# Comparison of Two Optimization Methods to Derive Energy Parameters for Protein Folding: Perceptron and *Z* Score

**Michele Vendruscolo,[1]* Leonid A. Mirny,[2] Eugene I. Shakhnovich,[2] and Eytan Domany[3]**
[1]*Oxford Centre for Molecular Sciences, New Chemistry Laboratory, Oxford, United Kingdom*
[2]*Department of Chemistry and Chemical Biology, Harvard University, Cambridge, Massachusetts*
[3]*Department of Physics of Complex Systems, Weizmann Institute of Science, Rehovot, Israel*

***ABSTRACT*** **Two methods were proposed recently to derive energy parameters from known native protein conformations and corresponding sets of decoys. One is based on finding, by means of a perceptron learning scheme, energy parameters such that the native conformations have lower energies than the decoys. The second method maximizes the difference between the native energy and the average energy of the decoys, measured in terms of the width of the decoys' energy distribution (*Z*-score). Whereas the perceptron method is sensitive mainly to "outlier" (i.e., extremal) decoys, the *Z*-score optimization is governed by the high density regions in decoy-space. We compare the two methods by deriving contact energies for two very different sets of decoys: the first obtained for model lattice proteins and the second by threading. We find that the potentials derived by the two methods are of similar quality and fairly closely related. This finding indicates that standard, naturally occurring sets of decoys are distributed in a way that yields robust energy parameters (that are quite insensitive to the particular method used to derive them). The main practical implication of this finding is that it is not necessary to fine-tune the potential search method to the particular set of decoys used. Proteins 2000;41:192–201.** © 2000 Wiley-Liss, Inc.

**Key words: protein folding; contact maps; protein potential; perceptron; *Z*-score**

## INTRODUCTION

To perform protein folding one assigns an energy $E$ to a protein sequence in a given conformation. One of the simplest approximations to the true energy is the pairwise contact approximation

$$E^{true}(\mathbf{a}, \mathbf{S}) \simeq E^{pair}(\mathbf{a}, \mathbf{S}, \mathbf{w}) = \sum_{i<j}^{N} \mathbf{S}_{ij}\, w(a_i, a_j), \qquad (1)$$

where we denoted by $\mathbf{a}$ the sequence of amino acids, by $\mathbf{S}$ the conformation (represented by its contact map,[1] and by $\mathbf{w}$ the set of energy parameters. If there is a contact between residues $i$ and $j$, then $\mathbf{S}_{ij} = 1$ and the parameter $w(a_i, a_j)$, which represents the energy gained by bringing amino acids $a_i$ and $a_j$ in contact, is added to the energy.

Given a set of proteins whose native structure is known, experimentally or otherwise, the set $\mathbf{w}$ must stabilize these structures against all the possible alternatives.

Optimization of the stability can be realized using different methods.[2–9] In this work, we compare two recently proposed approaches to this problem, the one by Mirny and Shakhnovich (MS)[10] and that of Vendruscolo and Domany (VD).[1]

The purpose of the present study is to compare the merits and possible shortcomings of the two methods, when applied to realistic situations and data, involving either real or artificial model proteins.

In the $Z$ score method, one determines the energy parameters by optimizing the gap between the native state and the average energy of alternative conformations, measured in units of standard deviations of the energy distribution. Imagine that a set of parameters with a very low $Z$-score has been found. If the number of decoys is large, the $Z$-score will not be affected by a small number of "outlier" configurations. If we now add a few conformations whose energy is *below* that of the native structure, this will not affect in a significant way the average energy and, hence, the $Z$-score. Therefore, for this new set of decoys the parameters that optimize the $Z$-score do not assign the lowest energy to the native state. We can easily create such a situation; it is not clear at all, however, whether such a mishap will or will not occur for a routinely obtained set of decoys.

The perceptron method is aimed at enforcing the condition

$$E_0 < E_\mu \qquad (2)$$

for one or more proteins. Here $E_0$ is the energy of the native state and $E_\mu$ ($\mu = 1, \ldots P$) are the energies of $P$ alternative conformations. This is a necessary condition for any energy function to be used for protein folding by energy minimization. When there exists a set of contact energy parameters $\mathbf{w}$ for which (2) holds for all $\mu$, we say

Grant sponsor: US-Israel Binational Science Foundation (BSF); Grant sponsor: Germany-Israel Science Foundation (GIF); Grant sponsor: Minerva Foundation; Grant sponsor: European Molecular Biology Organization (EMBO); Grant sponsor: National Institute of Health (NIH); Grant number: GM52126.

*Correspondence to: Michele Vendruscolo, Oxford Centre for Molecular Sciences, New Chemistry Laboratory, University of Oxford, South Parks Road, OX1 3QT Oxford UK. E-mail: michelev@bioch.ox.ac.uk

Received 28 February 2000; Accepted 27 June 2000

that the problem is *learnable*. For a learnable problem, however, the solution is not unique: there is a region (called "version space") in energy parameter space, whose points satisfy the $P$ inequalities (2). One can identify a subset of conformations that are of low energy for at least some of the points in version space. The result obtained by perceptron learning is sensitive only to such a (possibly very small) subset of conformations. Using a different set of conformations, even generated in the same way may, in principle, change the solution considerably.

Hence the perceptron solution may be influenced very strongly by a few low-energy "outlier" conformations. Therefore, it is possible to create a situation in which the energy parameters obtained by perceptron learning do satisfy (2) and stabilize the native fold but, at the same time, yield a relatively high value for the $Z$-score. Again we wish to find out whether for realistic decoys such a situation will or will not actually occur.

## OPTIMIZATION METHODS

### *Z*-Score Method

MS presented a method to derive a potential based on the optimization of the $Z$-score. The $Z$-score is defined by

$$Z = \frac{E_0 - \langle E \rangle}{\sigma} \tag{3}$$

where $E_0$ is the energy of the native state, and $\langle E \rangle$ and $\sigma$ are, respectively, the mean and the standard deviation of the energy distribution.

The procedure that they used to recover the true potential worked by optimizing the $Z$-score simultaneously for all the sequences as a function of the energy parameters $\mathbf{w}$. Using a Monte Carlo in parameter space they minimized the harmonic mean of the $Z$-scores

$$\langle Z \rangle_{\text{harm}} = \frac{M}{\sum_{m=1}^{M} 1/Z_m} \tag{4}$$

The procedure that they used to recover the true potential worked by optimizing the mean harmonic $Z$-score simultaneously for all proteins in the database as a function of the energy parameters $\mathbf{w}$. The reason for taking the harmonic mean of $Z$-scores as a function to optimize is that the harmonic mean is most sensitive to "outliers," i.e., it is a good approximation to obtain min$(\max_m Z_m)$. Physically it means that the optimization of the harmonic mean is likely to exclude the situation when few proteins are "over-optimized" while for most other proteins the derived potential is not satisfactory at all. More detailed discussion of this point is given in Mirny and Shakhnovich.[10]

### Computation of the *Z*-Score

The mean and the standard deviation of the energy distribution of decoys are given by

$$\langle E \rangle = \sum_{i<j}^{N} \langle S_{ij} \rangle w(a_i, a_j) \tag{5}$$

$$\sigma^2(E) = \sum_{i<j}^{N} \sum_{k<l}^{N} \text{cov}(S_{ij}, S_{kl}) w(a_i, a_j) w(a_k, a_l), \tag{6}$$

where $\langle S_{ij} \rangle$ is the frequency of a contact between residues $i$ and $j$ in the decoys and $\text{cov}(S_{ij}, S_{kl}) = \langle S_{ij} S_{kl} \rangle - \langle S_{ij} \rangle \langle S_{kl} \rangle$ is covariance of contacts between $i, j$ and $k, l$. For a given set of decoys, one can easily compute $\langle S_{ij} \rangle$ and $\text{cov}(S_{ij}, S_{kl})$. Importantly, the $Z$-score method allows derivation of a potential using no explicit decoys. *Assuming* a certain form of the distribution of contacts in the decoys $\langle S_{ij} \rangle$ and their correlations $\text{cov}(S_{ij}, S_{kl})$ one can compute the $Z$-score and optimize a potential against these *implicit* decoys.

However, when the $Z$-score method is compared with the perceptron learning (see below) a set of actual decoys is always present. In this case both $\langle S_{ij} \rangle$ and $\text{cov}(S_{ij}, S_{kl})$ are computed explicitly using these decoys.

### Optimization of Potential

A potential $\mathbf{w}$ is obtained by minimization of the $Z$-scores simultaneously for all proteins in a database. As explained above, this is achieved by using a harmonic mean of individual $Z$-scores as a function to be minimized

$$\langle Z \rangle_{\text{harm}} = \frac{M}{\sum_{m=1}^{M} 1/Z_m} \tag{7}$$

At each step of the Monte Carlo procedure, an element of $\mathbf{w}$ is chosen at random and a small random number $\epsilon \in [-0.1, 0.1]$ is added to it. This change is accepted or rejected according to the associated change in $\langle Z \rangle_{\text{harm}}$ and the Metropolis criterion with algorithmic temperature $T$.[11] At low temperature $T$, the procedure rapidly converges to the low values of $\langle Z \rangle_{\text{harm}}$.

To assess the quality of obtained potential $\mathbf{w}_{\text{opt}}$ one needs to compare the energy of the native conformation $E_0(\mathbf{w}_{\text{opt}})$ with the energy of each decoy $E_\mu(\mathbf{w}_{\text{opt}})$. If the actual decoys are present the procedure is straightforward. When potential is obtained using *implicit* decoys (see above), one cannot check whether $E_0 < E_\mu$ for all $\mu$. However, it is possible to *estimate* whether $E_0$ is below $E_C$, the bottom of the continuum part of the decoy's energy spectrum. Assuming the Gaussian energy distribution of the decoys one gets

$$E_C = \langle E \rangle - \sigma \sqrt{2 \ln M},$$

where $M$ is the estimated number of decoys. The value of $M$ depends on the procedure used to generate decoys: lattice or off-lattice folding, threading, and so on. Then the quality of a potential is given by the $E_0/E_C$ ratio. If $E_0/E_C < 1$, the native conformation is above the bottom of the continuum spectrum and there are lots of decoys with $E_\mu < E_0$. This is a strong indication that the problem is *unlearnable* (see below). On the contrary, $E_0/E_C > 1$

indicates that the native conformation is below in energy than a vast majority of decoys. However, even in this case some decoys can have the energy below $E_0$.

### Perceptron Method

VD used a perceptron learning technique to find energy parameters $\mathbf{w}$ for which the set of inequalities (2) is satisfied. The perceptron learning technique they used either converges to a solution $\mathbf{w}^*$ of the inequalities (2), or provides a proof for non-existence of such a solution.

For any conformation, the condition Eq. (2) can be expressed as

$$\mathbf{w} \cdot \mathbf{x}^\mu > 0 \tag{8}$$

To see this, just note that for any map $\mathbf{S}_\mu$ the energy (1) is a linear function of the 210 contact energies that can appear and it can be written as

$$E^{pair}(\mathbf{a}, \mathbf{S}_\mu, \mathbf{w}) = \sum_{c=1}^{210} N_c(\mathbf{S}_\mu) w_c \tag{9}$$

Here the index $c = 1, 2, \ldots 210$ labels the different contacts that can appear and $N_c(\mathbf{S}_\mu)$ is the total number of contacts of type $c$ that actually appear in map $\mathbf{S}_\mu$. The difference between the energy of this map and the native $\mathbf{S}_N$ is, therefore,

$$\Delta E_\mu = \sum_{c=1}^{210} x_c^\mu w_c = \mathbf{w} \cdot \mathbf{x}_\mu \tag{10}$$

where we used the notation

$$x_c^\mu = N_c(\mathbf{S}_\mu) - N_c(\mathbf{S}_0) \tag{11}$$

and $\mathbf{S}_0$ is the native map.

Each candidate map $\mathbf{S}_\mu$ is represented by a vector $\mathbf{x}^\mu$ and, hence, the question raised above regarding stabilization of a sequence $\mathbf{a}$ becomes

*Can one find a vector* $\mathbf{w}$ *such that condition (8)*

*holds for all* $\mathbf{x}^\mu$*?*

If such a $\mathbf{w}$ exists, it can be found by *perceptron learning.*

A perceptron is the simplest neural network.[12] It is aimed to solve the following task. Given $P$ patterns (also called input vectors, examples) $\mathbf{x}^\mu$, find a vector $\mathbf{w}$ of weights, such that the condition

$$h_\mu = \mathbf{w} \cdot \mathbf{x}^\mu > 0 \tag{12}$$

is satisfied for every example from a training set of $P$ patterns, $\mathbf{x}^\mu$, $\mu = 1, \ldots, P$. If such a $\mathbf{w}$ exists for the training set, the problem is *learnable;* if not, it is unlearnable. We assume that the vector of "weights" $\mathbf{w}$ is normalized,

$$\mathbf{w} \cdot \mathbf{w} = 1 \tag{13}$$

The vector $\mathbf{w}$ is "learned" in the course of a training session. The $P$ patterns are presented cyclically; after presentation of pattern $\mu$, the weights $\mathbf{w}$ are updated according to the following learning rule:

$$\mathbf{w}' = \begin{cases} \dfrac{w + \eta x^\mu}{|w + \eta x^\mu|} & \text{if} \quad \mathbf{w} \cdot \mathbf{x}^\mu < 0 \\ \mathbf{w} & \text{otherwise} \end{cases} \tag{14}$$

This procedure is called learning since when the present $\mathbf{w}$ misses the correct "answer" $h_\mu > 0$, for example, $\mu$, all weights are modified in a manner that reduces the error. No matter what initial guess for the $\mathbf{w}$ one takes, a convergence theorem guarantees that if a solution $\mathbf{w}$ exists, it will be found in a finite number of training steps.[12,13]

For learnable problems there is a continuous set of solutions, among which one can find the optimal one, the perceptron of *maximal stability.*[8,14,15] This solution maximizes the smallest gap between the native energy and the respective first excited state of the $M$ proteins in the learning set. In the algorithm, the condition (12) is replaced by

$$h_\mu = \mathbf{w} \cdot \mathbf{x}^\mu > c \tag{15}$$

where $c$ is a positive number that should be made as large as possible. At each time step, the "worst" example $\mathbf{x}^\nu$ is identified, namely the one such that

$$h_\nu = \mathbf{w} \cdot \mathbf{x}^\nu = \min_\mu \mathbf{w} \cdot \mathbf{x}^\mu \tag{16}$$

Such an example is used to update the weights according again to the rule (14). The field $h_\nu(t)$ keeps changing at each time step $t$; the procedure is iterated until it levels off to its asymptote.

## COMPARISON OF THE METHODS USING LATTICE PROTEINS

Lattice proteins constitute a simplified paradigm that represents many aspects of the real problem quite faithfully. Because of their relative simplicity, they were used to test a wide variety of ideas on proteins, ranging from sequence design, folding dynamics, calculation of free-energy landscapes, and many more. They form a well-controlled theoretical construct about which many basic questions can be asked, without the need to involve the added complexity of real polypeptide chains.

In particular, short fully compact lattice proteins were used by MS to test the $Z$-score methodology; hence, it is natural to use the same set as a testing ground for the perceptron method and for comparing it to the results obtained by $Z$-score. The database of $M = 200$ proteins used by MS was set up as follows. They randomly chose 200 conformations on a $3 \times 3 \times 3$ cube. Using the potential of Miyazawa and Jernigan (MJ)[16] (hereafter referred to as the "true" potential), they designed for each conformation a sequence that minimized the $Z$-score as a function of the sequence composition. The design method is standard Monte-Carlo optimization in sequence space.[17,18] A version of the method that directly optimizes the $Z$-score, without the requirement of constant amino acid composition was used. In the second part of their study, they used

the 200 sequences and structures in the database to optimize the $Z$-score as a function of $\mathbf{w}$. In this way, they found a solution $\mathbf{w}_{ZL}$. The 200 structures used in this study were the same as in earlier work by Mirny and Shakhnovich[10] and the procedure of parameter derivation and its relation to "true" input parameters are described in detail.[10]

We use here 198 of the 200 MS conformations and the corresponding optimal sequences.

For each of the 198 proteins, all the 103,346 conformations on the cube were considered as decoys, yielding $P = 198 \cdot 103345 \approx 20 \cdot 10^6$ inequalities. This is a very large number of examples to learn; fortunately, as we shall see, the structure of the problem facilitates our task considerably. Only a few of the $20 \cdot 10^6$ examples $\mathbf{x}^\mu$ are relevant to the learning procedure.

As our first attempt to derive energy parameters, we used the standard perceptron learning rule.[13] The procedure is the following. We initialized the vector $\mathbf{w}$ of parameters by drawing 210 random numbers uniformly distributed in the interval $[-1, 1]$. In this way, at the start $P/2$ examples are on average violating the $P$ inequalities Eq. (12). Then we ran cyclically through the $P$ inequalities, updating the vector $\mathbf{w}$ each time a violation of an inequality was found. We derived three different solutions $\mathbf{w}_1$, $\mathbf{w}_2$, and $\mathbf{w}_3$, each one obtained by starting from a different point in parameter space. Given the low complexity of this particular learning problem, the solutions $\mathbf{w}_1$ and $\mathbf{w}_3$ were found after only 1 sweep through the $P$ examples during which 8 updates of $\mathbf{w}$ were performed and the solution $\mathbf{w}_2$ was found after two sweeps, which involved 11 and 1 updates, respectively. We observe that in this context "low complexity" has the specific meaning that to change the sign of $P/2 \sim 10^7$ inequalities only about 10 updates are typically necessary. We found that the correlation coefficients between the solutions

$$\rho_{\alpha,\beta} = \mathbf{w}_\alpha \cdot \mathbf{w}_\beta \tag{17}$$

were quite small; respectively, $\rho_{1,2} = 0.65$, $\rho_{1,3} = 0.60$, and $\rho_{2,3} = 0.57$. This information is important since it measures the size of *version space,* i.e., that part of the parameter space whose points are solutions of Eqs. (2). A random initial guess for $\mathbf{w}$ lies outside version space and it "diffuses" towards it during the learning process. As soon as $\mathbf{w}$ enters version space, the learning process defined above stops. Hence our three solutions, which were generated starting from three uncorrelated random initial guesses, represent three typical vectors close to the boundary of version space; the angle between a pair of such vectors is about 53°.

As our second learning attempt we found the perceptron $\mathbf{w}_{PL}$ of maximal stability. This solution is near the centre of version space. From the previous attempt, we understood that only a very few examples are relevant for the learning process. Giving such insight, we followed a more economic procedure than the previous one that required to sweep each time through all the $P$ examples, the vast majority of which did not contribute to the learning. For each sequence, we generated 100 "important" low-energy

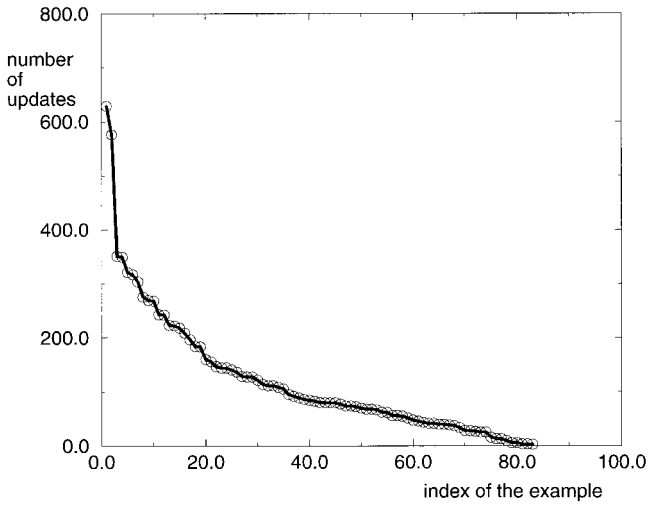


Fig. 1. Number of updates that were necessary for each example (identified in the figure by an "index") to converge to the solution of maximal stability. Only less than 100 examples participated to the learning process.

examples. One way to do this is, as before, to start from an initial random choice for the parameters $\mathbf{w}$ and to sweep once through the $P$ examples, updating $\mathbf{w}$. By using the updated $\mathbf{w}$ for each sequence, we identified the 100 examples of lowest energy. Then a second random set of parameters $\mathbf{w}$ was drawn and, again, the 100 examples of lowest energy were identified in the same way. Typically a few tens of structures are common to these two sets of 100. By taking 100 low-energy structures determined by either of the potentials, we are including the lowest 10 or so structures of any other reasonable pairwise potential function. In this way, we reduced the size of the learning task to $N_D = 19{,}800$ examples. Once these "hard" examples were learned, we turned back to the full set of $20 \cdot 10^6$ examples to ascertain that the solution obtained indeed satisfies the entire set of inequalities.

In Figure 1 we demonstrate that only less than $P^{hard} \sim 100$ examples participated in the learning process. The figure shows the number of updates for each example that were necessary to converge to the optimal solution, sorted in decreasing order. In practice, around one half of the sequences did not contribute at all to the total $P^{hard}$ and the remaining ones contributed one or very few examples. We found that the overlaps of $\mathbf{w}_{PL}$ with the three non-optimal solutions are $\rho_{P,1} = 0.74$, $\rho_{P,2} = 0.71$, and $\rho_{P,3} = 0.66$, corresponding to a smaller angle (about 45°). For $\mathbf{w}_{PL}$, the minimal gap between a native map and the lowest decoy above it is $\min_\mu \mathbf{w} \cdot \mathbf{x}^\mu = c_{PL} = 0.45$.

Next we investigated the influence of the database size on the derived potential. To this effect, we obtained new solutions $\mathbf{w}_{PM}$ using only a subset of $M$ proteins in the database. For example, for $M = 99$ proteins the correlation with the full solution $\mathbf{w}_{PL}$ is $\rho_{PL,PM} = 0.89$ and the stability $c_{PM} = 0.54$. The set $\mathbf{w}_{PM}$ is still a solution of the whole database of 198 proteins. However, the stability in the whole database is reduced to $c_{PM}^{198} = 0.035$, as shown in Figure 2. The stability as a function of $M$ apparently

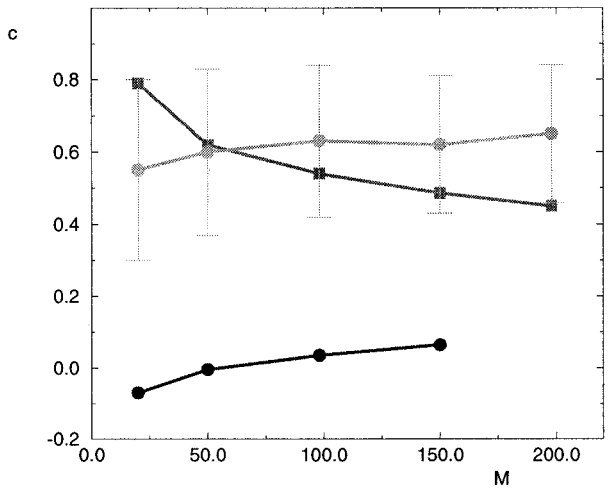


Fig. 2. Stability $c_{PM}$ as a function of the number M of proteins in the database (squares). The lower curve is the stability $c_{PM}^{198}$ in complete set of 198 proteins. We also show the stability for a single protein averaged over all the M proteins in the set (curve with error-bars).

asymptotes to a non-zero value. This fact might be regarded as a measure of the degree of design to which the proteins in the database have been subjected.

There are two questions that can be asked to compare the two methods of extraction of potentials

*Which is the best method to recover the true potential knowing only the sequences and their ground states?*

A possible answer is given by the correlation coefficient between the true potential and the derived one. The $Z$-score method gave $\rho = 0.84$ and the perceptron method $\rho = 0.69$. The correlation between $\mathbf{w}_{PL}$ and $\mathbf{w}_{ZL}$ is $\rho = 0.79$.

*Which is the method that gives a "better" potential?*

We considered four different measures of performance to answer to this question.

1. The $Z$-score measures the gap between the ground state and the average energy of a given sequence on a set of decoys. The perceptron method measures the gap between the ground state and the first excited state. Thus, the previous question can be rephrased as
   *How does* $\langle Z \rangle_{harm}$ *of* $\mathbf{w}_P$ *compare with* $\langle Z \rangle_{harm}$ *of* $\mathbf{w}_{ZL}$?

In the case of $\mathbf{w}_{PL}$, we obtained $\langle Z \rangle_{\rm harm} = -6.44$, and for $\mathbf{w}_{ZL}$ we obtained $\langle Z \rangle_{\rm harm} = -6.93$.

2. Another way to formulate the same question is
   *How does the stability* $c_{ZL}$ *of* $\mathbf{w}_{ZL}$ *compare with the stability* $c_{PL}$ *of* $\mathbf{w}_{PL}$?

We found $c_{ZL} = 0.05$ and $c_{PL} = 0.45$.

3. Beyond the $Z$-score and the gap to the first excited state, a third way to quantify the stability is to look at the correlation between the overlap $Q$ and the difference in energy with the ground state $\Delta E$. The overlap $Q$ is defined as

$$Q = \frac{N_p}{N_c} \tag{18}$$

where $N_p$ is the number of contacts present both in the native contact map and in the contact map of the decoy and $N_c$ is the number of contacts in the native contact map (contacts along the three main diagonals are not counted). A good potential should provide low energy to conformations close to the native state and high energy to those very different from it. As shown in Figure 3, $\mathbf{w}_{ZL}$ and $\mathbf{w}_{PL}$ provide approximately the same correlation. The $Z$-score method reduces the width of the distribution of the energy of the decoys, as shown in Figure 4. The perceptron, on the other hand, for large $Q$, pushes up the bottom of the energies, enlarging the gap to the ground state.

4. A fourth way to assess which is the "quality" of the recovered potential is to check whether the ground state obtained using it is, indeed, the correct ground state. For each of the 198 designed sequences, the corresponding compact structures were the ground states of the "true" MJ potential. The energy parameters obtained by the perceptron method depend on the particular set of decoys that were used. In the lattice case discussed above, we have used only maximally compact decoys. Performing a Monte Carlo energy minimization on the entire space of conformations, using the derived energy parameters $\mathbf{w}_{PL}$, we found that for 6 of the 198 sequences there were non-maximally compact conformations, whose energy was lower than the "true" ground state. Using the $Z$-score derived energy parameters, the same test gave 8 mistakes.

Finally, we observe that the database was obtained by minimizing the $Z$-score in the space of sequences at fixed conformation. The recovery of the parameter set was carried out by again minimizing the $Z$-score in the space of parameters. This procedure can introduce a bias, which complicates the comparison with the perceptron method to derive the energy parameter set.

## THREADING

We present here the results of an experiment of gapless threading, using the two methods. We considered a test set of 100 PDB proteins, for which decoys were derived by threading each sequence of every protein through the structure of all the longer ones. We used two sets of energy parameters; one, $\mathbf{w}_{PT}$, obtained by perceptron learning and the second, $\mathbf{w}_{ZT}$, obtained by $Z$-score optimization. The set $\mathbf{w}_{PT}$ was obtained by learning the solution of maximal stability for an independent set of 123 proteins and 836,020 decoys.[15] The set $\mathbf{w}_{ZT}$ was obtained in MS. For both potentials an all atoms definition of contacts was used with a threshold $R_c = 4.5$ Å. The correlation between $\mathbf{w}_{PT}$ and $\mathbf{w}_{ZT}$ is $\rho = 0.61$.

Testing the two methods on 100 proteins (that do not appear in the set that was used to derive the contact energies) gave results of similar quality (see Table I). The perceptron solution misclassified less decoys; the $Z$-score solution, on the other hand, assigned larger $Z$-score to the

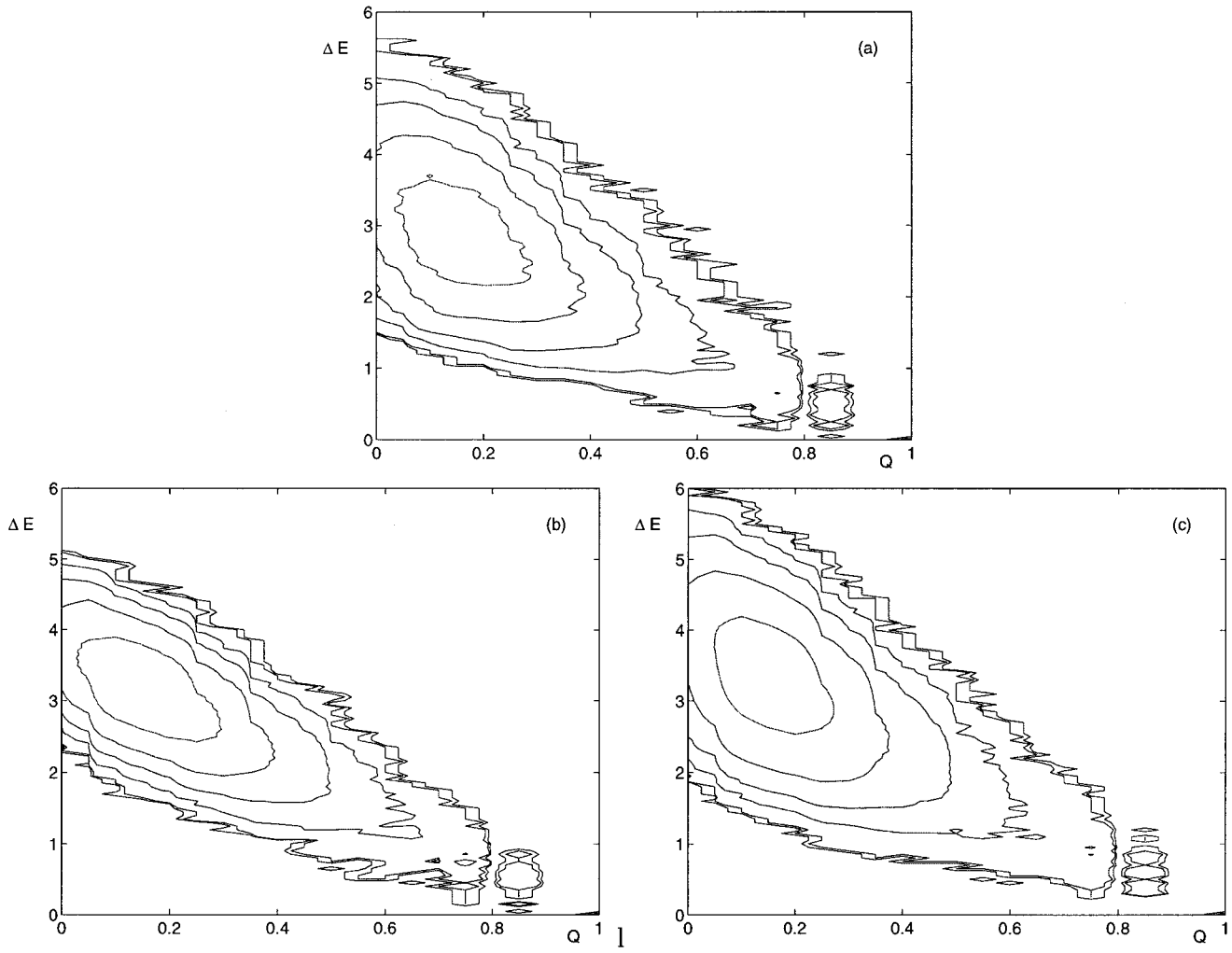


Fig. 3. Contour plot of the energy difference ΔE between decoys and native state and the overlap Q with the native state (**a**) using the set of true energy parameters, (**b**) using the set $\mathbf{w_{ZL}}$ of energy parameters, (**c**) using the set $\mathbf{w_{PL}}$ of energy parameters. Contour levels are spaced logarithmically.

native states, as expected. The distribution of the overlap $Q$ and of the energy $\Delta E$ are given for the two methods and for all the 100 proteins in Figure 5.

For both methods we considered the correlation between $Q$ and $\Delta E$ (see Fig. 6). In Figure 6 we compare two cases. The first is the protein 1mol, which was classified correctly against 11,191 decoys by both methods; it has 94 residues. The second case is the protein 1isu, of 62 residues, which both methods failed to classify correctly. Of the 14,016 decoys that were produced, the perceptron assigned lower than native energy to 138 decoys and the $Z$-score to 208. For the vast majority of the studied proteins (95%) we found a reasonably good correlation between $Q$ and $\Delta E$ (see in Fig. 6a,b) in that the single map with high $Q$ (e.g., the native one) has lower energy than the low-$Q$ decoys. We must add, however, the following note of caution. It is possible that this observed correlation is present only because gapless threading fails to generate challenging and high-$Q$ decoys. Within the present calculations, we are allowed to use only the pairwise contact approximation to some much more complicated "true" contact-map potential. One cannot rule out the possibility that this is such a poor approximation to the true potential, that had we generated better high-$Q$ decoys, the observed correlation would have disappeared. The case of lattice proteins cannot guide us to resolve this question. For lattice proteins, the "true potential" that produced the native folds was a *pairwise contact potential,* whereas the native structures of the threading experiment were stabilized by the (presumably much more complicated) "true" potential that governs protein folding under physiological conditions.

We also looked for some relationship between $Q$ and $\Delta E$ of the low-energy decoys. For each protein, we measured the quantity

$$\alpha = \min_k \arctan \frac{\Delta E_k / |E_k^0|}{1 - Q_k} \tag{19}$$

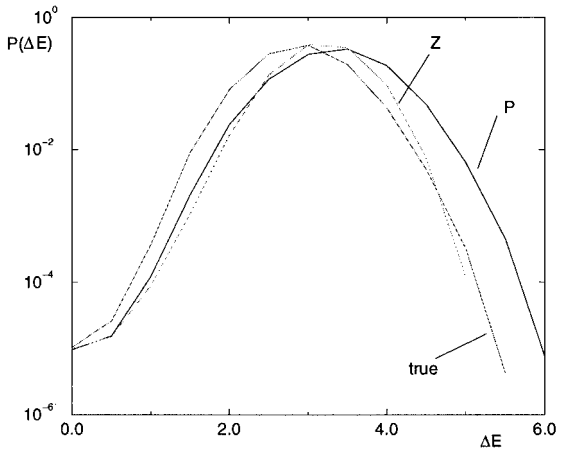


Fig. 4. Histogram of the energy differences between the decoys and the native states, for the true energy parameters and as obtained by the two methods.

where $k$ runs over all the decoys generated for such particular protein and $E_k^0$ is the energy of the native state of proteins $k$. For the cases shown in Figure 6, these minimal values are reached at the decoy of minimal energy. The distribution of the angles $\alpha$ is shown in Figure 7. For both methods, we get fairly similar distributions, with a large peak on the positive side corresponding to successful classifications and a broad tail on the negative side corresponding to misclassified folds.

Figure 8 shows potentials obtained by the two methods for real proteins. There are several important features shared by both potentials. (1) Cysteine, hydrophobic residues, and aromatic residues except proline (C, M, F, I, L, V, W, Y, H) attract each other. (2) Most of the polar residues are repelling from each other and from the hydrophobic ones. (3) Interactions between charged residues (D, E, K, R) are much weaker than between hydrophobic ones. Although most of the charged interactions have the right sign, they are hardly noticeable among other interactions of polar residues.

These properties are easy to understand since hydrophobic/aromatic residues tend to cluster in the protein core while polar ones are spread on the protein surface. Hence, contacts between hydrophobic/aromatic residues are found much more frequently than between polar ones. Cysteines frequently form stabilizing disulphide bonds and are usually located in the protein core. It is also clear why both potentials have weak electrostatic interactions. Salt bridges formed by pairs of oppositely charged residues although contributing to the stability of some proteins[21] are known to be rare and less important than hydrophobic interactions.[22]

Focusing on differences between the two potentials we notice that (1) $\mathbf{w}_{ZT}$ (Fig. 8B) has all interaction energies distributed much more evenly among residues. In contrast $\mathbf{w}_{PT}$ has very diverse interactions, especially those between polar residues. We suggest two possible explanation for the smoothness of interactions in $\mathbf{w}_{ZT}$ vs. diversity in $\mathbf{w}_{PT}$. First, it is known that potentials obtained by optimization of $Z$-scores (or similar functions) tend to underestimate repulsive interactions[10,23] and, hence, have smoother polar-polar interactions. Secondly, $\mathbf{w}_{PT}$ was obtained by discriminating the native fold from explicit decoys. The learning procedure focused on a few low-energy decoys must have learned certain features specific for these decoys and, thus, produced diverse pattern of polar-polar interactions. In summary, interactions between amino acids provided by both potentials agree well with physical and chemical properties of these amino acids and with known features of native proteins.

**TABLE I. Results of the Gapless Threading Fold Recognition Experiment†**

| Potential | Misclassified proteins | Misclassified decoys | $Z$-score |
|---|---|---|---|
| VD | 5 | 192 | −7.20 |
| MS | 7 | 1,261 | −8.44 |

†We used 100 proteins and 698,898 decoys. We report the number of proteins misclassified by the two methods, respectively, and the corresponding total number of which violated the conditions of Eq. (2).

## DISCUSSION

In this paper, we presented a detailed comparison of two methods to derive energy parameters from a known protein structure, the $Z$-score optimization[10] and the perceptron learning.[1] First, we chose an exactly solvable model: lattice 27-mers where sequences were designed to fold to their respective "native" conformations with certain "true" potentials. Our analysis showed that both methods recovered the potentials that were sufficiently close to "true" ones. Besides that, the maximum stability perceptron was able to find the potential with largest energy gap between the ground state and "first" excited state while the potentials derived using the $Z$-score optimization provided slightly lower $Z$-scores for the native structures of lattice proteins. This is as expected. It can be seen in Figure 3 that the large gap provided by the strongest perceptron is between the native state and the "first excited" that is structurally very similar to the native, having high overlap with the native state at $Q \approx 0.8$.

Which method may be better suited for practical applications? Each one has its strengths and weaknesses. The major strengths of the $Z$-score method are the possibility to use implicit decoys and and relative computational simplicity. The weakness is that it does not guarantee that the native state is lowest in energy with derived parameters, i.e., there are no "outliers" that feature lower energy than the native state. The strengths and weaknesses of the perceptron method are complementary to that of the $Z$-score method. The computational efficiency is at issue here especially since the perceptron method requires explicit decoys whose number can be great. In this regard, the observation that in practice the perceptron method used only a tiny fraction of all 103,346 lattice conformations is remarkable and telling. It certainly requires a deeper analysis that will be presented elsewhere.

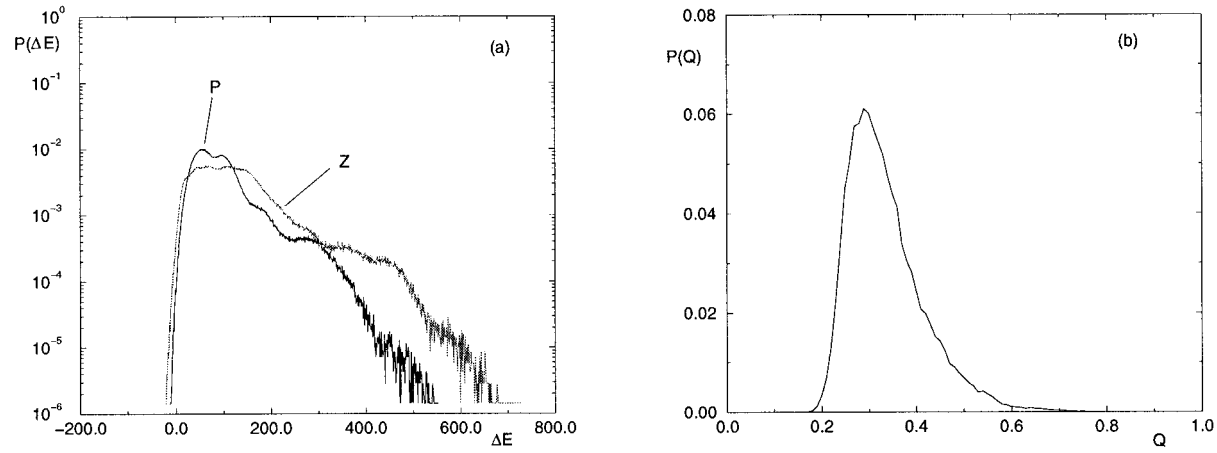


Fig. 5. **a:** Histogram of the energy differences between the decoys and the native states, as obtained by the two methods for all 100 proteins tested. **b:** Histogram of the overlap Q with the native state for the set of decoys used in the threading test.

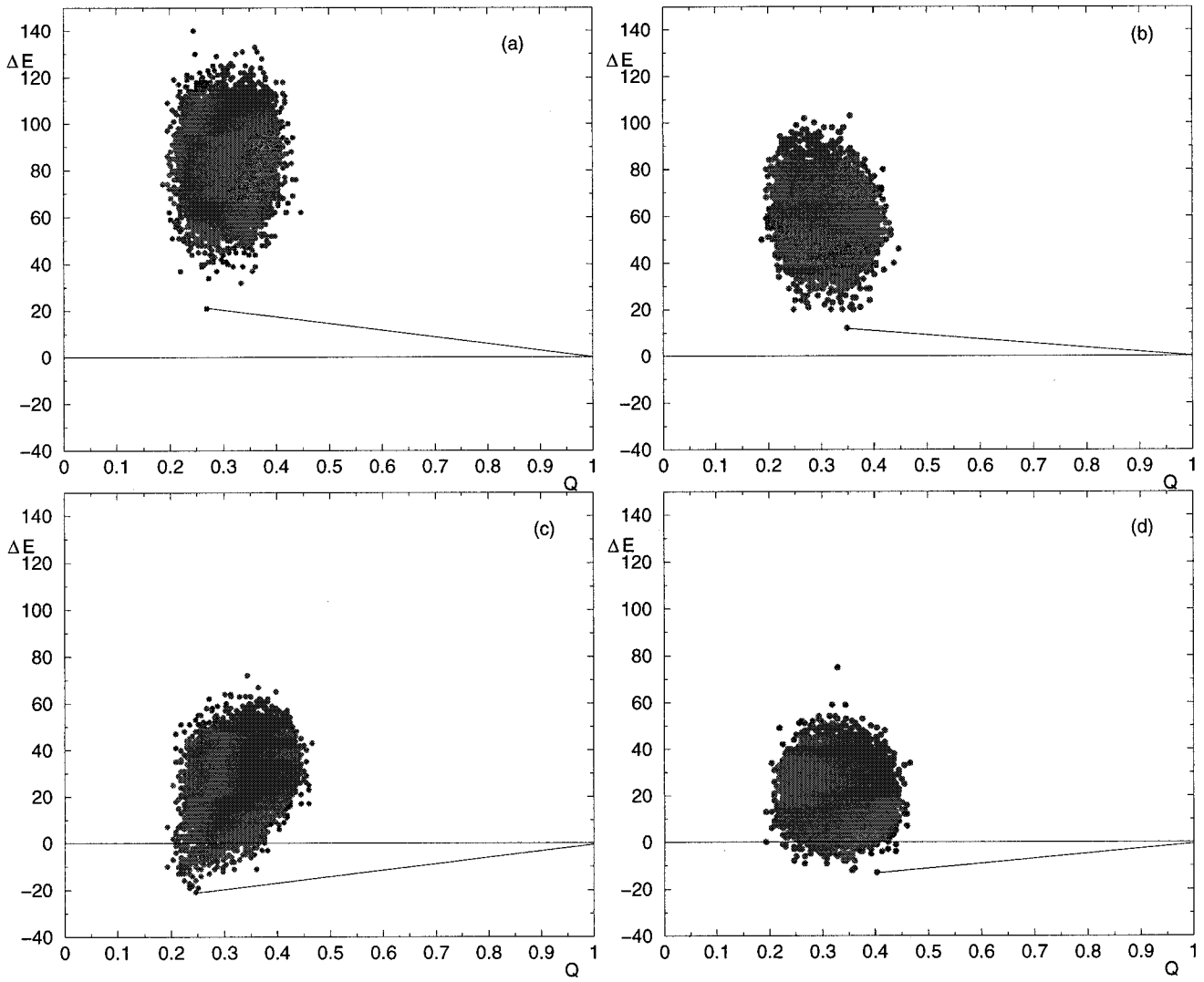


Fig. 6. Scatter plot of the energy difference ΔE between decoys and native state and the overlap Q with the native state (**a**) using the set $\mathbf{w_{ZT}}$ of energy parameters for protein 1mol, correctly classified by the Z-score method, (**b**) using the set $\mathbf{w_{PT}}$ of energy parameters for protein 1mol, correctly classified by the perceptron method, (**c**) using the set $\mathbf{w_{ZT}}$ of energy parameters for protein 1isu, incorrectly classified by the Z-score method, (**d**) using the set $\mathbf{w_{PT}}$ of energy parameters for protein 1isu, incorrectly classified by the perceptron method.

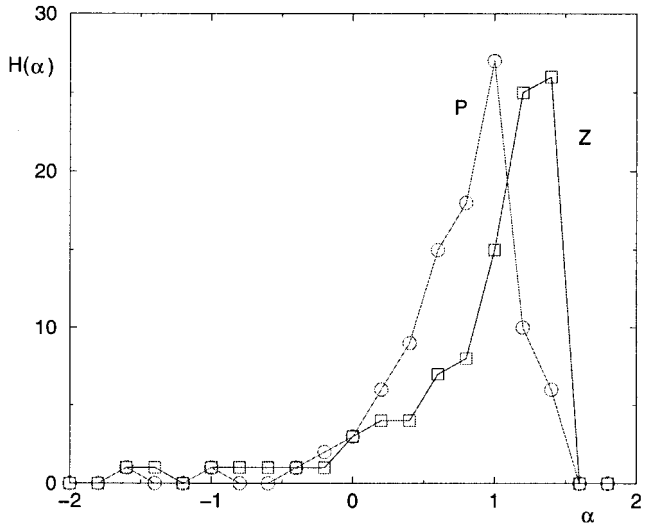


Fig. 7. Histogram of the angle α. The perceptron and the Z-score provide similar distributions.

Energy parameters for lattice proteins, for which the contact potential is the true potential, can be derived by both methods with no remarkable differences.

Further, we tested both methods on a gapless threading applications. Two potentials were used. One, generic, was derived earlier by MS to minimize the *Z*-score of native proteins against implicit decoys.[10] VD derived the other potential used here by perceptron learning of 836,020 decoys, obtained by gapless threading for 123 proteins, (using the all atoms definition of contact and a threshold of 4.5 Å).[15] The 100 proteins that were used in our calculations reported here, to test the performance of both potentials, were not included in the training set for perceptron learning, nor in the set of the *Z*-score-based derivation in Mirny and Shakhnovich.[10] Both potentials performed well in gapless threading tests, providing recognition of the native state in roughly 95% of all presented proteins. Importantly, most of the proteins whose native states were not recognized by either of the methods were "special" in the sense that they are stabilized by certain "extraneous" factors such as metal ions, quaternary interactions, and so on (see also Bastolla et al.[9]).

In this paper, we provided the analysis of two methods of derivation of potentials for protein structure predictions using most rigorous tests on lattice proteins and gapless threading. Both methods performed approximately with equal efficiency alleviating the major concerns that *Z*-score may not be able to provide potentials that discriminate against a few special lowest energy decoys and that the perceptron method may fail to deliver low *Z*-scores to native structures. Such cross-validation is important for application of either of the potential derivation methods to real protein structure prediction problems. Which method is preferable? The answer depends on the specific application. When explicit decoys are problematic to obtain, the *Z*-score method, that does not require them, can be used. On the other hand, in cases when explicit decoys are available the perceptron learning may provide a reliable set of potentials provided that the problem is "learnable."[1,19] Our study of the gapless threading application indicates that the learnability of the problem, for the perceptron, may depend on the inclusion of a small number of "outliers" in the training set, i.e., proteins that are stabilized by extraneous factors such as quaternary interactions or large number of disulfides. This is consistent with the situation in the *Z*-score optimization methods where addition of such proteins into the training set also rendered the problem unsolvable in a sense that no convergence to any potential was obtained. These findings teach us an important lesson, that the choice of the training set is crucial so that proteins in the training set should be stabilized by the same physical factors as those proteins whose structure is being determined using the derived potentials. Since such physical factors are not

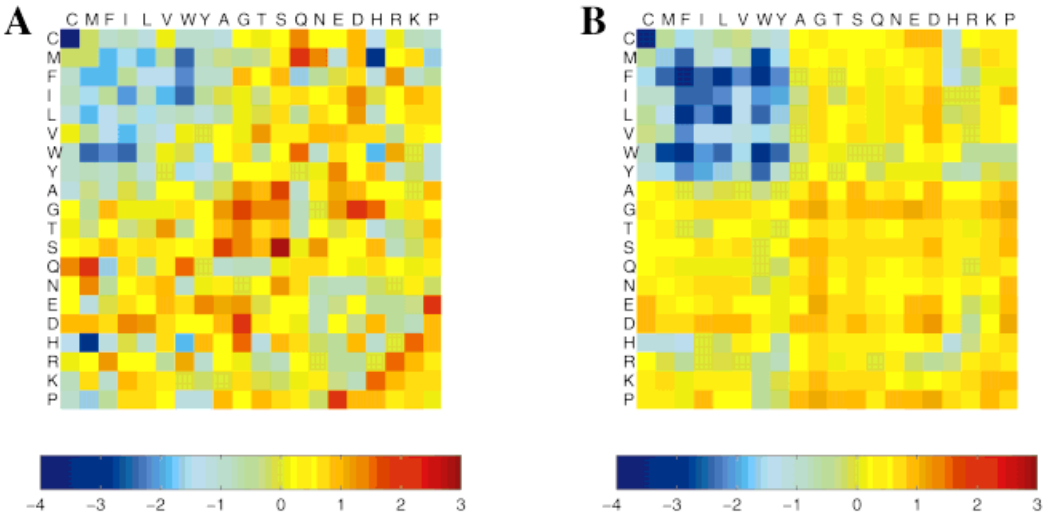


Fig. 8. Comparison of the two sets of pairwise contact energy parameters $w_{PT}$ (**A**) and $w_{ZT}$ (**B**).

known a priori for a new protein, high *Z*-score, or low maximal stability with the perceptron-derived potentials, may be an indication that such a situation is encountered.

We used gapless threading to test the methods of potential derivation. The advantage of this approach is in its extreme simplicity. However, we should note that gapless threading is not a very practical tool for real-life structure prediction application because actual native structure of the query sequence is never in the set of conformation scanned by threading simulation. An actual threading calculation aims to select analogs of the native state of the query sequence in the ensemble of structures scanned. Gapless threading is generally not capable to select or recognize analogs (Mirny and Shakhnovich, unpublished data). To this end, a more advanced threading technique should be used that allows gaps and insertions in sequence and structure.[20] This comes, however, at a price of increasing the number of decoys. The need to discriminate against a larger number of decoys requires better discriminating potentials and/or more detailed models of proteins. In future work, it will be interesting to explore combinations of perceptron learning for discriminating against the most difficult lowest energy decoys with the *Z*-score optimization to discriminate against a mass of "average" decoys. Such simultaneous optimization may be a way to address these very challenging problems of protein structure prediction.

## REFERENCES


1. Vendruscolo M, Domany E. Pairwise contact potentials are unsuitable for protein folding. J Chem Phys 1998;109:11101–11108.
2. Goldstein R, Luthey-Schulten ZA, Wolynes PG. Optimal protein-folding codes from spin-glass theory. Proc Natl Acad Sci USA 1992;89:4918–4922.
3. Goldstein R, Luthey-Schulten ZA, Wolynes PG. Protein tertiary structure recognition using optimized Hamiltonians with local interactions. Proc Natl Acad Sci USA 1992;89:9029–9033.
4. Crippen GM. Prediction of protein folding from amino-acid-sequence over discrete conformation spaces. Biochemistry 1991;30: 4232–4237.
5. Maiorov V, Crippen GM. Contact potential that recognizes the correct folding of globular proteins. J Mol Biol 1992;227:876–888.
6. Crippen GM. Easily searched protein folding potentials. J Mol Biol 1996;260:467–475.
7. Hao MH, Scheraga HA. How optimization of potential function affects protein folding. Proc Natl Acad Sci USA 1996;93:4984–4989.
8. Dima RI, Settanni G, Micheletti C, Banavar JR, Maritan A. Extraction of interaction potentials between amino acids from native protein structures. J Chem Phys 2000;112:9151–9166.
9. Bastolla U, Vendruscolo M, Knapp EW. Proc Natl Acad Sci USA 2000;97:3977–3981.
10. Mirny LA, Shakhnovich EI. How to derive a protein folding potential? A new approach to an old problem. J Mol Biol 1996;264: 1164–1179.
11. Metropolis N, Rosenbluth AW, Rosenbluth MN, Teller AN, Teller E. Equation of state calculations by fast computing machines. J Chem Phys 1953;21:1087–1092.
12. Rosenblatt F. Principles of neurodynamics. New York: Spartan Books; 1962.
13. Minsky ML, Papert SA. Perceptrons. Cambridge, MA: MIT Press; 1969.
14. Krauth W, Mezard M. Learning algorithms with optimal stability in neural networks. J Phys A 1987;20:L745–L752.
15. Vendruscolo M, Najmanovich R, Domany E. Can a pairwise contact potential stabilize native protein folds against decoys obtained by threading? Proteins 2000;38:134–148.
16. Miyazawa S, Jernigan RL. Estimation of effective inter-residue contact energies from protein crystal structures: quasi-chemical approximation. Macromolecules 1985;18:534–552.
17. Shakhnovich EI, Gutin A. Engineering of stable and fast-folding sequences of model proteins. Proc Natl Acad Sci USA 1993;90: 7195–7199.
18. Shakhnovich EI, Gutin A. A novel approach to design of stable proteins. Prot Eng 1993;6:793–800.
19. Vendruscolo M, Najmanovich R, Domany E. Protein folding in contact map space. Phys Rev Lett 1999;82:656–659.
20. Mirny LA, Shakhnovich EI. Protein structure prediction by threading. When it works and when it does not. J Mol Biol 1998;264:1164–1179.
21. Kumar S, Nussinov R. Salt bridge stability in monomeric proteins. J Mol Biol 1999;293:1241–1255.
22. Xu D, Lin SL, Nussinov R. Protein binding versus protein folding: the role of hydrophilic bridges. J Mol Biol 1997;265:69–94.
23. Zhang L, Skolnick J. How do potentials derived from structural databases relate to "true" potentials? Protein Sci 1998;7:112–122.