Protein Science (1994), *3*:567-574. Cambridge University Press. Printed in the USA.

De novo protein design using pairwise potentials and a genetic algorithm

DAVID T. JONES

Biomolecular Structure and Modelling Unit, Department of Biochemistry and Molecular Biology, University College, Gower Street, London WC1E 6BT, United Kingdom, and Laboratory of Mathematical Biology, National Institute for Medical Research, The Ridgeway, Mill Hill, London NW7 1AA, United Kingdom

(RECEIVED January 5, 1994; ACCEPTED February 8, 1994)

Abstract

One of the major goals of molecular biology is to understand how protein chains fold into a unique 3-dimensional structure. Given this knowledge, perhaps the most exciting prospect will be the possibility of designing new proteins to perform designated tasks, an application that could prove to be of great importance in medicine and biotechnology. It is possible that effective protein design may be achieved without the requirement for a full understanding of the protein folding process. In this paper a simple method is described for designing an amino acid sequence to fit a given 3-dimensional structure. The compatibility of a designed sequence with a given fold is assessed by means of a set of statistically determined potentials (including interresidue pairwise and solvation terms), which have been previously applied to the problem of protein fold recognition. In order to generate sequences that best fit the fold, a genetic algorithm is used, whereby the sequence is optimized by a stochastic search in the style of natural selection.

Keywords: algorithm; amino acid; artificial intelligence; computer; de novo protein design; molecular modeling; protein engineering; protein structure

The successful de novo design of a protein structural domain was first described by Regan and De Grado (1988). The problem they tackled was to design an amino acid sequence that would fold into a simple 4-helix bundle. The design was based on 4 identical helices whose sequence was designed by manual model building aimed at stabilizing the 20° interhelical angles observed in classical 4-helix bundles. This manual model building identified leucine as the ideal hydrophobic packing residue for this class of helix interaction, and glutamic acid and lysine residues (alternating 1 per helix turn to stabilize the helices by electrostatic interaction) were used to render the designed protein soluble. To date no structure has been determined for this design helix bundle, but experimental evidence has shown the expressed protein to be mostly helical, stable, and compactly folded. Other groups have attempted similar design experiments (Hecht et al., 1990; Sander et al., 1992), and although no detailed structures have been determined for any of the designed proteins, there is reasonable evidence that in several cases at least the design goals have been achieved.

Reprint requests to: David T. Jones, Biomolecular Structure and Modelling Unit, Department of Biochemistry and Molecular Biology, University College, Gower Street, London WC1E 6BT, UK; e-mail: jones@bsm.bioc.ucl.ac.uk.

The eventual pinnacle of protein engineering will be the fully automated design of a protein with novel structure and function. Achievement of this aim is far in the future, though some early progress has been made. Yue and Dill (1993) have described a simple strategy for designing a heteropolymer sequence (comprising just 2 species of monomer: 1 polar, 1 hydrophobic) such that its compatibility with a 2-dimensional lattice structure is optimized. The work described here is conceptually similar to that described by Yue and Dill, but in this case the objective is to design a real amino acid protein sequence such that its compatibility with a full 3-dimensional structure is optimized.

In designing a protein sequence, 2 primary considerations need to be taken into account. Firstly the designed sequence must be compatible with the specified fold. Secondly the designed sequence must be incompatible with folds other than the specified fold. After these constraints are satisfied, we might choose to narrow the search by favoring sequences that satisfy other less tangible constraints, such as ensuring a given sequence, has an amino acid composition that is typical of the protein's intended folding type, function, or location in the organism. Where applicable, a constraint that must override all the previously mentioned constraints is that of correctly positioning functionally important residues, for example the close proximity of the catalytic Asp, His, and Ser in a designed serine protease, or

selection of positively charged residues in a designed phosphate binding site.

Sequence-structure compatibility

Identifying sequences that are compatible with a given fold is sometimes called inverse protein folding after Drexler (1981), and recently several groups have described useful methods for determining sequence-to-structure compatibility (Sippl, 1990; Bowie et al., 1991; Jones et al., 1992; Maiorov & Crippen, 1992; Sippl & Weitckus, 1992; Godzik & Skolnick, 1992; see Jones & Thornton, 1993, for a review). At the heart of the evaluation function used here is a set of pairwise potentials (potentials of mean force), determined by the statistical analysis of highly resolved protein X-ray crystal structures. These potentials are similar to those originally described by Hendlich et al. (1990), though modified to exclude interactions beyond 10 Å (Jones et al., 1992).

For specified atoms (C$\beta \rightarrow$ Cβ for example) in a pair of residues *ab*, topological level (sequence separation) k, and distance interval s, the potential is given by the following expression:

$$\Delta E_k^{ab} = RT \ln(1 + m_{ab}\sigma) - RT \ln\left[1 + m_{ab}\sigma \frac{f_k^{ab}(s)}{f_k(s)}\right], \quad (1)$$

where m_{ab} is the number of pairs *ab* observed with sequence separation k, σ is the weight given to each observation, $f_k(s)$ is the frequency of occurrence of all residue pairs at topological level k and separation distance s, $f_k^{ab}(s)$ is the equivalent frequency of occurrence of residue pair *ab*, and RT is taken to be 0.582 kcal/mol. In this work, short (sequence separation, $k \leq 10$), medium ($11 \leq k \leq 30$), and long ($k > 30$) range potentials have been calculated between the following atom pairs: C$\beta \rightarrow$ Cβ, C$\beta \rightarrow$ N, C$\beta \rightarrow$ O, N $\rightarrow$ Cβ, N $\rightarrow$ O, O $\rightarrow$ Cβ, and O $\rightarrow$ N.

In addition to the pairwise potentials, a solvation potential for an amino acid residue a is defined as follows:

$$\Delta E_{solv.}^{a}(r) = -RT \ln\left[\frac{f^a(r)}{f(r)}\right], \quad (2)$$

where r is the % residue accessibility (relative to residue accessibility in GGXGG fully extended pentapeptide), $f^a(r)$ is the frequency of occurrence of residue a with accessibility r, and $f(r)$ is the frequency of occurrence of all residues with accessibility r. Residue accessibilities were calculated using the program DSSP (Kabsch & Sander, 1983), applied to Brookhaven coordinate files (Bernstein et al., 1977). Only monomeric proteins were included in this analysis.

In addition to these pseudoenergetic components to the objective function, other factors can be weighted in. Purely practical constraints can be applied to the designed sequence, such as limiting the number of mutations from a given reference sequence. In this way it is possible to answer questions along the lines of "What is the easiest way to make this sequence compatible with this fold?" A more general constraint that may be usefully applied is that of amino acid composition. It is now well established (Nakashima et al., 1986) that there is a significant correlation between the amino acid composition of a protein and its folding class ($\alpha\alpha$, $\alpha\beta$, $\beta\beta$). Although it is not yet known whether a protein must have an amino acid composition typical of its folding class in order to fold, it is at least reasonable to constrain the composition of the designed sequence in order that it is compatible with the folding class of the intended structure. In order to achieve this, an additional term is added to the objective function:

$$s_{comp.}(X) = \sum_{i=1}^{20} (X_i - Y_i)^2, \quad (3)$$

where X is a 20-dimensional vector representing the fractional composition of the 20 amino acids in the designed sequence, and Y is the average vector for the intended class of protein structure.

Genetic algorithms

Genetic algorithms (Goldberg, 1989) are similar in concept to simulated annealing, though their model of operation is different. Whereas simulated annealing is loosely based on the principles of statistical mechanics, genetic algorithms are based on the principles of natural selection. In a typical implementation, the variables to be optimized are encoded as a string of binary digits, and a population of random strings is created. This population is then subjected to the genetic operators of selection, mutation, and crossover. The probability of a string surviving from one generation to the next relates to its "fitness," where a "fit" string is a string relating to an optimal value of the target function. Each string may be randomly changed in 2 ways. The mutation operator simply selects a random bit in the string and changes it to a random value. An alternative means for generating new strings is the crossover operator. Here a randomly selected portion of one string is exchanged with a similar portion from another member of the string population. The crossover operator gives genetic search the ability to combine moderately good solutions so that "super individuals" may be created.

In this implementation of a genetic algorithm, a genome S is defined as a vector of m symbols, where m is the length of the protein sequence being designed:

$$S_1 \ldots S_m.$$

A single symbol here codes for 1 of the 20 standard amino acids, which is in contrast to classical genetic algorithms, where each symbol represents a single binary digit (bit), and where individual genes in the genome are encoded by groups of bits. This "base-20" symbolic representation is found to be far more convenient than a binary representation and avoids the problem of how to map the 20 amino acids onto specific bit patterns without bias. Purists will argue that by using a nonbinary representation the method can no longer be classed as a genetic algorithm, but it is hard to see how such a restrictive definition is helpful to understanding the principles of the method.

At the start of the simulation, a population of n genetic strings S^i ($i = 1 \ldots n$) is created, where the constituent symbols of each strings are selected either randomly, or set to predefined values. Given such a population of strings, a new generation of strings is created from the old set by a combination of mutation, crossover, and selection operators.

The simplest generation operator is that of mutation. A single mutational event is taken to be the change of 1 amino acid symbol, in 1 string, to a new symbol selected from the remain-

ing 19 alternatives. Generally more than 1 mutation is made at each generation, the number of mutations stemming from the mutation probability, which is an adjustable parameter. To choose which symbols to mutate, the following scheme is used. An integer k is set to an integral random number between 1 and mn, where the first symbol of the first string maps to $k = 1$ and the last symbol of the last string maps to $k = mn$. This symbol is set to a different randomly selected amino acid code. To find the next mutation site, k is incremented thus:

$$k' = 1 + k \bmod mn + \frac{\ln r}{\ln(1 - p)}, \tag{4}$$

where r is a uniformly distributed random number ($0 \geq r > 1$), and p is the mutation probability. Mutations are made until $k > mn$, at which point the final value of k is kept to seed the selections for the next generation.

A simple 2-point crossover operator has been utilized in this work. The population of strings is first sorted in descending order of fitness. The top $100C\%$ (where C is the crossover rate) of the population of strings is taken, 1 pair at a time, and for each pair, 2 random string positions, a and b are generated, such that $a \leq b \leq m$. Symbols are then exchanged between the 2 strings:

$$S_a^i \rightleftharpoons S_a^{i+1},\ S_{a+1}^i \rightleftharpoons S_{a+1}^{i+1},\ \ldots,\ S_b^i \rightleftharpoons S_b^{i+1}, \tag{5}$$

where $i = 1, 3, \ldots; i \leq nC$.

The selection strategy used is that suggested by Baker (1987) based on a hypothetical roulette wheel, which is "perfect" in the sense that it selects members of the population in the precise ratio of their respective fitnesses, though cannot easily be converted to a parallel multiprocessor implementation. A full coverage of this and other selection strategies is given in the above reference.

Implementation

The described method was implemented in ANSI C and should run on any Unix workstation (results shown here were obtained on a DEC Alpha 3000/400). The program reads as input a target protein structure in Brookhaven PDB format (Bernstein et al., 1977), a file containing secondary structure assignments and residue accessibilities calculated using the program DSSP (Kabsch & Sander, 1983), and a template sequence as a string of 1-letter codes and set closures. The template sequence is used to constrain the random selection of residues by the mutation operator. Each position in the template can be occupied either by an "X" character, indicating a free choice of amino acid, 1 of the 20 standard amino acid codes restricting the choice to the specified amino acid alone, or a set closure surrounded by square brackets. For example, the string

```
XXCXX[MLIV]XX
```

represents an 8-residue template, where positions 1, 2, 4, 5, 7, and 8 are unconstrained, position 2 is forced to be a cysteine residue, and position 6 can be any of the residues M (methionine), L (leucine), I (isoleucine), or V (valine). Using such a template, the resulting protein design can incorporate functionally important sequence patterns (a simple example being an exposed glycosylation site) or perhaps be forced to create an exposed hydrophobic patch.

The program EvolSeq is available from the author by e-mail (jones@bsm.bioc.ucl.ac.uk).

Results

As a first example, the problem of designing a sequence compatible with a 4-helix bundle structure is presented. As a target structure the coordinates for Felix-HMQ were taken, which was 1 of the 4-helix models built by Hecht et al. (1990). These coordinates have been deposited in the Brookhaven database (Bernstein et al., 1977) as entry 3FLX.

The genetic algorithm was set up with an initial population of 500 strings, a crossover rate of 0.1, and a mutation rate of 0.0013 (1/790). The structure of Felix was designed to incorporate a disulfide bridge between helix 1 (residue 11) and helix 4 (residue 71), and to accommodate this, the sequence template was set to force Cys residues to be located at these positions in the sequence. In this first example, the fitness function used included the previously described pairwise potentials alone. The simulation was run for 6 cycles (1,298 generations in cycle 1, 89 in cycle 2, 153 in cycle 3, 70 in cycle 4, 62 in cycle 5, and no change detected after 62 generations in cycle 6). The resultant "optimum" sequence was as follows:

```
LAAVLAALLACLAALLAAGIWAAILAILLALIALLLKGIMMAALAALL
ALLLALLLALHINAEALAALLACLLALLAAL.
```

Clearly, this designed sequence is not at all protein-like. Most of the sequence is comprised of alanine and leucine, with an apparent helical periodicity in the choice (leucine being used for core packing, and alanine for the exposed helix faces). The abundance of these amino acids is reasonable in the light of their very high helix-forming propensities, though clearly, the sequence is far too hydrophobic to be stable in an aqueous environment. The evident disregard of solvation requirements in the sequence is only to be expected given the fact that the long-distance (>10 Å) contributions to the potentials were excluded in favor of a specific solvation potential.

Another simulation was run using a fitness function from which all but the solvation potentials were excluded. Due to the simple 1-dimensional form of the solvation potentials, only 3 cycles (1,136, 70, and 70 generations) were required to optimize the sequence. Again, a fairly unrealistic sequence was obtained:

```
DKDYFDKWRKCFDDIDKDKRYKKWYKKIKKIFKWWKDRKKDDRWKDFFDRIRKW
FDDKKYKGKWYKKIYDCFKDFKDKK.
```

In this case, a similar periodicity is observed, but rather than alanine and leucine, lysine (with some aspartic acid) and tryptophan are used. Again this is reasonable in view of the fact that only solvation effects are under consideration. Tryptophan is the residue with the highest propensity to be found highly buried, and lysine and aspartic acid with the highest propensity to be found highly exposed. An easy solution to the optimization of the solvation potential terms alone is therefore to bury tryptophan and expose lysine or aspartic acid. Of course, there is insufficient room in the core of the target protein structure to

enable so many tryptophans to be packed together, and so the sequence could never fold into the required conformation.

In the third simulation, a combination of the pairwise and solvation potentials was used. To allow for the unequal distributions of the 2 potentials, the solvation terms were scaled-up by a factor of 15. This is the average ratio between the sum of pairwise terms and the sum of solvation terms in native sequence-structure relationships (based on a set of 102 folds as listed by Jones et al. [1992]). Interestingly, only 3 cycles were required to optimize the sequence in this case (1,368, 92, and 92 generations). The final sequence in this case appears to be a hybrid of the 2 previously shown:

PKEVLEQLRKCLEELAKEKLYEDYLKRLKELLKLLKEYTDEDALKALLEALRKL
LEEKKVSKQWIQELLECLQELQERK.

To the eye, this pairwise/solvation-based sequence looks to be a more plausible protein sequence. The myriad tryptophans have been generally replaced by leucines, which is altogether more acceptable in the light of packing requirements. Nevertheless, the amino acid composition is highly skewed toward leucine, glutamic acid, and lysine, which is unrealistic in comparison to natural protein sequences, though unsurprising considering the fact that these were the amino acids selected by Regan and De Grado (1988) for the $\alpha 4$ protein design described earlier. The final simulation was therefore run using the amino acid compositional bias (S_{comp}). The target amino acid composition used in this case was taken to be the average amino acid composition of the all-α protein chains in the January 1993 release of the Brookhaven database (Bernstein et al., 1977), as shown in Table 1. A (arbitrary) weight of 10,000 was applied to the compositional term in order to equalize its contribution relative to the other 2 terms.

Table 1. *Relative amino acid frequencies of occurrence for the proteins in the all-α ($\alpha\alpha$), all-β ($\beta\beta$), and mixed ($\alpha\beta$) structural classes*

	$\alpha\alpha$	$\beta\beta$	$\alpha\beta$
A	0.119	0.086	0.087
R	0.040	0.040	0.045
N	0.043	0.050	0.045
D	0.058	0.062	0.059
C	0.015	0.027	0.017
Q	0.040	0.034	0.035
E	0.060	0.049	0.060
G	0.072	0.087	0.084
H	0.024	0.020	0.021
I	0.044	0.046	0.055
L	0.100	0.069	0.081
K	0.072	0.047	0.059
M	0.024	0.011	0.020
F	0.039	0.040	0.039
P	0.037	0.051	0.045
S	0.058	0.083	0.064
T	0.053	0.071	0.059
W	0.013	0.016	0.015
Y	0.030	0.044	0.037
V	0.058	0.070	0.070

After running the genetic algorithm for 4 cycles (1,181, 71, 79, and 79 generations) the following sequence was obtained:

SPEVFEAMRKCLEALAQAGVPEKYYQTLKRIFEMYHNFTDDDVWKALLEAIRQL
LNSGGVSDDWLKRIAECLQALKARK.

The course of the sequence optimization for the first 600 generations is shown in Figure 1. The final Felix sequence has an amino acid composition very close to the average of an all-α protein chain, and to the eye looks like a plausible protein sequence. The final sequence has no detectable sequence similarity to any protein in SWISSPROT (Bairoch & Boeckmann, 1991) and is only 19% sequence identical to the sequence originally designed for the Felix-HMQ structure. Interestingly enough, the sequence designed without the compositional bias is 24% identical to the original Felix sequence, and given that the alignment requires no gaps to be inserted, despite the low overall similarity, some aspects of the sequence are evidently constrained by the target structure and can therefore be predicted.

The described optimization strategy ensures that the designed Felix sequence is highly compatible with the target structure. However, compatibility may not be enough to ensure that the sequence will fold into the required conformation. To increase the likelihood of the designed sequence folding correctly it is important to check if the designed sequence is incompatible with conformations other than the target. This check is accomplished by threading the designed sequence onto a library of decoy folds and observing whether the target conformation is significantly more favorable than the alternatives. The designed Felix sequence was threaded onto a library of 102 folds as described by Jones et al. (1992), along with the target fold, and the resulting pseudoenergy totals plotted in the form of a histogram (Fig. 2). The designed sequence clearly favors the target fold. It is interesting to observe that the sequence designed by Hecht et al. (1990) also clearly favors the target fold, though not to such a great extent.

To investigate the convergence properties of the design algorithm the program was run with different population sizes, re-

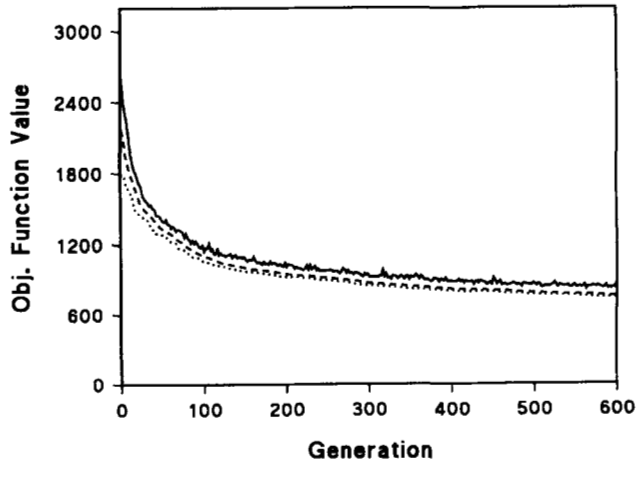

Fig. 1. Plot of objective function value against generation number for the Felix sequence design. The solid line represents the highest (worst) value in the population of 500 sequences, the dashed line the average value, and the dotted line the lowest (best) value.

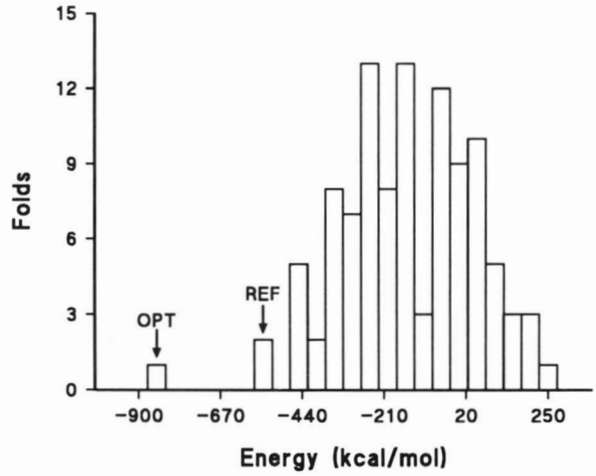

Fig. 2. Threading histogram for the Felix sequence designed by genetic algorithm optimization (OPT). The relative position of the sequence originally designed for Felix is also shown (REF).

peating the run 10 times for each size. The target structure used in each experiment was that of acylphosphatase determined by NMR (Pastore et al., 1992). The choice of acylphosphatase is fairly arbitrary, though its small size and compact $\alpha + \beta$ topology (shown in Fig. 3) are plus points. The coordinates used were those of the first (of 5) model in the Brookhaven file 1APS. To measure the overall degree of sequence conservation, percentage identity between every pair of sequences was calculated and averaged across all 45 sequence pairs. The overall conservation is shown plotted against population size in Figure 4, along with the average value of the objective function for each population. It may be concluded from these plots that no real benefit is gained from choosing a very large population size (1,000 or greater), and that a population of 200 is quite satisfactory. The highest degree of conservation was given by a population size of 500 sequences, albeit by a small margin, and this population size was used throughout the following experiments.

The 10 final sequences for 10 runs with a population of 500 sequences are shown in Figure 5, along with the native sequence. The highest sequence similarity between the native sequence and the 10 designed sequences is only 27% over 45 residues. Generally speaking the residues making the most contact with other residues are the ones that are the most highly constrained and

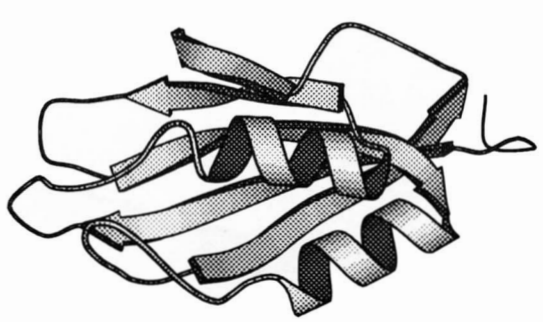

Fig. 3. Molscript (Kraulis, 1991) diagram of the target acylphosphatase structure.

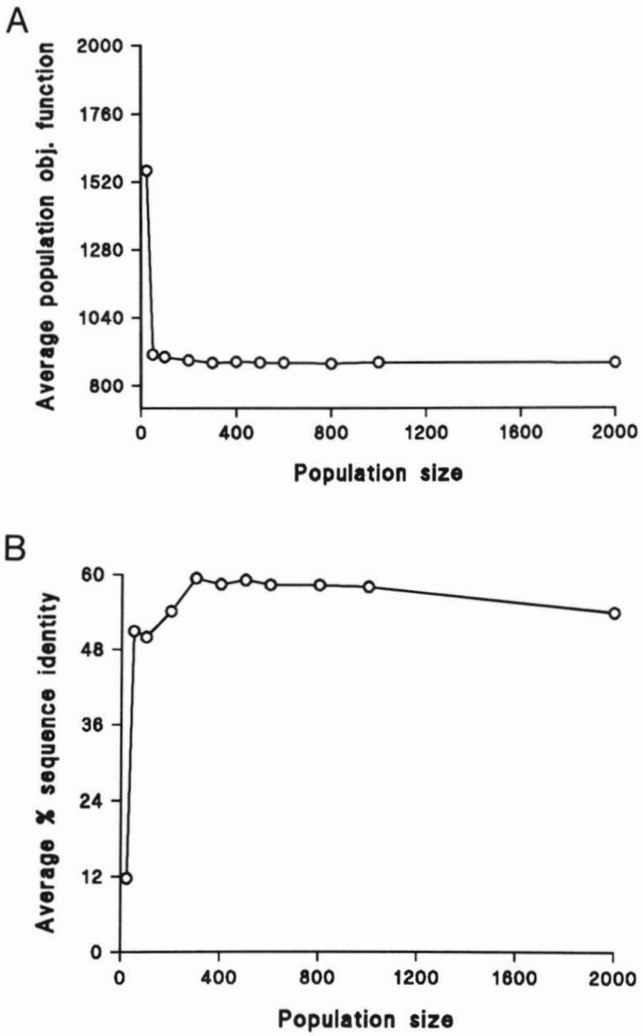

Fig. 4. Plot of average population energy (**A**) and average percentage of sequence identity (**B**) against population size for acylphosphatase.

are thus seen to be conserved between different runs of the program. Both the residue conservation and the 8-Å contact number (the number of surrounding Cα atoms within 8 Å of the residue Cα under consideration; Nishikawa & Ooi, 1986) are plotted against sequence position for the designed sequences in Figure 6.

As final examples, the redesign of 2 proteins originally designed during the 1991 EMBL Protein Design Workshop (Sander et al., 1992) will be considered. The first example is an 8-stranded (4-on-4) β-sandwich, called *shpilka*. The topology chosen for *shpilka* has not yet been observed, though its structure violates none of the known protein folding rules (Fig. 7). Apart from barring the use of cysteine, no constraints were imposed on the choice of residues for each structural site. After 9 cycles totalling 2,455 generations, the following sequence was obtained:

```
 1: GMSVTVTITMGGQKTEVSVSRPGPPPWRVTVTLTIGDGKIRIKLDTHDH
50: YEVPLTYTGGGTITVTIILHMGGRKVPLTVTTSDFGSVTVTIGLTHG.
```

```
Native: STARPLKSVDYEVFGRVQGVCFRMYAEDEARKIGVVGWVKNTSKGTVTGQVQGPEEKVNSMKSWLSKVGSPSSRIDRTNFSNEKTISKLEYSNFSVRY
Struct:      EEEEEEE        HHHHHHHHHHHH  EEEEEEE   EEEEEEEE HHHHHHHHHH          EEEEEEEEE       EEEEE

        NSEKPVVVVYVRIPMPGSRWGVSEWLMRRLKKYGIVFYIKKSSNGHVYIVVVLTQEALEALREFFRKKTDNSFKFGTVQVSSATSVYSTETDSIYGGN
        KPKSKSVYVVIIVPWPGNRWGVREWFAERLKKMGVVFYVTNSDNGYIYFIVVLSREALEALRKLIRKKTSPSYKFGTVTVTGQQSHSSSERDTVYSGN
        NSKKPVVYVYVPIPMPGDKYRWREWFARRLKEMGVVFWISNSSNGHVYVIVVLSQEALEALRELIKKKTDSGFKFGTVSVTQSTSVYSTETDSVYRGN
        NDEKPVVMVIVRVPIPNGRWGVREWLMERLKKYGIVFYISKSSNGHVYVVIVLTQEALEALRKFFREKTSPSYKFGTVVVSSTTTVWQTGSDSIYSSK
        NDETPVVYVVVRVPIPNDRWGVSEWLMERLKKMGVFFYIKKSSNGHVYIIVVLTQEALEALRKFFRKKRSTSYKFYKVQVSSSTTVWQTGSNSIYGGE
        NSKKKVVVVIVRVPMPGDRWGVEEWLMRRLKEYGIFFYIKKSSNGHVYIVVVLTREALEALREFFKKKTSPSYQFYTVVVSSSTSVWQTNTSSIYGGN
        KSKTPVPVVYVRVRMQNDGWGVSEWLAERLKKVGIFFYISKSSNGHVYVVVVLTQEALEALRKFIREKTDPNSKFYTITVRQGESVYSTNSPGIYSSK
        NSKTKVPVVYVAVYFSGGGWGVREWLMRRLKKYGIVWYIVPTENGVVVIFIVLSREALEALRQFIREKSDPNSKMYKVTFTQSSTVWSTESDPIYSSK
        NSKKKVPYVVIPIPMQNDGWGLREWLLRRLKEMGVVFYISKSSNGHVYVVVVMSEEAIEALREFIKKKTSPSYKFYTVTVTGRRSVWSTETDSVYSGN
        NSKGKVVMIYVRIPMPNDSYPVREWLAERLKKYGIFFWISKSSNGHVYVVVVLTEEALEALRKFIRQKYDPNSKVGTVTVRGTTTVWQSGSNSVYSSE
Consens:                      EW   RLK  G         NG      V   EA EALR     K                          Y
```

Fig. 5. Best designed sequences for acylphosphatase from 10 independent populations of 500 sequences. The native sequence is shown, along with a summary of the author-assigned secondary structure ("E" for strand, "H" for α-helix).

Figure 8 shows the threading histogram for the designed sequence, along with the position of the originally designed sequence. Even though the *shpilka* structure is fictional (in that it has not yet been observed in nature) and the original sequence was designed essentially by a process of human contemplation, it is remarkable that the original sequence does appear to be highly compatible with the structure, and a testament to the skill of the human designers. Despite this, the automatically designed sequence is found to be a far more favorable match for the given structure. This is particularly apparent when the pairwise energy and solvation energy sums are considered separately. The pairwise pseudoenergy sum for the original sequence is calculated to be -57 kcal/mol, but is as low as -207 kcal/mol for the designed sequence, which is far more typical of a native-sequence–native-structure threading energy. Both the original and designed sequences have suitably favorable solvation energies (-13 and -20 kcal/mol, respectively).

Another of the proteins designed during the 1991 EMBL Protein Design Workshop (Sander et al., 1992) was a minimal NAD-binding protein called leather comprising a parallel β-sheet with helices packed on either side, and was based on fragments of natural NAD-binding proteins (though mainly lactate dehydrogenase). In this case the residues required for NAD binding (defined as close contacts to the NAD group) were constrained. The following template was used:

```
  1: XXCXTXXXQXXXXXXXXXXXXXXXXXXIXVVGYGAVGXXXAXXXXXXXX
 50: XXXVXXXDGXXXYXXXXXXXXXXXTAGARXXCXXXXXXLXXXNXXIXXXI
100: XXXXXXXXXXXXXXXVSXXSXXXXXXXXXXXX.
```

The target structure and constrained residues are depicted graphically in Figure 9. After 8 cycles totaling 2,143 generations, the following sequence was obtained:

```
  1: GNCDTYAEQAKELLERLKEQQNKPMTVIIVVGYGAVGKALAQALKEAGV
 50: YDTVVVIDGGKPYESIRPLPVIIITAGARRSCGHPPDDLHQSNLDILKNI
100: LRALFKRFPNYTVYWVSYWSETIREFLAKAFG.
```

Figure 10 shows the threading histogram for the designed sequence, along with the position of the originally designed se-

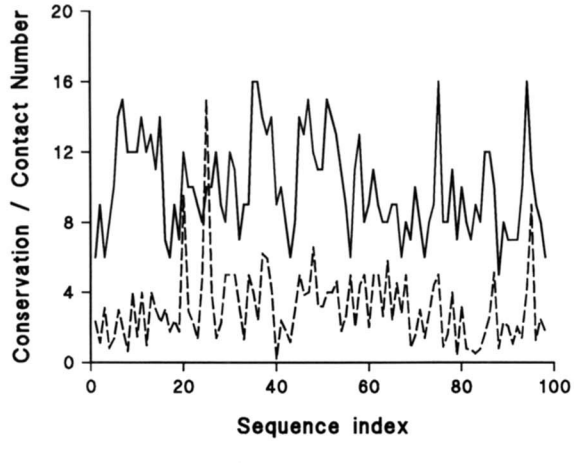

Fig. 6. Plot of sequence conservation and 8-Å contact number against sequence position index for designed acylphosphatase sequences.

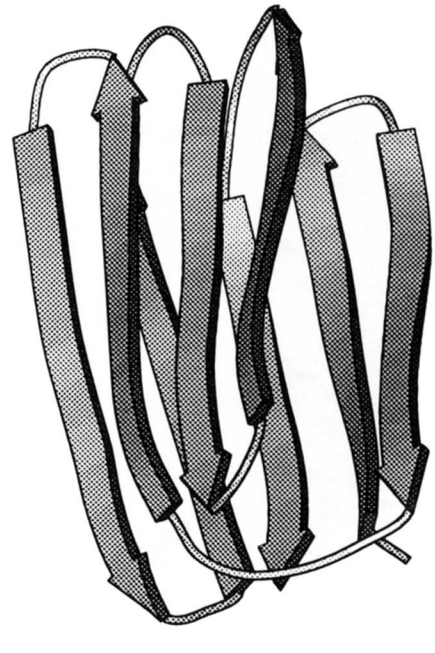

Fig. 7. Molscript (Kraulis, 1991) diagram of target structure for *shpilka* sequence.

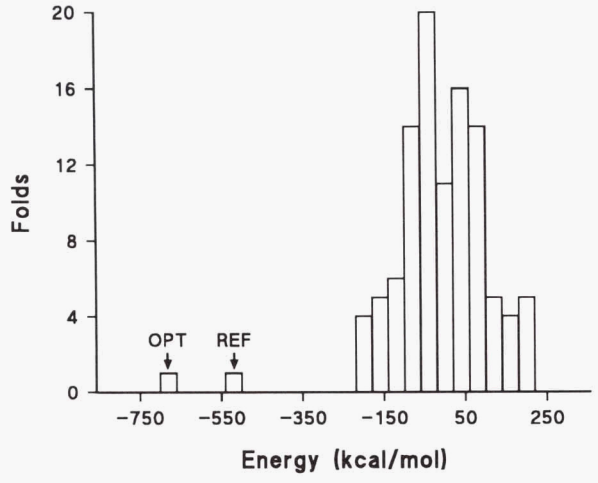

Fig. 8. Threading histogram for the *shpilka* sequence designed by genetic algorithm optimization. Positions of the optimized (OPT) and originally designed sequence (REF) are again shown.

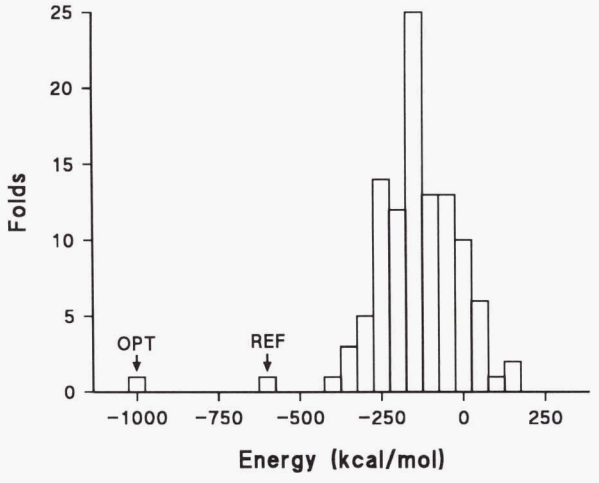

Fig. 10. Threading histogram for the leather sequence designed by genetic algorithm optimization. Positions of the optimized (OPT) and originally designed sequence (REF) are again shown.

quence. Again it is clear that the human protein designers did a good job of creating a compatible sequence, though not as well as the computer program.

Discussion

With reference to the chosen criteria, the described method is clearly able to generate seemingly realistic protein sequences that are highly compatible with their target 3-dimensional structures. The method is easy to apply to a wide range of protein design problems and does not require inordinate amounts of computer time to complete. The most glaring deficiency in the method is that, although it is clear that the designed sequences have a good chance of being stable if deliberately folded into the specified structure, there is absolutely no guarantee that they will be able to arrive at this fold by the normal processes of protein folding. Our present state of ignorance as to the mechanisms of protein folding is such that there is little that can be done to overcome this problem. Obviously the next step will be to synthesize these designed proteins and to test whether they are capable of folding at all. If they do form compact stable structures with the expected secondary structure compositions, then more explicit structural determinations may be attempted. Work along these lines is underway.

Fig. 9. Ribbon diagram of the target structure for leather sequence, rendered by Molscript and Raster3D (Bacon & Anderson, 1988; E. Merritt & M. Murphy, unpubl. results). Cα positions of constrained residues are indicated by yellow spheres.

Another point to consider with the described protein design strategy is whether or not the resultant sequences are overdesigned. As implemented, the genetic algorithm attempts to locate the global optimum sequence for a given structure, an optimum that often has a lower value of the objective function than the native sequence itself. Such overdesign might be beneficial in that the designed sequence might very readily form the required structure and be particularly stable in that conformation, but another possibility is that the global energy minimum for the protein chain is so deep and narrow that the chain never manages to locate the minimum in a biologically useful time. In view of this it may therefore be more appropriate to halt the optimization as soon as the value of the objective function is reduced to the value achieved by the native sequence and thus prevent the energy minimum from becoming too deep.

Away from the extreme case of de novo design of a complete protein sequence, more restricted use of the program may be on safer ground. In the above examples, the majority of the target protein sequence was arrived at by the artificial selection principles as described. Perhaps a more reasonable use of the program is in the redesign of natural proteins, where the native sequence is "tweaked" so as to optimize a given constraint. For example, consider crambin, a small plant protein of unknown function, which is a highly hydrophobic protein and is insoluble in water. A reasonable question that might be posed is whether the crambin sequence could be easily modified to render cram-

bin soluble in water while maintaining compatibility with its native fold. Starting with a homogeneous population of native crambin sequences, the protein design program was run with a template constraining the native cysteine residues that form disulfide bridges in the folded structure. Given the design of the solvation potentials, the exposed hydrophobic residues that render the protein insoluble are likely to be replaced by polar residues. Indeed, due to the presence of these exposed hydrophobic residues, the native sequence itself does not appear to be highly compatible with its fold. The sequence identity between the designed and native crambin sequences is 32% (most of this comes from the constrained cysteines) as shown in the following alignment:

```
            10        20        30        40
Design PYCCPTAQIAAALDRCRKPGITTEECYNAIGCITVNGPGCSSNTPT
       :::          :: ::     : . .:::   :  :
Native TTCCPSIVARSNFNVCRLPGTPEAICATYTGCIIIPGATCPGDYAN.
            10        20        30        40
```

By limiting the total number of mutations (10 in this case), sequences much closer to the native sequence may be generated:

```
            10        20        30        40
Design TYCCPSDEIRSNLNQCRKPGTPVAECATATGCIIIPGATCPGDYAN
      : ::::   ::::: :: :::: : ::: ::::::::::::::::::
Native TTCCPSIVARSNFNVCRLPGTPEAICATYTGCIIIPGATCPGDYAN.
            10        20        30        40
```

Improvements to the measures used to determine sequence-structure compatibility are under development. In particular the method is being expanded to explicitly take core packing into account. The simple pairwise potentials described here do encode some degree of packing information, though only very crudely. They are sufficient to ensure that gross overpacking of the protein core is avoided, though they are insufficient to avoid leaving cavities and are not sufficient to ensure that the core packing is really optimal. The route that is being investigated to overcome this limitation is to use the genetic algorithm to search for optimal side-chain conformations (from a library of rotamers) at the same time as it is searching for an optimal sequence. It should be noted that Tuffery et al. (1991) have already applied genetic algorithms to the problem of fitting side chains to protein main chains, though in their case the amino acid sequence was of course kept constant. Despite the fact that the large increase in the overall search space produced by adding side-chain information does result in slower convergence, the algorithm does appear to be capable of solving the combined problem of sequence optimization and side-chain rotamer selection.

Acknowledgments

I thank Janet Thornton, Willie Taylor, and Laurence Pearl for their helpful comments and useful discussion. This work was supported by a Wellcome Trust Biomathematics Fellowship.

References

Bacon DJ, Anderson WF. 1988. A fast algorithm for rendering space-filling molecule pictures. *J Mol Graphics 6*:219–220.

Bairoch A, Boeckmann B. 1991. The SWISS-PROT protein sequence data bank. *Nucleic Acids Res 19*:2247–2249.

Baker JE. 1987. Reducing bias and inefficiency in the selection algorithm. In: Grefenstette JJ, ed. *Genetic algorithms and their applications: Proceedings of the Second International Conference on Genetic Algorithms.* Hove, UK: Lawrence Erlbaum Associates. pp 14–21.

Bernstein FC, Koetzle TF, Williams GJB, Meyer EF Jr, Brice MD, Rodgers JR, Kennard O, Shimanouchi T, Tasumi M. 1977. The Protein Data Bank: A computer-based archival file for macromolecular structures. *J Mol Biol 112*:535–542.

Bowie JU, Lüthy R, Eisenberg D. 1991. A method to identify protein sequences that fold into a known three-dimensional structure. *Science 253*:164–170.

Drexler KE. 1981. Molecular engineering—An approach to the development of general capabilities for molecular manipulation. *Proc Natl Acad Sci USA 78*:5275–5278.

Godzik A, Skolnick J. 1992. Sequence-structure matching in globular proteins: Application to supersecondary and tertiary structure determination. *Proc Natl Acad Sci USA 89*:12098–12102.

Goldberg DE. 1989. *Genetic algorithms in search, optimization, and machine learning.* Reading, Massachusetts: Addison Wesley.

Hecht MH, Richardson JS, Richardson DC, Ogden RC. 1990. De novo design, expression, and characterization of Felix: A four-helix bundle protein of native-like sequence. *Science 249*:884–891.

Hendlich M, Lackner P, Weitckus S, Floeckner H, Froschauer R, Gottsbacher K, Casari G, Sippl MJ. 1990. Identification of native protein folds amongst a large number of incorrect models: The calculation of low energy conformations from potentials of mean force. *J Mol Biol 216*: 167–180.

Jones DT, Taylor WR, Thornton JM. 1992. A new approach to protein fold recognition. *Nature 358*:86–89.

Jones DT, Thornton JM. 1993. Protein fold recognition. *J Comp Mol Design 7*:439–456.

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

Kraulis PJ. 1991. MOLSCRIPT: A program to produce both detailed and schematic plots of protein structures. *J Appl Crystallogr 24*:946–950.

Maiorov VN, Crippen GM. 1992. Contact potential that recognizes the correct folding of globular proteins. *J Mol Biol 227*:876–888.

Nakashima H, Nishikawa K, Ooi T. 1986. The folding type of a protein is relevant to the amino acid composition. *J Biochem 99*:153–162.

Nishikawa K, Ooi T. 1986. Radial locations of amino acid residues in a globular protein: Correlation with the sequences. *J Biochem 100*:1043–1047.

Pastore A, Saudek V, Ramponi G, Williams RJP. 1992. Three-dimensional structure of acylphosphatase. Refinement and structure analysis. *J Mol Biol 224*:427–440.

Regan L, DeGrado WF. 1988. Characterization of a helical protein designed from first principles. *Science 241*:976–978.

Sander C, Vriend G, Bazan F, Horovitz A, Nakamura H, Ribas L, Finkelstein AV, Lockhart A, Merkl R, Perry LJ, Emery SC, Gaboriaud C, Marks C, Moult J, Verlinde C, Eberhard M, Elofsson A, Hubbard TJP, Regan L, Banks J, Jappelli R, Lesk AM, Tramontano A. 1992. Protein design on computers. Five new proteins: Shpilka, Grendel, Fingerclasp, Leather, and Aida. *Proteins Struct Funct Genet 12*:105–110.

Sippl MJ. 1990. Calculation of conformational ensembles from potentials of mean force. An approach to the knowledge-based prediction of local structures in globular proteins. *J Mol Biol 213*:859–883.

Sippl MJ, Weitckus S. 1992. Detection of native-like models for amino acid sequences of unknown three-dimensional structure in a data base of known protein conformations. *Proteins Struct Funct Genet 13*:258–271.

Tuffery P, Etchebest C, Hazout S, Lavery R. 1991. A new approach to the rapid-determination of protein side-chain conformations. *J Biomol Struct & Dyn 8*:1267–1289.

Yue KZ, Dill KA. 1993. Designing representations of protein conformations. *Abstr Pap Am Chem Soc 206*:115. [Abstr]