BIOINFORMATICS ORIGINAL PAPER Vol. 25 no. 19 2009, pages 2537–2543
doi:10.1093/bioinformatics/btp445

Structural bioinformatics

Fast and accurate predictions of protein stability changes upon mutations using statistical potentials and neural networks: PoPMuSiC-2.0

Yves Dehouck[1,*], Aline Grosfils[2], Benjamin Folch[1], Dimitri Gilis[1], Philippe Bogaerts[2] and Marianne Rooman[1]

[1]Bioinformatique génomique et structurale and [2]Modélisation et contrôle de bioprocédés, Université Libre de Bruxelles. Av Fr. Roosevelt 50, CP165/61, 1050 Brussels, Belgium

Received on March 18, 2009; revised on July 7, 2009; accepted on July 15, 2009

Advance Access publication August 3, 2009

Associate Editor: Anna Tramontano

ABSTRACT

Motivation: The rational design of proteins with modified properties, through amino acid substitutions, is of crucial importance in a large variety of applications. Given the huge number of possible substitutions, every protein engineering project would benefit strongly from the guidance of *in silico* methods able to predict rapidly, and with reasonable accuracy, the stability changes resulting from all possible mutations in a protein.

Results: We exploit newly developed statistical potentials, based on a formalism that highlights the coupling between four protein sequence and structure descriptors, and take into account the amino acid volume variation upon mutation. The stability change is expressed as a linear combination of these energy functions, whose proportionality coefficients vary with the solvent accessibility of the mutated residue and are identified with the help of a neural network. A correlation coefficient of $R = 0.63$ and a root mean square error of $\sigma_c = 1.15$ kcal/mol between measured and predicted stability changes are obtained upon cross-validation. These scores reach $R = 0.79$, and $\sigma_c = 0.86$ kcal/mol after exclusion of 10% outliers. The predictive power of our method is shown to be significantly higher than that of other programs described in the literature.

Availability: http://babylone.ulb.ac.be/popmusic

Contact: ydehouck@ulb.ac.be

Supplementary information: Supplementary data are available at *Bioinformatics* online.

1 INTRODUCTION

The ability to design more stable and robust proteins, that maintain their activity under unusual pH or temperature conditions, would be a precious asset in many industrial sectors. It can also be interesting to tune numerous other physicochemical or biological properties of proteins, such as for example their solubility or their specificity to various ligands. The considered mutations should not alter the stability too much, for they may otherwise cause the loss of the protein's structure and function. The experimental determination of the change in folding free energy upon mutation leads to the most reliable information, but is costly and time consuming, and therefore not adapted to explore extensively the mutational space of a protein. Predictive methods, capable of estimating rapidly the stability changes resulting from all possible mutations, are thus to become unavoidable in the implementation of any protein engineering project. The development of such methods also bears a considerable interest at the fundamental level, as it participates to a better understanding of the relationships between the sequence, structure, stability and function of proteins.

The earliest theoretical approaches for predicting stability changes caused by mutations were focused on a small number of mutations in a single protein, usually even at a single site (e.g. Basch *et al*., 1987; Eriksson *et al*., 1992; Miyazawa and Jernigan, 1994; Tidor and Karplus, 1991; van Gunsteren and Mark, 1992). The PoPMuSiC program (Gilis and Rooman, 2000; Kwasigroch *et al*., 2002) was the first to be general enough for predicting stability changes caused by mutations at any point in any protein. It relies on knowledge-based potentials, based on distances between residue pairs or backbone torsion angles. Since then, several other predictive methods have been developed. Some are also based on statistical potentials (Parthiban *et al*., 2006; Zhou and Zhou, 2002), while others rely on empirical potentials describing physically the interactions that contribute to protein stability (Guerois *et al*., 2002; Yin *et al*., 2007). Different types of machine learning tools, mostly support vector machines, have recently also been used in this context (Capriotti *et al*., 2005; Cheng *et al*., 2006; Masso and Vaisman, 2008). Note that all the above-mentioned methods require the structure of the target protein. Machine learning tools were also used to develop methods predicting stability changes in proteins of unknown structure, with reasonable but obviously more limited performances (Capriotti *et al*., 2005; Cheng *et al*., 2006; Huang *et al*., 2007).

When designing the first version of PoPMuSiC, it was shown that the prediction of stability changes is much more efficient when different energy functions are used for mutations in the core and on the surface of proteins (Gilis and Rooman, 1997). This highlights the fact that different types of interactions dominate in different protein environments. Based on this result, stability changes were estimated by linear combinations of torsion and distance potentials, whose coefficients were defined independently in three domains of solvent accessibility (A) of the mutated residue.

*To whom correspondence should be addressed.

Here, we keep the same basic idea, but design a completely new energy function. We use a whole new set of 24 statistical potentials, as well as terms modeling the volume changes upon mutation, and express the folding free energy change as a single linear combination of these terms, with weighting coefficients that depend on the solvent accessibility. To reproduce the smooth transition between the core and the surface of proteins, and to generalize the step functions used in the original version, we assume that these weighting coefficients have a sigmoid shape. The parameters defining these sigmoid functions are identified with the help of a neural network, which is also an important novelty of our algorithm, PoPMuSiC-2.0.

2 METHODS

2.1 Database of experimentally characterized mutations

We extracted, from the ProTherm database (Bava *et al*., 2004), a set of mutations whose impact on the stability of the protein structure has been measured experimentally. These data were carefully checked by consulting the original articles and a number of erroneous inputs had to be corrected or removed. We considered only single-site mutations, in globular proteins whose experimental structure (either X-ray or NMR) is available. We excluded mutations introduced in pseudo wild-type constructs, as well as mutations in heme-proteins (except if the measurements were performed on the apo form of the protein, and the structure of this apo form is available). Mutations that destabilize the structure by more than 5 kcal/mol and mutations involving a proline were not considered, as they are likely to induce significant structural modifications, which are not modeled by PoPMuSiC. In the case of homo-multimeric proteins, the measured free energy changes may correspond either to the whole protein or to a monomer only. Mutations for which this information could not be retrieved from the original papers were eliminated. All remaining values were adapted to correspond to the folding free energy change per mole of monomer.

With these criteria, 2648 different point mutations, in 131 proteins, were selected. For some mutants, several measures of the variation in folding free energy ($\Delta\Delta G$) have been performed, sometimes in different conditions. To avoid any redundancy in our database, we considered only one value per mutant, which is noted $\Delta\Delta G_{\mathrm{M}}$ and is defined as an average of all available measured $\Delta\Delta G$ values. Measurements performed with a pH close to 7, a temperature close to 25°C, and without additives, are given a higher weight in the averaging procedure. The complete database and details on this averaging procedure are given as Supplementary Material.

2.2 Simplified protein representation

The sequence of a protein is described by the nature of the amino acid at each position i, noted s_i. Its three-dimensional structure is represented by several descriptors. First, the conformation of a residue at position i is defined by its backbone torsion angles, t_i. Seven discrete domains of t are considered, grouping specific local organizations of the protein chain (Rooman *et al*., 1991). Secondly, the spatial distance between the two amino acids at positions i and j, is referred to as d_{ij}. The distance values are computed between the average geometric centers of the side chains and distributed in bins of 0.2 Å width, from 3.0 to 8.0 Å (Dehouck *et al*., 2006). A last bin is added for d_{ij}'s below 3.0 Å. When d_{ij} exceeds 8.0 Å, we consider that the two residues do not interact, and no energy is computed. Finally, the solvent-accessibility of the amino acid at position i, a_i, is defined as the ratio of its solvent-accessible surface in the considered structure, as computed by DSSP (Kabsch and Sander, 1983), and in an extended tripeptide Gly-X-Gly (Rose *et al*., 1985). Five discrete values of a are considered: 0–5, 5–15, 15–30, 30–50 and 50–100%.

Table 1. Selection of statistical potentials

Group of potentials	Potentials included in the group
G_1: Basic potentials	Local: ΔW_{st}, ΔW_{as} Non-local: ΔW_{sd}, ΔW_{sds}
G_2: Low-order coupling terms	Local: ΔW_{stt}, ΔW_{sst}, ΔW_{aas}, ΔW_{ass}, ΔW_{ast} Non-local: ΔW_{asd}, ΔW_{std}, $\Delta W_{\mathrm{asdas}}$, $\Delta W_{\mathrm{stdst}}$
G_3: High-order coupling terms	Local: ΔW_{sttt}, ΔW_{sstt}, ΔW_{ssst}, ΔW_{aaas}, ΔW_{aass}, ΔW_{asss}, ΔW_{aast}, ΔW_{asst}, ΔW_{astt} Non-local: ΔW_{astd}, $\Delta W_{\mathrm{astdast}}$
G_4: Volume terms	ΔV_+, ΔV_-
G_5: Independent term	1

2.3 Estimation of the folding free energy changes

In order to describe the different interactions contributing to protein stability, we rely on a set of statistical potentials, extracted from a database of known protein structures. A form commonly used for such potentials is

$$\Delta W(c_1,c_2) = -kT\log\frac{P(c_1,c_2)}{P(c_1)P(c_2)}, \tag{1}$$

where c_1 and c_2 are sequence or structure descriptors (i.e. s_i, t_i, a_i, or d_{ij}) of the same amino acids or of neighboring ones, and P are their relative frequencies of occurrence in a large, non-redundant, dataset of protein structures. Some of us previously generalized this relationship to derive complex potentials describing the correlations between more than two descriptors, while ensuring that each contribution is counted only once (Dehouck *et al*., 2006):

$$\Delta W(c_1,c_2,\ldots,c_n) = -kT\log \prod_{k=n,n-2,n-4,\ldots} \frac{\prod_{\substack{i_1,\ldots,i_k=1\\ i_1<\cdots<i_k}}^{n} P(c_{i_1},c_{i_2},\ldots,c_{i_k})}{\prod_{\substack{i_1,\ldots,i_{k-1}=1\\ i_1<\cdots<i_{k-1}}}^{n} P(c_{i_1},c_{i_2},\ldots,c_{i_{k-1}})}, \tag{2}$$

where n is the number of descriptors. In this work, we use 24 different potentials, with n ranging from 2 to 7; they are listed in Table 1, and grouped in several subsets G_i according to their complexity. They can be divided into two major classes: local and non-local potentials, which describe the correlations between descriptors attached to residues close to each other along the sequence, or close to each other in space, respectively. For example, $\Delta W_{\mathrm{st}} \equiv \Delta W(s_i,t_j)$ reflects the influence of the nature of an amino acid (s_i) on the conformation (t_j) of a neighboring residue, while $\Delta W_{\mathrm{std}} \equiv \Delta W(s_i,t_i,d_{ij})$ reflects the propensity of an amino acid s_i, in a conformation t_i, to be separated from another amino acid (whatever its nature and conformation) by a spatial distance d_{ij}. The main difference between these potentials and those used by other prediction algorithms is that they take simultaneously into account the correlations between several sequence and structure descriptors, offering thus a more complete picture of the complex ensemble of interactions that rule protein stability.

Another parameter used to predict the mutant stability is the volume difference ΔV between the mutant and wild-type amino acids. If the mutant amino acid is smaller ($\Delta V < 0$), a cavity is created, which usually destabilizes the protein (Eriksson *et al*., 1992). On the other hand, accommodating a large side-chain ($\Delta V > 0$) may induce stresses in the structure, which are also likely to have a destabilizing impact. Statistical potentials cannot describe correctly such effects, since they are derived from a dataset of native structures of wild-type proteins, with very few packing defects. As the amplitudes of

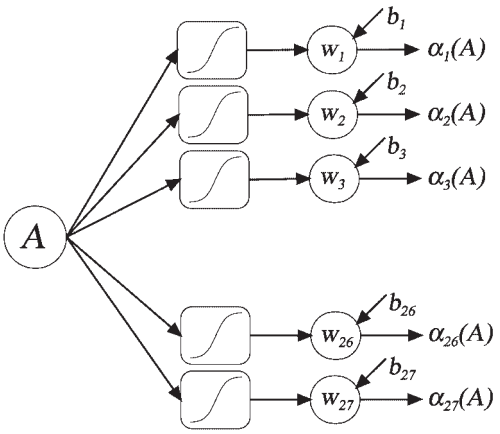

Fig. 1. Multilayer perceptron used to identify the $\alpha_i(A)$ functions.

both effects are not necessarily similar, we consider them separately, by introducing the two terms ΔV_+ and ΔV_-:

$$\Delta V_+ = \begin{cases} 0 & if\ \Delta V < 0 \\ \Delta V & if\ \Delta V \geq 0 \end{cases}, \quad \Delta V_- = \begin{cases} |\Delta V| & if\ \Delta V < 0 \\ 0 & if\ \Delta V \geq 0 \end{cases}. \tag{3}$$

The estimated stability change upon mutation, $\Delta\Delta G_P$, is expressed as a linear combination of the 26 energy functions described above plus an independent term:

$$\Delta\Delta G_{\mathrm{P}} = \sum_{i=1}^{24} \alpha_i(A)\Delta\Delta W_i + \alpha_{25}(A)\Delta V_+ + \alpha_{26}(A)\Delta V_- + \alpha_{27}(A). \tag{4}$$

As the relative weight of the different types of interactions may vary according to whether they concern residues at the protein surface or in the core (Gilis and Rooman, 1997), the 27 proportionality coefficients α_i are taken to depend on the solvent accessibility (A) of the mutated residue.

2.4 Neural network for parameter identification

To identify the 27 $\alpha_i(A)$ functions in Equation (4), a multilayer perceptron with sigmoid nodes was chosen. Sigmoid functions are well suited to our problem since the $\alpha_i(A)$'s are meant to describe the smooth transition between two different protein environments, the surface and the core of the protein. The network consists of 27 independent perceptrons with one sigmoid hidden neuron and a linear output (Fig. 1). Its mathematical expression is:

$$\alpha_i(A) = w_i \frac{1}{1+e^{-r_i(A-C_i)}} + b_i, \tag{5}$$

where C_i is the inflection point of the i-th sigmoid and r_i its slope.

This model structure is linear with respect to the weights w_i and biases b_i, but nonlinear with respect to the slopes r_i and inflection points C_i. To each set of values of the r_i and C_i parameters corresponds thus unique values of the weights w_i and biases b_i (easily computed with the function *lsqlin* in MatLab©), which minimize the mean square error on the $\Delta\Delta G$ predictions:

$$\hat{\boldsymbol{\theta}}_{wb} = \underset{\theta_{wb}}{ArgMin}\left(\sigma^2\right), \quad \text{with} \quad \sigma^2 = \frac{1}{N}\sum_{s=1}^{N}\left(\Delta\Delta G_{\mathrm{M},s} - \Delta\Delta G_{\mathrm{P},s}\right)^2, \tag{6}$$

where $\boldsymbol{\theta}_{wb}$ is a vector containing all parameters w_i and b_i (with i= 1, 2, …, 27), N the number of mutants, $\Delta\Delta G_{\mathrm{M},s}$ the experimentally measured folding free energy change of mutant s and $\Delta\Delta G_{\mathrm{P},s}$ its predicted value, which is a function of $\boldsymbol{\theta}_{wb}$ through Equations (4) and (5).

In order to estimate the parameters robustly with respect to the initial conditions, we use a *Levenberg–Marquardt* algorithm (function *lsqnonlin* in MatLab©) to identify the non-linear part of the model:

$$\hat{\boldsymbol{\theta}}_{Cr} = \underset{\theta_{Cr}}{ArgMin}\left(\sigma^2\right), \tag{7}$$

where $\boldsymbol{\theta}_{C,r}$ is a vector containing all parameters C_i and r_i (i = 1,2, …, 27). The inflection points C_i are initially fixed to 50%, the middle of the solvent accessibility range, and the initial slopes r_i are set to 1. Other initial values were tested, and turned out to have little influence on the results of the parameter estimation. At each step of the minimization, the weights and biases are deduced linearly, using Equation (6), from the C_i and r_i values.

This identification procedure allows to estimate the 108 parameters (27 C_j, 27 r_j, 27 w_i, and 27 b_i) of the vectors $\boldsymbol{\theta}_{Cr}$ and $\boldsymbol{\theta}_{wb}$. To avoid over fitting, the number of parameters is then iteratively reduced on the basis of the value of the normalized parameter covariance matrix $\mathbf{E} \equiv \mathbf{F}^{-1}$, defined as:

$$F_{ij} = \sum_{s=1}^{N}\left[\left.\frac{\partial\Delta\Delta G_{\mathrm{P,s}}}{\partial\theta_i}\right|_{\theta=\hat{\theta}} \hat{\theta}_i \left.\frac{\partial\Delta\Delta G_{\mathrm{P,s}}}{\partial\theta_j}\right|_{\theta=\hat{\theta}} \hat{\theta}_j\right], \tag{8}$$

with $\boldsymbol{\theta} = [\boldsymbol{\theta}_{bw}\ \boldsymbol{\theta}_{Cr}]$. The reduction procedure works as follows:

(1) After full identification of the model, the matrix $\mathbf{E}$ is computed.

(2) The parameter θ_i with the highest variance E_{ii} is identified, and the set of parameters θ_k whose variance E_{kk} is larger than 80% of this maximal value E_{ii} are selected. This set represents the parameters that are the least precisely identified in the model.

(3) Among this set of parameters, the parameter θ_k which has the largest covariance E_{kj} with another parameter $\theta_{j\neq k}$ is eliminated. Indeed, a large covariance indicates that the parameters are redundant, and thus that the one with the large variance can safely be eliminated. When eliminated, hardly assessable weights (w_i) and biases (b_i) are cancelled out, but slopes and inflection points are set to their initial values (r_i=1 and C_i=50). Note that the suppression of a weight w_i automatically implies the elimination of the corresponding C_i and r_i.

(4) The model is then re-identified with the remaining parameters according to the procedure described above.

The reduction procedure is then iteratively repeated to eliminate other parameters. It is ended when none of the remaining parameters has a variance larger than 10 mol^2/kcal^2 and a covariance with another parameter larger than 1 mol^2/kcal^2; these values were selected by trial and error.

3 RESULTS

3.1 Training and validation

We use a 5-fold validation procedure: the neural network is trained on 4/5 of the mutant dataset and validated on the remaining 1/5. Five different runs are performed, so that each mutant is included in one of the validation sets. The performances are assessed using the root mean square error σ [Equation (6)] and the Pearson correlation coefficient R between the measured ($\Delta\Delta G_{\mathrm{M}}$) and predicted ($\Delta\Delta G_{\mathrm{P}}$) values of the folding free energy changes. We also performed a 10-fold validation and observed no significant differences, which indicates that our dataset is large enough for 5-fold validation.

When all potentials are considered, the total number of parameters is quite large (N_{p}=85 after reduction). Still, the correlation coefficients and root mean square errors obtained on the training and validation sets are very similar (Table 2), indicating that the model is not over fitted. However, the results in Table 2 show that the potentials belonging to the group G_3 (Table 1) are not necessary and that removing them does not decrease the performances in cross-validation. This is not surprising since these are high-order coupling terms, reflecting weak correlations that are not already accounted for by the lower-order couplings. In addition, these G_3 potentials were previously shown to be very sensitive to the size of the database, and to be useful only in some specific applications (Dehouck *et al.*, 2006). In contrast, the removal of either the low-order coupling terms (G_2), the volume terms (G_4), or the independent term (G_5) leads to

Table 2. Performance in 5-fold validation

Potentials[a]	N_p^b	All mutants		Exclusion of 10% outliers	
		R_d/R_c^c	σ_d/σ_c(kcal/mol)[d]	R_d/R_c^c	σ_d/σ_c(kcal/mol)[d]
G_1,G_2,G_3,G_4,G_5	85/108	0.65/0.62	1.12/1.16	0.80/0.77	0.85/0.89
G_1, G_2, G_4, G_5	**52/64**	**0.64/0.62**	**1.13/1.16**	**0.79/0.78**	**0.85/0.88**
G_1,G_2,G_4	51/60	0.61/0.59	1.19/1.21	0.78/0.77	0.88/0.90
G_1,G_2,G_5	49/56	0.59/0.57	1.18/1.21	0.75/0.73	0.94/0.95
G_1,G_4,G_5	25/28	0.61/0.60	1.17/1.18	0.77/0.76	0.90/0.91
G_1, G_2, G_4, G_5^e	**52/64**	**–/0.63**	**–/1.15**	**–/0.79**	**–/0.86**

[a]Groups of potentials considered in the neural network. The potentials included in each group G_i are given in Table 1. The best selection of potentials is highlighted in bold, along with the corresponding results.
[b]Total number of parameters of the model, after/before reduction.
[c]Correlation coefficient between predicted and measured $\Delta\Delta G$s, in direct (R_d) and cross (R_c) validation. The reported values are averaged over five different runs.
[d]Root mean square error between predicted and measured $\Delta\Delta G$s, in direct (σ_d) and cross (σ_c) validation. The reported values are averaged over five different runs.
[e]Badly modeled mutants are removed from the training sets before parameter identification, but they are maintained in the validation sets.

a decrease in the predictive power. In summary, the best results are obtained by combining the basic potentials G_1, the low-order coupling terms G_2, the volume terms G_4, and the independent term G_5. This yields an R_c value of 0.62 and a σ_c value of 1.16 kcal/mol upon cross-validation.

Some mutations are consistently badly predicted, independently of the potentials chosen. This can of course be attributed to imperfections in our model, or to structural modifications provoked by the mutation and not taken into account by PoPMuSiC. But this can also be due to an experimental measure made in specific, non-physiological, conditions or affected by a significant error, to a poorly resolved structure, or to mistakes in the database indexing of the measured $\Delta\Delta G$ value. To account for this, we also computed the performance measures after removal of these outliers, with the aim of giving a more relevant evaluation of the predictive power. We chose here to exclude, one by one, the 10% mutants that impact most negatively the correlation coefficient. After this exclusion, we obtain a correlation coefficient R_c between predicted and measured stability changes as high as 0.78, and a root mean square error σ_c as low as 0.88 upon cross-validation (Table 2).

These outliers weaken the performance measures, but they may also have a negative impact on the identification of the model. We performed thus a second round of identification, after removal from the training sets of all mutants for which $|\Delta\Delta G_P - \Delta\Delta G_M|$ was larger than 1.5 kcal/mol in each of the five runs performed with the same set of potentials (G_1, G_2, G_4, G_5). These consistently badly predicted mutants were however maintained in the validation sets, so as to obtain comparable results. As can be seen in the last row of Table 2, the results are even better: the correlation coefficient reaches $R_c = 0.79$ and the root mean square error drops to $\sigma_c = 0.86$.

When designing our prediction method, we excluded from the dataset the mutants with a $\Delta\Delta G_M > 5$ kcal/mol, and those involving a proline, since they are likely to induce significant structural changes (Section 2.1). Yet we estimated their $\Delta\Delta G_P$ values using the parameters that were optimized without them. The root mean square error σ_c amounts to 1.33 kcal/mol for the 132 mutants involving a proline, which is slightly larger than the value of 1.15 kcal/mol obtained for the totality of other mutants (Table 2), but still very reasonable. The 41 mutants with $\Delta\Delta G_M > 5$ kcal/mol are all predicted as being strongly destabilizing, but not as much as the experiments suggest: the average $\Delta\Delta G_P$ is equal to 2.3 kcal/mol, and σ_c to 4.1 kcal/mol. Note that the values reported in Table 2 are affected by $<1\%$ when these 173 mutants are incorporated in the validation sets.

3.2 Biophysical significance of the parameters

The weight of the different energy terms in the folding free energy change $\Delta\Delta G_P$ [Equation (4)] are given by the α_i coefficients, which are functions of the solvent accessibility A. The shapes of most $\alpha_i(A)$ functions do not depend significantly on the training dataset, and are found to be well conserved in the different runs. This supports the idea that they have some biophysical meaning, and that they reflect the influence of the corresponding type of interactions at the protein surface and in the core. A few examples of the dependence of α_i on A are given in Figure 2.

For the basic potentials, included in set G_1 (Table 1), the interpretation of the α_i profiles is straightforward. The potential ΔW_{st} is computed from propensities of amino acids to be associated to torsion angle domains and thus describes interactions between amino acids close to each other in the sequence, which are at the basis of local (secondary) structure formation. As seen in Figure 2, the weight of these interactions is larger on the surface than in the core of proteins. In contrast, the weight of ΔW_{sd}, which is derived from propensities of amino acids of a given type to be close to any other amino acid and is dominated by the hydrophobic effect, is larger in the core. The value of the α_i coefficient corresponding to ΔW_{sds}, which describes specific interactions between amino acids close to each other in space, is also slightly lower on the surface. These trends are in agreement with a previous study showing that the influence of local (non-local) interactions increases (decreases) from the core to the surface (Gilis and Rooman, 1997).

The interpretation of the α_i functions corresponding to the coupling terms included in sets G_2 and G_3 is less straightforward since they are somehow correlated with each other and with the basic potentials from set G_1. Therefore, compensatory mechanisms may occur, which explain the negative values of some α_i's observed in Figure 2. The identification of the parameters of G_1 in the absence

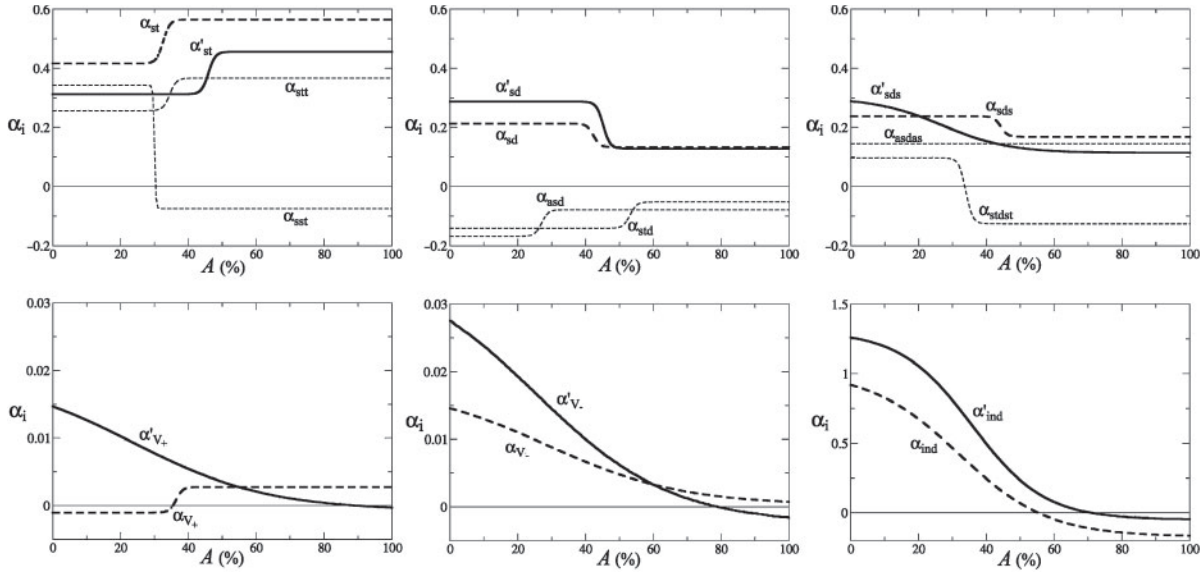

Fig. 2. Dependency of some of the weighting functions α_i on the solvent accessibility A. α'_{st}, α'_{sd} and α'_{sds} are obtained by training the network without coupling terms (G_2, G_3). α'_{V_+} and α'_{V_-} are obtained by training the network without the independent term (G_5). α'_{ind} is obtained by training the network without the volume terms (G_4). All α_i functions are dimensionless, except α_{V_+}, α_{V_-}, α'_{V_+} and α'_{V_-}, which are given in kcal/(mol Å^3), and α_{ind}, α'_{ind}, in kcal/mol.

of coupling terms from G_2 and G_3 yields the parameters noted α'_{st}, α'_{sd}, and α'_{sds}, which show almost the same profile as the original parameters α_{st}, α_{sd}, and α_{sds}. This attests that the overall shape of the α_i functions corresponding to basic potentials is modulated, but not radically modified, by the presence of coupling terms.

The α_{ind} parameter corresponding to the independent term is positive at small values of A, and tends to zero when A increases (Fig. 2). This indicates that mutations in the core are usually more destabilizing than expected, whereas this is not true for the surface mutations. This can be explained by the fact that local side chain rearrangements that limit strong destabilization are usually possible at the surface, but are much more difficult in the core. The α_{V_-} parameter corresponding to the ΔV_- term decreases as a function of the solvent accessibility, which correctly models the fact that the creation of a cavity is unfavourable in the core of a protein but not (as much) on the surface.

In contrast, the α_{V_+} profile corresponding to ΔV_+ is intriguing. It is indeed increasing as a function of increasing solvent accessibility whereas it is expected to decrease, for steric stresses resulting from the replacement of a small amino acid by a larger one should be more destabilizing in the core. This unexpected result is due to a partial compensation between the ΔV_+ contribution to the folding free-energy change $\Delta\Delta G_P$ and that of the independent term: when the latter is not included in the model, the corresponding α parameter, noted α'_{V_+}, shows the expected decreasing behavior. Note that ΔV_- is also somewhat compensated by the independent term, as can be seen from the larger values of α'_{V_-}, compared to α_{V_-}, at low A.

3.3 Comparison with other methods

Six other algorithms, previously published and available on the web, are tested against our new method for predicting stability changes upon mutation. PoPMuSiC-1.0 (Kwasigroch *et al.*, 2002) and CUPSAT (Parthiban *et al.*, 2006) rely on environment-dependent combinations of backbone torsion angle and distance potentials. The latter potentials are residue-based in PoPMuSiC-1.0 and atom-based in CUPSAT. Dmutant exploits atom-based distance potentials with a finite ideal gas reference state (Zhou and Zhou, 2002). Automute uses a four body, knowledge-based potential, combined with machine learning methodologies (Masso and Vaisman, 2008). I-mutant-2.0 is a support vector machine predictor, whose input vector consists of 42 elements, including the mutated and mutant amino acid, their structural environment, the temperature and pH (Capriotti *et al.*, 2005). Eris uses a physical force-field including contributions from van der Waals forces, solvation, hydrogen bonding and backbone-dependent statistical energies (Yin *et al.*, 2007). MuPRO (Cheng *et al.*, 2006) was not included in our analysis, since the web server predicts only the sign of the $\Delta\Delta G$.

To assess and compare the predictive power of these methods, we built a validation set by selecting all mutants that were not included in any of the databases used to devise or test these methods. Mutations in multimeric proteins were also excluded, since it is not always clear how the different prediction servers handle such mutants, and whether the predicted stability changes correspond to the multimer or to a single monomeric unit. These two criteria resulted in a dataset of 1181 mutations. As most servers require the introduction of each mutation separately, which requires a large amount of time, a subset of 350 mutants (corresponding to 67 different proteins) was randomly selected.

For each of these 350 mutations, the predicted value of $\Delta\Delta G$ was extracted from all six web servers. In the case of our novel program PoPMuSiC-2.0, the predicted values were obtained after training of the neural network on the remaining 2298 mutants of our dataset. The correlation coefficient and root mean square error between predicted and measured $\Delta\Delta G$'s are given in Table 3, for the

Table 3. Comparison of the performances of different prediction methods

Method	Number of predictions[a]	Complete set (350/309 mutants)[b]		Exclusion of 10% outliers (315/278 mutants)[b]	
		R	σ (kcal/mol)	R	σ (kcal/mol)
Automute	315	0.46/0.45	1.43/1.46	0.67/0.68	1.15/1.20
CUPSAT	346	0.37/0.35	1.91/1.96	0.62/0.61	1.31/1.33
Dmutant	350	0.48/0.47	1.81/1.87	0.74/0.74	1.17/1.21
Eris	334	0.35/0.34	4.12/4.28	0.67/0.67	2.89/2.96
I-mutant-2.0	346	0.29/0.27	1.65/1.69	0.57/0.56	1.23/1.27
PoPMuSiC-1.0	350	0.62/0.63	1.24/1.25	0.77/0.77	0.98/1.00
PoPMuSiC-2.0	350	0.67/0.67	1.16/1.19	0.81/0.81	0.94/0.96

[a]350 mutations were tested with each method. However, some servers failed to compute the $\Delta\Delta G_P$ for all mutants, resulting in a number of predictions lower than 350.

[b]Two values are given per column. The first corresponds to the whole validation set of 350 mutants (315 after exclusion of 10% outliers), with the unavailable $\Delta\Delta G_P$ values set to 0.0 kcal/mol. The second correspond to the 309 mutants (278 after exclusion of 10% outliers) for which a $\Delta\Delta G_P$ value is available for all predictors. Note that the outliers are selected independently for each method.

complete set of 350 mutants, and after removal of 10% outliers. Note that some of these web applications failed to provide predictions for all 350 mutants. In those cases, the missing $\Delta\Delta G_P$ values were set to 0.0 kcal/mol. To check that the overall comparison is not affected by these issues, we also computed the correlation coefficient and root mean square error on the 309 mutants for which a prediction was obtained from all servers (Table 3). A table including all $\Delta\Delta G_P$ values, obtained with the different methods, is available as Supplementary Material.

There is probably still some progress to be made in the prediction of stability changes resulting from mutations in proteins, as can be seen from the relatively low correlation coefficients obtained by the different methods on the 350 mutants of this blind validation test (Table 3). Indeed, R is lower than 0.5 for most methods. The highest value is 0.67, and is reached by our PoPMuSiC-2.0 program. The root mean square error is superior to 1.50 kcal/mol for most methods, and is the lowest for PoPMuSiC-2.0 (1.16 kcal/mol).

However, as discussed previously, the removal of consistently badly predicted mutants may give a more relevant view of the performances. We removed thus, for each method independently, the 10% of the mutants that lower most the correlation coefficient. Several methods then achieve a correlation coefficient close to (or higher than) 0.7 and a root mean square error close to (or lower than) 1.2 kcal/mol. We must emphasize that PoPMuSiC-2.0 shows a correlation coefficient of 0.81 and a root mean square error of 0.94 kcal/mol, and thus performs remarkably well, especially in comparison to other methods. The next best methods, as far as correlation coefficients are concerned, are PoPMuSiC-1.0 and Dmutant, with $R = 0.77$ and 0.74, respectively.

Note that the FoldX program (Guerois *et al*., 2002) was not included in this comparative test. Indeed, only the most recent—but not yet published—version (FoldX-3.0) supports mutations of small to large residues. This version is already available on a web server (http://foldx.crg.es): a correlation coefficient of 0.49 and a root mean square error of 2.16 kcal/mol were achieved on our dataset of 350 mutants ($R = 0.78$ and $\sigma = 1.07$ kcal/mol after removal of 10% outliers). FoldX-3.0 appears thus as one of the best methods after PoPMuSiC-2.0. However, the performance of FoldX-3.0 is likely to be overestimated, since we do not have access to the database of mutants used to devise the program, and no rigorous cross-validation is therefore possible.

The comparative results presented in Table 3 correspond to previously published algorithms that allow the fast prediction of mutant stability changes, on the basis of the protein structure. A few other methods were developed to carry out such predictions using the sequence only. Of course, they are not expected to perform as well as structure-based predictors, but are obviously useful in a different range of applications, when no experimental or modeled structure is available. To assess the benefits of using structural information in addition to sequence, we tested two sequence-based predictors. The sequence-only version of I-mutant-2.0 (Capriotti *et al*., 2005) achieved a correlation coefficient of 0.30 and a root mean square error of 1.70 kcal/mol on our dataset of 350 mutants ($R = 0.56$ and $\sigma = 1.27$ kcal/mol after removal of 10% outliers), while MuPRO (Cheng *et al*., 2006) achieved a correlation coefficient of 0.41 and a root mean square error of 1.44 kcal/mol ($R = 0.61$ and $\sigma = 1.20$ kcal/mol after removal of 10% outliers). Sequence-based methods appear thus as interesting but in general less performing than most structure-based methods, as expected.

4 CONCLUSIONS

Our enhanced set of statistical potentials, along with a neural network aimed at optimizing their weights as a function of the solvent-accessibility of the mutated residues, are the two basic ingredients of our new method for predicting protein stability changes upon mutation. In contrast with the original version of PoPMuSiC (Gilis and Rooman, 2000; Kwasigroch *et al*., 2002) and other methods such as CUPSAT (Parthiban *et al*., 2006), our method does not rely on discrete environmental compartments in which different energy functions are used. Rather, a single energy function, including components that vary with the environment (solvent-accessibility, secondary structure), is applied to all mutations. The high predictive power of our model, assessed by a rigorous cross-validation procedure, is illustrated by a correlation coefficient of 0.79 between the predicted and measured $\Delta\Delta G$ values, and a root mean square error of 0.86 kcal/mol, on 90% of the data. In addition, this new version of PoPMuSiC is shown to outperform

all (to our knowledge) previously published and freely available structure-based predictors, on an independent set of more than 300 mutants.

Despite the use of a neural network to optimize the weighting functions of the different potentials, our predictive model is very different from a pure black box model. Indeed, the parameters identified by the neural network define sigmoid functions describing the transition between two different environments, in which different types of interactions dominate. The overall shapes of these sigmoid functions were found to be very stable with respect to the composition of the training set, and may thus be considered as having a genuine biophysical significance. In particular, our results confirm previous observations that local interactions are more influential on the surface of the protein and tertiary interactions dominate in the core (Gilis and Rooman, 1997). They also indicate that changing the amino acid size upon mutation tends to be destabilizing, especially in the core, as this creates either stress or cavities, and that all mutations are on the average more destabilizing in the core than at the surface.

Packing defects caused by mutations in the core cannot be correctly taken into account by potentials derived from a database of wild-type proteins, since these proteins are usually very well packed. The volume terms included in our energy function, along with the independent term, provide a very coarse description of these defects but still enhance substantially the predictive power of our method. Among the further improvements that should be introduced in PoPMuSiC, one of the priorities is certainly a better description of the impact of packing defects. The modeling of possible local structural rearrangements of the side-chains or backbone upon mutation should improve the performances of our method, and allow us to consider multiple mutations in a systematic way. However, taking into account structural flexibility is usually very hungry for computing power, while the ability of our current algorithm to predict within a minute (on a contemporary desktop computer) the stability changes resulting from all possible mutations in an average-sized protein is definitely one of its advantages.

In summary, our results suggest that PoPMuSiC-2.0 is—at present—the most reliable method for predicting stability changes of mutant proteins, as compared to the other available structure-based predictors that are able to test rapidly all possible mutations in a given protein. We believe that every rational design of modified proteins would benefit from the use of PoPMuSiC-2.0 in a first stage, so as to identify a small set of mutations likely to present the desired (de)stabilization properties. These mutations are then to be studied further, either by more time-consuming computational methods, or directly by experimental means (see for example Cabrita *et al*., 2007; Gilis *et al*., 2003).

ACKNOWLEDGEMENTS

The authors thank Jean Marc Kwasigroch for his contribution to the web server.

Funding: Belgian State Science Policy Office through an Interuniversity Attraction Poles Programme (DYSCO); the Belgian Fund for Scientific Research (FRS) (FRFC project, FRIA grants to A.G. and B.F.); the Brussels Region (TheraVip project); the Walloon Region and BioXpr bioinformatics company (First-Postdoc grant to Y.D.). M.R. is Research Director at the FRS.

Conflict of Interest: none declared.

REFERENCES

Bash,P.A. *et al*. (1987) Free energy calculations by computer simulation. *Science,* **236**, 564–568.

Bava,K.A. *et al*. (2004) ProTherm, version 4.0: Thermodynamic Database for Proteins and Mutants. *Nucleic Acids Res.*, **32**, D120–D121.

Cabrita,L.D. *et al.* (2007) Enhancing the stability and solubility of TEV protease using in silico design. *Protein Sci*., **16**, 2360–2367.

Capriotti,E. *et al*. (2005) I-Mutant2.0: predicting stability changes upon mutation from the protein sequence or structure. *Nucleic Acids Res.*, **33**, W306–W310.

Cheng,J. *et al*. (2006) Prediction of protein stability changes for single-site mutations using support vector machines. *Proteins,* **62**, 1125–1132.

Dehouck,Y. *et al*. (2006) A new generation of statistical potentials for proteins, *Biophys. J.*, **90**, 4010–4017.

Eriksson,A.E. *et al*. (1992) Response of a protein structure to cavity-creating mutations and its relation to the hydrophobic effect. *Science*, **255**, 178–183.

Gilis,D. and Rooman,M. (1997) Predicting protein stability changes upon mutation using database-derived potentials: solvent accessibility determines the importance of local versus non-local interactions along the sequence. *J. Mol. Biol*., **272**, 276–290.

Gilis,D. and Rooman,M. (2000) PoPMuSiC, an algorithm for predicting protein mutant stability changes. Application to prion proteins. *Protein Eng.*, **13**, 849–856.

Gilis,D. *et al*. (2003) In vitro and in silico design of α_1-antitrypsin mutants with different conformational stabilities. *J. Mol. Biol*., **325**, 581–589.

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.

Huang,L.T. *et al*. (2007) Sequence analysis and rule development of predicting protein stability change upon mutation using decision tree model. *J. Mol. Model,* **13**, 879–890.

Kabsch,W. and Sander,C. (1983) Dictionary of protein secondary structure: pattern recognition of hydrogen-bonded and geometrical features. *Biopolymers*, **22**, 2577–2637.

Kwasigroch,J.M. *et al*. (2002) PoPMuSiC, rationally designing point mutations in protein structures. *Bioinformatics,* **18**, 1701–1702.

Masso,M. and Vaisman,I.I. (2008) Accurate prediction of stability changes in protein mutants by combining machine learning with structure based computational mutagenesis. *Bioinformatics,* **24**, 2002–2009.

Miyazawa,S., and Jernigan,L. (1994) Protein stability for single substitution mutants and the extent of local compactness in the denatured state. *Protein Eng*., **7**, 1209–1220.

Parthiban,V. *et al*. (2006) CUPSAT: prediction of protein stability upon point mutations. *Nucleic Acids Res*., **34**, W239–W242.

Rooman,M.J. *et al*. (1991) Prediction of protein backbone conformation based on seven structure assignments. Influence of local interactions, *J. Mol. Biol*., **221**, 961–979.

Rose,G.D. *et al*. (1985) Hydrophobicity of amino acid residues in globular proteins. *Science*, **29**, 834–838.

Tidor,B., and Karplus,M. (1991) Simulation analysis of the stability mutant R96H of T4 lysozyme. *Biochemistry,* **30**, 3217–3228.

van Gunsteren,W.F. and Mark,A.E. (1992) Prediction of the activity and stability effects of site-directed mutagenesis on a protein core. *J. Mol. Biol*., **227**, 389–395.

Yin,S. *et al*. (2007) Modeling backbone flexibility improves protein stability estimation. *Structure,* **15**, 1567–1576.

Zhou,H. and Zhou,Y. (2002) Distance-scaled, finite ideal-gas reference state improves structure-derived potentials of mean force for structure selection and stability prediction. *Protein Sci*., **11**, 2714–2726.