Computational design of PDZ domains: parameterization and performance of a simple model

David Mignon[1], Nicolas Panel[1], Xingyu Chen and Thomas Simonson*

†Laboratoire de Biochimie (UMR CNRS 7654), Ecole Polytechnique, Palaiseau, France

[1]Joint first authors. *Corresponding author: thomas.simonson@polytechnique.fr

Abstract

Short title: Computational design of PDZ domains

Keywords:

1 Introduction

PDZ domains are small, globular protein domains that help establish protein-protein interaction networks in the cell [**?**]. To do this, they form specific interactions with other, target proteins, usually by recognizing a few amino acids at the C-terminus of the targets. Due to their biological importance, PDZ domains and their ligands have been extensively studied and engineered. Thus, peptide ligands have been identified or developed that modulate the activity of PDZ domains involved in pathologies like cancer, rabies, and cystic fibrosis [**?**]. Engineered PDZ domains have also been used to elucidate principles of protein folding and evolution [**?**], and to test the engineering methods themselves.

One emerging method that has been applied to several PDZ domains is computational protein design (CPD) [1–7]. Starting from a 3D structural model, CPD explores a large space of possible sequences and conformations, to identify protein variants that have certain predefined properties, such as stability or ligand binding. Conformational space is usually defined by a library of sidechain rotamers, which can be discrete or continuous, and by a finite set of backbone conformations or a specific repertoire of allowed backbone deformations. The energy function usually combines physical and empirical terms [8–10]. Both solvent and the unfolded protein state are described implicitly.

Here, we consider a simple but important class of CPD variants; we optimize selected parameters of the energy function for PDZ proteins, we test the quality of the model, and we use it for two applications. Our CPD variant is implemented in the Proteus software [11–13]. It uses an "MMGBSA" energy function, which combines a molecular mechanics protein energy function with a Generalized Born + Surface Area implicit solvent treatment. The folded protein structure is represented by a single, fixed, backbone conformation and a discrete sidechain rotamer library. The unfolded state energy depends only on sequence composition, not an explicit structural model. With this CPD variant, the main adjustable model parameters are the protein dielectric constant ϵ, a small set of atomic surface energy coefficients σ_i, and a collection of amino acid chemical potentials, or "reference energies" E_t^r that each represent the contribution of a single amino acid of type t to the unfolded state energy.

We specifically optimize the reference energies E_t^r, using a maximum likelihood formalism and a set of eight PDZ test proteins. For two of the proteins, Tiam1 and Cask, we compare two sets of surface coefficients and two values of the dielectric constant. The

resulting parameter sets are tested by generating designed sequences for all eight proteins and comparing them to natural sequences, as well as sequences generated with the Rosetta energy function and software [**?**]. We also perform 100-400 nanosecond molecular dynamics simulations for a few designed sequences, to help assess their stability.

We then apply one of the optimized models to the Tiam1 protein and two problems. First, we do a series of Monte Carlo simulations of Tiam1 where the chemical potential of the hydrophobic amino acid types is gradually increased, artificially biasing the protein composition. As a result of the bias, hydrophobic amino acids gradually invade the protein interior, forming a hydrophobic core that is initially smaller, then becomes larger than the natural one. The propensity of each core position to become hydrophobic at a high or low level of bias can be seen as a structure-dependent hydrophobicity index, providing information on the designability of the protein core. The second application consists in designing four Tiam1 positions that are known to be involved in specific target recognition, and that have been experimentally mutated so as to modify the preferred Tiam1 target [14], switching its preference from the natural target peptide syndecan-1 (Sdc1) to another peptide, Caspr4. We mutate these positions through Monte Carlo simulations of either the apo-protein or the protein in complex with either the natural peptide ligand, Sdc1, or the ligand preferred by the quadruple mutant, Caspr4.

2 The unfolded state model

2.1 Maximum likelihood reference energies

We use Monte Carlo to generate a Markov chain of states [15, 16], such that the states are populated according to a Boltzmann distribution. One possible elementary move is a "mutation": we modify the sidechain type $t \to t'$ at a chosen position i in the folded protein, assigning a particular rotamer r' to the new sidechain. At the same time, we perform the reverse mutation in the unfolded protein, $t' \to t$. The corresponding energy change has the form:

$$\Delta E = \Delta E^f - \Delta E^u = \left(E^f(...t'_i, r'_i...) - E^f(...t_i, r_i...)\right) - \left(E^u(t'_i) - E^u(t_i)\right) \qquad (1)$$

ΔE measures the stability change due to the mutation. For a particular sequence S, the unfolded state energy has the form:

$$E^u_S = \sum_{i \in S} E^r(t_i). \qquad (2)$$

The type-dependent quantities $E^r(t) \equiv E^r_t$ are essential parameters in the simulation model, referred to as "reference energies". Our goal here is to choose them empirically so that the simulation produces amino acid frequencies that match a set of target values, for example experimental values in the Pfam database. Specifically, we will choose them so as to maximize the probability, or likelihood of the target sequences.

Let S be a particular sequence. Its Boltzmann probability is

$$p(S) = \frac{1}{Z} \exp(-\beta \Delta G_S), \tag{3}$$

where $\Delta G_S = G^f_S - E^u_S$ is the folding free energy of S, G^f_S is the free energy of the folded form, and Z is a normalizing constant (the partition function). We then have

$$kT \ln p(S) = \sum_{i \in S} E^r(t_i) - G^f_S - kT \ln Z = \sum_{t \in \text{aa}} n_S(t) E^r_t - G^f_S - kT \ln Z, \tag{4}$$

where the sum on the right is over the amino acid types and $n_S(t)$ is the number of amino acids of type t within the sequence S.

We now consider a set $\mathcal{S}$ of N target sequences S; we denote $\mathcal{L}$ the probability of the entire set, which depends on the model parameters E^r_t; we refer to $\mathcal{L}$ as their likelihood [17]. We have

$$kT \ln \mathcal{L} = \sum_S \sum_{t \in \text{aa}} n_S(t) E^r_t - \sum_S G^f_S - NkT \ln Z = \sum_{t \in \text{aa}} N(t) E^r_t - \sum_S G^f_S - NkT \ln Z, \tag{5}$$

where $N(t)$ is the number of amino acids of type t in the whole dataset $\mathcal{S}$. The normalization factor or partition function Z is a sum over all possible sequences R:

$$Z = \sum_R \exp(-\beta \Delta G_R) = \sum_R \exp(-\beta \Delta G^f_R) \Pi_{t \in aa} \exp(\beta n_R(t) E^r_t) \tag{6}$$

In view of maximizing $\mathcal{L}$, we consider the derivative of Z with respect to one of the E^r_t:

$$\frac{\partial Z}{\partial E^r_t} = \sum_R \beta n_R(t) \exp(-\beta \Delta G^f_R) \prod_{s \in aa} \exp(\beta n_R(s) E^r_s) \tag{7}$$

We then have

$$\frac{kT}{Z} \frac{\partial Z}{\partial E^r_t} = \frac{\sum_R n_R(t) \exp(-\beta \Delta G_R)}{\sum_R \exp(-\beta \Delta G_R)} = \langle n(t) \rangle. \tag{8}$$

The quantity on the right is the Boltzmann average of the number $n(t)$ of amino acids t over all possible sequences. In practice, this is the average population of t we would

obtain in a long MC simulation. We note that, as usual in statistical mechanics [18], the derivative of $\ln Z$ with respect to one quantity (E_t^r) is equal to the ensemble average of the conjugate quantity ($\beta n_S(t)$).

A necessary condition to maximize $\ln \mathcal{L}$ is that its derivatives with respect to the E_t^r should all be zero. We see that

$$\frac{1}{N}\frac{\partial}{\partial E_t^r} \ln \mathcal{L} = \frac{1}{N} \sum_S n_S(t) - \langle n(t) \rangle = \frac{N(t)}{N} - \langle n(t) \rangle \tag{9}$$

so that

$$\mathcal{L} \text{ maximum} \Longrightarrow \frac{N(t)}{N} = \langle n(t) \rangle, \ \ \forall t \in \text{aa} \tag{10}$$

Thus, to maximize $\mathcal{L}$, we should choose $\{E_t^r\}$ such that a long simulation gives the same amino acid frequencies as the target database.

2.2 Searching for the maximum likelihood

To approach the maximum likelihood $\{E_t^r\}$ values, starting from a current guess $\{E_t^r(n)\}$, we will compare three methods. With the first method, we step along the gradient of $\ln \mathcal{L}$, using the update rule [17]:

$$E_t^r(n+1) = E_t^r(n) + \alpha \frac{\partial}{\partial E_t^r} \ln \mathcal{L} = E_t^r(n) + \delta E \left(n_t^{\text{exp}} - \langle n(t) \rangle_n \right) \tag{11}$$

Here, α is a constant; $n_t^{\text{exp}} = N(t)/N$ is the mean population of amino acid type t in the target database; $\langle \rangle_n$ indicates an average over a simulation done using the current reference energies $\{E_t^r(n)\}$, and δE is an empirical constant with the dimension of an energy, referred to as the update amplitude. This update procedure is repeated until convergence. We refer to this method as the linear update method.

The second method, used previously [11, 12], employs a logarithmic update rule:

$$E_t^r(n+1) = E_t^r(n) + kT \ln \frac{\langle n(t) \rangle_n}{n_t^{\text{exp}}} \tag{12}$$

where kT is a thermal energy, set empirically to 0.5 kcal/mol. We refer to this as the logarithmic update method. Both the linear and logarithmic update methods converge to the same optimum, specified by (Eq. 10).

In the later iterations, some E_t^r values tended to converge slowly, with an oscillatory behavior. Therefore, we sometimes used a modified update rule, where the $E_t^r(n+1) -$

$E_t^r(n)$ value computed with the linear or logarithmic method for iteration n was mixed with the value computed at the previous iteration, with the $(n-1)$ value having a weight of 1/3 and the current value a weight of 2/3.

3 Computational methods

3.1 Effective energy function for the folded state

The energy matrix was computed with the following effective energy function for the folded state:

$$E = E_{\text{bonds}} + E_{\text{angles}} + E_{\text{dihe}} + E_{\text{impr}} + E_{\text{vdw}} + E_{\text{Coul}} + E_{\text{solv}} \tag{13}$$

The first six terms in (13) represent the protein internal energy. They were taken from the Amber ff99SB empirical energy function [19], slightly modified for CPD (see below). The last term on the right, E_{solv}, represents the contribution of solvent. We used a "Generalized Born + Surface Area", or GBSA implicit solvent model [20]:

$$E_{\text{solv}} = E_{\text{GB}} + E_{\text{surf}} = \frac{1}{2}\left(\frac{1}{\epsilon_W} - \frac{1}{\epsilon_P}\right) \sum_{ij} q_i q_j \left(r_{ij}^2 + b_i b_j \exp[-r_{ij}^2/4b_i b_j]\right)^{-1/2} + \sum_i \sigma_i A_i \tag{14}$$

Here, ϵ_W, ϵ_P are the solvent and protein dielectric constants; r_{ij} is the distance between atoms i, j and b_i is the "solvation radius" of atom i [20, 21]. A_i is the exposed solvent accessible surface area of atom i; σ_i is a parameter that reflects each atom's preference to be exposed or hidden from solvent. The solute atoms were divided into 4 groups with specific σ_i values (see below): unpolar, aromatic, polar, and ionic. Hydrogen atoms were assigned a surface coefficient of 0. Surface areas were computed by the Lee and Richards algorithm [22], implemented in the XPLOR program [23], using a 1.5 Å probe radius. Most of the MC simulations used a protein dielectric of $\epsilon_P = 4$ or 8 (see Results).

In the GB energy term, the atomic solvation radius b_i approximates the distance from i to the protein surface and is a function of the coordinates of all the protein atoms. The particular b_i form corresponds to a GB variant we call GB/HCT, after its original authors [20], with model parameters optimized for use with the Amber force field [21]. Since b_i depends on the coordinates of all the solute atoms [20], an additional approximation is needed to make the GB energy term pairwise additive and define the energy matrix. We use a "Native Environment Approximation", or NEA, where the solvation radius b_i of

each particular group (backbone, sidechain or ligand) is computed ahead of time, with the rest of the system having its native sequence and conformation [12, 24].

The surface energy contribution E_{surf} is not pairwise additive either, because in a protein structure, surface area buried by one sidechain may also be buried by another. To make this energy pairwise, Street et al proposed a simple procedure [25]. The buried surface of a sidechain is computed by summing over the neighboring sidechain and backbone groups. For each neighboring group, the contact area with the sidechain of interest is computed, independently of other surrounding groups. The contact areas are then summed. To avoid overcounting of buried surface area, a scaling factor is applied to the contact areas involving buried sidechains. Previous work showed that a scaling factor of 0.65 works well [21, 24].

The Amber force field ff99SB is slightly modified for CPD, with the original backbone charges replaced by a unified set, obtained by averaging over all amino acid types and adjusting slightly to make the backbone portion of each amino acid neutral [26].

3.2 Reference energies in the unfolded state

In the unfolded state, the energy depends on the sequence composition through a set of reference energies E_t^r (Eq. 2). The values are assigned based on amino acid types t, taking into account also the position of each amino acid in the folded structure, through its buried or solvent-exposed character. Thus, for a given type (Ala, say), there are two distinct E_t^r values: a buried and an exposed value. This is done even though the reference energies are used to represent the unfolded, not the folded state. There are three rationales for this. First, we assume residual structure is present in the unfolded state, so that amino acids partly retain their buried/exposed character. Second, we hypothesize that the unfolded state model compensates in a systematic way for errors in the folded state energy function, so that the folded structure matters. Third, this strategy makes the model less sensitive to variations in the length of surface loops, and to the proportion of surface vs. buried residues, which can vary widely among homologs (see below); as a result, the model should be more transferable within a protein family.

Distinguishing buried/exposed positions doubles the number of adjustable E_t^r parameters. Conversely, to reduce the number of adjustable parameters, we group amino acids into homologous classes (given in Results). Within each class c, and for each type

of position (buried or exposed), the reference energies have the form

$$E_t^r = E_c^r + \delta E_t^r \tag{15}$$

Here, E_c^r is an adjustable parameter while δE_t^r is a constant, computed as the molecular mechanics energy difference between amino acid types within the class c, assuming an unfolded conformation where each amino acid interacts only with itself and with solvent. During likelihood maximization, E_c^r is optimized while δE_t^r is held fixed. Numerical values are listed further on.

3.3 Experimental sequences

We considered a set of eight PDZ domains, whose PDB codes are listed in Table 1. To define the target amino acid frequencies for likelihood maximization, we collected homologous sequences for each of the eight. We started by a Blast search of Uniprot with the PDB sequence as the query and the Blosum62 matrix, and retained homologs with a sequence identity, relative to the query, above a certain threshold, around 60–80% depending on the test protein. Within the retained homologs, sequences that were over 95% or 85% identical were pruned, keeping just one of the redundant variants. This led to about 40–120 homologs per test protein; see details inTable 1. For each set, amino acid frequencies were computed, distinguishing buried and exposed positions. Buried positions were defined to have a solvent-accessible surface area below 30% of that obtained for the amino acid alone (over 70% burial), which led to similar numbers of buried/exposed positions. The eight sets of mean frequencies were themselves averaged, giving the overall target amino acid frequencies (see below).

3.4 Structural models

Structures were prepared and energy matrices computed using procedures described previously [27, 28]. For Tiam1, two missing segments (residues 851-854 and 868-869) were modelled using the Modeller program [**?**]. In the energy matrix calculations, for each residue pair, interaction energies were computed after 15 steps of energy minimization, with the backbone fixed and only the interactions of the pair with each other and the backbone included. This alleviates the discrete rotamer approximation. Sidechain rotamers were described by a slightly expanded version of the library of Tuffery et al [29], which has a total of 228 rotamers (sum over all amino acid types). The expansion consists

in allowing additional hydrogen orientations for OH and SH groups [24]. This rotamer library was chosen for its simplicity and because it gives very good performance in sidechain placement tests, comparable to the specialized Scwrl4 program (which uses a much larger library) [30, 31].

3.5 Monte Carlo simulations

The Monte Carlo simulations use one- and two-position moves, where either rotamers, types, or both are changed. For two-position moves, the second position is selected among those that have a significant (unsigned) interaction energy with the first one, meaning that there is at least one rotamer conformation where their interaction is 10 kcal/mol or more. In addition, we mostly perform Replica Exchange Monte Carlo (REMC), where several simulations ("replicas" or "walkers") are propagated in parallel, at different temperatures; periodic swaps are attempted between the conformations of two walkers i, j (adjacent in temperature). The swap is accepted with probability

$$acc(\text{swap}_{ij}) = \text{Min}\left[1, e^{(\beta_i - \beta_j)(\Delta E_i - \Delta E_j)}\right] \tag{16}$$

where β_i, β_j are the inverse temperatures of the two walkers and ΔE_i, ΔE_j are the changes in their folding energies due to the conformation change [32, 33]. We use eight walkers, with thermal energies kT_i that range from 0.125 to 2 kcal/mol, and are spaced in a geometric progression: T_{i+1}/T_i = constant [32]. Simulations are done with the proteus program [12]; REMC uses an efficient, shared-memory, OpenMP parallelization [34].

3.6 Rosetta sequence generation

Monte Carlo simulations were also done using the Rosetta program and energy function [**?**], for comparison. The simulations were done using version 2015.38.58158 of Rosetta (freely available online), using the command

```
fixbb -s Tiam1.pdb -resfile Tiam1.res -nstruct 10000 -ex1 -ex2 -linmem_ig 10
```

Simulations were run for each protein until 10000 low energy sequences were identified, corresponding to run times of about 5 minutes per sequence on a single core of a recent Intel processor, for a total of 10 hours (per protein) using 80 cores. This is comparable to the cost of the Proteus calculations (energy matrix plus Monte Carlo).

3.7 Sequence characterization

Designed sequences were compared to the Pfam alignment for the PDZ family, using the Blosum40 scoring matrix and a gap penalty of -6. Each Pfam sequence was also compared to its own Pfam alignment. For these Pfam/Pfam comparisons, if a test protein T was part of the Pfam alignment, the T/T self comparison was left out, to be more consistent with the designed/Pfam comparisons. The Pfam alignment was the "RP55" alignment, with 12255 sequences. Similarities were computed for 14 core residues and 16 surface residues, defined by their near-complete burial or exposure (listed in Results) and for the entire protein.

Designed sequences were submitted to the Superfamily library of Hidden Markov Models [35, 36], which attempts to classify sequences according to the SCOP classification [37]. Classification was based on SCOP version 1.75 and version 3.5 of the Superfamily tools. Superfamily executes the hmmscan program, which implements a Hidden Markov model for each SCOP family and superfamily; here hmmscan was executed with an E-value threshold of 10^{-10}, using a total of 15438 models to represent the SCOP database.

3.8 Molecular dynamics simulations

For wildtype Tiam1 and a few designed sequences, to help assess the stability of the designed proteins, we ran MD simulations with explicit solvent. Starting structures were taken from the MC trajectory or the crystal structure (wildtype protein) and slightly minimized through xxx steps of conjugate gradient minimization. The protein was immersed in a large box of water; waters overlapping protein were eliminated, and the solvated system was truncated to the shape of a truncated octahedral box using the Charmm graphical interface or GUI [**?**]; final models included about 10000 water molecules. A few sodium or chloride ions were included to ensure overall electroneutrality. Protonation states of histidines were assigned to be neutral, based on visual inspection. MD was done at room temperature and pressure, using a Nose-Hoover thermostat and barostat. Long-range electrostatic interactions were treated with a Particle Mesh Ewald approach [38]. The Amber ff99SB forcefield was used for the protein; the TIP3P model [39] was used for water. Simulations were run for 100–400 nanoseconds, depending on the sequence, using the Charmm and NAMD programs [40, 41].

4 Results

4.1 Experimental structures and sequences

The 3D structures of four of our PDZ domains are shown in Fig. 1A. While 14 core residues (identified by their C_β atoms, shown as spheres) superimpose well between structures, there are large deviations in loops and chain termini, and the Tiam1 α_2 helix is rotated slightly outwards compared to the other three structures. Fig. 1B illustrates the similarity between all 8 domains, measured by the rms deviation between structurally-aligned C_α atoms. Structure pairs with rms deviations of 1 Å or less and 60 aligned residues or more are linked; 2BYG forms the center of the group, with five links; Cask is not quite linked to 1IHJ (rms deviation of 1.3 Å over 63 residues) and Tiam1 is isolated. Sequence identities are also shown, including those between Tiam1 and its closest homologs. Overall, six proteins form a tight group, with deviations of 1 Å or less, while Cask and Tiam1 are further away. The Tiam1/Cask sequence identity is 33%; their structural deviation is 1.7 Å based on a 42-residue alignment.

Sequence conservation within our eight proteins and the Pfam seed alignment is shown in Fig. 2. The 14 positions we use to define the hydrophobic core are highly, though not totally conserved within the Pfam seed alignment. Arg, Lys and Gln appear at some of the positions, since in a small PDZ protein, the long hydrophobic portion of these sidechains can be buried in the core while still allowing the polar tip of the sidechain to be exposed to solvent. Small amounts of Asp, and Glu also appear, in places where the sequence alignment may not reflect closely the 3D sidechain superposition (or lack thereof).

4.2 Optimizing the unfolded state model

We optimized the reference energies E_t^r for six of the eight proteins, with the other two (Tiam1, Cask) left for cross validation. The protein dielectric constant was $\epsilon_P = 8$, and a first set of atomic surface coefficients was used. We refer to the resulting model either as model A or as the (ϵ_P=8, S1, n=6) model (S1 for "set 1"; n=6 being the number of proteins used to define the target amino acid frequencies). We repeated the optimization for Tiam1 and Cask alone, to see what improvement is obtained, if any, when a smaller set of proteins is specifically optimized. The resulting model is called A' or (ϵ_P=8, S1, n=2). Finally, we repeated the E_t^r optimization using a second set of surface coefficients

and an ϵ_P of either 8 or 4; these calculations were done for Tiam1 and Cask alone, due to the cost of repeated optimizations with multiple proteins. The corresponding models are called B (ϵ_P=8, S2, n=2) and B' (ϵ_P=4, S2, n=2). Model names are listed in Table 2. The optimizations all converged to within 0.05 kcal/mol after about 20 iterations for most amino acid types, and within 0.1 kcal/mol for the others (the weakly-populated types), using either the linear or the logarithmic method. Table 3 indicates the final reference energies for models A, B. The E_t^r values are compared to, and agree qualitatively with the energies computed from an extended peptide structure, which provides a less empirical model of the unfolded state. Table 4 compares the amino acid frequencies from experiment and the simulations. We see that while the theoretical populations of the different amino acid classes agree well with experiment, the agreement for some of the amino acid types is not perfect.

4.3 Assessing designed sequence quality

Family recognition tests Proteus simulations used Replica Exchange Monte Carlo (REMC) with eight replicas at temperatures between 0.236 and 3 kcal/mol; the 10000 lowest energies among those sampled by any of the the replicas were retained for analysis, along with the 10000 Rosetta sequences. These sequences were submitted to the Superfamily fold recognition tool. Results are given in Table 5. For 6 of 8 proteins, all 10000 Rosetta sequences are assigned by Superfamily to the correct superfamily and family, with E-values between 0.5 10^{-3} and 4 10^{-3} for the family assignments. For two, the family success rates are 90/98%, with slightly higher E-values. With model A = (ϵ=8,S1,n=6), the Proteus results are also good, with 6 out of 8 proteins giving 100% correct family assignments, with E-values between 2 10^{-3} and 15 10^{-3} for the family assignments. For the 1R6J and Tiam1 proteins, only 13% and 4 %, respectively, of the top sequences were assigned to the correct family. For these two proteins, for the Proteus sequences, Superfamily only recognizes about half of the sequence length. For the other 6 proteins, the Superfamily match lengths are similar to those seen with the Rosetta sequences. Notice that Tiam1 was not part of the reference energy optimization set (nor was Cask). Specifically optimizing the reference energies for Tiam1 and Cask (model A') did not improve the Superfamily performance, with just 0.1% of correct family assignments for Tiam1 and 63% for Cask (vs. 4% and 100% with model A; Table 5).

Switching to a second set of surface coefficients (model B) gave significantly improved Proteus results for Tiam1 and Cask, even though these values (set 2) were optimized earlier

using a *different* solvent model (Coulombic instead of Generalized Born electrostatics) [42]. Using these coefficients and E_t^r values optimized for Tiam1 and Cask (model B or ϵ=8,S2,n=2), we obtained a high percentage of sequences assigned to the correct family: 91% for Tiam1 and 100% for Cask, slightly better than Rosetta. Changing the protein dielectric constant to ϵ_P=4 (model B') gave results for Tiam1 that were somewhat poorer than model B but still much better than model A.

Sequences and sequence diversity Tiam1 and Cask sequences predicted by Proteus, using model B, and by Rosetta are shown as sequence logos, both for the 14 core residues (Fig. 3) and 16 surface residues (Fig. 4), and compared to natural sequences. Agreement with experiment for the core residues is very good, while agreement for the surface residues is much poorer, as seen in many previous CPD studies. The behavior of the surface positions is also illustrated by designing each position individually, with the rest of the protein free to explore rotamers but not mutations (single-position design). The corresponding logo shows an excess of Arg and Lys residues, suggesting that their chemical potentials, or reference energies are not yet optimal, despite extensive optimization. Sequence similarity scores are given in the next subsection.

The diversity of the natural and designed sequences can be characterized by a mean, exponentiated sequence entropy (see Methods), which corresponds to a mean number of sampled sequence classes per position. The large, RP55 set of experimental sequences has a mean entropy of 3.4 (exponentiated, per-position). Pooling the computed Tiam1 and Cask sequences gives an entropy of 2.2 with Rosetta and 2.2 with Proteus and model A, or 2.0 with model B, indicating that these two backbone geometries cannot support as much diversity as the much larger RP55 set. Taking the 10000 lowest energy sequences sampled with the *room temperature* Monte Carlo replica (instead of the 10000 lowest energies sampled by all replicas) and pooling Tiam1 and Cask as before gives a higher overall entropy of 3.0/2.9 with models A/B. Entropy in the core is only slightly below average (2.1 and 2.0) with Rosetta and model A, but only 1.25 with model B, vs. 1.8 for Pfam-RP55.

Blosum similarity scores We also computed Blosum40 similarity scores between theoretical and experimental sequences, shown in Figs. 5 and 6. With model A = (ϵ=8,S1,n=6), for the 14 core residues, the scores for all but one protein overlap with the scores seen within the RP55 set of experimental sequences, with scores between 20

and 40. The 1R6J protein, which gave low Superfamily scores, does well in terms of Blosum40 scores. Only for 3K82 are the Proteus scores in the very low range of experimental scores. Rosetta does somewhat better for this protein. For the seven others, results for the Rosetta sequences are similar on average to Proteus, with some cases a bit better and others a bit worse. The Proteus sequences for Tiam1 and Cask score mostly above the Rosetta sequences, even though these proteins were not part of the E_r^t optimization set.

With model B = (ϵ=8,S2,n=2), the Cask results are similar to model A and the Tiam1 results slightly improved (whereas the Superfamily results were much better). With model B, we also computed surface and overall similarities; the overall values overlap with the bottom of the peak of experimental scores, and are very similar to the values for the Rosetta sequences. For the surface residues, similarity to the experimental sequences is low (scores below zero), both for Proteus and Rosetta. Model B' (not shown) performs about as well as model B.

For certain applications, we may need to specifically explore the sequence space region that is very similar to Pfam, beyond the similarity that is provided by our approximate energy function. This can be achieved by adding to the energy function an umbrella potential that explicitly favors high sequence scores. Fig. 6 includes results that use such a bias potential (see Methods): by construction, it leads to very high similarity scores, higher than the mean similarity between the Pfam sequences themselves.

We also analyzed the similarity between theoretical sequences generated with the different models. With model B, overall similarity scores between the Proteus sequences (model B vs. model B) are 453±22 for Tiam1 and 491±20 for Cask. Changing the dielectric constant to 4 (model B') gives sequences less similar to model B, with B-B' scores of 389±18 and 412±30 for Tiam1 and Cask, respectively. Changing the surface coefficients to set 1 (model A) changes the sequences more dramatically, with A-B scores of 173±11 and 150±14 for Tiam1 and Cask, respectively.

4.4 Stability of designed sequences in molecular dynamics simulations

A few designed Tiam1 sequences were chosen for testing in molecular dynamics simulations (with an explicit solvent environment). The sequences were produced using the best design models, B and B' (ϵ=8/4,S2,n=2). Among the 2500 top energy sequences, we chose ones that had a nonneutral isoelectric point, that were assigned to correct SCOP family by

Superfamily with good E-values, that had good Pfam similarty scores, and not too many mutations that drastically change the amino acid type compared to the wildtype proteins. This left us with 66 sequences from model B and 45 from model B'. In addition, we eliminated two mutations that created a buried cavity and several that led to net protein charges of +6 or more. Six sequences were chosen; four were modified further by hand to eliminate charged residues in the exposed loop 852–856 (lysines changed to alanine), for a total of ten sequences for MD. Simulations were run for 100 ns, then extended to 200 ns for the cases that did not exhibit instability within 100 ns (5 sequences). 3 of the 5 were stable after 200 ns; one was extended to 400 ns and remained stable. The native Tiam1 protein was also simulated for 400 ns and remained stable. While 200-400 ns simulation times are still short compared to experimental unfolding times (microseconds), they are long enough to challenge the structural models, in the context of a high quality protein force field and explicit solvent model. Some of the sequences rapidly drift away from the starting, designed structure, as measured by their rms deviation (Fig. 7), suggesting they may be unstable or weakly stable. Two move a few Ångstroms away from the starting, designed structure, stay put over 200 ns, suggesting a higher stability. Two sequences appear highly stable, with a structure that is very close to the initial, designed model and to the native Tiam1 structure. This includes sequence 6, which stays within 2 Å of the wildtype structure over 400 ns, without visible drift. Results are similar if the rms deviation is computed including 90% of the amino acids, omitting those that deviate most.

4.5 Application to Tiam1: growing the hydrophobic core

As an application of our parameterized models, we examined the designability of the Tiam1 hydrophobic core. We submitted the protein to Replica Exchange Monte Carlo simulations with a succession of slightly different energy functions, which increasingly favor hydrophobic residues. The first one includes a bias energy δ = xxx kcal/mol that penalizes hydrophobic amino acid types (ILMVACWFY). The last one includes a bias energy δ = -xxx kcal/mol that favors hydrophobic types by the same amount. Intermediate bias values δ = xxx, 0, and xxx kcal/mol were also simulated. Results are shown in Fig. 8. With the largest δ value, the Tiam1 hydrophobic core is depleted, with xxx positions changed to polar types; these positions mostly lie on the outer edge of the core. With the intermediate values, the hydrophobic core is native-like. With the most negative δ value, the hydrophobic core has expanded outwards into surface regions, with xxx (ini-

tially polar) positions changed to hydrophobic types. The observed propensities of each position to become polar or hydrophobic in the presence of a large or small penalty δ can be thought of as a hydrophobic designability measure.

4.6 Application to Tiam1: designing specificity positions

As a second, practical application, we have redesigned four positions in Tiam1 that are known to contribute to its specific peptide binding. These positions were engineered experimentally to alter Tiam1 specificity, leading to a quadruple mutant that preferentially binds the Caspr4 peptide instead of the natural, syndecan-1 (Sdc1) peptide [14]. The native/mutant types at these positions are xxx/xxx. We did REMC simulations where all four positions could mutate simultaneously, in the absence of a peptide ligand, and in the presence of either the Sdc1 or the Caspr4 peptide. The simulations lead to amino acid types at the four positions that are very similar to the experimental ones, either in the native or the mutant protein. Results are shown in Fig. 9, as sequence logos that show the types that are predominantly sampled by the various replicas in the REMC simulations. With no ligand and with the native, Sdc1 ligand, we mostly retrieve native types. With the Caspr4 ligand, we sample the experimental mutant types.

5 Discussion

References

[1] Dantas, G., Kuhlman, B., Callender, D., Wong, M., and Baker, D. A large test of computational protein design: Folding and stability of nine completely redesigned globular proteins. *J. Mol. Biol. 332* (2003), 449–460.

[2] Butterfoss, G. L., and Kuhlman, B. Computer-based design of novel protein structures. *Ann. Rev. Biophys. Biomolec. Struct. 35* (2006), 49–65.

[3] Lippow, S. M., and Tidor, B. Progress in computational protein design. *Curr. Opin. Biotech. 18* (2007), 305–311.

[4] Saven, J. G. Computational protein design: engineering molecular diversity, nonnatural enzymes, nonbiological cofactor complexes, and membrane proteins. *Curr. Opin. Chem. Biol. 15* (2011), 452–457.

[5] Feldmeier, K., and Hoecker, B. Computational protein design of ligand binding and catalysis. *Curr. Opin. Chem. Biol. 17* (2013), 929–933.

[6] 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., and Baker, D. Computational design of ligand-binding proteins with high affinity and selectivity. *Nature 501* (2013), 212–218.

[7] Nisonoff, P. G. H. M., and Donald, B. R. Algorithms for protein design. *Curr. Opin. Struct. Biol. 39* (2016), 16–26.

[8] Pokala, N., and Handel, T. Energy functions for protein design I: Efficient and accurate continuum electrostatics and solvation. *Prot. Sci. 13* (2004), 925–936.

[9] Samish, I., MacDermaid, C. M., Perez-Aguilar, J. M., and Saven, J. G. Theoretical and computational protein design. *Ann. Rev. Phys. Chem. 62* (2011), 129—149.

[10] Li, Z., Yang, Y., Zhan, J., Dai, L., and Zhou, Y. Energy functions in de novo protein design: Current challenges and future prospects. *Ann. Rev. Biochem 42* (2013), 315–335.

[11] Schmidt am Busch, M., Lopes, A., Mignon, D., and Simonson, T. Computational protein design: software implementation, parameter optimization, and performance of a simple model. *J. Comput. Chem. 29* (2008), 1092–1102.

[12] Simonson, T. Protein:ligand recognition: simple models for electrostatic effects. *Curr. Pharma. Design 19* (2013), 4241–4256.

[13] Polydorides, S., Michael, E., Mignon, D., Druart, K., Archontis, G., and Simonson, T. Proteus and the design of ligand binding sites. In *Methods in Molecular Biology: Design and Creation of Protein Ligand Binding Proteins*, B. Stoddard, Ed., vol. 1414. Springer Verlag, New York, 2016, p. 0000.

[14] Shepherd, T. R., Hard, R. L., Murray, A. M., Pei, D., and Fuentes, E. J. Distinct ligand specificity of the Tiam1 and Tiam2 PDZ domains. *Biochemistry 50* (2011), 1296–1308.

[15] Frenkel, D., and Smit, B. *Understanding molecular simulation, Chapter 3.* Academic Press, New York, 1996.

[16] Grimmett, G. R., and Stirzaker, D. R. *Probability and random processes.* Oxford University Press, 2001.

[17] Kleinman, C. L., Rodrigue, N., Bonnard, C., Philippe, H., and Lartillot, N. A maximum likelihood framework for protein design. *BMC Bioinf. 7* (2006), Art. 326.

[18] Fowler, R. H., and Guggenheim, E. A. *Statistical Thermodynamics.* Cambridge University Press, 1939.

[19] Cornell, W., Cieplak, P., Bayly, C., Gould, I., Merz, K., Ferguson, D., Spellmeyer, D., Fox, T., Caldwell, J., and Kollman, P. A second generation force field for the simulation of proteins, nucleic acids, and organic molecules. *J. Am. Chem. Soc. 117* (1995), 5179–5197.

[20] Hawkins, G. D., Cramer, C., and Truhlar, D. Pairwise descreening of solute charges from a dielectric medium. *Chem. Phys. Lett. 246* (1995), 122–129.

[21] Lopes, A., Aleksandrov, A., Bathelt, C., Archontis, G., and Simonson, T. Computational sidechain placement and protein mutagenesis with implicit solvent models. *Proteins 67* (2007), 853–867.

[22] Lee, B., and Richards, F. The interpretation of protein structures: estimation of static accessibility. *J. Mol. Biol. 55* (1971), 379–400.

[23] Brünger, A. T. X-plor *version 3.1, A System for X-ray crystallography and NMR.* Yale University Press, New Haven, 1992.

[24] Gaillard, T., and Simonson, T. Pairwise decomposition of an MMGBSA energy function for computational protein design. *J. Comput. Chem. 35* (2014), 1371–1387.

[25] Street, A. G., and Mayo, S. Pairwise calculation of protein solvent-accessible surface areas. *Folding and Design 3* (1998), 253–258.

[26] Aleksandrov, A., Polydorides, S., Archontis, G., and Simonson, T. Predicting the acid/base behavior of proteins: A constant-pH Monte Carlo approach with Generalized Born solvent. *J. Phys. Chem. B 114* (2010), 10634–10648.

[27] Schmidt am Busch, M., Mignon, D., and Simonson, T. Computational protein design as a tool for fold recognition. *Proteins 77* (2009), 139–158.

[28] Schmidt am Busch, M., Sedano, A., and Simonson, T. Computational protein design: validation and possible relevance as a tool for homology searching and fold recognition. *PLoS One 5(5)* (2010), e10410.

[29] Tuffery, P., Etchebest, C., Hazout, S., and Lavery, R. A new approach to the rapid determination of protein side chain conformations. *J. Biomol. Struct. Dyn. 8* (1991), 1267.

[30] Krivov, G. G., Shapalov, M. V., and Dunbrack, R. L. Improved prediction of protein side-chain conformations with SCWRL4. *Proteins 77* (2009), 778–795.

[31] Gaillard, T., Panel, N., and Simonson, T. Protein sidechain conformation predictions with an MMGBSA energy function. *Proteins 84* (2016), 803–819.

[32] Kofke, D. A. On the acceptance probability of replica-exchange Monte Carlo trials. *J. Chem. Phys. 117* (2002), 6911–6914.

[33] Earl, D., and Deem, M. W. Parallel tempering: theory, applications, and new perspectives. *Phys. Chem. Chem. Phys. 7* (2005), 3910–3916.

[34] Mignon, D., and Simonson, T. Comparing three stochastic search algorithms for computational protein design: Monte Carlo, Replica Exchange Monte Carlo, and a multistart, steepest-descent heuristic. *J. Comput. Chem. 37* (2016), 1781–1793.

[35] Gough, J., Karplus, K., Hughey, R., and Chothia, C. Assignment of homology to genome sequences using a library of hidden Markov models that represent all proteins of known structure. *J. Mol. Biol. 313* (2001), 903–919.

[36] Wilson, D., Madera, M., Vogel, C., Chothia, C., and Gough, J. The SUPERFAMILY database in 2007: families and functions. *Nucl. Acids Res. 35* (2007), D308––D313.

[37] Andreeva, A., Howorth, D., Brenner, S. E., Hubbard, J. J., Chothia, C., and Murzin, A. G. SCOP database in 2004: refinements integrate structure and sequence family data. *Nucl. Acids Res. 32* (2004), D226–229.

[38] Darden, T. Treatment of long-range forces and potential. In *Computational Biochemistry & Biophysics*, O. Becker, A. Mackerell Jr., B. Roux, and M. Watanabe, Eds. Marcel Dekker, N.Y., 2001, ch. 4.

[39] Jorgensen, W., Chandrasekar, J., Madura, J., Impey, R., and Klein, M. Comparison of simple potential functions for simulating liquid water. *J. Chem. Phys. 79* (1983), 926–935.

[40] Brooks, B., Brooks III, C. L., Mackerell Jr., A. D., Nilsson, L., Petrella, R. J., Roux, B., Won, Y., Archontis, G., Bartels, C., Boresch, S., Caflisch, A., Caves, L., Cui, Q., Dinner, A. R., Feig, M., Fischer, S., Gao, J., Hodoscek, M., Im, W., Kuczera, K., Lazaridis, T., Ma, J., Ovchinnikov, V., Paci, E., Pastor, R. W., Post, C. B., Pu, J. Z., Schaefer, M., Tidor, B., Venable, R. M., Woodcock, H. L., Wu, X., Yang, W., York, D. M., and Karplus, M. CHARMM: The biomolecular simulation program. *J. Comp. Chem. 30* (2009), 1545–1614.

[41] Phillips, J. C., Braun, R., Wang, W., Gumbart, J., Tajkhorshid, E., Villa, E., Chipot, C., Skeel, R. D., Kale, L., and Schulten, K. Scalable molecular dynamics with NAMD. *J. Comput. Chem. 26* (2005), 1781–1802.

[42] Schmidt am Busch, M., Lopes, A., Amara, N., Bathelt, C., and Simonson, T. Testing the Coulomb/Accessible Surface Area solvent model for protein stability, ligand binding, and protein design. *BMC Bioinformatics 9* (2008), 148–163.

Table 1: Test proteins and their homologs

PDB code	residue numbers	# active positions	number of homologs	E-value threshold	identity % range	protein acronym
1G9O	9-99	76	62	1e-32	67-95	NHERF
1IHJ	12-105	82	42	1e-10	38-95	INAD
1N7E	667-761	79	48	1e-45	84-95	GRIP
1R6J	192-273	72	85	1e-43	85-95	syntenin
2BYG	186-282	82	43	1e-41	78-95	DLG2
3K82	305-402	80	50	1e-46	81-95	PSD95
1KWA	487-568	74	126	7e-28	60-85	Cask
4GVD	837-930	84	50	2e-23	60-85	Tiam1

).

Table 2: Model variants

model name	model symbol	dielectric constant	σ_i set	# target proteins	σ_i values (cal/mol/Å^2) (unp,aro,pol,ion)
A	(ϵ_P=8,S1,n=6)	8	set 1	6	-5, -12, -8, -9
A'	(ϵ_P=8,S1,n=2)	8	set 1	2	-5, -12, -8, -9
B	(ϵ_P=8,S2,n=2)	8	set 2	2	-5,-40,-80,-100
B'	(ϵ_P=4,S2,n=2)	4	set 2	2	-5,-40,-80,-100

).

Table 3: Unfolded state reference energies

		Model A		Model B	
Residues	Peptide	Buried	Exposed	Buried	Exposed
ALA	0.00	0.00	0.00	0.00	0.00
CYS	-1.04	-1.04	-1.04	-0.85	-0.85
THR	-3.82	-3.82	-3.82	-5.44	-5.44
ASP	-9.17	-9.19	-9.80	-11.90	-15.88
GLU	-7.88	-7.90	-8.51	-11.97	-15.95
ASN	-5.64	-5.94	-6.00	-7.82	-10.22
GLN	-4.42	-4.72	-4.78	-7.07	-9.47
HIP	15.72	14.53	14.96	12.53	9.73
HIE	12.62	11.43	11.85	10.49	7.69
HID	13.16	11.96	12.39	10.86	8.06
ILE	1.63	4.72	2.11	4.63	3.63
VAL	-2.25	0.83	-1.77	0.26	-0.74
LEU	-1.92	1.17	-1.44	-0.12	-1.12
LYS	-4.21	-4.56	-4.47	-6.76	-10.17
MET	-2.44	-2.78	-3.54	-2.05	-2.40
ARG	-25.30	-28.29	-28.90	-32.00	-35.18
SER	-2.85	-3.73	-2.80	-3.71	-4.74
PHE	-1.42	-0.37	-2.55	-0.23	-4.17
TRP	-2.66	-1.61	-3.79	-2.21	-6.15
TYR	-4.56	-4.20	-6.10	-5.80	-9.82

).

Table 4: Amino acid composition

R	Model A				Exper. n=6				Exper. n=2				Model B			
	Buried		Exposed		Buried		Exposed		Buried		Exposed		Buried		Exposed	
A	11.1	17.0	4.4	12.0	10.9	16.9	4.6	13.4	5.9	11.2	4.6	13.4	4.1	12.7	7.2	13.6
C	0.0	[0.1]	0.3	[-1.4]	1.3		0.5		1.5		1.2		8.6	[1.5]	5.8	[0.2]
T	5.9		7.3		4.7		8.3		3.8		7.6		0.0		0.6	
D	4.5	6.7	5.6	16.7	4.3	6.8	6.0	17.9	3.5	9.6	6.2	16.7	7.4	9.4	8.0	16.1
E	2.2	[-0.1]	11.1	[-1.2]	2.5		11.9		6.1		10.5		2.0	[-0.2]	8.1	[-0.6]
N	2.5	4.7	7.5	14.0	2.6	4.7	6.7	12.2	1.9	2.7	7.4	16.1	1.8	2.8	8.6	17.1
Q	2.2	[0.0]	6.5	[1.8]	2.1		5.5		0.8		8.7		1.0	[0.1]	8.5	[1.0]
H^+	1.0	1.1	5.2	5.6	1.2	1.2	5.0	5.0	0.7	0.7	4.7	4.7	0.1	0.9	1.8	4.5
H_ϵ	0.1	[-0.1]	0.4	[0.6]	0.0		0.0		0.0		0.0		0.6	[0.2]	2.2	[-0.2]
H_δ	0.0		0.0		0.0		0.0		0.0		0.0		0.2		0.5	
I	16.9	52.1	4.1	14.0	16.0	50.7	4.2	14.0	15.7	49.6	4.1	14.4	25.1	46.7	8.4	15.3
V	16.7	[1.4]	5.6	[0.0]	16.5		5.4		13.5		5.5		12.8	[-2.9]	3.3	[0.9]
L	18.5		4.3		18.2		4.4		20.4		4.8		8.8		3.6	
K	1.5	1.5 [-1.0]	13.0	13.0 [2.1]	2.5	2.5	10.9	10.9	6.5	6.5	10.1	10.1	5.5	5.5 [-1.0]	10.8	10.8 [0.7]
M	1.6	1.6 [0.7]	1.4	1.4 [-0.1]	0.9	0.9	1.5	1.5	5.0	5.0	1.4	1.4	5.9	5.9 [0.9]	1.4	1.4 [0.0]
R	2.5	2.5 [-0.3]	6.1	6.1 [-2.6]	2.8	2.8	8.7	8.7	1.8	1.8	9.5	9.5	2.2	2.2 [0.4]	9.1	9.1 [-0.4]

Table 5: Superfamily results

Protein	Model	Match/seq length	Superfamily E-value	Superfamily success #	Family E-value	Family success #
1G9O	A	78/91	2.5e-3	10000	3.0e-3	10000
1IHJ	A	86/94	5.6e-7	10000	2.3e-3	10000
1N7E	A	81/95	1.1e-6	10000	2.4e-3	10000
1R6J	A	41/82	1.5	1350	2.6e-2	1350
2BYG	A	77/97	1.0e-2	10000	2.3e-3	10000
3K82	A	79/97	5.8e-10	10000	3.6e-3	10000
Tiam1	A	43/94	1.3	442	4.0e-2	374
Cask	A	72/83	2.3e-4	10000	1.5e-2	10000
Tiam1	B	64/94	1.2e-4	9920	5.2e-2	9058
Cask	B	71/83	3.2e-7	10000	8.2e-3	10000
1G9O	Rosetta	79/91	1.3e-13	10000	2.2e-3	10000
1IHJ	Rosetta	85/94	7.4e-14	10000	3.7e-3	10000
1N7E	Rosetta	84/95	2.2e-10	10000	1.2e-3	10000
1R6J	Rosetta	76/82	7.3e-13	10000	1.8e-3	10000
2BYG	Rosetta	86/97	1.3e-9	10000	9.6e-4	10000
3K82	Rosetta	90/97	3.7e-23	10000	5.2e-4	10000
Tiam1	Rosetta	65/94	4.4e-4	9035	2.8e-2	9030
Cask	Rosetta	68/83	2.8e-5	9832	7.5e-3	9832

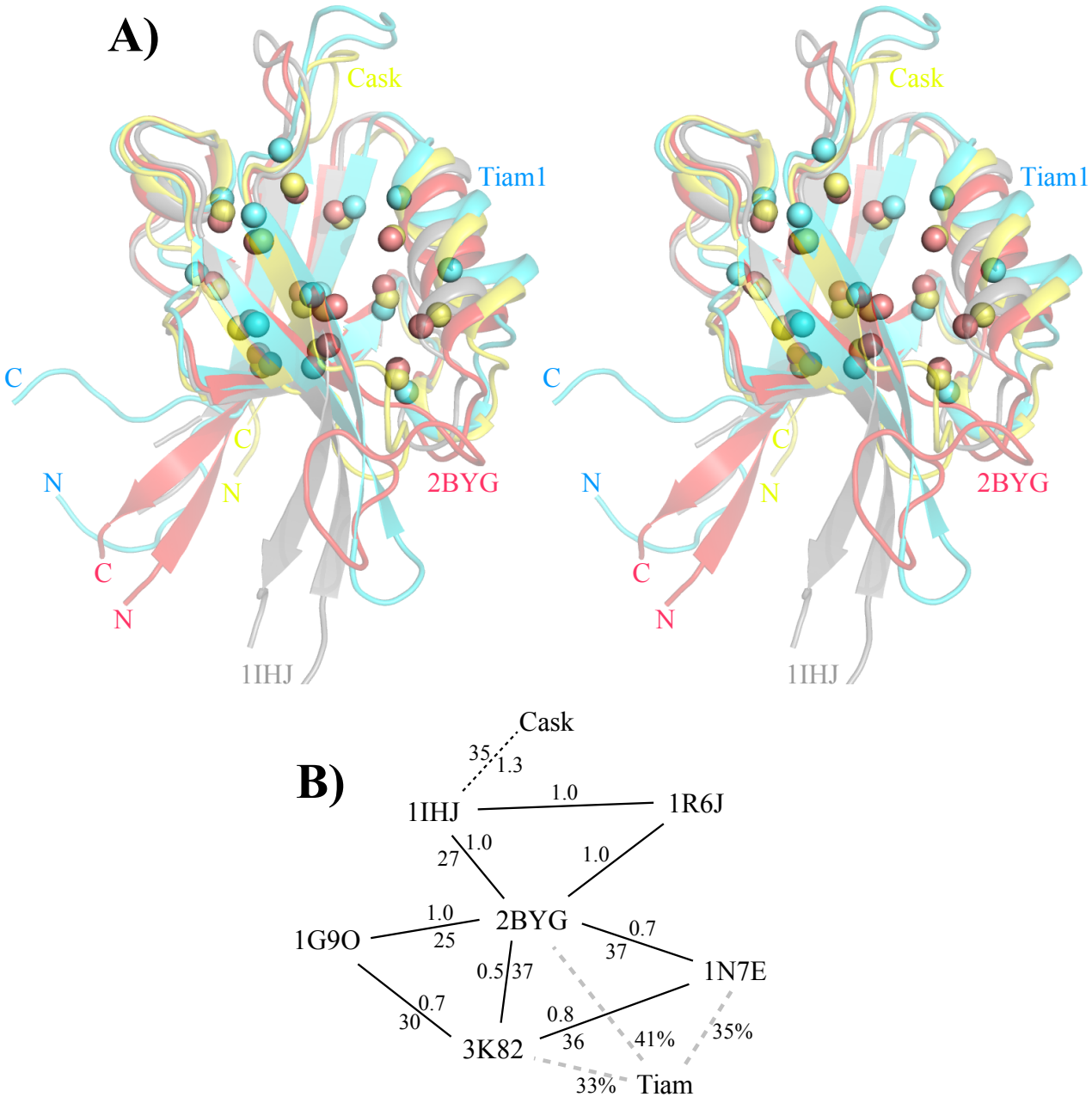

Figure 1: **A)** 3D view of four PDZ domains. **B)** Cluster representation.

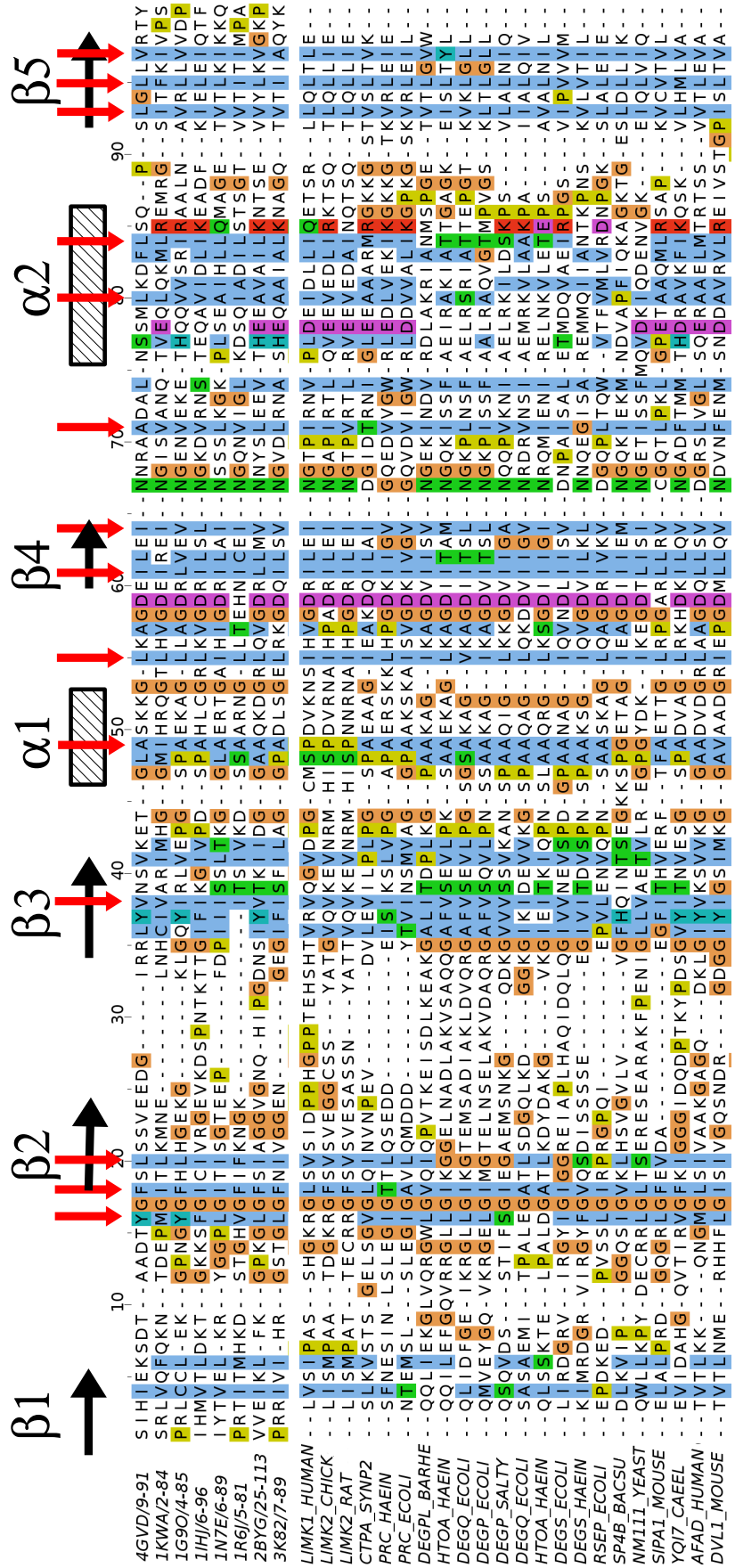

Figure 2: Alignment.

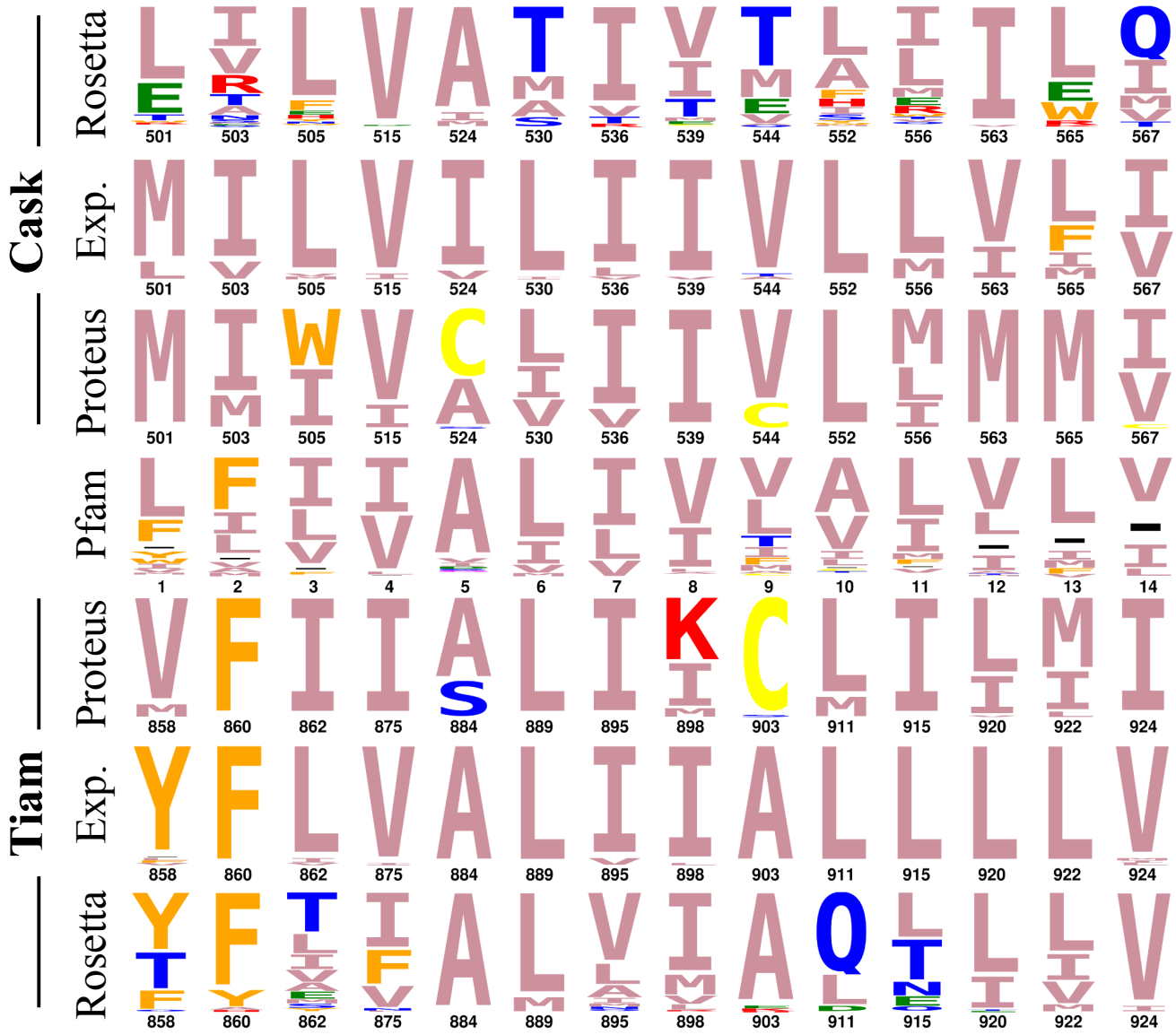

Figure 3: Core logos.

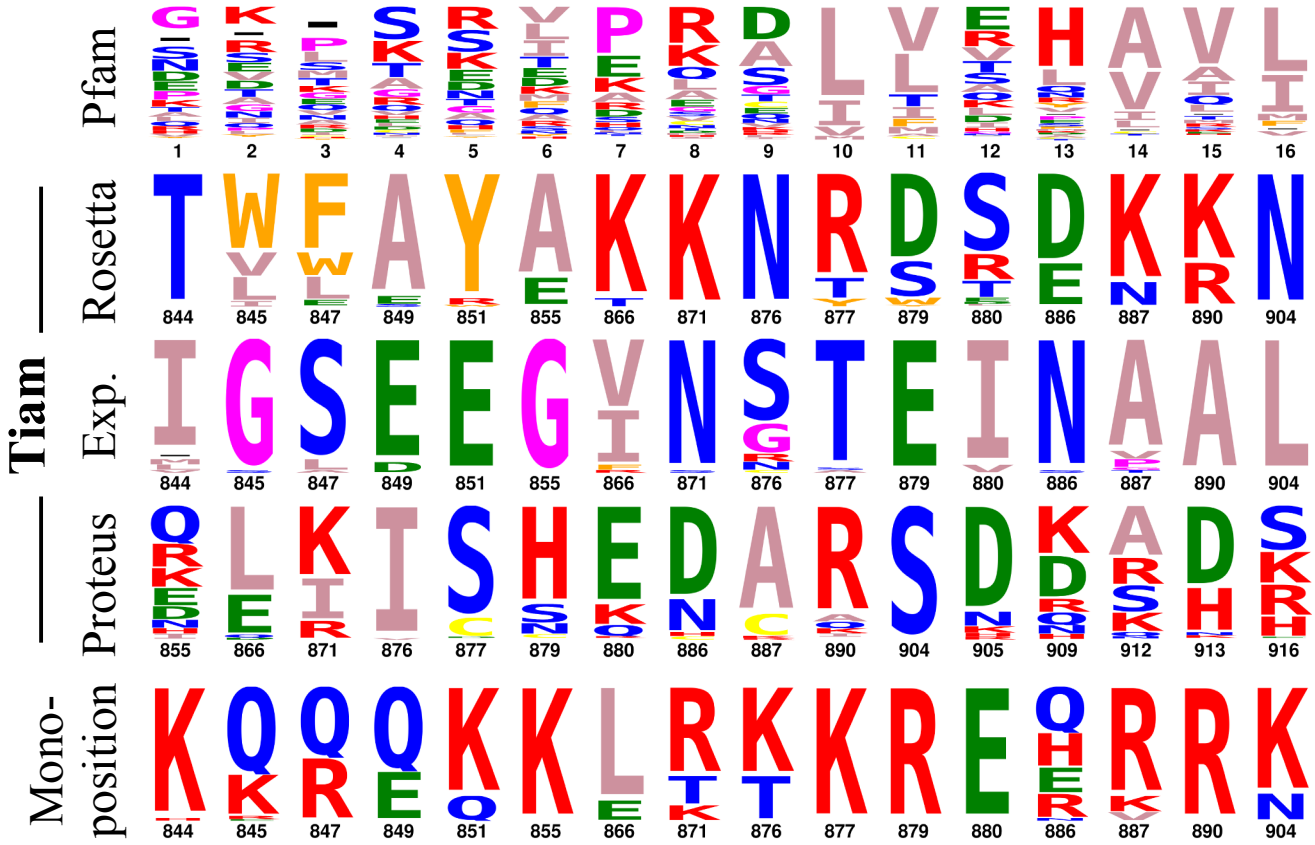

Figure 4: Surface logos.

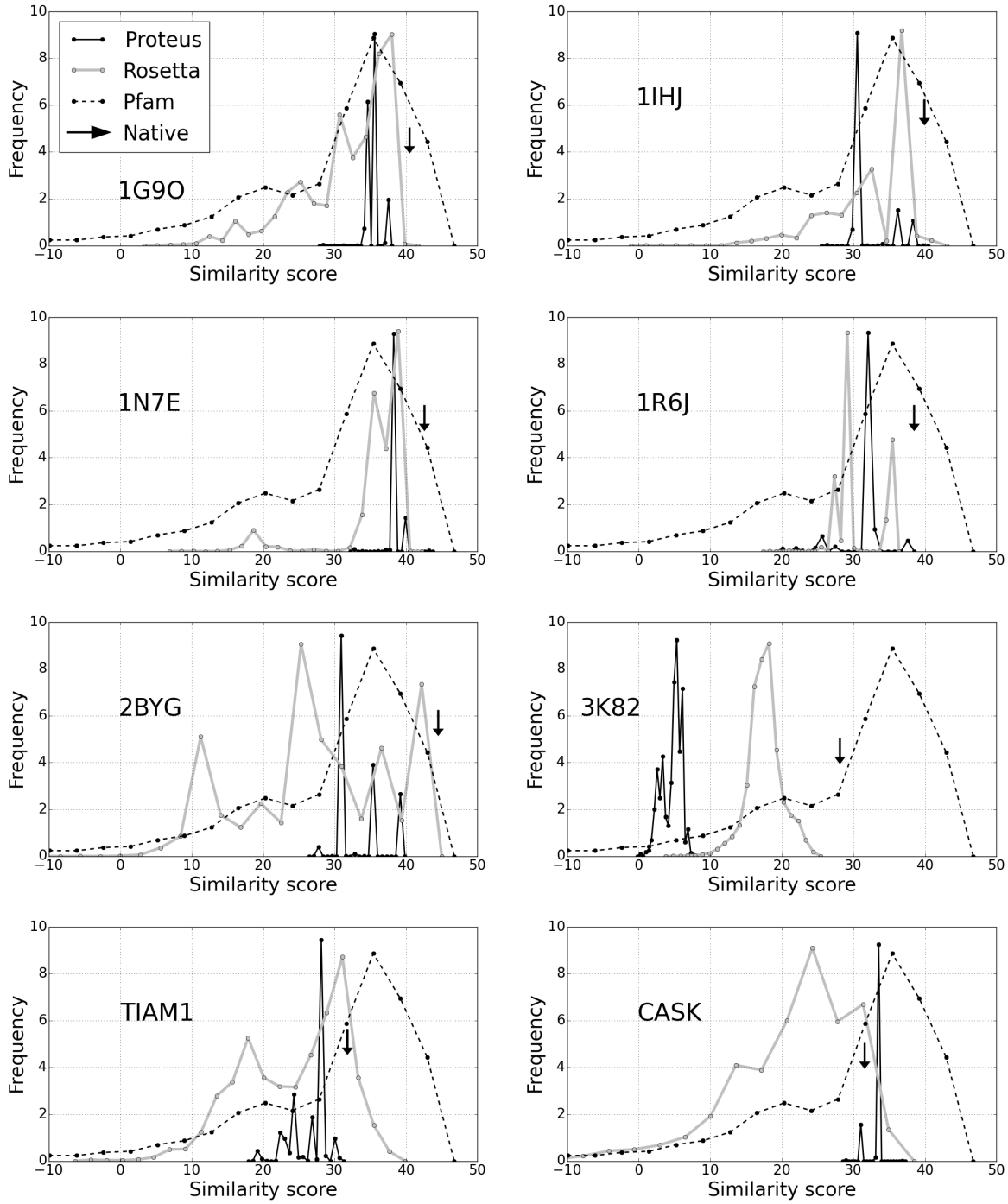

Figure 5: Similarity scores.

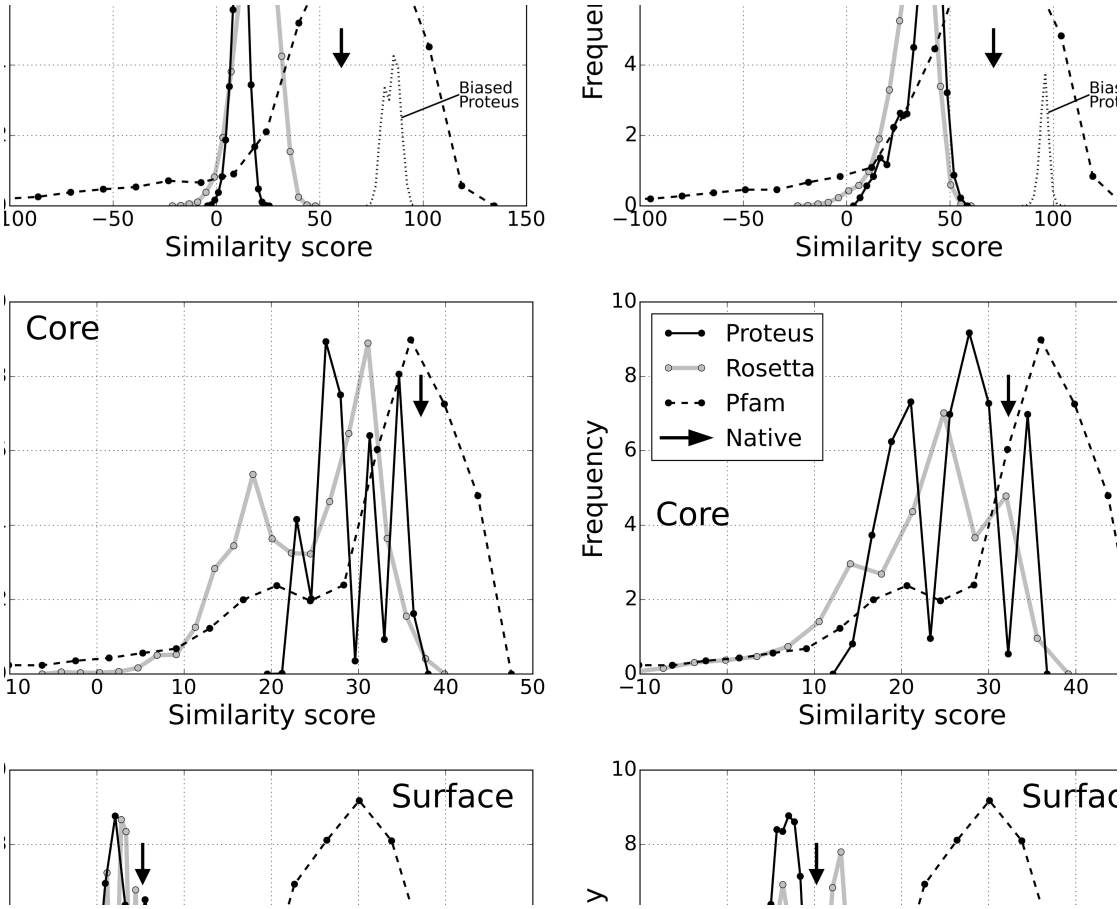

Figure 6: Similarity scores.

Figure 7: MD figure.

Figure 8: Titration figure.

Figure 9: QM figure.