

Structural bioinformatics

Cost function network-based design of protein–protein interactions: predicting changes in binding affinity

Clément Viricel^{1,2}, Simon de Givry², Thomas Schiex^{2,*} and Sophie Barbe^{1,*}

¹Laboratoire d'Ingénierie des Systèmes Biologiques et des Procédés, Université de Toulouse, CNRS, INRA, INSA, 31400 Toulouse, France and ²Unité de Mathématiques et Informatique Appliquées de Toulouse, INRA, 31326 Castanet Tolosan cedex, France

*To whom correspondence should be addressed.

Associate Editor: Alfonso Valencia

Received on August 2, 2017; revised on February 6, 2018; editorial decision on February 15, 2018; accepted on February 16, 2018

Abstract

Motivation: Accurate and economic methods to predict change in protein binding *free energy* upon mutation are imperative to accelerate the design of proteins for a wide range of applications. Free energy is defined by enthalpic and entropic contributions. Following the recent progresses of Artificial Intelligence-based algorithms for guaranteed NP-hard energy optimization and partition function computation, it becomes possible to quickly compute minimum energy conformations and to reliably estimate the entropic contribution of side-chains in the change of free energy of large protein interfaces.

Results: Using guaranteed Cost Function Network algorithms, Rosetta energy functions and Dunbrack's rotamer library, we developed and assessed EasyE and JayZ, two methods for binding affinity estimation that ignore or include conformational entropic contributions on a large benchmark of binding affinity experimental measures. If both approaches outperform most established tools, we observe that side-chain conformational entropy brings little or no improvement on most systems but becomes crucial in some rare cases.

Availability and implementation: as open-source Python/C++ code at sourcesup.renater.fr/projects/easy-jayz.

Contact: sophie.barbe@insa-toulouse.fr or thomas.schiex@inra.fr

Supplementary information: [Supplementary data](#) are available at *Bioinformatics* online.

1 Introduction

Protein–Protein Interactions (PPI) play an essential role in all life processes. The ability to rationally design protein–protein interfaces with tight or otherwise modified binding affinity is a stringent test of our understanding of molecular recognition and of great interest for a wide range of industrial, technological and biomedical applications (Haidar *et al.*, 2009; Pierce *et al.*, 2014; Tinberg *et al.*, 2013). Experimental evaluation of the effect of mutations on protein binding is an expensive and time-consuming process. There is thus a high demand for fast and accurate *in silico* methods to predict protein binding affinity changes upon amino acid mutations.

The binding affinity of two proteins is governed by the change in free energy upon binding. At temperature T , free energy (G) has two components coding enthalpy (H) and entropy (S) as $G = H - TS$. Standard computational methods based on physics-based energy functions estimate enthalpic binding contributions and may sometimes include implicit solvation free energy terms in the definition of the energy function in order to take into account solvent entropy contributions. The estimation of the missing protein conformational entropy is a challenging computational problem, of extreme complexity.

In theory, computational methods such as free energy perturbation, thermodynamic integration, umbrella sampling or potential of mean force approaches (Åqvist *et al.*, 2002; Beveridge and DiCapua, 1989; Chipot, 2014; Dixit and Chipot, 2001) can be used to approximate the conformational entropy contribution to the binding energy. Though in principle rigorous, these methods require extensive sampling of protein conformations (e.g. via molecular dynamics simulations) which is associated with a high computational cost. They are thus not suitable to study large complexes or a substantial number of mutants which is needed for the design of protein–protein interfaces.

Therefore, for the design of PPI, the most commonly used approaches neglect changes in conformational entropy. These methods predict enthalpic changes by approximating the internal energy E of the system. They mainly rely on pairwise physics-based or/and statistical energy functions, a discrete set of low-energy side-chain conformations (called rotamers) and energy optimization algorithms (Kortemme and Baker, 2002). Amino-acid side-chains in the binding interface are replaced according to the new sequence and the surrounding (or all) amino-acid side-chains are repacked while the protein backbone is kept fixed. This process relies on optimization algorithms that try to identify the combination of side-chain rotamers of global minimum energy or GMEC. The binding energy is then estimated by the difference in energy of the bound and unbound proteins in their optimal conformation and the effect of a mutation is evaluated as the change in binding energies between the wild type and the mutant. The effect of a mutation on binding is therefore reduced to changes in intermolecular energies, each computed on a single conformation. These GMEC-based approaches omit the conformational entropy contribution to the binding free energy. Some alternate approaches use empirically-derived or statistical potentials in the energy function to account for the crude change in conformational entropy (Dehouck *et al.*, 2013; Guerois *et al.*, 2002; Li *et al.*, 2016; Schymkowitz *et al.*, 2005). Statistical approaches consider changes in coarse structural features such as the change in overall volume. Though computationally advantageous, these implicit entropy terms neglect details of interactions between atoms and are highly dependent on the availability of case-dependent experimental training and independent testing data. Additionally, most of these methods optimize energy using stochastic local search procedures that offer no guarantee that the lowest energy conformation will be identified in finite time. These routines may instead end up trapped in local minima. Our previous work has showed that the accuracy of an optimized and popular implementation of Monte Carlo simulated annealing in the well-known Rosetta software degrades as problem size increases and that the probability of finding the GMEC drops quickly close to 0 as problems get harder (Simoncini *et al.*, 2015). In contrast to these approaches based on a single low-energy conformation, some recent methods explicitly account for conformational entropy by exploiting the connection between the partition function Z and the free energy $G = -k_B T \log(Z)$ (Georgiev *et al.*, 2008; Kamisetty *et al.*, 2011; Ojewole *et al.*, 2017; Sciretti *et al.*, 2009; Silver *et al.*, 2013; Viricel *et al.*, 2015). Since it is not currently possible to compute the exact partition functions for a protein complex, they are computed over rotamer-based conformational ensembles using pairwise decomposed energy functions. Computing the partition function of a system at constant temperature requires to sum up exponential functions of the energy over all conformational states. With discrete rotamers, this continuous integral is replaced by a discrete sum with an exponential number of terms. Computing Z under these simplifying assumptions is still a P-complete problem, a class of problems

with extreme computational complexity (Valiant, 1979; Toda, 1989). Despite the hardness of computing partition function, some algorithms can compute the partition function with guaranteed approximations on the input model. The K^* algorithm (Georgiev *et al.*, 2008) uses a combination of Dead-End Elimination (DEE) pruning (Goldstein, 1994) and A^* tree-search based gap free conformation enumeration (Leach and Lemon, 1998) to provably approximate the partition function. However, it suffers of the worst-case exponential time and space complexity of A^* , exacerbated by a loose admissible heuristics (Viricel *et al.*, 2015) and does not enable the enumeration of many low energy conformations for large protein design problems (Traoré *et al.*, 2013), thus decreasing the accuracy of the predicted partition function. Following their amazing progress on NP-complete solving, propositional SAT(isifiability) solvers and knowledge compilation approaches have also been used to exactly compute the partition function (Chavira and Darwiche, 2008; Sang *et al.*, 2005) but are still limited to small problems. Forgetting about guarantees, the GOBLIN method uses Loopy Belief Propagation (LBP) to estimate the partition function formulated as an inference problem over a probabilistic graphical model (Kamisetty *et al.*, 2011). Though computationally cheap, LBP algorithms do not provide any guarantee, asymptotic or not on the accuracy of their estimations.

Recently, we showed that Cost Function Network (CFN) techniques can be exploited to speed-up partition function computation with deterministic guarantees of quality and thus open new routes for provably-accurate approximations of partition functions for estimating binding free energy of large set of protein mutants (Viricel *et al.*, 2015, 2016). In this paper, we present EasyE and JayZ, two novel GMEC-based and partition function-based computational approaches to estimate binding energy changes upon amino-acid mutations and investigate the role of entropy on PPI. Built upon our previous CFN-based computational protein design methods (Allouche *et al.*, 2014; Traoré *et al.*, 2016), these PPI design approaches use the state-of-the-art Rosetta energy functions (Alford *et al.*, 2017; Park *et al.*, 2016) and Dunbrack's rotamer library (Shapovalov and Dunbrack, 2011) to predict the effect of mutations on protein–protein binding. Being based on provable optimization and counting methods, their estimation error is known to come only from the assumptions in the protein model. We applied these methods to study the role of conformational entropy in PPI on a benchmark set extracted from the PROXiMATE database (Jemimah *et al.*, 2017) which includes SKEMPI (Moal and Fernández-Recio, 2012) and contains experimentally measured binding affinity of PPI mutants. We also compared these two methods to the GOBLIN, BindProfX and mCSM programs and to the empirical force field-based FoldX approach, widely used to calculate protein–protein binding energy (Guerois *et al.*, 2002; Kiel and Serrano, 2014; Pires *et al.*, 2014; Schymkowitz *et al.*, 2005; Xiong *et al.*, 2017). We showed that EasyE and JayZ achieve good accuracy for predicting binding energy change upon mutations. Compared to previous computational methods our approaches often show better correlation coefficients between predicted and experimental values. At the same time, EasyE can efficiently handle large number of mutants.

2 Materials and methods

The Cost Function Network-based PPI design strategies developed in this work are described in Figure 1. They rely on a target protein–protein complex structure defining a fixed protein backbone, a

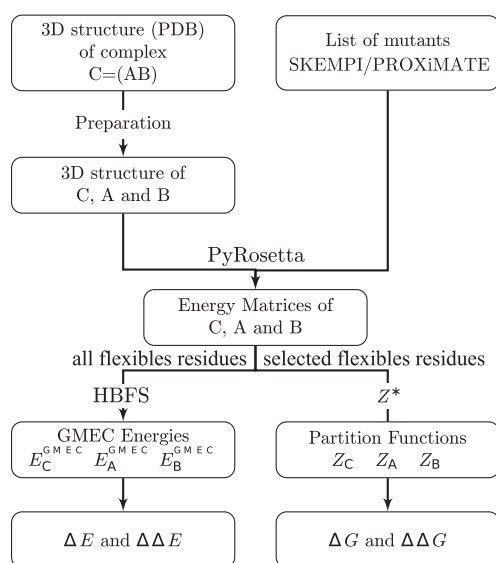


Fig. 1. Workflow for affinity estimation using either EasyE (the GMEC-based, left) or JayZ (Partition function-based, right) approach

discrete set of rotamers and a pairwise decomposable energy function. As is common in such approaches, only the side-chain conformational variability of flexible or mutable amino-acid residues is considered. From this input biophysical model, an optimization algorithm providing guaranteed minimum-energy conformations and a counting algorithm providing a guaranteed approximation of the partition function have been used to estimate the change in binding energy upon mutation on a benchmark set of 1098 PPI systems.

2.1 PPI design using Cost Function Networks

In our settings (fixed backbone, discrete rotamer library and a pairwise decomposable energy), the energy of a given sequence-conformation of a protein in bound or unbound state can be written as:

$$E_{total} = E_c + \sum_i E_i(i_r) + \sum_{j>i} E_{ij}(i_r, j_s)$$

where E_c is a constant energy contribution capturing interactions between fixed parts of the model, $E_i(i_r)$ captures internal side-chain energies and rotamer-backbone interactions for rotamer r at position i , and $E_{ij}(i_r, j_s)$ captures pairwise interactions between rotamers r and s at position i and j respectively. These single and pair energy terms can be precomputed and stored in energy matrices. This global energy function can be represented as a Markov Random Field (Kamisetty *et al.*, 2011), a specific type of stochastic graphical model (Koller and Friedman, 2009).

Cost Function Networks (CFNs) are deterministic graphical models that generalize Constraint Satisfaction problems (Schiex *et al.*, 1995). A CFN allows to concisely define a complex global cost distribution over many variables as a sum of local cost functions each involving some variables (Cooper *et al.*, 2010). CFNs have been used to solve various combinatorial optimization problems in bioinformatics and resource allocation (Cabon *et al.*, 1999; Thébault *et al.*, 2006; Zytynicki *et al.*, 2008).

Formally, a CFN $P = (X, D, C)$ is defined by a set of variables $X = \{X_1, \dots, X_n\}$ each with a discrete domain of possible values $D_i \in D$ and a set C of local cost functions. Each cost function $c_S \in C$ is defined over a subset of variables $S \subset X$ (called its scope),

has domain $\prod_{x_i \in S} D_i$ and takes its values in $\mathbb{N} \cup \infty$. It is usually assumed that C contains one constant cost function, with an empty scope, denoted as c_\emptyset .

Cost functions can be arbitrary and are often described by cost tables. The infinite cost identifies infeasible combinations of values. A labeling $\mathbf{x} = (X_1 \leftarrow x_1, \dots, X_n \leftarrow x_n)$ is a mapping from variables to values from their domains. The global cost $C(\mathbf{x})$ of a labeling is the sum of the costs $c_S(\mathbf{x})$ over all local cost functions. It defines a joint cost distribution over \mathbf{X} . Notice that since all cost functions in a CFN are non-negative, the constant function c_\emptyset defines a lower bound of the joint global cost $C(\mathbf{x})$.

Modeling the energy of a protein defined with a rigid backbone, discrete rotamers and decomposable energy as a CFN is straightforward. Each flexible amino acid residue AA_i is represented by a variable X_i . The set of possible rotamers for the residue AA_i defines the domain D_i of the variable X_i . Each energy term in E_{total} is represented as a cost function: the constant term E_c is captured as the constant cost function with empty scope c_\emptyset and E_i and E_{ij} terms are respectively represented by unary c_i and binary c_{ij} cost functions, involving the variables of the corresponding residues. These cost functions are integer cost matrices defined from precomputed energy matrices. Floating point energy terms can be mapped to non-negative integers through shifting, scaling and rounding according to desired precision. The joint cost distribution defined by the corresponding CFN is then equal to the energy, up to a known shift and scale. In the rest of the paper, we therefore confound energy and cost.

2.2 EasyE, a GMEC-based approach

CFN are mostly known for their exact cost minimization algorithms (Cooper *et al.*, 2010; Hurley *et al.*, 2016). These algorithms solve the NP-hard problem of computing a labeling \mathbf{x}^* that minimizes the joint cost distribution $C(\mathbf{x})$. In the case of protein modeling, such an optimal labeling defines a Global Minimum Energy Conformation (GMEC) which is also NP-hard to find (Pierce and Winfree, 2002). The most usual exact minimization algorithms for CFN are based on tree-search. These algorithms explore a binary tree where each node is associated with a CFN with possibly restricted domains. The root of the tree corresponds to the original CFN problem and the sons of a node are obtained by either restricting the domain of a variable X_i to a single value x_i or by removing x_i from D_i . The leaves of the tree therefore define complete labelings, describing all possible side-chain conformations. Because this tree has exponential size and bounded depth, Depth First Search Branch and Bound (DFS-BB) is used to avoid a complete exploration. Compared to A*, DFS is polynomial space and allows to immediately dive in the tree which quickly and continuously provides a ‘currently best known labeling’. The energy of this labeling defines an increasingly tight upper bound on the energy of the GMEC. The specificity of CFN algorithms lies in the use of a family of lower bounds computed by polynomial time/space filtering algorithms, each associated with a specific ‘local consistency’ property (Cooper *et al.*, 2010; Schiex, 2000). Each filtering algorithm transforms a CFN P into an equivalent CFN P' (defining the same joint cost distribution) with a possibly increased global lower bound c_\emptyset . At a given node, if c_\emptyset is equal to or larger than the best known energy, then the node can be pruned. Compared to our previous work (Traoré *et al.*, 2013), and because we are not dealing with full redesign problems, we used the fast Existential Directional Arc Consistency bound (Larrosa *et al.*, 2005) for preprocessing and during search. We exchanged DFS-BB for the more recent Hybrid Best First Search (HBFS) algorithm

(Allouche *et al.*, 2015). HBFS combines the advantage of Depth First and Best First Search (used in A*) to provide a space-bounded search algorithm (A* has exponential space complexity) that supplies a sequence of quickly decreasing energy conformations (usually faster than DFS while A* outputs a labeling only when it finishes). HBFS also has the advantage of providing progress feedback during search through an increasingly tight energy gap to optimality.

For a given protein–protein complex ($C=AB$) and a given sequence, we used HBFS to identify the GMEC of the bound and unbound states. All residues were considered as flexible in all states. This means the number of variables in the network is exactly the number of residues of the protein(s). The energy values of the three GMECs were then used to compute the binding energy, $\Delta E = E_C^{GMEC} - (E_A^{GMEC} + E_B^{GMEC})$. The binding energies were computed for both the wild-type ΔE_{WT} and mutant ΔE_{MUT} protein complexes. The changes in binding energy upon mutations, $\Delta\Delta E$, were then calculated as $\Delta\Delta E = \Delta E_{WT} - \Delta E_{MUT}$. This approach follows the same principle as the ddG Rosetta protocol (Kortemme and Baker, 2002) but relies on a provable method instead of stochastic search, removing all uncertainty except those that come from modeling.

2.3 JayZ, a partition function-based approach

In contrast to GMEC computation, partition function computation requires weighted counting instead of optimization. Assuming w.l.o.g. that no symmetry or state degeneracy exists in the considered system, the partition function of a protein at temperature T with a set of micro-states $l \in \Lambda$ each with energy E_l is equal to (k_B is Boltzmann's constant):

$$Z = \sum_{l \in \Lambda} \exp\left(\frac{-E_l}{k_B T}\right)$$

In our context, computing Z therefore requires to sum $\exp\left(\frac{-E_l}{k_B T}\right)$ over an exponential number of possible side-chain conformations [To account for the fact that this number increases with the rotamer library density, a log-probability distribution over rotamers must be included in the energy (as the fa_dun term in Rosetta energies)]. To achieve this, we developed JayZ, a partition function-based PPI design method, built up on our recently introduced Z^* weighted counting algorithm that offers a guaranteed maximum error (Viricel *et al.*, 2015, 2016). Z^* explores the same tree as for minimization. Each time it visits a leaf with an associated complete conformation, it adds the corresponding contribution $\exp\left(\frac{-E_l}{k_B T}\right)$ to Z in a running estimate \hat{Z} . To avoid exploring the tree in its entirety, Z^* exploits proofs that the sum $Z(n)$ of all contributions below the current node n will be negligible compared to the true (yet unknown) value of Z . To achieve this, it uses the lower bound c_{\emptyset} provided by filtering to compute an upper bound $U(n)$ of $Z(n)$. If this upper bound is sufficiently small, it prunes the branch and adds $U(n)$ to a running sum of errors \bar{U} . By pruning only if the invariant $\bar{U} \leq \varepsilon \hat{Z}$ is not broken, Z^* eventually provides a guaranteed ε -approximation of Z using polynomial space, where K^* uses exponential space (Lilien *et al.*, 2005). To further accelerate counting, Z^* performs on-the-fly sum-product variable elimination (Larrosa, 2000; Koller and Friedman, 2009) each time a variable X_i of small degree or s.t. $|D_i| = 1$ appears.

The Z^* algorithm has already shown good performances on various protein partition function computations, a domain where it outperforms K^* as well as SAT solver-based or knowledge

compilation-based exact counters (Viricel *et al.*, 2015, 2016). We used it to estimate the binding affinity of a protein complex ($C=AB$) through its association constant $K_a = \frac{Z_C}{Z_A Z_B}$ related to the change in free binding energy by $\Delta G = -k_B T \log(K_a)$. Despite its relative efficiency, Z^* is still exponential time in the worst case and it is usually necessary to define a subset of amino acids that will be considered as flexible for partition function computation. Eventually, we compute $\Delta\Delta G^c = \Delta G_{WT} - \Delta G_{MUT}$.

EasyE optimization and JayZ counting algorithms are implemented in our open source solver toulbar2 (version 0.9.8). Beyond the sources of our solver, we give access to Python scripts to extract energy matrices through PyRosetta, and compute ΔE and ΔG .

2.4 Comparison to experimental data

To evaluate the quality of the estimations of a method that predicts binding free energy, we considered three different measures: the quality of a linear regression between experimental and predicted value estimated through the Pearson correlation coefficient R , a Root Mean Square Error (RMSE) between experimental values and predicted values and the Area Under the Curve (AUC) of a sensibility/specificity ROC curve using predicted values to decide if mutations improve (or not) the binding free energy over the Wild Type (WT) sequence.

To determine if the predictions bring real added value, we also compute a P -value where H_0 assumes an optimal random prediction. For a given benchmark set, we consider a predictor that randomly samples from a Bernoulli distribution of parameter μ to predict a ΔG_S which is lower (or higher) than ΔG_{WT} . For a given system, if σ is the ratio of sequences S with experimental $\Delta G_S \leq \Delta G_{WT}$, then a random predictor with optimal accuracy is obtained with $\mu = \sigma$. The P -value associated with H_0 is the probability that such an optimal random predictor performs at least as good as our predictor. If we have n mutants, this means that n repetitions of Bernoulli sampling will lead to a count of positive $\#T$ larger than or equal to the observed number of correct predictions ($\#CP$). The P -value associated with H_0 is therefore:

$$P(\#T \geq \#CP) = \sum_{i=\#CP}^n P(\#T = i) = \sum_{i=\#CP}^n \binom{n}{i} \mu^i (1 - \mu)^{n-i}$$

Finally, since the measurement of association constants has a typical precision of 20% (Lu *et al.*, 1997), each experimentally determined binding affinity K_a was represented by a random variable following a normal distribution with mean K_a and a standard deviation of 20% of K_a , truncated to values between $\pm 20\%$ of K_a . For each evaluation measure, we took the empirical average of the measure over a thousand samples. E.g. the correlation coefficient R was computed 1,000 times, each with a sampled experimental value and we report the average of all R over these thousand samples.

2.5 Benchmark set

Protein–protein complexes were selected from the PROXiMATE database (Jemimah *et al.*, 2017), the union of the SKEMPI database (Moal and Fernández-Recio, 2012) and other experimentally measured effects of mutations on protein complexes for which a crystal structure of the WT is available. This database is not limited to the results of alanine scanning mutagenesis and includes single-point or multiple simultaneous mutations to any type of amino acid. A subset of the database excluding redundant measures was used (see Supplementary Table S1).

We removed non-peer-reviewed data and mutants in which substituted residues were not resolved in the wild-type protein crystal structure. To be able to evaluate prediction quality on each system independently and reliably, we selected protein–protein complexes for which experimental binding affinities measured with the same technique were available for at least 10 mutants and redundant mutant entries were avoided. Because experimental measures of association constants have limited precision and may include non-negligible lab or technical biases, we did not pool all measures together as this may lead to fits (or misfits) that would result more from experimental biases than actual variation in the association constant.

For comparability, we removed few systems that could not be handled by the GOBLIN software because of out of memory issues. Eventually, our main dataset consists of 1098 experimental ΔG measures (including wild-type and mutants with up to 8 substitutions/mutant) on 21 different protein–protein complexes. It comprises a range of experimental protein–protein binding free energies from -17.35 to -3.09 kcal mol $^{-1}$ and includes well-studied proteins such as the Bovine α -chymotrypsin (PDB ID 1cho), Human leukocyte elastase (1ppf), Subtilisin Carlsberg (1r0r) and *Streptomyces griseus* proteinase B (3sgb). Our dataset includes 163 mutants for each of these 4 systems, that together represent almost 60% of the overall dataset. Because of the additional limitations of BindProfX (location of the mutations) and mCSM (only single-point mutation), two additional subsets compatible with each of these limitations were defined.

2.6 Preparing complexes and energy matrices

To compute binding energies, the three-dimensional structure of the protein–protein complex and the unbound proteins is needed. The structure of the wild-type protein–protein complexes were extracted from the Protein Data Bank (Berman *et al.*, 2000) (see Supplementary Table S1) and water molecules were removed. As usual in high-throughput approaches, we considered that the native backbone in the wild-type complex is a good approximation for the wild-type unbound states as well as the bound and unbound states of mutants. The two chains of wild-type protein–protein complex X-ray structures were separated and missing hydrogens were added using PyRosetta version 144 (Chaudhury *et al.*, 2010). Energy matrices were computed for the wild-type proteins and the mutants in bound and unbound states using PyRosetta, Rosetta energy functions and Dunbrack backbone dependent rotamer library (Alford *et al.*, 2017; Park *et al.*, 2016; Shapovalov and Dunbrack, 2011). These energy matrices storing single and two bodies energy terms were then used as input to the CFN solver toulbar2 (Cooper *et al.*, 2010; Hurley *et al.*, 2016).

We experimented with the *hard* and *soft-rep* versions of beta_{nov15} and beta_{nov16} Rosetta energy functions (Alford *et al.*, 2017; Park *et al.*, 2016). In the *hard-rep* variant, the Lennard-Jones repulsive terms are not damped and atomic clashes incur huge energetic penalties. The *soft-rep* energy function has the repulsive interactions at short atomic separations damped and thus, small atomic overlaps are not penalized.

Besides the systematic inclusion of the side-chain conformations in the wild type (Through the use_input_sc flag in rosetta initialization.), we investigated the inclusion of additional rotamers using the ex1/ex2 flags available within (Py)Rosetta. ex1/ex2 specify that for each conformation in the rotamer library, extra samples at ± 1 std-dev for respectively χ_1/χ_2 angles should be included. We denote by EX0 the default rotamer library, EX1 the library with extra χ_1 samples and EX2 the library with oversampled χ_1 and χ_2 .

Because Z^* remains exponential time, we had to restrict the set of residues that are considered as flexible during partition function computations. We decided to consider all positions that may have a side-chain atom within 3 Å of any atom of any mutable side chain. All other side-chains were frozen to their GMEC conformation. The number of variables in the network reduces significantly but the remaining networks remain very dense (see Supplementary Table S8). The temperature for partition function computation was set to the experimental temperature when known, otherwise it was set to 298 K.

2.7 Comparison to other methods

On our full dataset, we compared our methods with two other methods: FoldX (version 4) and GOBLIN. These methods were also compared to BindProfX and mCSM on specific subsets defined by their respective limitations. FoldX uses an empirical energy function which is parametrized on experimental changes of unfolding free energy and includes an implicit entropy term. It estimates the effect of mutations on the stability of a protein or a protein complex (Guerois *et al.*, 2002; Kiel and Serrano, 2014; Schymkowitz *et al.*, 2005). Although it is not parameterized to predict changes in protein binding, it is widely used in PPI design. We used the RepairPDB function within FoldX to perform a quick optimization in X-ray structures of native protein complexes. The BuildModel function was then used to generate protein mutants. Finally, the AnalyseComplex function was used to estimate the binding energy for a given protein–protein complex ($C=AB$), as $\Delta\Delta G_{\text{bind}} = \Delta\Delta G_{\text{fold}}^C - (\Delta\Delta G_{\text{fold}}^A + \Delta\Delta G_{\text{fold}}^B)$.

GOBLIN uses Loopy Belief Propagation and a simple energy function without electrostatics to approximate, with no guarantee, the partition functions for the protein bound and unbound states. The changes in free binding energy are derived from the partition functions (Kamisetty *et al.*, 2011). The temperature for FoldX and GOBLIN was set to the experimental temperature when known, otherwise it was set to 298 K.

BindProfX predicts change in binding affinity by computing a structural score using an ensemble of structurally similar complexes (Xiong *et al.*, 2017). The score reflects the probability of finding an amino acid in a specific position and is computed from the frequencies of each amino acid in a multiple sequence alignment (Brender and Zhang, 2015).

mCSM is a machine learning predictor exploiting atomic-distance patterns around residues captured by cumulative distributions over distance of various atom-types capturing essential physico-chemical properties. A Gaussian process regression model trained on Skempi is used for $\Delta\Delta G$ prediction (Pires *et al.*, 2014).

3 Results and discussion

As described in detail in Section 2, we experimented with different methods and protocols for computing binding energy changes upon mutations. We exploited 1098 experimental ΔG measures on the wild-type and mutants of 21 protein–protein complexes. The mutations (if any) and experimental binding free energies ΔG^e were extracted from the PROXiMATE/SKEMPI databases and 1077 experimental $\Delta\Delta G^e$ values computed and compared to predicted values.

Energy functions are defined by weights that govern the relative contribution of the different energy terms in the calculation of the protein energy. In most studies of assessment of binding energy prediction methods, an initial training database of mutants is used to

obtain a set of optimal weights that best fits the experimental $\Delta\Delta G_e$ (Kamisetty et al., 2011). If cross-validation on random subsets may be useful to prevent weights over-fitting, it remains extremely difficult to build truly independent subsets for estimation and testing when many samples typically share related or even identical backbones. Moreover, when such methods are applied for a given PPI design problem, experimental binding affinity data on related protein complexes are generally unavailable. We thus decided to assess the performance of the methods in the same conditions than their standard usage: we did not fit the weights to avoid bias towards the training mutant database. Blind tests were thus carried out on all mutants.

3.1 The EasyE approach

We tested EasyE using the four energy functions beta_nov16 (*hard* and *soft*) and three levels of rotamer sampling (EX0, EX1 and EX2). Despite the fact that it relies on an exact algorithm for an NP-hard problem, optimization with EX0 was always efficient (Supplementary Table S7): on the largest complex with 628 flexible residues and the EX0 library, the GMEC was found and proved optimal in 8.18 cpu-time seconds of one Xeon core. No system took more than 20.96 s. Higher rotamer densities are computationally more demanding but remain solvable despite the huge search space they define ($\approx 10^{927}$ conformations for 1ahw using EX2).

EasyE performances were assessed by comparing computed $\Delta\Delta E$ to $\Delta\Delta G^e$ for all mutants (Fig. 2 and Supplementary Fig. S5, Tables S2 and S3). $\Delta\Delta E$ values were computed by performing *in silico* mutations, repeating the binding energy calculation and computing the difference $\Delta E_{WT} - \Delta E_{MUT}$. As already shown in (Kellogg et al., 2011) for monomer $\Delta\Delta G_{fold}$ estimation, we observed that the *soft* version is better suited for predicting free binding energy change upon mutations (Fig. 2, Supplementary Tables S2–S4).

Using the *hard* version, the best correlation coefficient and RMSE obtained are respectively 0.19 and 32 kcal mol⁻¹ against 0.41 and 2.18 kcal mol⁻¹ using the *soft* version. The inclusion of additional rotamers (using EX1 and EX2) only slightly improves the predictions made using the *hard* version. We assume that the damped Lennard-Jones repulsive interaction in the *soft* version

compensates for rotamer discretization and rigid backbone: it makes room for rotamers that could avoid a clash with a small conformational change but which are otherwise eliminated by the *hard* version. Moreover, the beta_nov16_*soft* energy function always gives better results than beta_nov15_*soft* (RMSE of 2.18 versus 2.28 and correlation of 0.41 versus 0.37, respectively). All the results presented later were obtained using the default rotamer library (EX0) and the *soft* version of beta_nov15 or beta_nov16 energy function.

3.2 EasyE compared to other methods

On our full dataset, we compared the prediction performance of EasyE to predict binding energy change upon mutations with two approaches, FoldX and GOBLIN (Table 1 and Supplementary Table S4). FoldX uses an implicit entropy term in its energy function while GOBLIN explicitly accounts for rotamer conformational entropy through unguaranteed approximate partition function estimation by LBP. Notice that the results of FoldX and GOBLIN presented here cannot be compared with results in (Dourado and Flores, 2014, 2016; Guerois et al., 2002; Kamisetty et al., 2011; Li et al., 2016; Xiong et al., 2017) which included training on dataset of changes in protein stability or protein–protein interaction and excluded outliers (which cannot be identified *a priori* in usual conditions, see Supplementary Tables S9 and S12–S14 for results with 10% outliers removed).

Table 1 presents the average over all systems of the averaged AUC, correlation and RMSE computed for each system. EasyE using either the beta_nov15_*soft* or beta_nov16_*soft* energy function performs better than FoldX and GOBLIN in predicting quantitative values of binding energy change upon mutation in terms of correlation coefficient and AUC. We detail below the performances of EasyE using the beta_nov16_*soft* energy function compared to those of FoldX and GOBLIN.

The correlation obtained with EasyE over all systems is 0.41. It decreases to 0.24 for FoldX and drops to 0.18 for GOBLIN. EasyE shows a better correlation with $\Delta\Delta G^e$ than FoldX and GOBLIN on respectively 16 and 18 of the overall set of 21 systems (Supplementary Table S4). With EasyE, the correlation is ≥ 0.50 for 10 over 21 systems while FoldX and GOBLIN reach the same level of performance for only 5 and 3 systems respectively. Among the systems with correlation < 0.50 with EasyE, 2 have correlation close to 0.50 and only two are negative against 4 for GOBLIN and FoldX.

The AUC for EasyE over all systems is 70% (corresponding per system ROC curves are available in Supplementary Fig. S1). It decreases to 58% with FoldX and to 65% with GOBLIN. EasyE

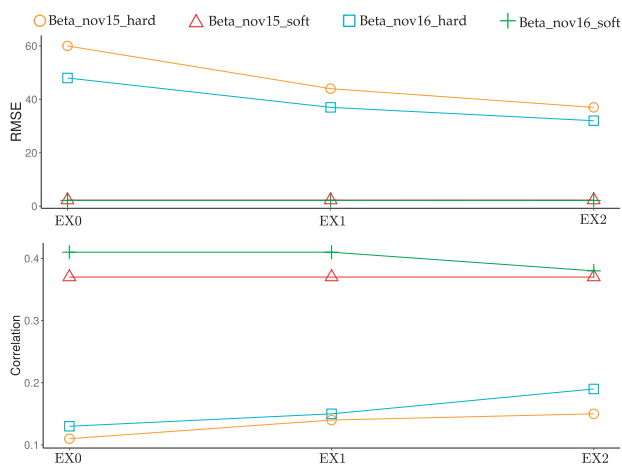


Fig. 2. Comparison of $\Delta\Delta G^e$ and $\Delta\Delta E$ predicted by the EasyE using beta_nov15_*hard* (circle), beta_nov15_*soft* (triangle), beta_nov16_*hard* (square) or beta_nov16_*soft* (cross) and three levels of rotamer sampling (EX0, EX1 or EX2). Average on the 21 systems of the averaged Root Mean Square Error by system (RMSE in kcal.mol⁻¹, top) and of the averaged Pearson Correlation coefficient by system (bottom)

Table 1. Comparison between EasyE, FoldX and GOBLIN methods: the average over all systems of the per-system averaged Pearson correlation coefficient, Area Under the Curve (AUC in %) of the ROC and Root Mean Square Error (RMSE in kcal.mol⁻¹) are given

		Correlation	AUC	RMSE
EasyE	beta_nov15_ <i>soft</i>	0.37	69	2.28
method	beta_nov16_ <i>soft</i>	0.41	70	2.18
FoldX		0.24	58	1.94
GOBLIN		0.18	65	2.20

Note: These three measures capture different desirable properties of the predictors. The RMSE measures how close to experimental values the predicted values are. The correlation coefficient does the same but after a possible ‘shift and scale’ correction of the predicted values. Finally, the AUC shows how reliably the predictor detects stabilizing or destabilizing mutations.

shows a higher AUC than FoldX and GOBLIN on respectively 14 and 12 of the 21 systems (Supplementary Table S4). The AUC is > 50% for 19 of the 21 systems with EasyE and for 13 systems with FoldX and 17 systems with GOBLIN. With EasyE, the worst AUC (31%) was obtained for the largest complex (1ahw). The AUC of FoldX and GOBLIN on this system is also low, 31 and 42% respectively. Moreover, for the 13 systems with more than 20 mutants, the *P*-value remains below 0.1% in 12, 11 and 9 cases for EasyE, FoldX and GOBLIN respectively (Supplementary Table S9).

EasyE shows an average RMSE (over all systems) of the experimental versus predicted values of 2.18 kcal mol⁻¹, slightly larger than FoldX (1.94 kcal mol⁻¹) and marginally better than GOBLIN (2.20 kcal mol⁻¹).

Additionally, BindProfX and mCSM were evaluated each on the specific subset adapted to their intrinsic limitations (Supplementary Tables S5, S6, S10 and S11, Supplementary Fig. S1). While BindProfX provides worse results than EasyE on all measures, mCSM shows a better correlation (*R* = 0.51 versus 0.48) and RMSE (1.03 versus 2.10) than EasyE and the same AUC of 74% over this subset. It is important to remember that mCSM is a machine-learning based prediction system which has been trained on Skempi (≈80% of our dataset). An inflated performance on the training set is expected. We therefore evaluated mCSM on a subset excluding Skempi systems (leaving 7 systems). On this subset, the two prediction tools have incomparable performances with a better RMSE for mCSM (1.18 versus 1.60) but a better correlation coefficient (0.31 versus 0.13) and AUC (0.70 versus 0.65) for EasyE (see Supplementary Table S6 for details). Besides the fact that it avoids training altogether, EasyE also has the advantage that it can handle multiple simultaneous mutations.

Eventually, we analyzed the influence of the change in side-chain size upon mutation (Supplementary Figs S2–S4, Table S15) for all methods on the four systems with the largest number of mutations (all from Skempi, other systems do not contain a sufficient variety of mutations to be informative). It is clear that the lack of backbone flexibility makes mutations to large side-chain more challenging to analyze with a fixed backbone approach such as in EasyE. An ensemble approach using a set of perturbed backbones could possibly alleviate this weakness, with an increased complexity scaling linearly with the size of the ensemble. This does not affect mCSM performance to the same extent, but given that the four systems were all present in the training set of mCSM, it's hard to conclude that such performance would generalize to other systems.

3.3 The JayZ approach

We evaluated the performances of JayZ using the four energy functions beta_nov15 and beta_nov16 in their *hard* and *soft* variants and EX0 rotamer sampling, for predicting free binding energy change upon mutations on our benchmark set. Out of the 21 systems, JayZ using the *soft* version of the energy functions failed to complete within the 48h time-out per partition function computation for 9 systems for beta_nov15 and 11 systems for beta_nov16 (Supplementary Table S8). The size of search space here can reach 10¹⁰³, with systems including up to 80 tightly interacting variables (with a median degree of 16 for the denser problem, see Supplementary Table S8). The *hard* version of these energy functions defines problems which are computationally much easier to solve.

We compared the $\Delta\Delta G^c$ predicted by JayZ with the $\Delta\Delta E$ computed by EasyE. Surprisingly, the two scores behave very similarly in most cases. Figure 3 shows this on the 3sgb system: the $\Delta\Delta G^c$ score

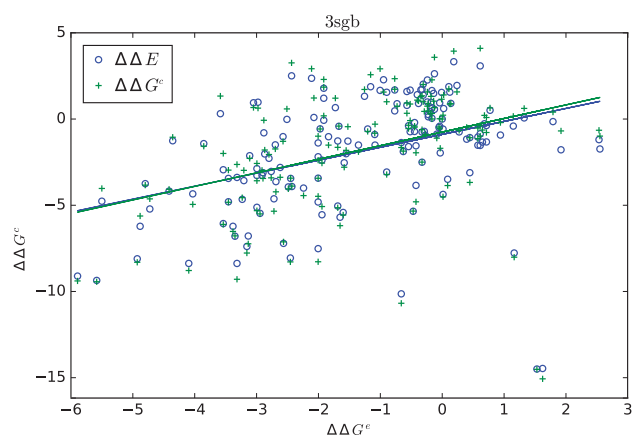


Fig. 3. Scatter plot of $\Delta\Delta G^c$ versus the $\Delta\Delta E$ (circle) and $\Delta\Delta G^c$ scores (cross) with linear regression for 3sgb

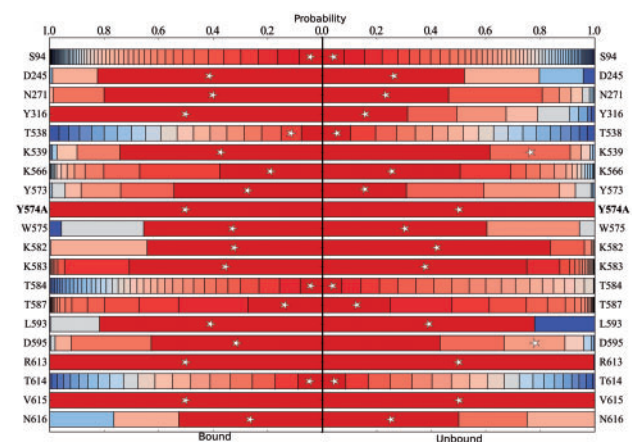


Fig. 4. Marginal distribution of rotamer probabilities on all flexible residues for the bound (left) and unbound (right) mutant Y574A of 1ahw. GMCE rotamers are indicated by a white star

looks essentially like a slightly perturbed version of the $\Delta\Delta E$ score. Side-chain conformational entropic effects remain extremely small. A linear regression of the two scores leads to a correlation coefficient of 0.99. The only exception is observed for the largest system (1ahw) for which JayZ estimation gives a positive correlation coefficient, clearly better than the negative coefficient of EasyE estimation (from -0.18 to 0.65 for beta_nov15_soft and from -0.08 to 0.32 beta_nov16_soft).

By combining the ability of CFN algorithms to enumerate all conformations within a threshold of the optimum (Traoré et al., 2013) and the knowledge of the partition function provided by Z^* , it becomes possible to estimate the marginal probability distribution over rotamers at each residue. Using HBFs, we enumerated enough conformations to cover between 80 and 90% of the partition functions of the Y574A mutant of 1ahw, a worst case for ΔE prediction. Although Z^* took only 5 min to compute the partition functions, enumerations took a bit more than 5 days on the same machine, showing the importance of on-the-fly sum-product variable elimination in Z^* .

Figure 4 shows these marginal probabilities for the bound and unbound states. Optimal rotamers predicted by EasyE are indicated with a white star. For most residues, the optimal rotamer represents the largest similar probability mass in both states, bound and

unbound. This is not true however for few residues, more notably for K539, D595 or Y316. For these residues, the optimal rotamer represents a smaller fraction in the unbound form than in the bound state. Additionally, for K539 and D595 residues, in the unbound state, the largest probability of mass is not represented by the optimal rotamer predicted by EasyE. In our benchmark set, such examples are however rare.

4 Conclusion

By relying on recently introduced guaranteed computational methods for NP-hard optimization and #P-complete discrete integration on graphical models, we have shown that it is now possible to design structure-based affinity estimation methods that remove all the uncertainty that algorithms with no guarantee may introduce, while preserving the capability of dealing with large systems. In numerous cases, it also becomes possible to explicitly account for side-chain conformational entropy.

Our evaluation on a large benchmark shows that, even in highly realistic conditions, with no outlier exclusion of extra parameter fitting, a pure guaranteed energetic approach, with no explicit side-chain conformational entropy computation, performs better than existing established methods. Amazingly, the use of guaranteed optimization techniques on NP-hard energy optimization problems can be performed on problems with several hundreds of flexible residues at very small computational cost.

In rare cases, accounting for side-chain conformational entropy allows to improve predictions, at non-negligible computational costs. All the computational methods used in this paper being guaranteed, it is important to combine their capabilities while accounting for continuous flexibility in their side-chains and their backbone. While progress in this direction has recently been made (Hallen et al., 2013, 2015), the size of problems that can be handled guaranteedly still needs to be improved.

Acknowledgements

This work relied on the HPC resources of the Computing Center of Region Midi-Pyrénées (CALMIP, Toulouse, France) and the GenoToul Bioinformatics Platform of INRA-Toulouse.

We sincerely thank the RosettaCommons/PyRosetta teams for making PyRosetta available. This work could not have been done without it.

Funding

The PhD thesis of Clément Viricel was funded by the French ‘Région Occitanie and by INRA.

Conflict of Interest: none declared.

References

Alford,R.F. et al. (2017) The rosetta all-atom energy function for macromolecular modeling and design. *J. Chem. Theory Comput.*, **13**, 3031–3048.

Allouche,D. et al. (2014) Computational protein design as an optimization problem. *Artif. Intell.*, **212**, 59–79.

Allouche,D. et al. (2015) Anytime hybrid best-first search with tree decomposition for weighted CSP. In: *International Conference on Principles and Practice of Constraint Programming*. Springer, pp. 12–29.

Åqvist,J. et al. (2002) Ligand binding affinities from MD simulations. *Accounts Chem. Res.*, **35**, 358–365.

Berman,H.M. et al. (2000) The protein data bank. *Nucleic Acids Res.*, **28**, 235–242.

Beveridge,D.L. and DiCapua,F. (1989) Free energy via molecular simulation: applications to chemical and biomolecular systems. *Annu. Rev. Biophys. Biophys. Chem.*, **18**, 431–492.

Brender,J.R. and Zhang,Y. (2015) Predicting the effect of mutations on protein–protein binding interactions through structure-based interface profiles. *PLoS Comput. Biol.*, **11**, e1004494.

Cabon,B. et al. (1999) Radio link frequency assignment. *Constraints J.*, **4**, 79–89.

Chaudhury,S. et al. (2010) Pyrosetta: a script-based interface for implementing molecular modeling algorithms using rosetta. *Bioinformatics*, **26**, 689–691.

Chavira,M. and Darwiche,A. (2008) On probabilistic inference by weighted model counting. *Artif. Intell.*, **172**, 772–799.

Chipot,C. (2014) Frontiers in free-energy calculations of biological systems. *Wiley Interdisc. Rev. Comput. Mol. Sci.*, **4**, 71–89.

Cooper,M. et al. (2010) Soft arc consistency revisited. *Artif. Intell.*, **174**, 449–478.

Dehouck,Y. et al. (2013) Beatmusic: prediction of changes in protein–protein binding affinity on mutations. *Nucleic Acids Res.*, **41**, W333–W339.

Dixit,S.B. and Chipot,C. (2001) Can absolute free energies of association be estimated from molecular mechanical simulations? the biotin–streptavidin system revisited. *J. Phys. Chem. A*, **105**, 9795–9799.

Dourado,D.F. and Flores,S.C. (2014) A multiscale approach to predicting affinity changes in protein–protein interfaces. *Proteins Struct. Funct. Bioinf.*, **82**, 2681–2690.

Dourado,D.F. and Flores,S.C. (2016) Modeling and fitting protein–protein complexes to predict change of binding energy. *Sci. Rep.*, **6**, 25406.

Georgiev,I. et al. (2008) 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. *J. Comput. Chem.*, **29**, 1527–1542.

Goldstein,R.F. (1994) Efficient rotamer elimination applied to protein side-chains and related spin glasses. *Biophys.J.*, **66**, 1335–1340.

Guerois,R. et al. (2002) Predicting changes in the stability of proteins and protein complexes: a study of more than 1000 mutations. *J. Mol. Biol.*, **320**, 369–387.

Haidar,J.N. et al. (2009) Structure-based design of a t-cell receptor leads to nearly 100-fold improvement in binding affinity for pepmhc. *Proteins Struct. Funct. Bioinf.*, **74**, 948–960.

Hallen,M.A. et al. (2013) Dead-end elimination with perturbations (DEEPer): a provable protein design algorithm with continuous sidechain and backbone flexibility. *Proteins Struct. Funct. Bioinf.*, **81**, 18–39.

Hallen,M.A. et al. (2015) Compact representation of continuous energy surfaces for more efficient protein design. *J. Chem. Theory Comput.*, **11**, 2292–2306.

Hurley,B. et al. (2016) Multi-language evaluation of exact solvers in graphical model discrete optimization. *Constraints*, **21**, 413–434.

Jemimah,S. et al. (2017) Proximate: a database of mutant protein–protein complex thermodynamics and kinetics. *Bioinformatics*, **33**, 2787–2788.

Kamisetty,H. et al. (2011) Accounting for conformational entropy in predicting binding free energies of protein–protein interactions. *Proteins Struct. Funct. Bioinf.*, **79**, 444–462.

Kellogg,E.H. et al. (2011) Role of conformational sampling in computing mutation-induced changes in protein structure and stability. *Proteins Struct. Funct. Bioinf.*, **79**, 830–838.

Kiel,C. and Serrano,L. (2014) Structure-energy-based predictions and network modelling of rasopathy and cancer missense mutations. *Mol. Syst. Biol.*, **10**, 727.

Koller,D. and Friedman,N. (2009) *Probabilistic Graphical Models: Principles and Techniques*. The MIT Press Cambridge, Massachusetts London, England.

Kortemme,T. and Baker,D. (2002) A simple physical model for binding energy hot spots in protein–protein complexes. *Proc. Natl. Acad. Sci. USA*, **99**, 14116–14121.

Larrosa,J. (2000) Boosting search with variable elimination. In: Dechter R. (ed) *Principles and Practice of Constraint Programming – CP 2000*. CP 2000, Lecture Notes in Computer Science, vol. 1894. Springer, Berlin, Heidelberg.

Larrosa,J. et al. (2005) Existential arc consistency: getting closer to full arc consistency in weighted CSPs. In: *Proc. of the 19th IJCAI*, Edinburgh, Scotland, pp. 84–89.

- Leach,A.R. and Lemon,A.P. (1998) Exploring the conformational space of protein side chains using dead-end elimination and the A* algorithm. *Proteins*, **33**, 227–239.
- Li,M. *et al.* (2016) Mutabind estimates and interprets the effects of sequence variants on protein–protein interactions. *Nucleic Acids Res.*, **44**, 494–501.
- Lilien,R.H. *et al.* (2005) A novel ensemble-based scoring and search algorithm for protein redesign and its application to modify the substrate specificity of the gramicidin synthetase a phenylalanine adenylation enzyme. *J. Comput. Biol.*, **12**, 740–761.
- Lu,W. *et al.* (1997) Binding of amino acid side-chains to s 1 cavities of serine proteinases. *J. Mol. Biol.*, **266**, 441–461.
- Moal,I.H. and Fernández-Recio,J. (2012) Skempi: a structural kinetic and energetic database of mutant protein interactions and its use in empirical models. *Bioinformatics*, **28**, 2600–2607.
- Ojewole,A.A. *et al.* (2017) Bbk^{*}(branch and bound over k^{*}): A provable and efficient ensemble-based algorithm to optimize stability and binding affinity over large sequence spaces. In: *International Conference on Research in Computational Molecular Biology*. Springer, pp. 157–172.
- Park,H. *et al.* (2016) Simultaneous optimization of biomolecular energy functions on features from small molecules and macromolecules. *J. Chem. Theory Comput.*, **12**, 6201–6212. PMID: 27766851.
- Pierce,B.G. *et al.* (2014) Computational design of the affinity and specificity of a therapeutic t cell receptor. *PLoS Comput. Biol.*, **10**, e1003478.
- Pierce,N.A. and Winfree,E. (2002) Protein design is np-hard. *Protein Eng.*, **15**, 779–782.
- Pires,D.E. *et al.* (2014) mcsm: predicting the effects of mutations in proteins using graph-based signatures. *Bioinformatics*, **30**, 335–342.
- Sang,T. *et al.* (2005) Solving Bayesian networks by weighted model counting. In: *Proceedings of the Twentieth National Conference on Artificial Intelligence (AAAI-05)*, Vol. 1, pp. 475–482.
- Schiex,T. (2000) Arc consistency for soft constraints. In: *Principles and Practice of Constraint Programming – CP 2000, Volume 1894 of LNCS*, Singapore, pp. 411–424.
- Schiex,T. *et al.* (1995) Valued constraint satisfaction problems: hard and easy problems. *IJCAI*, **95**, 631–639.
- Schymkowitz,J. *et al.* (2005) The FoldX web server: an online force field. *Nucleic Acids Res.*, **33**, W382–W388.
- Sciretti,D. *et al.* (2009) Computational protein design with side-chain conformational entropy. *Proteins Struct. Funct. Bioinf.*, **74**, 176–191.
- Shapovalov,M.V. and Dunbrack,R.L. (2011) A smoothed backbone-dependent rotamer library for proteins derived from adaptive kernel density estimates and regressions. *Structure*, **19**, 844–858.
- Silver,N.W. *et al.* (2013) Efficient computation of small-molecule configurational binding entropy and free energy changes by ensemble enumeration. *J. Chem. Theory Comput.*, **9**, 5098–5115.
- Simoncini,D. *et al.* (2015) Guaranteed discrete energy optimization on large protein design problems. *J. Chem. Theory Comput.*, **11**, 5980–5989.
- Thébault,P. *et al.* (2006) Searching RNA motifs and their intermolecular contacts with constraint networks. *Bioinformatics*, **22**, 2074–2080.
- Tinberg,C.E. *et al.* (2013) Computational design of ligand-binding proteins with high affinity and selectivity. *Nature*, **501**, 212–216.
- Toda,S. (1989) On the computational power of PP and $\oplus P$. In: *30th Annual Symposium on Foundations of Computer Science, 1989*. IEEE, pp. 514–519.
- Traoré,S. *et al.* (2013) A new framework for computational protein design through cost function network optimization. *Bioinformatics*, **29**, 2129–2136.
- Traoré,S. *et al.* (2016) Fast search algorithms for computational protein design. *J Comput Chem.*, **37**, 1048–1058.
- Valiant,L.G. (1979) The complexity of computing the permanent. *Theor. Comput. Sci.*, **8**, 189–201.
- Viricel,C. *et al.* (2015) Approximate counting with deterministic guarantees for affinity computation. In: *Modelling, Computation and Optimization in Information Systems and Management Sciences*. Springer, pp. 165–176.
- Viricel,C. *et al.* (2016) Guaranteed weighted counting for affinity computation: Beyond determinism and structure. In: *International Conference on Principles and Practice of Constraint Programming*. Springer, pp. 733–750.
- Xiong,P. *et al.* (2017) BindProfX: assessing mutation-induced binding affinity change by protein interface profiles with pseudo-counts. *J. Mol. Biol.*, **429**, 426–434.
- Zytnicki,M. *et al.* (2008) Darn! a weighted constraint solver for RNA motif localization. *Constraints*, **13**, 91–109.