.82	59	1.593
1128-0	0	58.1	0	0	0	0	7.4	-54.145	0.80	61.0	99.0	5.19	122.121	-0.02764	0.029612	1	2	0	14	40.07	140.262	2	1	0.86	58	2.714
1128-1	1	58.1	8	94	90	44	7.9	-52.593	1.39	61.9	98.9	5.19	122.121	-0.09961	0.029612	1	2	0	14	40.07	140.262	2	1	0.86	58	2.714
1128-2	2	58.1	7	93	25	68	7.8	-52.924	1.16	63.7	96.9	5.19	122.121	-0.02255	0.029612	1	2	0	14	40.07	140.262	2	1	0.86	58	2.714
1128-3	3	58.1	0	100	60	40	8.4	-51.941	1.86	66.9	93.3	5.19	122.121	-0.04287	0.029612	1	2	0	14	40.07	140.262	2	1	0.86	58	2.714
1128-4	4	58.1	1	99	42	56	8.6	-51.928	1.76	67.6	92.4	5.19	122.121	-0.09179	0.029612	1	2	0	14	40.07	140.262	2	1	0.86	58	2.714
1128-5	5	58.1	1	99	88	12	9.0	-52.957	2.32	51.5	48.5	5.19	122.121	-0.09972	0.029612	1	2	0	14	40.07	140.262	2	1	0.86	58	2.714
1127-0	0	59.0	0	0	0	0	9.1	-52.267	2.32	47.1	52.9	9.60	242.916	-0.05275	0.032724	0	9	2	57	56.74	814.256	5	3	2.17	68	1.189
1127-1	1	59.0	6	96	6	90	9.5	-52.221	2.76	49.2	50.8	9.60	242.916	-0.09255	0.032724	0	9	2	57	56.74	814.256	5	3	2.17	68	1.189
1127-2	2	59.0	40	60	32	8	9.6	-52.955	3.56	49.7	50.3	9.60	242.916	-0.09299	0.032724	0	9	2	57	56.74	814.256	5	3	2.17	68	1.189
1127-3	3	59.0	58	42	35	7	9.5	-52.664	3.26	49.2	50.8	9.60	242.916	-0.09132	0.032724	0	9	2	57	56.74	814.256	5	3	2.17	68	1.189
1127-4	4	59.0	52	48	41	8	9.6	-52.019	2.76	49.7	50.3	9.60	242.916	-0.09958	0.032724	0	9	2	57	56.74	814.256	5	3	2.17	68	1.189
1127-5	5	59.0	41	59	33	20	10.0	-51.927	2.45	51.8	48.2	9.60	242.916	-0.04708	0.032724	0	9	2	57	56.74	814.256	5	3	2.17	68	1.189
1148-0	0	22.4	0	0	0	0	10.4	-14.957	2.99	66.4	93.6	6.59	226.291	-0.09638	0.029862	2	2	2	33	62.46	296.281	4	2	1.71	67	1.727
1148-1	1	22.4	2	98	76	22	10.0	-14.999	2.90	66.6	93.4	6.59	226.291	-0.09294	0.029862	2	2	2	33	62.46	296.281	4	2	1.71	67	1.727
1148-2	2	22.4	7	93	53	40	11.2	-14.901	3.86	50.0	50.0	6.59	226.291	-0.09089	0.029862	2	2	2	33	62.46	296.281	4	2	1.71	67	1.727
1148-3	3	22.4	16	84	45	39	10.0	-14.128	3.05	44.6	55.4	6.59	226.291	-0.09139	0.029862	2	2	2	33	62.46	296.281	4	2	1.71	67	1.727
1148-4	4	22.4	19	87	58	29	11.4	-15.559	3.91	50.9	49.1	6.59	226.291	-0.09852	0.029862	2	2	2	33	62.46	296.281	4	2	1.71	67	1.727
1148-5	5	22.4	16	89	56	31	11.7	-15.979	3.16	52.2	47.8	6.59	226.291	-0.09858	0.029862	2	2	2	33	62.46	296.281	4	2	1.71	67	1.727
1179-0	0	53.9	0	0	0	0	8.8	-56.702	1.02	68.8	91.2	7.79	75.067	-0.22169	0.139177	1	3	1	10	67.71	102.139	4	0	-1.99	47	4.702



```
METHOD_WEIGHTS ref 0.0497258 -2.46822 0.817984 0.234042 0.349976 -0.379306 1.68114 -0.4552
0.548527 -0.392527 -0.615286 0.0697327 -1.36984 -0.383895 1.0298 0.963313 0.180077 -0.436006
0.445645 0.130317
fa_atr 0.8
fa_rep 0.4
fa_sol 0.6
hack_elec 0.25
pro_close 1
fa_pair 0.8
hbond_sr_bb 2
hbond_lr_bb 2
hbond_bb_sc 2
hbond_sc 2
dslf_ss_dst 0.5
dslf_cs_ang 2
dslf_ss_dih 5
dslf_ca_dih 5
atom_pair_constraint 1
coordinate_constraint 1
angle_constraint 1
dihedral_constraint 1
omega 0.5
fa_dun 0.4
p_aa_pp 0.5
chainbreak 1
ref 1
```

Ligand weights. The ref line corresponds to each residue (in alphabetical order), and following those are the weight terms with their weights.

```
-in
        -path
                -database /insert_path/rosetta_database/
        -file
                -s /insert_path/XXXXX.pdb
                -extra_res_fa / insert_path /INH.params
-out
        -pdb
        -nstruct 9

-parser
        -protocol /insert_path/xml_script.xml

-packing
        -ex1
        -ex2
        -linmem_ig 10
```

Options file, used to specify the input, output, parser file, and packing options. For input files, one must specify the path to the Rosetta database, path to the ligand PDB file(s), and the path

to the ligand params file. For output, one must specify the type of file to be output (pdb) and number of structures to generate. For the parser, one must specify the path to the XML parser file. For packing, these are standard options to include for side chain repacking.

```
<ROSETTASCRIPTS>

        <SCOREFXNS>
                <ligand_soft_rep weights=ligand_soft_rep
                </ligand_soft_rep>
                <hard_rep weights=ligandprime>
                </hard_rep>
        </SCOREFXNS>
        <TASKOPERATIONS>
                <DetectProteinLigandInterface name=design_interface cut1=6.0 cut2=8.0 cut3=10.0
cut4=12.0 design=1 resfile="/insert_path/Resfile"/>
        </TASKOPERATIONS>
        <LIGAND_AREAS>
                <docking_sidechain chain=X cutoff=6.0 add_nbr_radius=true all_atom_mode=true
minimize_ligand=10/>
                <final_sidechain chain=X cutoff=6.0 add_nbr_radius=true all_atom_mode=true/>
                <final_backbone chain=X cutoff=7.0 add_nbr_radius=false all_atom_mode=true
Calpha_restraints=0.3/>
        </LIGAND_AREAS>
        <INTERFACE_BUILDERS>
                <side_chain_for_docking ligand_areas=docking_sidechain/>
                <side_chain_for_final ligand_areas=final_sidechain/>
                <backbone ligand_areas=final_backbone extension_window=3/>
        </INTERFACE_BUILDERS>
        <MOVEMAP_BUILDERS>
                <docking sc_interface=side_chain_for_docking minimize_water=true/>
                <final sc_interface=side_chain_for_final bb_interface=backbone
minimize_water=true/>
        </MOVEMAP_BUILDERS>
        <MOVERS>
        single movers
                <FavorNativeResidue name=favor_native bonus=1.25/>
                <ddG name=calculateDDG jump=1 symmetry=0 per_residue_ddg=1 repack=0
scorefxn=hard_rep/>
                <Translate name=translate chain=X distribution=uniform angstroms=2.0 cycles=50/>
                <Rotate name=rotate chain=X distribution=uniform degrees=360 cycles=1000/>
                <SlideTogether name=slide_together chain=X/>
                <HighResDocker name=high_res_docker cycles=6 repack_every_Nth=3
scorefxn=ligand_soft_rep movemap_builder=docking/>
                <PackRotamersMover name=designinterface scorefxn=hard_rep
task_operations=design_interface/>
                <FinalMinimizer name=final scorefxn=hard_rep movemap_builder=final/>
                <InterfaceScoreCalculator name=add_scores chains=X scorefxn=hard_rep
native="/insert_path/2087.pdb"/>
        compound movers
                <ParsedProtocol name=low_res_dock>
                        <Add mover_name=translate/>
                        <Add mover_name=rotate/>
                        <Add mover_name=slide_together/>
                </ParsedProtocol>
                <ParsedProtocol name=high_res_dock>
                        <Add mover_name=high_res_docker/>
                        <Add mover_name=final/>
                </ParsedProtocol>
        </MOVERS>
        <PROTOCOLS>
                <Add mover_name=low_res_dock/>
                <Add mover_name=favor_native/>
                <Add mover_name=designinterface/>
                <Add mover_name=high_res_dock/>
                <Add mover_name=calculateDDG/>
                <Add mover_name=add_scores/>
```

```
        </PROTOCOLS>
</ROSETTASCRIPTS>
```

XML Script, for the experiments discussed in this study, used to assign values for the cut-off points to detect the protein-small molecule interface, a value for the favor native residue bonus, and values for ligand translation and rotation. In TASKOPERATIONS:DetectProteinLigandInterface, one must specify values to determine which residues surrounding the ligand are allowed to be designed and/or repacked (details in the text above), and specify the path to the resfile. In MOVERS:FavorNativeResidue, one must assign a value to the bonus. In MOVER:Translate, one must assign a value (in Å) to how much the ligand is allowed to translate from its original position. In MOVER:Rotate, one must assign a value to how much the ligand is allowed to rotate (in deg) from its original position (360 being full rotation allowed).

```
ALLAA
AUTO

start
```

Resfile, used to indicate that residues considered for design and repack are limited to the cut-off points specified above.

```
/insert_path/rosetta_scripts.default.linuxgccrelease -out:prefix 2028_$f-
@/insert_path/flags.txt >& log.$f.txt
```

Executable, which is the script to run the program. One must specify the path to the executable and the path to the options files (seen as 'flags.txt'). One may choose to specify an output prefix, which was useful for scripts used in analysis because all designs generated from the same parent complex had the same handle at the beginning of the filename.

```
# needs native pdb XXXX.pdb
# needs list of score files XXXX*score.sc
# option 1: 4-letter PDB code (XXXX)
# option 2: number X of amino acids mutated

# calculate the ddg. outputs average, standard deviation, etc...
/insert_path/ddg_mover_collater.py $1*score.sc > $1_scores.txt;

# construct map of residue names to ids
grep "ATOM   .... CA" $1.pdb | awk '(aaid = int( substr( $0, 23, 4)); aaname = $4; print "AAID",
aaid, "AANAME", aaname)' > $1_map.txt;
```

```
# find X residues with strongest interaction to ligand to be mutated
cat $1_scores.txt | awk '(if( pr == 1) print $0; if($1 == "ref") pr = 1;)' | sort -nk2 | awk -v
num=$2 '(lin++; if( lin > 1 && lin <= num+1) print "AAMUTATE", $1, $2 )' > $1_top_$2_mutate.txt;

# create human readable file describing mutants
cat $1_map.txt $1_top_$2_mutate.txt | awk -v pdb=$1 '( if ($1 == "AAID") aa_name[$2] = $4; if( $1
== "AAMUTATE") print "MUTATION " aa_name[$2], $2 " ALA FILE " pdb "_" ++i ".pdb ENERGY " $3
".";)' > $1_top_$2.txt;

# copy native pdb into XXXX_0.pdb (no mutations)
cp $1.pdb $1_0.pdb;

# create X mutant pdb files adding the mutations one by one in order of strong to weaker
interactions
cat $1_top_$2.txt | awk -v pdb=$1 '( lin++; system("/home/allisoba/scripts/mutate_to_ala.awk "
pdb "_" lin-1 " " $3 " > " pdb "_" lin ".pdb;");)';
```

Script to create Alanine mutant pdbs. This script is run in the directory with the native pdb code and score files, and needs the 4-letter code (as designated in the executable script, which matches the native code), and the number of subsequent Alanine mutated PDBs you wish to create (in our case, 5). This script sorts the score file by per-residue-ddg, to determine which residues contribute most to the protein-small molecule interface. These residues are then mutated to Alanine in a new file designated by XXXX_1.pdb (one Alanine mutation), XXXX_2.pdb (two Alanine mutations), and so on until the number specified has been reached. This script creates the XXXX_top_Y.txt file which will be needed later.

```
#!/usr/bin/env python2.5

from operator import itemgetter
import numpy
import sys
from optparse import OptionParser
from  Bio.PDB import *
import warnings
from Bio.PDB import PDBExceptions
warnings.simplefilter('ignore',PDBExceptions.PDBConstructionWarning)

usage = "%prog [options] <list of score files>"
parser = OptionParser(usage)
parser.add_option("--num_top_total",dest="num_top_total",help="first, filter out the top total
score structures using this many models, default=10",default="100000")
parser.add_option("--num_top_ddg",dest="num_top_ddg",help="second, take the top X by ddg and
generate the statistics,default=10",default="100000")
parser.add_option("--pdb",dest="pdb",help="input pdb for mapping the per-residue ddgs. Probably
best to use the input or output struc from the ddg analysis",default="")
parser.add_option("--stdev_putty",dest="stdev_putty",help="map the mean per-residue ddg to
occupancy and stdev of ddgs to bfactor so it can be shown as putty",action="store_true")
(options,args)=parser.parse_args()

def line_of_floats(line_of_strings):
        line_of_numbers = []
        for string in line_of_strings:
                number = float(string)
                line_of_numbers.append(number)
        return line_of_numbers
```

```
header = []
models = []
for score_file in args:
	score_lines = open(score_file,'r').readlines()

	for line in score_lines:
		line = line.strip().split()
		if line[0] == 'SEQUENCE:':
			continue
		if line[1] == 'total_score':
			trash = line.pop(0)
			trash = line.pop(-1)
			header = line
		else:
			trash = line.pop(0)
			trash = line.pop(-1)
			models.append(line_of_floats(line))

sort_total_score = sorted(models, key=itemgetter(0),reverse=True)
top_scores = sort_total_score[0:int(options.num_top_total)]

sort_total_ddg = sorted(top_scores, key=itemgetter(1),reverse=True)

top_ddgs = sort_total_ddg[0:int(options.num_top_ddg)]

top_ddgs = numpy.array(top_ddgs,dtype=float)

mean = numpy.mean(top_ddgs, axis=0)
st_dev = numpy.std(top_ddgs, axis=0)
max = numpy.max(top_ddgs, axis=0)
min = numpy.min(top_ddgs,axis=0)
median = numpy.median(top_ddgs,axis=0)
#print mean

def strip_path(pdb):
	return pdb.split('/')[-1]

per_res_dict = ()

print '%20s %10s %10s %10s %10s %10s' % ('score','mean','st_dev','min','max','median')
min_mean = 0
for i in range(len(header)):
	if header[i].split('_')[0] == 'residue':
		field = header[i].split('_')[2]
		per_res_dict[field] = (mean[i],st_dev[i])
		if min_mean > mean[i]:
			min_mean = mean[i]
	else:
		field = header[i]

	print '%20s %10s %10s %10s %10s %10s' % (field, str(round(mean[i],3)),
str(round(st_dev[i],3)), str(round(min[i],3)), str(round(max[i],3)), str(round(median[i],3)))

if not options.pdb == "":
	print 'mapping b-factors to',options.pdb
	PDBParse = PDBParser(PERMISSIVE=1)
	struct = PDBParse.get_structure('X',options.pdb)
	atoms = struct.get_atoms()
	curr_residue_id = 0
	prev_res_id = 0
	for atom in atoms:
		residue_id = atom.get_parent().get_id()[1]
		if residue_id is not prev_res_id:
			curr_residue_id += 1
			prev_res_id = residue_id
		#print residue_id, curr_residue_id, prev_res_id,per_res_dict[str(curr_residue_id)]
		if options.stdev_putty:
			atom.set_occupancy(per_res_dict[str(curr_residue_id)][0])
			atom.set_bfactor(per_res_dict[str(curr_residue_id)][1])
```

```
                else:
                        atom.set_bfactor(per_res_dict[str(curr_residue_id)][0])
        io=PDBIO()
        io.set_structure(struct)
        io.save(strip_path(options.pdb)[0:-4]+'_ddg_b-factor.pdb')
        print 'writing',strip_path(options.pdb)[0:-4]+'_ddg_b-factor.pdb'

        pymol_script = open(strip_path(options.pdb)[0:-4]+'_ddg_b-factor.pml','w')
        pymol_script.write("hide everything\n"
                "show cartoon\n"
                "cartoon putty\n")
        if options.stdev_putty:
                pymol_script.write("spectrum q, minimum="+str(min_mean)+",
maximum="+str(abs(min_mean))+"\n")
        else:
                pymol_script.write("spectrum b, minimum="+str(min_mean)+",
maximum="+str(abs(min_mean))+"\n")
        pymol_script.close()
```

The ddg_mover_collater.py script that is called in the preceding script.

```
# parameter 1 - four letter PDB code of native
# parameter 2 - how many models to keep?
echo "+++0 usage: get_best_design_models.inp <four letter PDB code of native> <how many models to
keep>";
#
# grep score, interface score, and rmsd
echo "+++1 computing score, interface score, and rmsd";
/bin/ls $1_*_????.pdb | awk '(system ("/home/allisoba/scripts/interf_tot_rmsd.inp " $1 ";"))' >
interf_tot_rmsd.out;
#
# get top $2 by interface score
echo "+++2 copy best interface score models to best_interface_models";
mkdir best_interface_models;
sort -nk3 interf_tot_rmsd.out | head -n $2 > best_interface_models.txt;
sort -nk3 interf_tot_rmsd.out | head -n $2 | awk '(system("cp " $1 " best_interface_models"))';
#
# tar and zip all models
echo "+++4 tar and zip all models into all_models.tgz";
tar -czf $1_models.tgz $1*_????.pdb;
rm -f $1*_????.pdb;
```

Script to get the best design models by interface energy. Once all the models for a complex have been generated (in our case, we generate 1000 designs per input pdb), one should then filter these by interface energy, and then only keep the top models for analysis. The script should be run as: get_best_design_models.inp <four letter PDB code of native> <how many models to keep>. This script takes the filename and score from every design and puts in them in a separate file which is sorted by value, then the script copies the top designs into a new directory, best_interface_models, and then all designs are zipped into a file in case one needs them again at a later point. At this point, the top 50 designs are in their own directory, ready for analysis.

```
more $1_top_*.txt $1.pdb $3/*.pdb | awk -v mut_num=$2 '(
        if( $1 == "MUTATION")
        (
                mut++;
                mut_num_seq[mut] = $3;
                mut_num_aa[mut] = $2;
        )
        if( substr($1,length($1)-2,3) == "pdb") file = $1;
        if( $1 == "ATOM" && $3 == "N")
        (
                tmp_seq[++num_pdb] = $4;
        )
        if( $1 != "ATOM" && $1 != "REMARK" && last == "ATOM")
        (

                num_seq++;
                filename[num_seq] = file;
                score[num_seq] = 0;
                rep[num_seq] = 0;
                atr[num_seq] = 0;
                hbb[num_seq] = 0;
                hsc[num_seq] = 0;
                cou[num_seq] = 0;
                sol[num_seq] = 0;
                rms[num_seq] = 0;
                for( i = 1; i <= num_pdb; i++)
                        if( tmp_seq[i] == "ASN") seq[num_seq,i] = "N";
                        else if( tmp_seq[i] == "ASP") seq[num_seq,i] = "D";
                        else if( tmp_seq[i] == "GLU") seq[num_seq,i] = "E";
                        else if( tmp_seq[i] == "GLN") seq[num_seq,i] = "Q";
                        else if( tmp_seq[i] == "TYR") seq[num_seq,i] = "Y";
                        else if( tmp_seq[i] == "TRP") seq[num_seq,i] = "W";
                        else if( tmp_seq[i] == "LYS") seq[num_seq,i] = "K";
                        else if( tmp_seq[i] == "PHE") seq[num_seq,i] = "F";
                        else if( tmp_seq[i] == "ARG") seq[num_seq,i] = "R";
                        else seq[num_seq,i] = substr(tmp_seq[i],0,1);
                tmp_num = num_pdb;
                num_pdb = 0;
                ala_rev[num_seq] = 0;
                ala_ala[num_seq] = 0;
                ala_mut[num_seq] = 0;
                for( i = 1; i <= mut_num; i++)
                (
                                if( tmp_seq[mut_num_seq[i]] == mut_num_aa[i]) ala_rev[num_seq]++;
                        else if( tmp_seq[mut_num_seq[i]] == "ALA") ala_ala[num_seq]++;
                        else ala_mut[num_seq]++;
                )
        )
        if( $1 == "interface_delta_X") score[num_seq] = $2;
        if( $1 == "if_X_fa_rep") rep[num_seq] = $2;
        if( $1 == "if_X_fa_atr") atr[num_seq] = $2;
        if( $1 == "if_X_hbond_bb_sc") hbb[num_seq] = $2;
        if( $1 == "if_X_hbond_sc") hsc[num_seq] = $2;
        if( $1 == "if_X_hack_elec") cou[num_seq] = $2;
        if( $1 == "if_X_fa_sol") sol[num_seq] = $2;
        if( $1 == "ligand_rms_no_super_X") rms[num_seq] = $2;
        last = $1;
) END (
        min_score_min_rms = 0;
        min_score_max_rms = 0;
        for( j = 1; j <= tmp_num; j++)
        (
                p[j] = 0;
                for(k=2;k<=num_seq;k++) if(seq[1,j]!=seq[k,j]) p[j] = 1;
        )
        print tmp_num;
        printf " 100 x        ";
        for( j = 1; j <= tmp_num; j++) if( p[j]) printf int(j/100);
        printf "   0 -99.000\n";
        printf "  10 x        ";
        for( j = 1; j <= tmp_num; j++) if( p[j]) printf (int(j/10))%10;
```

```
        printf "   0 -99.000\n";
        printf "   1 x        ";
        for( j = 1; j <= tmp_num; j++) if( p[j]) printf j%10;
        printf "   0 -99.000     rms ala_tot ala_rev ala_ala ala_mut     rep     atr     sol
hbb     hsc     cou filename\n";
        for( i = 1; i <= num_seq; i++)
        (
                printf "%3d: %7.3f  ", i, score[i];
                eq_seq = 0;
                for( j = 1; j <= tmp_num; j++) if(p[j])
                (
                        printf seq[i,j];
                        if( seq[i,j] != seq[1,j]) eq_seq++;
                )
                if( eq_seq != 0) sc_seq = score[i] / eq_seq;
                else sc_seq = -98;
                printf " %3d %7.3f %7.2f %7d %7d %7d %7d %7.3f %7.3f %7.3f %7.3f %7.3f %7.3f
%s\n", eq_seq, sc_seq, rms[i], mut_num, ala_rev[i], ala_ala[i], ala_mut[i], rep[i], atr
[i], sol[i], hbb[i], hsc[i], cou[i], filename[i];
                if( i > 1)
                (
                        tot_mut     += eq_seq;
                        tot_ala_rev += ala_rev[i];
                        tot_ala_ala += ala_ala[i];
                        tot_ala_mut += ala_mut[i];
                        tot_rms += rms[i];
                        if( i == 2 || rms[i] < min_rms) min_rms = rms[i];
                        if( i == 2 || rms[i] > max_rms) max_rms = rms[i];
                        tot_score += score[i];
                        if( i == 2 || score[i] < min_score) min_score = score[i];
                        if( i == 2 || score[i] > max_score) max_score = score[i];
                        if( rms[i] <= 2)
                        (
                                if( score[i] < min_score_min_rms) min_score_min_rms = score[i];
                        )
                        else
                        (
                                if( score[i] < min_score_max_rms) min_score_max_rms = score[i];
                        )
                )
        )
        pos_num = 0; for( j = 1; j <= tmp_num; j++) if( p[j]) pos_num++;
        printf "NUMBER_MODELS                        %5d\n", num_seq-1;
        printf "SEQUENCE_LENGTH                      %5d\n", tmp_num;
        printf "POSITIONS_MUTATED_FROM_WT_BY_ROSETTA %5d\n", pos_num;
        printf "POSITIONS_MUTATED_TO_ALA_MANUALLY    %5d\n", mut_num;
        printf "SCORE_AVERAGE                        %7.3f\n", tot_score / ( num_seq-1);
        printf "SCORE_MIN                            %7.3f\n", min_score;
        printf "SCORE_MAX                            %7.3f\n", max_score;
        printf "RMS_AVERAGE                          %5.2f\n", tot_rms / ( num_seq-1);
        printf "RMS_MIN                              %5.2f\n", min_rms;
        printf "RMS_MAX                              %5.2f\n", max_rms;
        printf "SCORE_DELTA_RMS_BELOW_2A             %7.3f\n", min_score_min_rms -
min_score_max_rms;
        printf "AVERAGE_MUTATIONS_IN_BINDING_SITE    %5.1f %5.0f %%\n", tot_mut     / ( num_seq-
1), 100 * tot_mut     / ( num_seq-1) / pos_num;
        printf "ALA_REVERTED_TO_WILDTYPE             %5.1f %5.0f %%\n", tot_ala_rev / ( num_seq-
1), 100 * tot_ala_rev / ( num_seq-1) / mut_num;
        printf "ALA_NOT_REVERTED_TO_WILDTYPE         %5.1f %5.0f %%\n", mut_num - ( tot_ala_rev /
( num_seq-1)), 100 * (1 - tot_ala_rev / ( num_seq-1) / mut_num);
        printf "ALA_KEPT_INCORRECT_ALA               %5.1f %5.0f %%\n", tot_ala_ala / ( num_seq-
1), 100 * tot_ala_ala / ( num_seq-1) / mut_num;
        printf "ALA_MUTATED_TO_OTHER_AA              %5.1f %5.0f %%\n", tot_ala_mut / ( num_seq-
1), 100 * tot_ala_mut / ( num_seq-1) / mut_num;
)'
```

Script that performs sequence recovery and Alanine to wild type recovery analysis, called "seq_complex_compare.inp". This script is run in the directory that contains the native PDB,

the Alanine mutants file XXXX_top_Y.txt, and the best_interface_models directory. The script is run as: seq_complex_compare.inp <4-letterPDB code> <number of Alanine mutations in original PDB> <best_interface_models>. The results can be output into a file by adding "> analysis.txt" at the end of the command. The output file includes the statistics averaged from the top 50 best models, providing results such as sequence recovery, Alanine to wild type recovery, interface score, and RMSD. From here, these numbers can be input into an Excel sheet to make graphs.

Appendix B: Experimental and Computational Identification of Naïve Binders to a TIM-Barrel Protein Scaffold

Discussion

12 binders that were not included in set of 'naïve binders'. Some of the ligands identified as hits were not included in the group of 'naïve binders'. Although these ligands caused significant (interesting) peak shifts in the NMR spectra of residues in the binding pocket, the binding curves either fit more of a straight line (saturation could not be guesstimated) and/or indicated a binding event other than one-site specific binding (Figure 20 (SF3)). These were included in the search for matching ligands, but excluded in the overall naïve binder set.

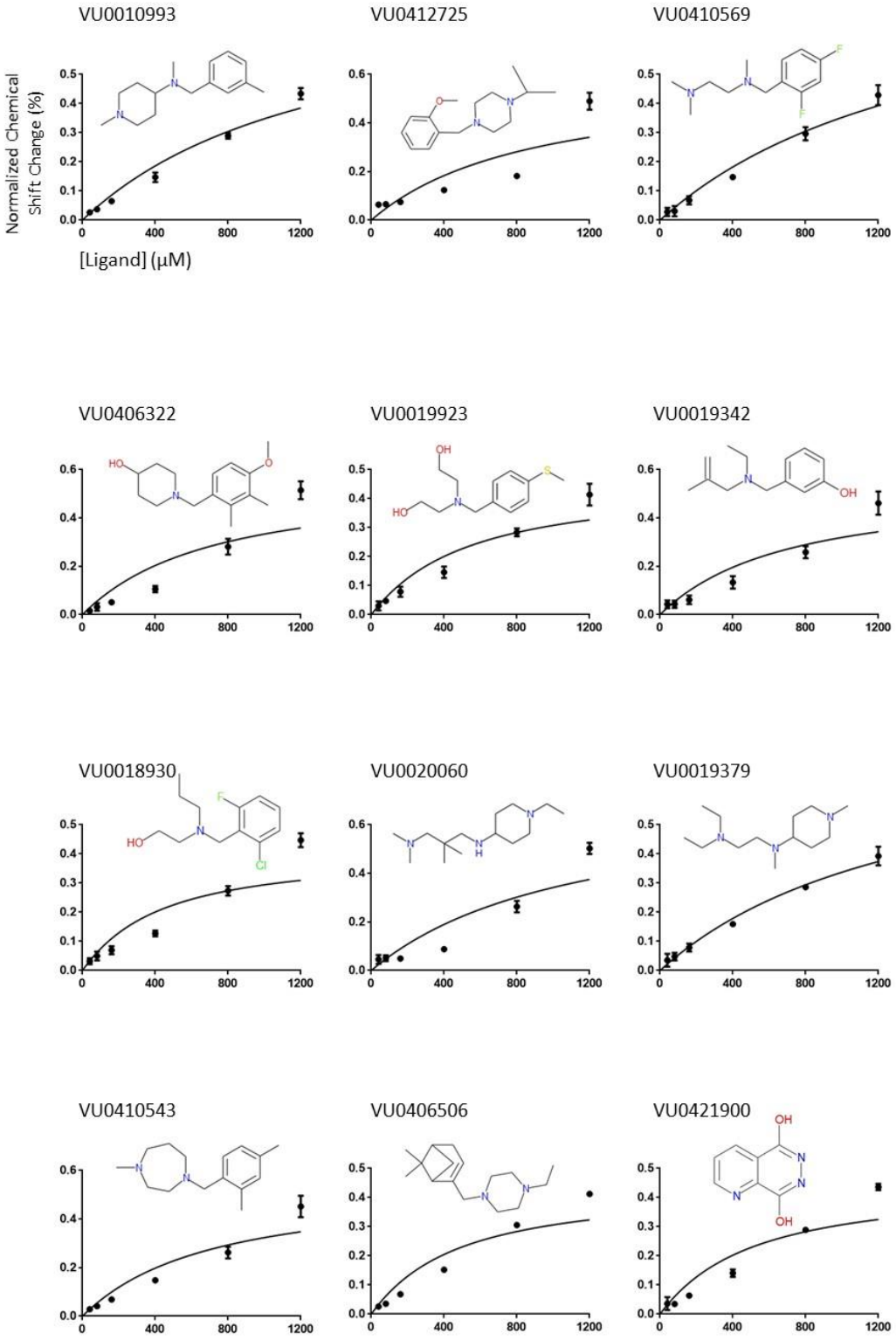


Figure 20 (SF3)Average normalized chemical shift (%) vs [Ligand] (μM) for compounds that induced peak shifts but were not included in the 'naïve binders' set.

Inclusion vs exclusion of Val127 and Leu170 in significant shift recovery

Upon an in-depth look at the recovery of specific residues, it was noticed that Val127 and Leu170 were never recovered by Rosetta (0% recovery) (Table 8 (ST6)), despite the fact that both of these residues are often identified experimentally as residues interacting with the ligand. Looking at these residues in pymol reveals that that both of these residue side chains point away from the ligand. Because the HMQC-NMR experiment measures the backbone H and NH, we believe that these residues contribute to the ligand binding interaction via the backbone atoms, not the side chain atoms. Considering that ddg is the predicted interaction of a residue with the ligand, and because these side chains point away, we suspect this is the reason why Rosetta excluded these residues. We conducted a follow-up analysis of recovery, excluding Val127 and Leu170, to assess how this may change the landscape of significant shift recovery (Figure 21 (SF4)). The average recovery jumped to 67%, a 14% increase. Rosetta achieved 100% recovery for an additional 5 protein-ligand complexes, bringing the total to 7 complexes. In comparing the recovery against the same metrics as above, surprisingly no new or different trends emerge (Figure 22 (SF5)). The only change is ligand number of atoms 'clustering' trend reverts to no trend at all. As expected, the significant shift recovery for most of the complexes jumps higher, allowing for the majority of the complexes to have a 75% or higher recovery.

	# of significant shift occurrences	# of times Rosetta recovered	% recovery
L169	5	4	80
D51	7	7	100
G80	8	8	100
Y143	9	9	100
G145	9	4	44
S144	12	11	92
V127	13	0	0
D130	13	13	100
A128	16	16	100
L170	16	0	0
S201	16	3	19

Table 8 (ST6). Residue recovery. Broken down by individual residues, Rosetta recovery of the residues identified in the ligand-binding hot spot.

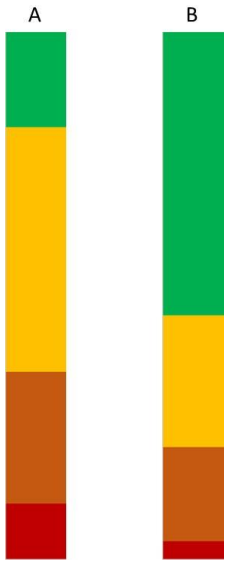


Figure 21 (SF4). Comparison of significant shift recovery with inclusion (A) and exclusion (B) of V127 and L170. Percent recovery, 0-24% (red), 25-49% (orange), 50-74% (yellow), and 75-100% (green).

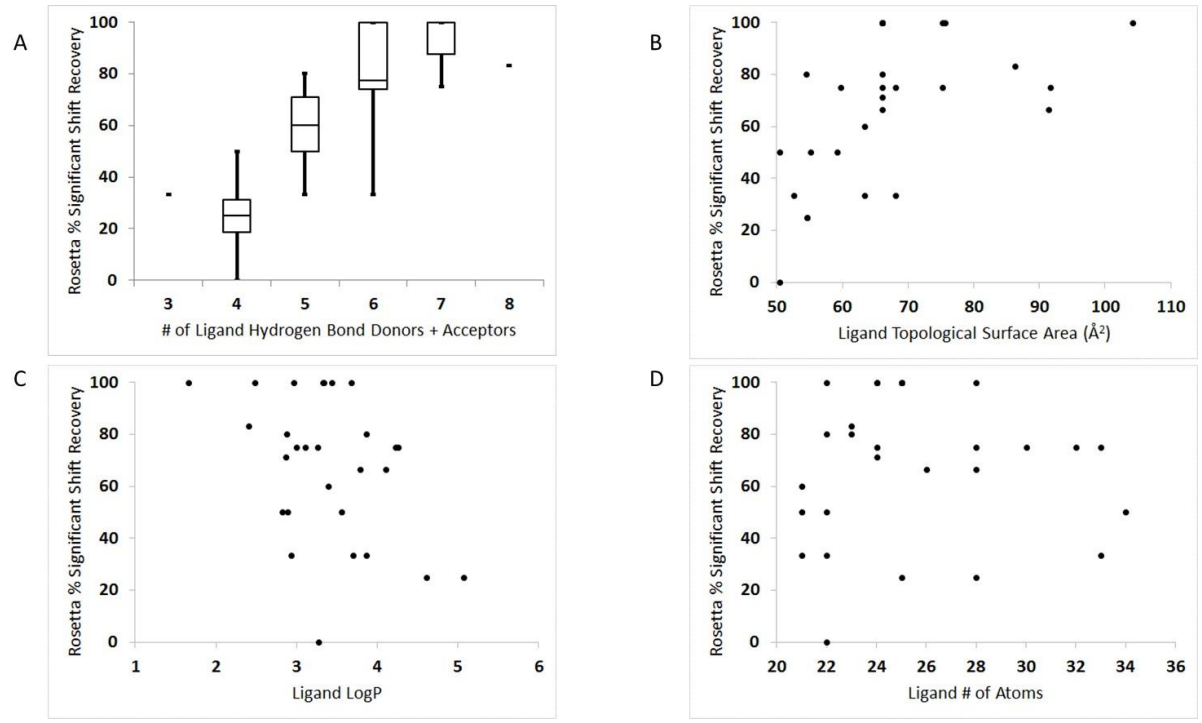


Figure 22 (SF5). Significant shift recovery by Rosetta, excluding V127 and L170. Trends remain consistent as previously shown, with the exception of ligand number of atoms which now shows no trend.

Scripts and Commands

Options file, used to specify the input, output, parser file, and packing options. For input files, one must specify the path to the Rosetta database, path to the ligand PDB file(s), and the path to the ligand params file. For output, one must specify the type of file to be output (pdb) and number of structures to generate. For the parser, one must specify the path to the XML parser file. For packing, these are standard options to include for side chain repacking.

```
-in
        -path
                -database /path_to_database/database/
        -file
                -s /path_to_pdb/filename.pdb
                -extra_res_fa /path_to_params/ligand.params
-out
        -level 300
        -pdb_gz
```

```
        -path
                -pdb    /path_to_output_files/
                -score  /path_to_output_files
        -nstruct 100
        -mute all
        -unmute protocols.jd2.JobDistributor

-parser
        -protocol /path_to_RosettaScripts/RosettaScripts.xml

-packing
        -ex1
        -ex2
        -linmem_ig 10
```

XML Script, for the experiments discussed in this study, used to assign values for the cut-off points to detect the protein-small molecule interface, a value for the favor native residue bonus, and values for ligand translation and rotation. In TASKOPERATIONS:DetectProteinLigandInterface, one must specify values to determine which residues surrounding the ligand are allowed to be designed and/or repacked (details in my first manuscript), and specify the path to the resfile. In Transform, must specify how much ligand movement is allowed.

```
<ROSETTASCRIPTS>

        <SCOREFXNS>
                <ligand_soft_rep weights=ligand_soft_rep>
                        <Reweight scoretype=fa_elec weight=0.42/>
                        <Reweight scoretype=hbond_bb_sc weight=1.3/>
                        <Reweight scoretype=hbond_sc weight=1.3/>
                        <Reweight scoretype=rama weight=0.2/>
                </ligand_soft_rep>
                <hard_rep weights=ligandprime>
                        <Reweight scoretype=fa_intra_rep weight=0.004/>
                        <Reweight scoretype=fa_elec weight=0.42/>
                        <Reweight scoretype=hbond_bb_sc weight=1.3/>
                        <Reweight scoretype=hbond_sc weight=1.3/>
                        <Reweight scoretype=rama weight=0.2/>
                </hard_rep>
        </SCOREFXNS>
        <SCORINGGRIDS ligand_chain="X" width="16">
                <vdw grid_type="ClassicGrid" weight="1.0"/>
        </SCORINGGRIDS>
        <TASKOPERATIONS>
                <DetectProteinLigandInterface name=design_interface cut1=6.0 cut2=8.0 cut3=10.0
cut4=12.0 design=1 resfile="/path_to_resfile/Resfile_dock"/>
        </TASKOPERATIONS>
        <LIGAND_AREAS>
                <docking_sidechain chain=X cutoff=6.0 add_nbr_radius=true all_atom_mode=true
minimize_ligand=10/>
                <final_sidechain chain=X cutoff=6.0 add_nbr_radius=true all_atom_mode=true/>
                <final_backbone chain=X cutoff=7.0 add_nbr_radius=false all_atom_mode=true
Calpha_restraints=0.3/>
        </LIGAND_AREAS>
        <INTERFACE_BUILDERS>
                <side_chain_for_docking ligand_areas=docking_sidechain/>
```

```
                <side_chain_for_final ligand_areas=final_sidechain/>
                <backbone ligand_areas=final_backbone extension_window=3/>
        </INTERFACE_BUILDERS>
        <MOVEMAP_BUILDERS>
                <docking sc_interface=side_chain_for_docking minimize_water=true/>
                <final sc_interface=side_chain_for_final bb_interface=backbone
minimize_water=true/>
        </MOVEMAP_BUILDERS>
        <MOVERS>
        single movers
                <StartFrom name=start_from_X chain=X>
                        <Coordinates x=25.325 y=35.021 z=22.716/>
                </StartFrom>
                <FavorNativeResidue name=favor_native bonus=1.0/>
                <ddG name=calculateDDG jump=1 per_residue_ddg=1 repack=0 scorefxn=hard_rep/>
                <Transform name="transform" chain="X" box_size="5.0" move_distance="1.0"
angle="360" cycles="500" temperature="5" initial_perturb="5.0"/>
                <HighResDocker name=high_res_docker cycles=1 repack_every_Nth=1
scorefxn=ligand_soft_rep movemap_builder=docking/>
                <PackRotamersMover name=designinterface scorefxn=hard_rep
task_operations=design_interface/>
                <FinalMinimizer name=final scorefxn=hard_rep movemap_builder=final/>
                <InterfaceScoreCalculator name=add_scores chains=X scorefxn=hard_rep/>
        </MOVERS>
        <PROTOCOLS>
                <Add mover_name=start_from_X/>
                <Add mover_name=transform/>
                <Add mover_name=favor_native/>
                <Add mover_name=high_res_docker/>
                <Add mover_name=final/>
                <Add mover_name=calculateDDG/>
                <Add mover_name=add_scores/>
        </PROTOCOLS>
</ROSETTASCRIPTS>
```

Resfile, used to indicate that residues considered for design and repack are limited to the cut-off points specified above.

```
#These commands will be applied to all residue positions that lack a specified behavior in the
body:

NATAA              # allow only native residues (for docking only; no design allowed)

AUTO

start
```

Appendix C: Designed C9S_HisF Binds VU0068924 More Tightly

VU0068924 was identified as a naïve binder to C9S_HisF with a binding affinity of 442 ± 10 μM. VU0068924 properties include: 245 g/mol, 6 hydrogen bond donors + acceptors, 2 rotatable bonds, 3 rings, contains an acid group, LogP 4.22, and TPSA 68 Å^2. Using RosettaLigand, the six mutations L50V, G80A, A128T, D130M, S144W, T171A were

introduced into the protein, 6166_C9S_HisF (computational analysis done by lab member Brian Bender). The program suggested these mutations would induce tighter binding. 6166_C9S_HisF was expressed and purified, and I carried out the ^{15}N-HMQC-NMR titration style experiments. The NMR samples were set up as: 6166_C9S_HisF at 50 μM with VU0068924 at 0 μM (reference spectrum), and 2.5, 5, 10, 20, 30, 40, 60, 80, 200, 400, and 600 μM. Due to the tight binding nature of this protein-ligand complex, the ligand concentrations had to be lowered and measured at smaller increases compared to the setup of the naïve binders measured with C9S_HisF. Raw data processing, excel analysis, dissociation constant, and binding curves were calculated as described previously. Indeed, the binding affinity increased to 23 ± 2 μM (Figure 23 (SF6)). The next step was to further investigate this tighter binder, and assess which mutations contribute most to the binding. Protein variants were created for a 'back titration', where each mutation was reverted back to wild type one by one, giving 6 daughter proteins from the original designed 6 mutation protein: MinusL50V_6166_C9S_HisF, MinusG80A_6166_C9S_HisF, MinusA128T_6166_C9S_HisF, MinusD130M_6166_C9S_HisF, MinusS144W_6166_C9S_HisF, and MinusT171A_6166_C9S_HisF. These daughter proteins were expressed and purified as previously described, ^{15}N-HMQC-NMR titration style experiments performed, each protein measured with 0, 2.5, 5, 10, 20, 30, 40, 60, 80, 150, 250, and 500 μM ligand, and data analysis carried out in the same way as 6166_C9S_HisF. Dissociation constants were calculated: MinusL50V_6166_C9S_HisF Kd = 36 ± 2 μM; MinusG80A_6166_C9S_HisF Kd = 38 ± 2 μM, MinusA128T_6166_C9S_HisF Kd = 30 ± 4 μM, MinusD130M_6166_C9S_HisF Kd = 187 ± 32 μM, MinusS144W_6166_C9S_HisF Kd = 106 ± 7 μM, and MinusT171A_6166_C9S_HisF Kd = 62 ± 3 μM (Figure 24 (SF7)). This project is still in

progress, as 6166_C9S_HisF continues to be further characterized, and the other naïve binders being docked/designed into C9S_HisF with RosettaLigand.

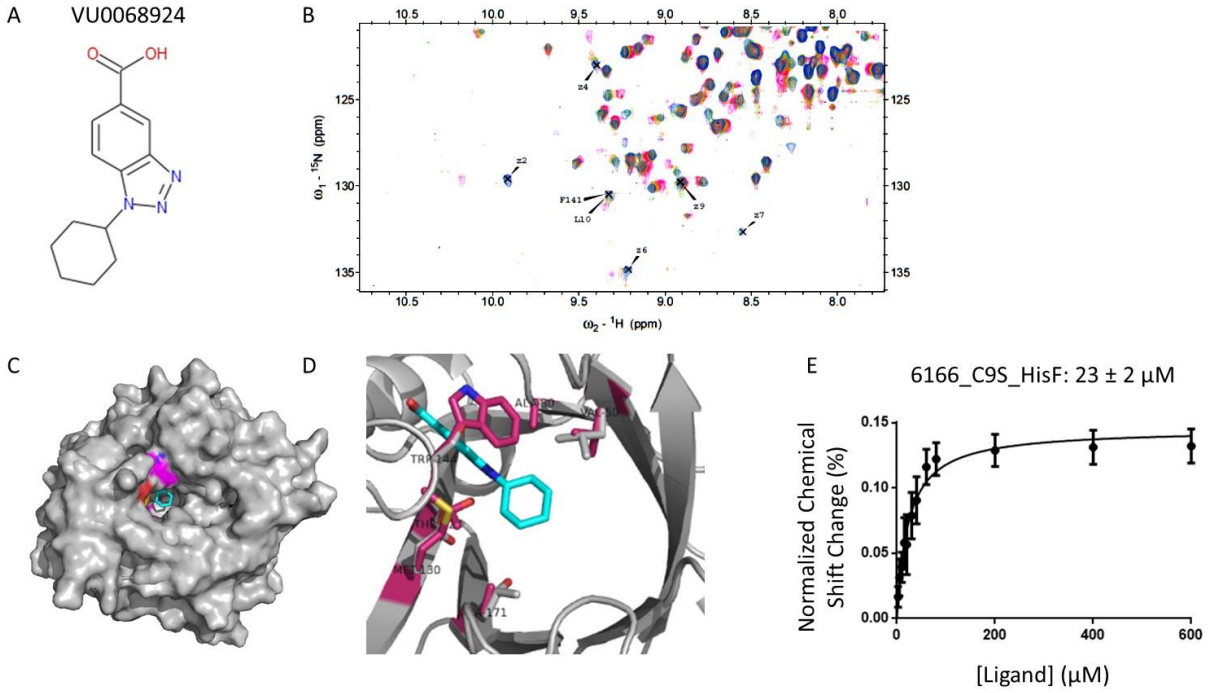


Figure 23 (SF6). Designed 6166_C9S_HisF binds VU0068924 more tightly. (A) Chemical structure of VU0068924. (B) Overlay of titration style spectra, 6166_C9S_HisF measured with 0, 2.5, 5, 10, 20, 30, 40, 60, 80, 200, 400, and 600 µM ligand. (C) Surface representation of Rosetta model, VU0068924 docked 6166_C9S_HisF. (D) Cartoon representation of Rosetta model, mutations L50V, G80A, A128T, D130M, S144W, T171A highlighted in magenta. (E) Binding curve and binding affinity calculated.

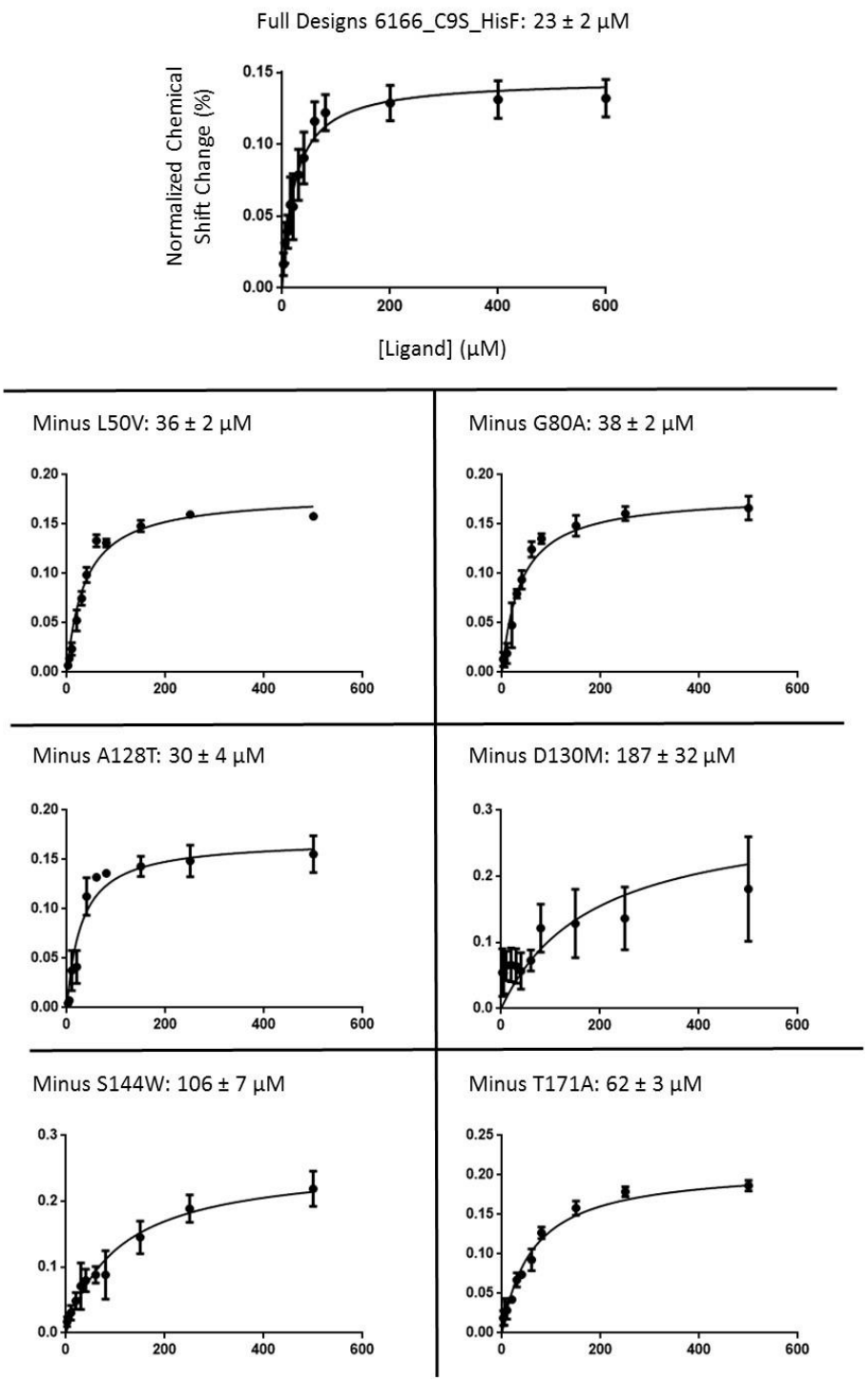


Figure 24 (SF7). Back titration of 6166_C9S_HisF with each mutation reverted back to wild type. Binding curves and binding affinities calculated. These results suggest that mutations D130M, S144W, and T171A contribute most to the tighter interaction

Summarization of key findings and future directions: Designed_C9S_HisF binds VU0068924 more tightly. The key finding of this project thus far, is that RosettaLigand was used to dock naïve binder VU0068924 while allowing mutations in the C9S_HisF binding pocket. RosettaLigand designed in 8 mutations that would induce tighter binding, and we decided upon 6 (L50V, G80A, A128T, D130M, S144W, T171A) to experimentally test, designed_C9S_HisF. The binding affinity increased from ~442 μM to ~23 μM. The back titration of designed_C9S_HisF suggests that mutations D130M, S144W, and T171A contribute most to the tighter interaction. In moving forward with computationally designing C9S_HisF to more tightly bind the other naïve binders, I propose some thoughts based on knowledge gained through this experiment and previous experiments. On the computational side, I suggest optimizing the favor native residue bonus. With designing in mutations, it is imperative that this metric is benchmarked and optimized. When analyzing designs, the ddg should be output and reviewed but not necessarily taken into account, based on results above. On the experimental side, I suggest a fresh 2D resonance assignment transfer from HisF to C9S_HisF. With the subsequent designs, even more assignments are lost, therefore we should start off with as many peak assignments as possible. I suggest expressing and purifying, and performing the NMR experiments all in triplicate to establish confidence that the results are consistent and reproducible. Thorough and confirmed sequencing at each step is also necessary. For the designs that induce binding that approaches the lower end of the micro-molar scale, other approaches to calculate Kd should be considered. In this range (~5 – 50 μM), intermediate exchange becomes as issue, and the NMR spectrum resonances broaden which cause the peaks to disappea[51]. Also, once there are many designs that indicate tighter binding, all designs and

their ligands should be tested against the other designs computationally and experimentally, to produce a matrix of results to analyze. This matrix would allow us to elucidate how small changes in a ligand affect RosettaLigand design choices, and also if RosettaLigand is unnecessarily inputting or excluding certain mutations from the designs.

REFERENCES

1. Leader, B.; Baca, Q. J.; Golan, D. E., Protein therapeutics: a summary and pharmacological classification. *Nature Reviews Drug Discovery* **2008,** *7* (1), 21-39.
2. Schreier, B.; Stumpp, C.; Wiesner, S.; Höcker, B., Computational design of ligand binding is not a solved problem. *Proceedings of the National Academy of Sciences* **2009,** *106* (44), 18491-6.
3. Morin, A.; Kaufmann, K. W.; Fortenberry, C.; Harp, J. M.; Mizoue, L. S.; Meiler, J., Computational design of an endo-1,4-beta-xylanase ligand binding site. *Protein Engineering, Design and Selection* **2011,** *24* (6), 503-16.
4. Stehlin, C.; Dahm, A.; Kirschner, K., Deletion mutagenesis as a test of evolutionary relatedness of indoleglycerol phosphate synthase with other TIM barrel enzymes. *FEBS Letters* **1997,** *403* (3), 268-72.
5. Lang, D.; Thoma, R.; Henn-Sax, M.; Sterner, R.; Wilmanns, M., Structural evidence for evolution of the beta/alpha barrel scaffold by gene duplication and fusion. *Science* **2000,** *289* (5484), 1546-50.
6. Meiler, J.; Baker, D., ROSETTALIGAND: protein-small molecule docking with full side-chain flexibility. *Proteins* **2006,** *65* (3), 538-48.
7. Davis, I. W.; Baker, D., RosettaLigand docking with full ligand and receptor flexibility. *Journal of Molecular Biology* **2009,** *385* (2), 381-92.
8. Kuhlman, B.; Dantas, G.; Ireton, G. C.; Varani, G.; Stoddard, B. L.; Baker, D., Design of a novel globular protein fold with atomic-level accuracy. *Science* **2003,** *302* (5649), 1364-8.
9. Schueler-Furman, O.; Wang, C.; Bradley, P.; Misura, K.; Baker, D., Progress in modeling of protein structures and interactions. *Science* **2005,** *310* (5748), 638-42.
10. Shuker, S. B.; Hajduk, P. J.; Meadows, R. P.; Fesik, S. W., Discovering high-affinity ligands for proteins: SAR by NMR. *Science* **1996,** *274* (5292), 1531-4.
11. Hajduk, P. J.; Gerfin, T.; Boehlen, J. M.; Häberli, M.; Marek, D.; Fesik, S. W., High-throughput nuclear magnetic resonance-based screening. *Journal of Medicinal Chemistry* **1999,** *42* (13), 2315-7.
12. Strohl, W. R.; Knight, D. M., Discovery and development of biopharmaceuticals: current issues. *Current Opinion in Biotechnology* **2009,** *20* (6), 668-72.
13. Aggarwal, R. S., What's fueling the biotech engine-2012 to 2013. *Nature Biotechnology* **2014,** *32* (1), 32-9.
14. Jones, D. S.; Silverman, A. P.; Cochran, J. R., Developing therapeutic proteins by engineering ligand-receptor interactions. *Trends in Biotechnology* **2008,** *26* (9), 498-505.
15. Gebauer, M.; Skerra, A., Engineered protein scaffolds as next-generation antibody therapeutics. *Current Opinion in Chemical Biology* **2009,** *13* (3), 245-55.
16. Kariolis, M. S.; Kapur, S.; Cochran, J. R., Beyond antibodies: using biological principles to guide the development of next-generation protein therapeutics. *Current Opinion Biotechnology* **2013,** *24* (6), 1072-7.

17. Huang, P. S.; Feldmeier, K.; Parmeggiani, F.; Fernandez Velasco, D. A.; Höcker, B.; Baker, D., De novo design of a four-fold symmetric TIM-barrel protein with atomic-level accuracy. *Nature Chemical Biology* **2016,** *12* (1), 29-34.
18. Huang, Y. M.; Banerjee, S.; Crone, D. E.; Schenkelberg, C. D.; Pitman, D. J.; Buck, P. M.; Bystroff, C., Toward Computationally Designed Self-Reporting Biosensors Using Leave-One-Out Green Fluorescent Protein. *Biochemistry* **2015,** *54* (40), 6263-73.
19. Penchovsky, R., Computational design of allosteric ribozymes as molecular biosensors. *Biotechnology Advances* **2014,** *32* (5), 1015-27.
20. Pierce, B. G.; Hellman, L. M.; Hossain, M.; Singh, N. K.; Vander Kooi, C. W.; Weng, Z.; Baker, B. M., Computational design of the affinity and specificity of a therapeutic T cell receptor. *PLOS Computational Biology* **2014,** *10* (2), e1003478.
21. Procko, E.; Hedman, R.; Hamilton, K.; Seetharaman, J.; Fleishman, S. J.; Su, M.; Aramini, J.; Kornhaber, G.; Hunt, J. F.; Tong, L.; Montelione, G. T.; Baker, D., Computational design of a protein-based enzyme inhibitor. *Journal of Molecular Biology* **2013,** *425* (18), 3563-75.
22. Bjelic, S.; Nivón, L. G.; Çelebi-Ölçüm, N.; Kiss, G.; Rosewall, C. F.; Lovick, H. M.; Ingalls, E. L.; Gallaher, J. L.; Seetharaman, J.; Lew, S.; Montelione, G. T.; Hunt, J. F.; Michael, F. E.; Houk, K. N.; Baker, D., Computational design of enone-binding proteins with catalytic activity for the Morita-Baylis-Hillman reaction. *ACS Chemical Biology* **2013,** *8* (4), 749-57.
23. Mills, J. H.; Khare, S. D.; Bolduc, J. M.; Forouhar, F.; Mulligan, V. K.; Lew, S.; Seetharaman, J.; Tong, L.; Stoddard, B. L.; Baker, D., Computational design of an unnatural amino acid dependent metalloprotein with atomic level accuracy. *Journal of the American Chemical Society* **2013,** *135* (36), 13393-9.
24. Fleishman, S. J.; Whitehead, T. A.; Ekiert, D. C.; Dreyfus, C.; Corn, J. E.; Strauch, E. M.; Wilson, I. A.; Baker, D., Computational design of proteins targeting the conserved stem region of influenza hemagglutinin. *Science* **2011,** *332* (6031), 816-21.
25. Ashworth, J.; Taylor, G. K.; Havranek, J. J.; Quadri, S. A.; Stoddard, B. L.; Baker, D., Computational reprogramming of homing endonuclease specificity at multiple adjacent base pairs. *Nucleic Acids Research* **2010,** *38* (16), 5601-8.
26. Siegel, J. B.; Zanghellini, A.; Lovick, H. M.; Kiss, G.; Lambert, A. R.; St Clair, J. L.; Gallaher, J. L.; Hilvert, D.; Gelb, M. H.; Stoddard, B. L.; Houk, K. N.; Michael, F. E.; Baker, D., Computational design of an enzyme catalyst for a stereoselective bimolecular Diels-Alder reaction. *Science* **2010,** *329* (5989), 309-13.
27. Chen, C. Y.; Georgiev, I.; Anderson, A. C.; Donald, B. R., Computational structure-based redesign of enzyme activity. *Proceedings of the National Academy of Sciences* **2009,** *106* (10), 3764-9.
28. Grigoryan, G.; Reinke, A. W.; Keating, A. E., Design of protein-interaction specificity gives selective bZIP-binding peptides. *Nature* **2009,** *458* (7240), 859-64.
29. Calhoun, J. R.; Liu, W.; Spiegel, K.; Dal Peraro, M.; Klein, M. L.; Valentine, K. G.; Wand, A. J.; DeGrado, W. F., Solution NMR structure of a designed metalloprotein and complementary molecular dynamics refinement. *Structure* **2008,** *16* (2), 210-5.
30. Röthlisberger, D.; Khersonsky, O.; Wollacott, A. M.; Jiang, L.; DeChancie, J.; Betker, J.; Gallaher, J. L.; Althoff, E. A.; Zanghellini, A.; Dym, O.; Albeck, S.; Houk, K. N.; Tawfik, D. S.; Baker, D., Kemp elimination catalysts by computational enzyme design. *Nature* **2008,** *453* (7192), 190-5.

31. Jiang, L.; Althoff, E. A.; Clemente, F. R.; Doyle, L.; Röthlisberger, D.; Zanghellini, A.; Gallaher, J. L.; Betker, J. L.; Tanaka, F.; Barbas, C. F.; Hilvert, D.; Houk, K. N.; Stoddard, B. L.; Baker, D., De novo computational design of retro-aldol enzymes. *Science* **2008,** *319* (5868), 1387-91.
32. Sood, V. D.; Baker, D., Recapitulation and design of protein binding peptide structures and sequences. *Journal of Molecular Biology* **2006,** *357* (3), 917-27.
33. Humphris, E. L.; Kortemme, T., Design of multi-specificity in protein interfaces. *PLOS Computational Biology* **2007,** *3* (8), e164.
34. Kortemme, T.; Morozov, A. V.; Baker, D., An orientation-dependent hydrogen bonding potential improves prediction of specificity and structure for proteins and protein-protein complexes. *Journal of Molecular Biology* **2003,** *326* (4), 1239-59.
35. Feldmeier, K.; Höcker, B., Computational protein design of ligand binding and catalysis. *Current Opinion in Chemical Biology* **2013,** *17* (6), 929-33.
36. Chakrabarti, R.; Klibanov, A. M.; Friesner, R. A., Computational prediction of native protein ligand-binding and enzyme active site sequences. *Proceedings of the National Academy of Sciences* **2005,** *102* (29), 10153-8.
37. Tinberg, C. E.; Khare, S. D.; Dou, J.; Doyle, L.; Nelson, J. W.; Schena, A.; Jankowski, W.; Kalodimos, C. G.; Johnsson, K.; Stoddard, B. L.; Baker, D., Computational design of ligand-binding proteins with high affinity and selectivity. *Nature* **2013,** *501* (7466), 212-6.
38. Waszkowycz, B.; Clark, D. E.; Gancia, E. Outstanding challenges in protein-ligand docking and structure-based virtual screening *Wiley Interdisciplinary Reviews: Computational Molecular Science* [Online], 2011, p. 229-259. http://onlinelibrary.wiley.com/doi/10.1002/wcms.18/abstract (accessed 01).
39. Khare, S. D.; Whitehead, T. A., Introduction to the Rosetta Special Collection. *PLoS One* **2015,** *10* (12), e0144326.
40. Allison, B.; Combs, S.; DeLuca, S.; Lemmon, G.; Mizoue, L.; Meiler, J., Computational design of protein-small molecule interfaces. *Journal of Structural Biology* **2014,** *185* (2), 193-202.
41. Lemmon, G.; Meiler, J., Rosetta Ligand docking with flexible XML protocols. *Methods in Molecular Biology* **2012,** *819*, 143-55.
42. Douangamath, A.; Walker, M.; Beismann-Driemeyer, S.; Vega-Fernandez, M. C.; Sterner, R.; Wilmanns, M., Structural evidence for ammonia tunneling across the (beta alpha)(8) barrel of the imidazole glycerol phosphate synthase bienzyme complex. *Structure* **2002,** *10* (2), 185-93.
43. Nagano, N.; Orengo, C. A.; Thornton, J. M., One fold with many functions: the evolutionary relationships between TIM barrel families based on their sequences, structures and functions. *Journal of Molecular Biology* **2002,** *321* (5), 741-65.
44. Dvir, H.; Harel, M.; McCarthy, A. A.; Toker, L.; Silman, I.; Futerman, A. H.; Sussman, J. L., X-ray structure of human acid-beta-glucosidase, the defective enzyme in Gaucher disease. *EMBO Reports* **2003,** *4* (7), 704-9.
45. Pisano, C.; Vlodavsky, I.; Ilan, N.; Zunino, F., The potential of heparanase as a therapeutic target in cancer. *Biochemical Pharmacology* **2014,** *89* (1), 12-9.
46. Huang, C. J.; Guo, S. H.; Chung, S. C.; Lin, Y. J.; Chen, C. Y., Analysis of the involvement of chitin-binding domain of ChiCW in antifungal activity, and engineering a novel chimeric chitinase with high enzyme and antifungal activities. *Journal of Microbiology and Biotechnology* **2009,** *19* (10), 1169-75.

47. Fortenberry, C.; Bowman, E. A.; Proffitt, W.; Dorr, B.; Combs, S.; Harp, J.; Mizoue, L.; Meiler, J., Exploring symmetry as an avenue to the computational design of large protein domains. *Journal of the American Chemical Society* **2011,** *133* (45), 18026-9.
48. Sperl, J. M.; Rohweder, B.; Rajendran, C.; Sterner, R., Establishing catalytic activity on an artificial (βα)8-barrel protein designed from identical half-barrels. *FEBS Letters* **2013,** *587* (17), 2798-805.
49. Lipchock, J. M.; Loria, J. P., 1H, 15N and 13C resonance assignment of imidazole glycerol phosphate (IGP) synthase protein HisF from Thermotoga maritima. *Biomolecular NMR Assignments* **2008,** *2* (2), 219-21.
50. Chaudhuri, B. N.; Lange, S. C.; Myers, R. S.; Davisson, V. J.; Smith, J. L., Toward understanding the mechanism of the complex cyclization reaction catalyzed by imidazole glycerolphosphate synthase: crystal structures of a ternary complex and the free enzyme. *Biochemistry* **2003,** *42* (23), 7003-12.
51. Harner, M. J.; Frank, A. O.; Fesik, S. W., Fragment-based drug discovery using NMR spectroscopy. *Journal of Biomolecular NMR* **2013,** *56* (2), 65-75.
52. Krishnamoorthy, J.; Yu, V. C.; Mok, Y. K., Auto-FACE: an NMR based binding site mapping program for fast chemical exchange protein-ligand systems. *PLoS One* **2010,** *5* (2), e8943.
53. Khoury, G. A.; Smadbeck, J.; Kieslich, C. A.; Floudas, C. A., Protein folding and de novo protein design for biotechnological applications. *Trends in Biotechnology* **2014,** *32* (2), 99-109.
54. Knudsen, K. E.; Scher, H. I., Starving the addiction: new opportunities for durable suppression of AR signaling in prostate cancer. *Clinical Cancer Research* **2009,** *15* (15), 4792-8.
55. Hao, J.; Serohijos, A. W.; Newton, G.; Tassone, G.; Wang, Z.; Sgroi, D. C.; Dokholyan, N. V.; Basilion, J. P., Identification and rational redesign of peptide ligands to CRIP1, a novel biomarker for cancers. *PLOS Computational Biology* **2008,** *4* (8), e1000138.
56. Istivan, T. S.; Pirogova, E.; Gan, E.; Almansour, N. M.; Coloe, P. J.; Cosic, I., Biological effects of a de novo designed myxoma virus peptide analogue: evaluation of cytotoxicity on tumor cells. *PLoS One* **2011,** *6* (9), e24809.
57. Bellows, M. L.; Taylor, M. S.; Cole, P. A.; Shen, L.; Siliciano, R. F.; Fung, H. K.; Floudas, C. A., Discovery of entry inhibitors for HIV-1 via a new de novo protein design framework. *Biophysical Journal* **2010,** *99* (10), 3445-53.
58. Sievers, S. A.; Karanicolas, J.; Chang, H. W.; Zhao, A.; Jiang, L.; Zirafi, O.; Stevens, J. T.; Münch, J.; Baker, D.; Eisenberg, D., Structure-based design of non-natural amino-acid inhibitors of amyloid fibril formation. *Nature* **2011,** *475* (7354), 96-100.
59. Nannemann, D. P.; Birmingham, W. R.; Scism, R. A.; Bachmann, B. O., Assessing directed evolution methods for the generation of biosynthetic enzymes with potential in drug biosynthesis. *Future Medicinal Chemistry* **2011,** *3* (7), 809-19.
60. Skerra, A., Engineered protein scaffolds for molecular recognition. *Journal of Molecular Recognition* **2000,** *13* (4), 167-87.
61. Gill, D. S.; Damle, N. K., Biopharmaceutical drug discovery using novel protein scaffolds. *Current Opinion in Biotechnology* **2006,** *17* (6), 653-8.
62. Skerra, A., Alternative binding proteins: anticalins - harnessing the structural plasticity of the lipocalin ligand pocket to engineer novel binding activities. *FEBS Journal* **2008,** *275* (11), 2677-83.

63. Carter, P. J., Introduction to current and future protein therapeutics: a protein engineering perspective. *Experimental Cell Research* **2011,** *317* (9), 1261-9.
64. Mayr, L. M.; Bojanic, D., Novel trends in high-throughput screening. *Current Opinion in Pharmacology* **2009,** *9* (5), 580-8.
65. Leach, A. R.; Shoichet, B. K.; Peishoff, C. E., Prediction of protein-ligand interactions. Docking and scoring: successes and gaps. *Journal of Medicinal Chemistry* **2006,** *49* (20), 5851-5.
66. Cheng, T.; Li, Q.; Zhou, Z.; Wang, Y.; Bryant, S. H., Structure-based virtual screening for drug discovery: a problem-centric review. *American Association of Pharmaceutical Scientists Journal* **2012,** *14* (1), 133-41.
67. Warren, G. L.; Andrews, C. W.; Capelli, A. M.; Clarke, B.; LaLonde, J.; Lambert, M. H.; Lindvall, M.; Nevins, N.; Semus, S. F.; Senger, S.; Tedesco, G.; Wall, I. D.; Woolven, J. M.; Peishoff, C. E.; Head, M. S., A critical assessment of docking programs and scoring functions. *Journal of Medicinal Chemistry* **2006,** *49* (20), 5912-31.
68. Ó Conchúir, S.; Barlow, K. A.; Pache, R. A.; Ollikainen, N.; Kundert, K.; O'Meara, M. J.; Smith, C. A.; Kortemme, T., A Web Resource for Standardized Benchmark Datasets, Metrics, and Rosetta Protocols for Macromolecular Modeling and Design. *PLoS One* **2015,** *10* (9), e0130433.
69. Kumar, A.; Zhang, K. Y., Computational fragment-based screening using RosettaLigand: the SAMPL3 challenge. *Journal of Computer-Aided Molecular Design* **2012,** *26* (5), 603-16.
70. Lemmon, G.; Kaufmann, K.; Meiler, J., Prediction of HIV-1 protease/inhibitor affinity using RosettaLigand. *Chemical Biology & Drug Design* **2012,** *79* (6), 888-96.
71. Davis, I. W.; Raha, K.; Head, M. S.; Baker, D., Blind docking of pharmaceutically relevant compounds using RosettaLigand. *Protein Science* **2009,** *18* (9), 1998-2002.
72. DeLuca, S.; Khar, K.; Meiler, J., Fully Flexible Docking of Medium Sized Ligand Libraries with RosettaLigand. *PLoS One* **2015,** *10* (7), e0132508.
73. Zhu, X.; Dickerson, T. J.; Rogers, C. J.; Kaufmann, G. F.; Mee, J. M.; McKenzie, K. M.; Janda, K. D.; Wilson, I. A., Complete reaction cycle of a cocaine catalytic antibody at atomic resolution. *Structure* **2006,** *14* (2), 205-16.
74. Clifton, M. C.; Corrent, C.; Strong, R. K., Siderocalins: siderophore-binding proteins of the innate immune system. *Biometals* **2009,** *22* (4), 557-64.
75. Baeumner, A. J., Biosensors for environmental pollutants and food contaminants. *Analytical and Bioanalytical Chemistry* **2003,** *377* (3), 434-45.
76. Zanghellini, A.; Jiang, L.; Wollacott, A. M.; Cheng, G.; Meiler, J.; Althoff, E. A.; Röthlisberger, D.; Baker, D., New algorithms and an in silico benchmark for computational enzyme design. *Protein Science* **2006,** *15* (12), 2785-94.
77. Wang, L.; Althoff, E. A.; Bolduc, J.; Jiang, L.; Moody, J.; Lassila, J. K.; Giger, L.; Hilvert, D.; Stoddard, B.; Baker, D., Structural analyses of covalent enzyme-substrate analog complexes reveal strengths and limitations of de novo enzyme design. *Journal of Molecular Biology* **2012,** *415* (3), 615-25.
78. Althoff, E. A.; Wang, L.; Jiang, L.; Giger, L.; Lassila, J. K.; Wang, Z.; Smith, M.; Hari, S.; Kast, P.; Herschlag, D.; Hilvert, D.; Baker, D., Robust design and optimization of retroaldol enzymes. *Protein Science* **2012,** *21* (5), 717-26.

79. Allert, M.; Rizk, S. S.; Looger, L. L.; Hellinga, H. W., Computational design of receptors for an organophosphate surrogate of the nerve agent soman. *Proceedings of the National Academy of Sciences* **2004,** *101* (21), 7907-12.
80. Hayden, E. C., Key protein-design papers challenged. *Nature* **2009,** *461* (7266), 859.
81. Weng, Z.; DeLisi, C., Protein therapeutics: promises and challenges for the 21st century. *Trends in Biotechnology* **2002,** *20* (1), 29-35.
82. Meiler, J.; Baker, D., ROSETTALIGAND: Protein-small molecule docking with full side-chain flexibility. *Proteins* **2006,** *65* (3), 538-548; Baker, D.; Davis, I. W., ROSETTALIGAND Docking with Full Ligand and Receptor Flexibility. *J Mol Biol* **2009,** *385* (2), 381-392.
83. Kaufmann, K. W.; Dawson, E. S.; Henry, L. K.; Field, J. R.; Blakely, R. D.; Meiler, J., Structural determinants of species-selective substrate recognition in human and Drosophila serotonin transporters revealed through computational docking studies. *Proteins* **2009,** *74* (3), 630-42.
84. DeLuca, S.; Dorr, B.; Meiler, J., Design of native-like proteins through an exposure-dependent environment potential. *Biochemistry* **2011,** *50* (40), 8521-8.
85. Gainza, P.; Roberts, K. E.; Donald, B. R., Protein design using continuous rotamers. *PLOS Computational Biology* **2012,** *8* (1), e1002335.
86. Malisi, C.; Schumann, M.; Toussaint, N. C.; Kageyama, J.; Kohlbacher, O.; Höcker, B., Binding pocket optimization by computational protein design. *PLoS One* **2012,** *7* (12), e52505.
87. Roberts, K. E.; Cushing, P. R.; Boisguerin, P.; Madden, D. R.; Donald, B. R., Computational design of a PDZ domain peptide inhibitor that rescues CFTR activity. *PLOS Computational Biology* **2012,** *8* (4), e1002477.
88. Frey, K. M.; Georgiev, I.; Donald, B. R.; Anderson, A. C., Predicting resistance mutations using protein design algorithms. *Proceedings of the National Academy of Sciences* **2010,** *107* (31), 13707-12.
89. Hallen, M. A.; Keedy, D. A.; Donald, B. R., Dead-end elimination with perturbations (DEEPer): a provable protein design algorithm with continuous sidechain and backbone flexibility. *Proteins* **2013,** *81* (1), 18-39.
90. Keedy, D. A.; Georgiev, I.; Triplett, E. B.; Donald, B. R.; Richardson, D. C.; Richardson, J. S., The role of local backrub motions in evolved and designed mutations. *PLOS Computational Biology* **2012,** *8* (8), e1002629.
91. Georgiev, I.; Lilien, R. H.; Donald, B. R., The minimized dead-end elimination criterion and its application to protein redesign in a hybrid scoring and search algorithm for computing partition functions over molecular ensembles. *Journal of Computational Chemistry* **2008,** *29* (10), 1527-42.
92. Sidhu, S. S.; Koide, S., Phage display for engineering and analyzing protein interaction interfaces. *Current Opinion in Structural Biology* **2007,** *17* (4), 481-7.
93. Sidhu, S. S.; Lowman, H. B.; Cunningham, B. C.; Wells, J. A., Phage display for selection of novel binding peptides. *Methods in Enzymology* **2000,** *328*, 333-63.
94. Dunbar, J. B.; Smith, R. D.; Yang, C. Y.; Ung, P. M.; Lexa, K. W.; Khazanov, N. A.; Stuckey, J. A.; Wang, S.; Carlson, H. A., CSAR benchmark exercise of 2010: selection of the protein-ligand complexes. *Journal of Chemical Information and Modeling* **2011,** *51* (9), 2036-46.

95. Lemmon, G.; Meiler, J., Towards ligand docking including explicit interface water molecules. *PLoS One* **2013,** *8* (6), e67536.
96. Butkiewicz, M.; Lowe, E. W.; Mueller, R.; Mendenhall, J. L.; Teixeira, P. L.; Weaver, C. D.; Meiler, J., Benchmarking ligand-based virtual High-Throughput Screening with the PubChem database. *Molecules* **2013,** *18* (1), 735-56.
97. Ertl, P.; Rohde, B.; Selzer, P., Fast calculation of molecular polar surface area as a sum of fragment-based contributions and its application to the prediction of drug transport properties. *Journal of Medicinal Chemistry* **2000,** *43* (20), 3714-7.
98. Xing, L.; Glen, R. C., Novel methods for the prediction of logP, pK(a), and logD. *Journal of Chemical Information and Modeling* **2002,** *42* (4), 796-805.
99. Altschul, S. F.; Gish, W.; Miller, W.; Myers, E. W.; Lipman, D. J., Basic local alignment search tool. *J Mol Biol* **1990,** *215* (3), 403-10.
100. Reisinger, B.; Bocola, M.; List, F.; Claren, J.; Rajendran, C.; Sterner, R., A sugar isomerization reaction established on various $(\beta\alpha)_8$-barrel scaffolds is based on substrate-assisted catalysis. *Protein Engineering, Design and Selection* **2012,** *25* (11), 751-60.
101. Copley, R. R.; Bork, P., Homology among (betaalpha)(8) barrels: implications for the evolution of metabolic pathways. *Journal of Molecular Biology* **2000,** *303* (4), 627-41.
102. Höcker, B., Design of proteins from smaller fragments-learning from evolution. *Current Opinion in Structural Biology* **2014,** *27*, 56-62.
103. Beismann-Driemeyer, S.; Sterner, R., Imidazole glycerol phosphate synthase from Thermotoga maritima. Quaternary structure, steady-state kinetics, and reaction mechanism of the bienzyme complex. *Journal of Biological Chemistry* **2001,** *276* (23), 20387-96.
104. Höcker, B.; Lochner, A.; Seitz, T.; Claren, J.; Sterner, R., High-resolution crystal structure of an artificial (betaalpha)(8)-barrel protein designed from identical half-barrels. *Biochemistry* **2009,** *48* (6), 1145-7.
105. Eisenbeis, S.; Proffitt, W.; Coles, M.; Truffault, V.; Shanmugaratnam, S.; Meiler, J.; Höcker, B., Potential of fragment recombination for rational design of proteins. *Journal of the American Chemical Society* **2012,** *134* (9), 4019-22.
106. Williamson, M. P., Using chemical shift perturbation to characterise ligand binding. *Progress in Nuclear Magnetic Resonance Spectroscopy* **2013,** *73*, 1-16.
107. Combs, S. A.; Deluca, S. L.; Deluca, S. H.; Lemmon, G. H.; Nannemann, D. P.; Nguyen, E. D.; Willis, J. R.; Sheehan, J. H.; Meiler, J., Small-molecule ligand docking into comparative models with Rosetta. *Nature Protocols* **2013,** *8* (7), 1277-98.
108. Schanda, P.; Brutscher, B., Very fast two-dimensional NMR spectroscopy for real-time investigation of dynamic events in proteins on the time scale of seconds. *Journal of the American Chemical Society* **2005,** *127* (22), 8014-5.
109. Liebold, C.; List, F.; Kalbitzer, H. R.; Sterner, R.; Brunner, E., The interaction of ammonia and xenon with the imidazole glycerol phosphate synthase from Thermotoga maritima as detected by NMR spectroscopy. *Protein Science* **2010,** *19* (9), 1774-82.
110. Farmer, B. T.; Constantine, K. L.; Goldfarb, V.; Friedrichs, M. S.; Wittekind, M.; Yanchunas, J.; Robertson, J. G.; Mueller, L., Localizing the NADP+ binding site on the MurB enzyme by NMR. *Nature Structural Biology* **1996,** *3* (12), 995-7.
111. Wu, B.; Zhang, Z.; Noberini, R.; Barile, E.; Giulianotti, M.; Pinilla, C.; Houghten, R. A.; Pasquale, E. B.; Pellecchia, M., HTS by NMR of combinatorial libraries: a fragment-based approach to ligand discovery. *Chemical Biology* **2013,** *20* (1), 19-33.

112. Fielding, L., NMR methods for the determination of protein-ligand dissociation constants. *Current Topics in Medicinal Chemistry* **2003,** *3* (1), 39-53.
113. Fleishman, S. J.; Leaver-Fay, A.; Corn, J. E.; Strauch, E. M.; Khare, S. D.; Koga, N.; Ashworth, J.; Murphy, P.; Richter, F.; Lemmon, G.; Meiler, J.; Baker, D., RosettaScripts: a scripting language interface to the Rosetta macromolecular modeling suite. *PLoS One* **2011,** *6* (6), e20161.
114. Morin, A.; Meiler, J.; Mizoue, L. S., Computational design of protein-ligand interfaces: potential in therapeutic development. *Trends in Biotechnology* **2011,** *29* (4), 159-66.
115. Jäckel, C.; Kast, P.; Hilvert, D., Protein design by directed evolution. *Annual Review of Biophysics* **2008,** *37*, 153-73.
116. Leaver-Fay, A.; Tyka, M.; Lewis, S. M.; Lange, O. F.; Thompson, J.; Jacak, R.; Kaufman, K.; Renfrew, P. D.; Smith, C. A.; Sheffler, W.; Davis, I. W.; Cooper, S.; Treuille, A.; Mandell, D. J.; Richter, F.; Ban, Y. E.; Fleishman, S. J.; Corn, J. E.; Kim, D. E.; Lyskov, S.; Berrondo, M.; Mentzer, S.; Popović, Z.; Havranek, J. J.; Karanicolas, J.; Das, R.; Meiler, J.; Kortemme, T.; Gray, J. J.; Kuhlman, B.; Baker, D.; Bradley, P., ROSETTA3: an object-oriented software suite for the simulation and design of macromolecules. *Methods in Enzymology* **2011,** *487*, 545-74.
117. Koga, N.; Tatsumi-Koga, R.; Liu, G.; Xiao, R.; Acton, T. B.; Montelione, G. T.; Baker, D., Principles for designing ideal protein structures. *Nature* **2012,** *491* (7423), 222-7.
118. Sammond, D. W.; Bosch, D. E.; Butterfoss, G. L.; Purbeck, C.; Machius, M.; Siderovski, D. P.; Kuhlman, B., Computational design of the sequence and structure of a protein-binding peptide. *Journal of the American Chemical Society* **2011,** *133* (12), 4190-2.
119. O'Boyle, N. M.; Banck, M.; James, C. A.; Morley, C.; Vandermeersch, T.; Hutchison, G. R., Open Babel: An open chemical toolbox. *Journal of Cheminformatics* **2011,** *3*, 33.
120. Kothiwale, S.; Mendenhall, J. L.; Meiler, J., BCL::Conf: small molecule conformational sampling using a knowledge based rotamer library. *Journal of Cheminformatics* **2015,** *7*, 47.
121. Allen, F. H., The Cambridge Structural Database: a quarter of a million crystal structures and rising. *Acta Crystallographica Section B* **2002,** *58* (Pt 3 Pt 1), 380-8.
122. Hawkins, P. C.; Skillman, A. G.; Warren, G. L.; Ellingson, B. A.; Stahl, M. T., Conformer generation with OMEGA: algorithm and validation using high quality structures from the Protein Databank and Cambridge Structural Database. *Journal of Chemical Information and Modeling* **2010,** *50* (4), 572-84.
123. Labute, P., LowModeMD--implicit low-mode velocity filtering applied to conformational search of macrocycles and protein loops. *Journal of Chemical Information and Modeling* **2010,** *50* (5), 792-800.
124. Ebejer, J. P.; Morris, G. M.; Deane, C. M., Freely available conformer generation methods: how good are they? *Journal of Chemical Information and Modeling* **2012,** *52* (5), 1146-58.
125. Nivón, L. G.; Moretti, R.; Baker, D., A Pareto-optimal refinement method for protein design scaffolds. *PLoS One* **2013,** *8* (4), e59004.
126. Pettersen, E. F.; Goddard, T. D.; Huang, C. C.; Couch, G. S.; Greenblatt, D. M.; Meng, E. C.; Ferrin, T. E., UCSF Chimera--a visualization system for exploratory research and analysis. *Journal of Computational Chemistry* **2004,** *25* (13), 1605-12.
127. Sheffler, W.; Baker, D., RosettaHoles: rapid assessment of protein core packing for structure prediction, refinement, design, and validation. *Protein Science* **2009,** *18* (1), 229-39.

128. Lawrence, M. C.; Colman, P. M., Shape complementarity at protein/protein interfaces. *Journal of Molecular Biology* **1993,** *234* (4), 946-50.
129. Stranges, P. B.; Kuhlman, B., A comparison of successful and failed protein interface designs highlights the challenges of designing buried hydrogen bonds. *Protein Science* **2013,** *22* (1), 74-82.
130. Nivón, L. G.; Bjelic, S.; King, C.; Baker, D., Automating human intuition for protein design. *Proteins* **2014,** *82* (5), 858-66.
131. Song, Y.; DiMaio, F.; Wang, R. Y.; Kim, D.; Miles, C.; Brunette, T.; Thompson, J.; Baker, D., High-resolution comparative modeling with RosettaCM. *Structure* **2013,** *21* (10), 1735-42.
132. Henrich, S.; Salo-Ahen, O. M.; Huang, B.; Rippmann, F. F.; Cruciani, G.; Wade, R. C., Computational approaches to identifying and characterizing protein binding sites for ligand design. *Journal of Molecular Recognition* **2010,** *23* (2), 209-19.
133. Smith, R. D.; Damm-Ganamet, K. L.; Dunbar, J. B.; Ahmed, A.; Chinnaswamy, K.; Delproposto, J. E.; Kubish, G. M.; Tinberg, C. E.; Khare, S. D.; Dou, J.; Doyle, L.; Stuckey, J. A.; Baker, D.; Carlson, H. A., CSAR Benchmark Exercise 2013: Evaluation of Results from a Combined Computational Protein Design, Docking, and Scoring/Ranking Challenge. *Journal of Chemical Information and Modeling* **2015**.
134. Dunbar, J. B.; Smith, R. D.; Damm-Ganamet, K. L.; Ahmed, A.; Esposito, E. X.; Delproposto, J.; Chinnaswamy, K.; Kang, Y. N.; Kubish, G.; Gestwicki, J. E.; Stuckey, J. A.; Carlson, H. A., CSAR data set release 2012: ligands, affinities, complexes, and docking decoys. *Journal of Chemical Information and Modeling* **2013,** *53* (8), 1842-52.
135. Hennig, M.; Darimont, B. D.; Jansonius, J. N.; Kirschner, K., The catalytic mechanism of indole-3-glycerol phosphate synthase: crystal structures of complexes of the enzyme from Sulfolobus solfataricus with substrate analogue, substrate, and product. *Journal of Molecular Biology* **2002,** *319* (3), 757-66.
136. Henn-Sax, M.; Thoma, R.; Schmidt, S.; Hennig, M.; Kirschner, K.; Sterner, R., Two (betaalpha)(8)-barrel enzymes of histidine and tryptophan biosynthesis have similar reaction mechanisms and common strategies for protecting their labile substrates. *Biochemistry* **2002,** *41* (40), 12032-42.