# The Application of the Genetic Algorithm to the Minimization of Potential Energy Functions

SCOTT M. LE GRAND and KENNETH M. MERZ JR.
*Department of Molecular and Cell Biology and Department of Chemistry, The Pennsylvania State University, University Park, Pennsylvania 16802, U.S.A.*

(Accepted: 14 November 1991)

**Abstract.** We adapted the genetic algorithm to minimize the AMBER potential energy function. We describe specific recombination and mutation operators for this task. Next we use our algorithm to locate low energy conformation of three polypeptides (AGAGAGAGA, A9, and [Met]-enkephalin) which are probably the global minimum conformations. Our potential energy minima are −94.71, −98.50, and −48.94 kcal/mol respectively. Next, we applied our algorithm to the 46 amino acid protein crambin and located a non-native conformation which had an AMBER potential energy ~150 kcal/mol lower than the native conformation. This is not necessarily the global minimum conformation, but it does illustrate problems with the AMBER potential energy function. We believe this occurred because the AMBER potential energy function does not account for hydration.

**Key words.** Genetic algorithms, protein folding, AMBER, minimization

## 1. Introduction

A protein is a polymer of amino acids linked by peptide bonds (Figure 1). The amino acids are identical except at the points labeled $S_1$, $S_2$, and $S_3$, which are called the sidechains. The sidechain of an amino acid determines its identity; there are 20 different naturally occurring amino acids, hence 20 possible sidechains, each of which has an individual chemical character and structure. The sequence of amino acids in a protein from one end to the other is known as its "primary" structure; there can be anywhere from a few to several thousand amino

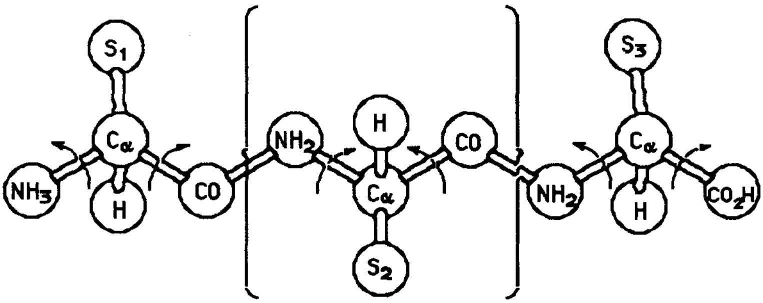

Fig. 1. A three amino acid protein. The $NH_n$'s are called amino groups while CO and $CO_2H$ are called carboxyl groups. $C\alpha$ refers to the carbon where the sidechain ($S_n$) attaches. Torsion angles controlling the conformation are represented by curved arrows. The part of the molecule in brackets is the repeating amino acid monomer.

acids in a given protein. When a protein is synthesized, it rapidly assumes a unique conformation determined by its primary structure (Anfinsen, 1959, 1961, 1973) which is called its "tertiary" or "native" structure. Determining the tertiary structure from the primary structure has been termed the "protein folding" problem.

Although there are $3n-6$ (where $n$ is the number of atoms present) independent variables which determine the tertiary structure of a protein, the bond lengths and bond angles, which account for 2/3 of these independent variables, are relatively fixed (Richardson, 1981). The major determinant of the tertiary structure is the rotation angle around the bonds, which is called the "torsion" or "dihedral" (Figure 1); each amino acid has anywhere from two to ten independent dihedral angles.

Many attempts have been made to reveal the underlying rules governing the protein folding process. Some have focused on predicting the secondary structure, an intermediate structural level between primary and tertiary. Secondary structure breaks the primary structure into three classes: $\alpha$-helix, $\beta$-sheet, or random coil. It has been shown that the amino acids each show different tendencies to be in each structural class (Chou and Fasman, 1974). Secondary structure prediction is assumed to be a somewhat simpler task than determining the tertiary structure which could assist later tertiary structure predictions (Fasman, 1989). So far, it is at best approximately 65% accurate (Stolorz *et al.*, 1991). Despite the application of neural networks (Qian and Sejnowksi, 1988; Holley and Karplus, 1989; Bohr *et al.*, 1988; Kneller *et al.*, 1990; Stolorz *et al.*, 1991), and information theory (Garner *et al.*, 1978; Gibrat *et al.*, 1987; Lambert and Scheraga, 1989b) to this problem, there has been little improvement on this base rate of success nor is this level of knowledge of secondary structure been adequate to determine tertiary structure (Jaenicke, 1991; Stolorz *et al.*, 1991). These methods appear to be limited by their failure to account for long range interactions (Jaenicke, 1991).

Other methods focus on determination of the tertiary structure from first principles, or expand on the results of secondary structure predictions by incorporating long range interactions (Skolnick and Kolinski, 1990; Cohen and Kuntz, 1989). Many of these methods include a potential energy function which represents the stability of a given polypeptide conformation or "conformer" (Weiner *et al.*, 1986, Brooks *et al.*, 1983, Momany *et al.*, 1975). These functions consist of a set of $\sim n^2$ interaction terms depending on the form of the function which describe pairwise interactions between pairs of different atoms for the $n$ atoms in the molecule. The function represents a classical approximation to the many-bodied Schrödinger equation reduced to a set of two-body terms. In some functions such as CHARMM (Brooks *et al.*, 1983) and AMBER (Weiner *et al.*, 1986), every possible pairwise interaction is considered. In others such as ECEPP (Momany *et al.*, 1975), certain interactions are ignored because parts of the molecule (most bond lengths and bond angles) are held rigid. Most of the interaction terms are "non-bonded" interactions. In AMBER, CHARMM, and ECEPP, these interactions are described by a Lennard–Jones potential which accounts for van der Waals attraction and repulsion (1):

$$E_{\mathrm{VDW}} = \left[\left(\frac{A_{ij}}{r_{ij}}\right)^{12} - \left(\frac{B_{ij}}{r_{ij}}\right)^{6}\right], \tag{1}$$

where $E_{\mathrm{VDW}}$ is the van der Waals potential energy between atoms $i$ and $j$, $r_{ij}$ is the distance between atoms $i$ and $j$, and $A_{ij}$ and $B_{ij}$ are empirically determined constants for atoms $i$ and $j$. Additionally, electrostatic attraction/repulsion is modeled by a Coulomb equation (2):

$$E_{\mathrm{electrostatic}} = \frac{q_i q_j}{\varepsilon r_{ij}}, \tag{2}$$

where $E_{\mathrm{electrostatic}}$ is the electrostatic potential energy between atoms $i$ and $j$, $q_i$ and $q_j$ are the atomic charges on atoms $i$ and $j$, $r_{ij}$ is the distance atoms $i$ and $j$, and $\varepsilon$ is the dielectric constant of the medium in which the molecule is located. The Lennard–Jones potential dominates these interactions at short range, while the Coulomb equation dominates them at long range. As can be seen, there is a singularity in this equation when $r_{ij}$ is zero. This leads to $\sim n^2$ singularities in the overall potential energy function of a given molecule. This is the main source of difficulty in locating the global minima of these potential energy functions. Terms which describe interactions between atoms connected by a path of three or fewer bonds are calculated differently. In AMBER and CHARMM, bonded atom interaction terms are calculated by using a harmonic oscillator equation with an equilibrium bond distance and a relatively large force constant to hold the bond distance fairly constant (3):

$$E_{\mathrm{bond}} = K_{\mathrm{bond}}(r_{ij} - r_{\mathrm{eq}})^2, \tag{3}$$

where $E_{\mathrm{bond}}$ is the potential energy of the bond between atoms $i$ and $j$, $K_{\mathrm{bond}}$ is a constant dependent on the type of bond, $r_{ij}$ is the distance between atoms $i$ and $j$, and $r_{\mathrm{eq}}$ is the equilibrium atomic distance for this type of bond. These are mostly ignored in ECEPP as most bond lengths are held constant. The only bond interactions which are considered are those involving two sulfur atoms in a disulfide bridge. In AMBER and CHARMM, an harmonic oscillator equation is also used for the interaction terms of atoms separated by two bonds, except that the angle defined by the three atoms in the path along the two bonds from one end atom to the other is used instead of the bond length to maintain equilibrium bond angles (4):

$$E_{\mathrm{angle}} = \frac{K_{\theta_{ijk}}}{2}(\theta_{ijk} - \theta_{\mathrm{eq}})^2, \tag{4}$$

where $E_{\mathrm{angle}}$ is the potential energy of the bond angle between atoms $i$, $j$, and $k$, $\theta_{ijk}$ is the bond angle $i$-$j$-$k$, $K_{\theta_{ijk}}$ is a constant depending on atoms $i$, $j$ and $k$, and $\theta_{\mathrm{eq}}$ is the equilibrium bond angle. Again, these are mostly ignored in ECEPP except for bond angle terms around a disulfide bridge. In AMBER and CHARMM, interaction terms for atoms separated by three bonds are given dihedral terms; these are modeled as a truncated Fourier series (5):

$$E_{\text{dihedral}} = \sum_{m} \frac{V_{m_{ijkl}}}{2} \left[1 + \cos(m\phi_{ijkl} - \gamma_{ijkl})\right], \tag{5}$$

where $E_{\text{dihedral}}$ is the potential energy of the dihedral angle between atoms $i$, $j$, $k$, and $l$, $m$ ranges over all terms of the series, $\phi_{ijkl}$ is the dihedral angle between atoms $i$, $j$, $k$, and $l$, $V_{mijkl}$ is an empirically determined constant for $\phi_{ijkl}$, and $\gamma_{ijkl}$ is a phase angle for $\phi_{ijkl}$. In addition, this class of interaction terms is sometimes also given a partial non-bonded character by including a scaled contribution from (1) and (2) for atoms $i$ and $l$. In ECEPP, only a subset of the dihedrals are allowed to vary, hence only a subset of the possible dihedral interactions are calculated. The sum of all of these pairwise interaction terms represents the total potential energy of the molecule (6).

$$\begin{aligned} E_{\text{Total}} = {} & \sum_{\text{bonds}} \frac{K_{r_{ij}}}{2} (r_{ij} - r_{\text{eq}})^2 + \sum_{\text{angles}} \frac{K_{\theta_{ijk}}}{2} (\theta_{ijk} - \theta_{\text{eq}})^2 \\ & + \sum_{\text{dihedrals}} \sum_{m} \frac{V_{m_{ijkl}}}{2} \left[1 + \cos(m\phi_{ijkl} - \gamma_{ijkl})\right] \\ & + \sum_{\text{non-bonds}} \left[\left(\frac{A_{ij}}{r_{ij}}\right)^{12} - \left(\frac{B_{ij}}{r_{ij}}\right)^{6} + \frac{q_i q_j}{\varepsilon r_{ij}}\right]. \end{aligned} \tag{6}$$

Techniques based on the use of these potential functions are used to search for the conformation which returns the potential energy function's global minimum. In these techniques, it is assumed that the global minimum energy corresponds to the conformation representing the correct tertiary structure of the protein. This is a controversial assumption (Jaenicke, 1991). Simulated annealing (Wilson and Cui, 1990) and Monte Carlo techniques (Li and Scheraga, 1987, 1988; Skolnick and Kolinski, 1990), dynamic programming (Vajda and Delisi, 1990), distance geometry (Crippen, 1977), the ellipsoid algorithm (Billeter *et al.*, 1987), gradient descent with constraints (Levitt, 1983), and a variety of other minimization techniques (Piela and Scheraga, 1989; Purisima and Scherga, 1987; Dudek and Scheraga, 1990; Lambert and Scheraga, 1989a, 1989b, 1989c; Crippen and Havel, 1990; Lipton and Still, 1988) have been applied to such functions and have succeeded for relatively small proteins of 20 or fewer amino acids. In this paper, we investigate the application of the genetic algorithm to the minimization of the AMBER potential energy function which has the same form as (6) with partial non-bonded character in the dihedral terms as described above and an additional term similar to (1) that handles non-bonded interactions between polar hydrogen atoms and nitrogen and oxygen atoms (7).

$$\begin{aligned} E_{\text{Total}} = {} & \sum_{\text{bonds}} \frac{K_{r_{ij}}}{2} (r_{ij} - r_{\text{eq}})^2 + \sum_{\text{angles}} \frac{K_{\theta_{ijk}}}{2} (\theta_{ijk} - \theta_{\text{eq}})^2 \\ & + \sum_{\text{dihedrals}} \left[ \sum_m \frac{V_{m_{ijkl}}}{2} [1 + \cos(m\phi_{ijkl} - \gamma_{ijkl})] \right. \\ & \left. + \left(\frac{A_{ij}}{r_{il}}\right)^{12} - \left(\frac{B_{ij}}{r_{il}}\right)^{6} + \frac{q_i q_l}{\varepsilon r_{il}} \right] \\ & + \sum_{\text{non-bonds}} \left[ \left(\frac{A_{ij}}{r_{ij}}\right)^{12} - \left(\frac{B_{ij}}{r_{ij}}\right)^{6} + \frac{q_i q_j}{\varepsilon r_{ij}} \right] + \sum_{\text{H-bonds}} \left[ \frac{C_{ij}}{r_{ij}^{12}} - \frac{D_{ij}}{r_{ij}^{10}} \right]. \end{aligned} \tag{7}$$

For a more comprehensive introduction to the protein folding problem, see Richards (1991), Richardson and Richardson (1990) and Jaenicke (1991).

## 2. The Genetic Algorithm

The genetic algorithm is an optimization technique derived from the principles of evolutionary theory (Holland, 1975; Goldberg, 1989). It has been applied to a myriad of optimization problems such as the Traveling Salesman Problem, neural network optimization (Montana and Davis, 1989; Whitley *et al.*, 1989a, 1989b, 1990a, 1990b), scheduling (Cleveland and Smith, 1989), machine vision, pattern recognition, and the solution of non-linear equations. See Goldberg (1989) for a more complete review of these applications. Recently, it was applied to NMR refinement of small nucleotides (Lucasius and Kateman, 1989; Lucasius *et al.*, 1990).

Figure 2 illustrates a traditional genetic algorithm (TGA) as described in

```
procedure GA
begin
    t = 0;
    initialize P(t);
    evaluate structures in P(t);
    while termination condition not satisfied do
    begin
        t = t + 1;
        select P(t) from P(t-1);
        recombine structures in P(t);
        evaluate structures in P(t);
    end
end.
```

Fig. 2. A genetic algorithm.

Grefenstette and Baker (1989). First, the $k$ independent variables of a multivariate function are encoded in some fashion as genes on a chromosome. Next, a population of $N$ chromosomes (hereafter known as $P(t)$) is initialized randomly. After this *initialization* step, the function value of the point in parameter space represented by each chromosome $x$ is evaluated and called the chromosome's fitness $u(x)$. Next, the algorithm cycles through rounds of *selection*, *recombination* and *evaluation* until termination conditions are met.

During *selection*, a new population $P(t+1)$ is selected from $P(t)$. A popular method is known as proportionate selection. This selects a given chromosome $x$ for $P(t+1)$ with probability $p(x)$ which is proportional to the ratio of its fitness relative to the mean fitness of the population $\bar{u}(t)$ (7).

$$p(x) = \frac{u(x)}{\bar{u}(t)} \,. \tag{7}$$

There are numerous other methods in use (Goldberg, 1989; Whitley and Hanson, 1989a).

During *recombination*, the genes in pairs of chromosomes in $P(t+1)$ (hereafter known as *parents*) are mixed together to produce hybrid chromosomes (hereafter known as *children*) via operators analogous to genetic crossover. There are many crossover operators in use (Booker, 1987; Schaffer and Morishima, 1987; Sirag and Weisser, 1987; Davidor, 1989; Goldberg, 1989). A typical operator, known as simple two-point crossover, creates a child containing all the genes from the beginning of one parent's chromosome up to a cut point, and the rest of its genes from that cut point to the end of the chromosome from the second parent. A second child can be created from the genes in both parental chromosomes which are not in the first child. After crossover, parts of the child's chromosome are altered slightly by operators analogous to genetic mutation. As with crossover, there are many mutation operators in use (Fogarty, 1989; Whitley and Hanson, 1989a; Goldberg, 1989). A typical method is to give each of a child's genes a 3–5% chance of being changed to a random value based on the value of a random variable.

Finally, during *evaluation*, the fitnesses of the chromosomes in $P(t+1)$ are evaluated in the same manner as after initialization, and the process repeats until user specified termination conditions are met. Limited convergence theorems have been proven (Ankenbrandt, 1991; Davis and Principe, 1991). However, the success of the genetic algorithm up to this point has been shown mainly through empirical demonstrations.

There are many variations on this basic theme, and there are several good introductions to the subject which cover both this basic approach (Wayner, 1991; Walbridge, 1989; Radcliffe and Wilson, 1990) and many of the variations (Goldberg, 1989). Holland (1975) presents a rigorous derivation of the genetic algorithm.

## 3. Our Genetic Algorithm

Our genetic algorithm is known as a steady state genetic algorithm (SSGA). The main difference between TGA and SSGA is that $P(t+1)$ is identical to $P(t)$ except for the possible introduction of a single new chromosome created by the selection of two chromosomes in $P(t)$ and their subsequent recombination. In a TGA, the entire population is regenerated at each step. This means that it is possible to lose a relatively fit chromosome and hinder the optimization process. Since most of the population is maintained in an SSGA, this will not happen. This frequently tends to speed up the optimization process when compared to TGA (Ackley, 1987).

We have based our genetic algorithm on Whitley's GENITOR algorithm (1989a,b; 1990a,b) which has two distinctive features. The first feature is a rank based rather than fitness based selection function which awards many more reproductive opportunities to the fittest chromosomes in $P(t)$ (8):

$$p(x) = \int_{r_x - 1/N}^{r_x/N} \frac{bias - \sqrt{bias^2 - 4.0*(bias - 1.0)*rnd()}}{2.0*(bias - 1.0)}, \tag{8}$$

where $r_x$ is the rank of a given chromosome $x$ if the population is sorted in order of decreasing fitness, *bias* is a user definable variable for the degree of focusing selection on the fittest members of the population, and rnd() is a random variable between zero and one. The selection function integrated in (8) is plotted in Figure 3. The probability of selection, $p(x)$, decreases linearly with decreasing rank. The

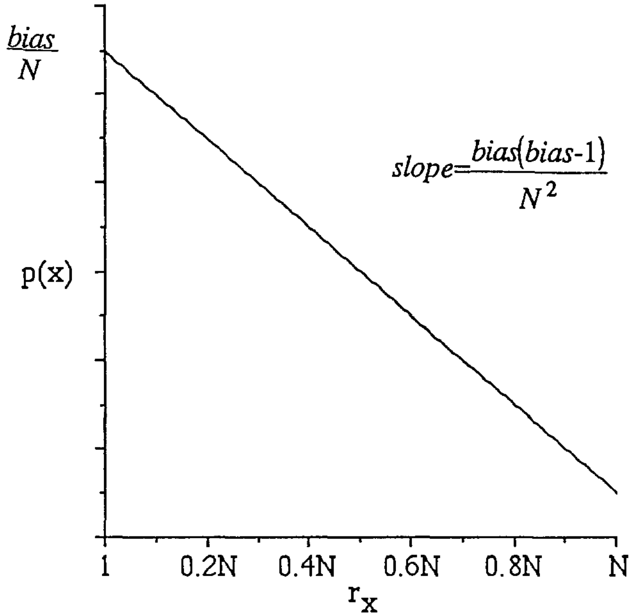

Fig. 3. Probability of selection versus rank using equation 8 with *bias* = 1.5.

second feature is a mutation operator which bases its rate of mutation on the similarity of the two parents used to create the child it will operate on. However, we have not followed GENITOR verbatim. We have modified it in several ways which are described below.

### 3.1. INITIALIZATION

For a given molecule, we encode each conformationally dependent dihedral angle as a gene. We encode the dihedrals of the amino acids of a polypeptide in the order of the primary sequence. For each amino acid, we use the internal order $\phi$, $\varphi$, $\chi_1, \chi_2, \ldots, \chi_n$. We have obtained our best results by encoding dihedral angles as floating point numbers. In all of the runs reported here, $N$ is 200. After we generate each chromosome, we compare the conformation it represents to a library of ECEPP based local minima conformers (Vasquez *et al.*, 1983). We find conformer it resembles the most, $c$, by the following metric (9):

$$d = \sum_{i}^{\text{dihedrals}} \frac{min(abs(\theta_{ci} - \theta_{li}), 360 - abs(\theta_{ci} - \theta_{li}))}{dihedrals*180}, \tag{9}$$

where $\theta_{xi}$ represents the value of dihedral $i$ in $x$ and $\theta_{li}$ represents the value of dihedral $i$ in a given conformer $l$ in the minima library. The value of $d$ can range from zero to one. Zero indicates the two conformers are identical, and one indicates they are maximally different (all angles differ by exactly 180°). Once we have located $c$, we check that all dihedral angles in $x$ are within ±20° of the respective dihedral in $c$. For each dihedral $\theta_{xi}$ that is not, we set it to $\theta_{ci} \pm 20°*rnd()$. This step is a heuristic based on the assumption that in the global minimum conformation of the entire molecule, each of the individual amino acids should be in a conformation close to an individual local minimum. It reduces the size of the search space of each amino acid by a factor of 10 for small amino acids, and by as much as a factor of 10,000 for large ones. Finally, we evaluate the AMBER potential energy of the conformation represented by each chromosome $x$ and use that as its fitness $u(x)$.

### 3.2. SELECTION

Selection in our algorithm is identical to GENITOR and uses (8). We use 1.5 as our value for *bias* because that is what empirically gave the best performance. Lower values led to longer optimization times, and higher values generally caused our algorithm to fail.

### 3.3. RECOMBINATION

We use three different crossover operators here with adaptive probabilities (Davis, 1989). Initially these are chosen with equal probability. Each time a new minimum energy conformer is created by one of these operators, the decrease in potential energy from the previous minimum is stored in a counter $s_i$ with one counter for each crossover operator $i$. Every 50 iterations of the main loop, we update 15% of the net probabilities for each of the three crossover operators based on the changes in the $s_i$ during the last 50 iterations. The probability of choosing crossover operators which caused the most improvement are increased at the expense of the probabilities of choosing the less successful crossover operators. A minimum probability for each crossover operator of 3% is enforced to insure that each of the crossover operators are at least occasionally applied. The three crossover operators we used are illustrated in Figure 4. Figure 4(a) illustrates simple two-point crossover as described above. Figure 4(b) illustrates two-point wraparound crossover. In two-point wraparound crossover, the chromosome is treated as a ring: a child's chromosomes is created from an arc segment out of the first parent's chromosome and the complementary arc segment from the second parent's chromosome. This is thought to help transfer genes together which are on opposite ends of the chromosome which would otherwise

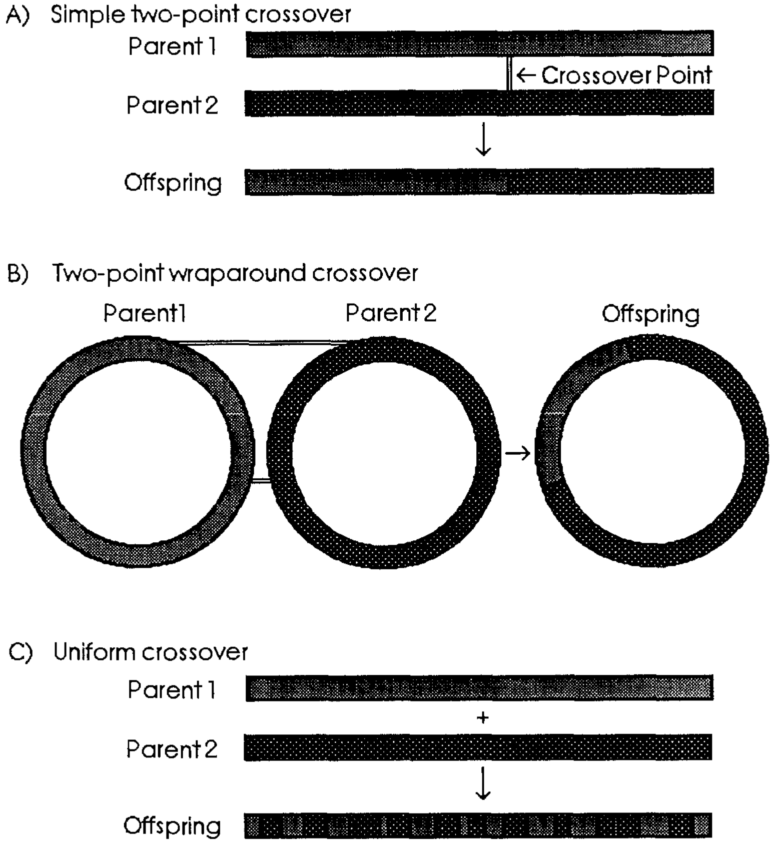

Fig. 4. The three modes of crossover we use in our algorithm.

tend to be broken apart by simple two-point crossover. Figure 4(c) illustrates uniform crossover. In uniform crossover, each gene is taken from either parent with equal probability based on the value of a random variable. Uniform crossover also helps to solve distance dependent crossover problems, but it can also disrupt pairs of genes near one another that would otherwise be likely to be transferred together during crossover.

Additionally, all three operators perform crossover only on the subset of the chromosome at which the parent's genes have an absolute difference greater than five degrees out of the maximum possible difference of 180. This subset of the chromosome is known as the *reduced-surrogate* (Booker, 1987). The use of reduced-surrogate based crossover helps maintain genetic variation in the population during long searches by forcing the child to be different from both of its parents. The genes of the child which are not part of the reduced-surrogate are set to the same values as the first parent.

Although we also use a parental similarity dependent mutation rate as in GENITOR, we vary the similarity dependence during the course of the run. The mutation rate $m(x)$ is calculated two ways depending on the stage of the run. Initially, it is calculated using:

$$m(x) = 0.3^* d^{2^{e(t)}}, \tag{10}$$

where $d$ is the genetic distance between the child's as measured by (9), and $e(t)$ is initially zero, but is incremented by one every 5,000 iterations during which no new minimum energy conformation is located. Once 25,000 iterations have passed during which no new minimum energy conformation is located $m(x)$ is set to 0.1. When a mutation is performed on a gene, we bias it using (8) and use $r_x$ allowed to range from one to 180 with a randomly determined sign as a displacement added to the mutated gene's current value. This favors small displacements in the dihedral angles over large ones and seems to improve performance. This behavior has been seen in other genetic algorithms using floating-point representation for genes (Janikow and Michaelewicz, 1991).

After crossover and mutation have been performed on the child, it is compared to the conformer library as described in the initialization and adjusted to resemble the most similar conformer in this library if necessary.

### 3.4. EVALUATION

The AMBER potential energy of the conformation represented by the child's chromosome is now calculated and used to determine whether it will be inserted into $P(t)$. Initially, this is determined by locating the member of $P(t)$ most similar to the newly created child $x$ via (9). If $x$ has a lower potential energy than this member of $P(t)$, it replaces it. Otherwise, $x$ is discarded. This implements a variation of phenotypic sharing (Goldberg, 1987) because it helps to maintain diversity by rewarding the exploration of unexploited low energy regions of the search space. After 25,000 iterations have passed without locating a new minimum energy conformer, only the members of $P(t)$ with higher potential energies than $x$ are considered for replacement. This is done to focus the search and converge the population so the program will terminate.

### 3.5. TERMINATION

The program is terminated if any of the following conditions are met: 100,000 iterations have passed without locating a new minimum energy conformer, the variance of the potential energies of $P(t)$ is less than 0.1, or the average distance between 200 randomly selected pairs of chromosomes in $P(t)$ as measured by (9) is less than 0.1. Otherwise, the cycle of selection, recombination, and evaluation continues.

## 4. Results

We applied this algorithm to the nine amino acid polypeptide ala-gly-ala-gly-ala-gly-ala-gly-ala and obtained almost identical results in nine out of ten runs (Figure 5) with an average RMS displacement of less than 0.01 Å between successful runs. The final conformation we obtained is an $\alpha$-helix. In the successful runs, the final potential energies ranged from −94.69 to −94.78 kcal. In every run, most of the backbone is oriented as an $\alpha$-helix, but in the one failure, the only substantial difference from the nine other runs appears at the carboxyl end of the helix, where the final carboxyl group is improperly oriented relative to the rest of the helix. Glycine helices have been found as global minimum structures by other groups (Ripoli *et al.*, 1991), but glycine is known as a helix-breaking residue which suggests that this is an unrealistic structure for this molecule in solution.

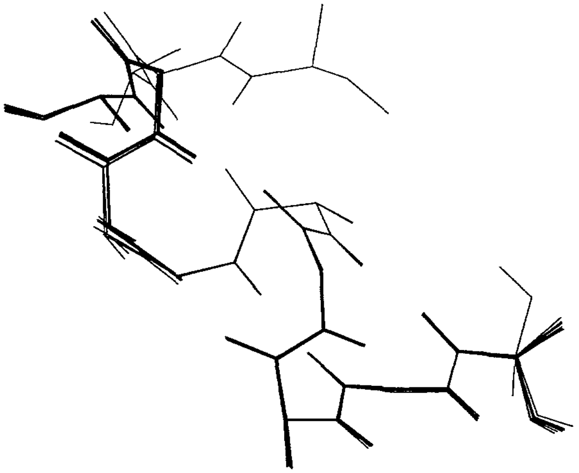

Fig. 5. Superimposition of 10 runs on AGAGAGAGA.

Glycine is the most flexible amino acid, and it is probable that there are many low energy structures which would be sampled by this polypeptide were it synthesized and placed in solution, outweighing this helical structure by sheer number (Chakrabarty *et al.*, 1991). Each of these runs took an average of two hours of CPU time on an SGI IRIS 4D/220 (240,000 iterations).

Similarly, we consistently located the same minimum energy conformation of $Ala_9$ in nine out of ten runs. The final potential energies of these successful runs ranged from −98.42 to −98.50 kcal. The final conformation we obtained is an $\alpha$-helix (Figure 6). The average RMS displacement between the successful runs is less than 0.01 Å. The one unsuccessful run is identical to the nine others except that it has a slightly misaligned carboxyl terminal which raised its potential energy by approximately 0.7 kcal. Contrary to the above polypeptide, alanine rich polypeptides are known to form a helical structure in solution (Marqusee *et al.*, 1989). These runs also consumed approximately two hours of CPU time on the same machine (240,000 iterations).

Next, we applied our algorithm to [Met]enkephalin which has been minimized under ECEPP by several groups (Li and Scheraga, 1987; Purisima and Scheraga, 1987; Vajda and Delisi, 1990) who have located the same minimum energy structure for this molecule. Initially, we consistently converged to a structure which bore little resemblance to theirs, and which had a higher potential energy (~−42 kcal) than their structure (~−47.2 kcal) after extensive gradient minimization of its AMBER potential energy. However, when we slightly altered several bond angles and bond distances of our amino acids near the alpha carbon which are not otherwise varied during a run, we obtained a structure more similar to that found by the other groups in eight out of ten runs (Figure 7) all of which had lower potential energy than their structure, with basically identical backbone structure in all ten runs. Although we have a similar hydrogen bond geometry to the ECEPP minimum energy structure, ours is basically different from theirs. Given that we are working with a completely different force field, and in the absence of solvent, this is not worrisome. This may also serve as a caveat against restricting conformational search solely to dihedral angles. It may help to give bond angles which are observed to vary in crystal structures a small degree of

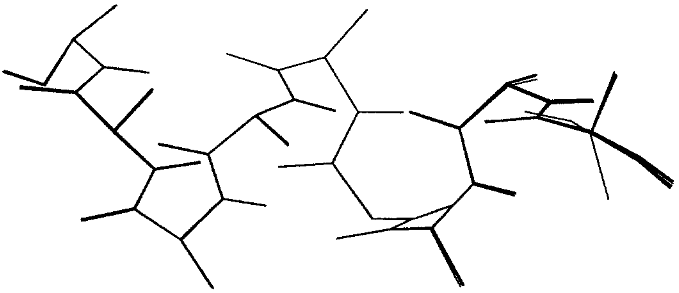

Fig. 6. Superimposition of 10 runs on AAAAAAAAA.

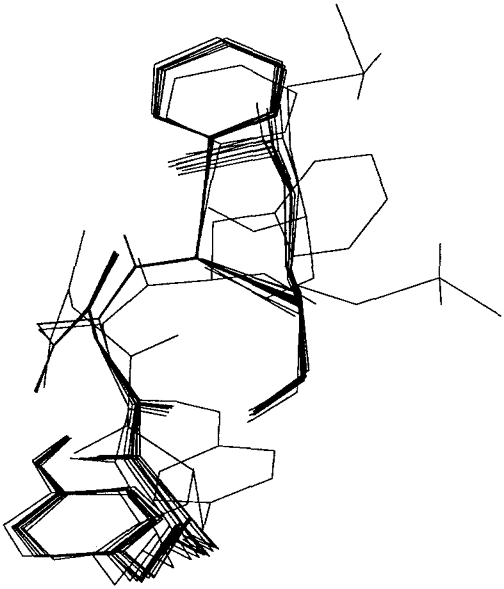

Fig. 7. Superimposition of 10 runs on [Met]Enkephalin.

flexibility. In the nine successful runs, our final potential energies ranged from −48.67 to −48.94 kcal, and had an average RMS displacement of <0.02 Å between successful runs. The conformation generated by the one completely unsuccessful run has a properly oriented backbone, but improperly oriented sidechains. A second run properly oriented everything except for the methionine sidechain. As with the two previous polypeptides, these runs consumed an average of approximately two hours of CPU time (250,000 iterations) on our IRIS.

One of our hopes was that the encoding of the dihedrals in the order of the primary sequence would favor the creation of small groups of linked dihedral genes. In all three of these minimizations, individuals which strongly resemble the final minimum energy conformation, but with much higher potential energy, appear early on in the population. These conformers can be observed in the playback of the minimization process. This indicates that such groups of dihedral genes do arise and that this is a suitable task for the genetic algorithm.

Next, we applied our algorithm to the 46 amino acid protein crambin and obtained ambiguous results. This was only run once because it required a week of CPU time on our heavily used IRIS and therefore our algorithm took a month to complete. Although the final conformation is not necessarily the global minimum conformation, it has an AMBER potential energy 150 kcal lower than that of the known crystal structure (Figure 8 and 9). Unfortunately, our final conformation bears little resemblance to the known crystal structure. Brooks *et al.* has stated that the minimum of a potential energy function for a large molecule in the

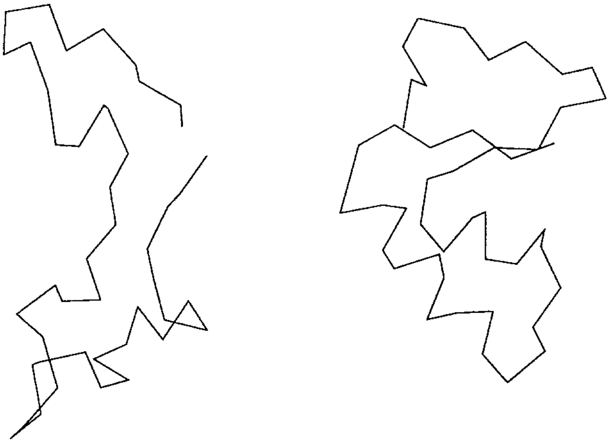

Fig. 8. Alpha carbon trace of our crambin structure (left) and the correct one (right).

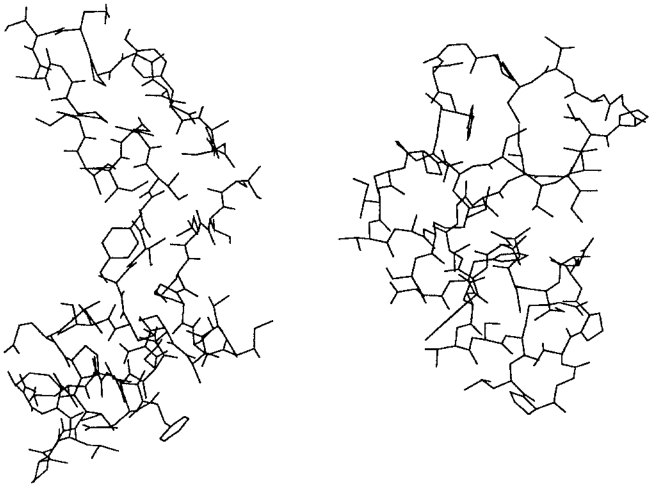

Fig. 9. Our final crambin structure (left) and the correct structure (right).

absence of solvent would be an "inside-out" protein (1988). Large sidechains would stick out of the protein, and small polar sidechains would be closely paired with other such sidechains in its interior. Our protein resembles an inside-out protein so our results may correspond to this situation. Our final structure appears to be mainly stabilized by electrostatic interactions and hydrogen bonds between the backbone and polar sidechain groups (Figure 9). This may indicate that protein folding algorithms must account for hydration, which is believed to be a driving force for protein folding (Baldwin and Eisenberg 1987; Baldwin 1989; Khechinashvili 1990).

## 5. Conclusions

The genetic algorithm is an effective method of searching the conformational space of small molecules which may eventually be successfully applied to small proteins. This will require accounting for hydration. Several models for this exist which treat it as an additional potential energy term (Eisenberg and McLachlan, 1986, Ooi *et al.*, 1987). We are currently experimenting with these models, and we are also investigating a parallel implementation of our genetic algorithm (Tanese, 1989; Whitley and Starkweather, 1990b; Mühlenbein, 1989) for use on larger molecules and more elaborate potential energy functions.

## Acknowledgements

We would like to acknowledge the help of Dr. Max Vasquez and Stephanie Wukovitz in conducting this research. We thank the Pittsburgh Supercomputing Center for Cray YMP time. Molecular graphics images were produced using the MidasPlus software system from the Computer Graphics Laboratory, University of California, San Francisco (Ferrin, 1988). Finally, we thank the Office of Naval Research for support of this project (N00014-90-3-4002).

## References

Ackley, D. (1987), *A Connectionist Machine for Genetic Hillclimbing*, Kluwer, Boston.

Anfinsen, C. (1959), *The Molecular Basis of Evolution*, John Wiley & Sons, New York.

Anfinsen, C. (1961), The kinetics of formation of native ribonuclease during oxidation of the reduced polypeptide chain, *PNAS* **47**(9), 1309–1314.

Anfinsen, C. (1973), Principles that govern the folding of protein chains, *Science* **181**, 223–230.

Ankenbrandt, C. (1991), An extension to the theory of convergence and a proof of the time complexity of genetic algorithms, in *Foundations of Genetic Algorithms*, Morgan Kaufmann, San Mateo, CA.

Baldwin, R. (1989), How does protein folding get started?, *Trends Biochem. Sci.* **14**, 291–294.

Baldwin, R. and Eisenberg, D. (1987), Protein stability, in *Protein Engineering*, Alan R. Liss, Inc., NY.

Billeter, M., Havel, T. F., and Wüthrich, K. (1987), The ellipsoid algorithm as a method of determination of polypeptide conformations from experimental distance constraints and energy minimization, *J. Comp. Chem.* **8**(2), 132–141.

Bohr, H., Bohr, J., Brunak, S., Cotterill, R., Lautrup, B., Norskov, L., Oslen, O., and Petersen, S. (1988), Protein secondary structure and homology by neural networks: The a-helices in rhodopsin, *FEBS Letters* **241**, 223–228.

Booker, L. (1987), Improving search in genetic algorithms, in *Genetic Algorithms and Simulated Annealing*, Morgan Kaufmann, San Mateo, CA.

Brooks, B., Bruccoleri, R., Olafson, B., States, D., Swaminathan, S., and Karplus, M. (1983), CHARMM: A program for macromolecular energy, minimization, and dynamics calculations, *J. Comp. Chem.* **4**, 187–217.

Brooks, C., III, Karplus, M., and Montgomery Pettitt, B. (1988), *Proteins: A Theoretical Perspective of Dynamics, Structure and Thermodynamics*, Wiley Interscience, New York.

Chakrabarty, A., Schellman, J. A., and Baldwin, R. L. (1991), Large differences in the helix propensities of alanine and glycine, *Nature* **351**, 586–588.

Chou, P. and Fasman, G. (1974), Prediction of protein conformation, *Biochemistry* **13**, 222–244.

Cleveland, G. and Smith, S. (1989), Using genetic algorithms to schedule flow shop releases, in *Proceedings of the Third International Conference on Genetic Algorithms*, Morgan Kaufmann, San Mateo, CA.

Cohen, F. and Kuntz, I. (1989), Tertiary Structure Prediction, in *Prediction of Protein Structure and the Principles of Protein Conformation*, Plenum Press, New York.

Crippen, G. (1977), A novel approach to calculation of conformation: distance geometry, *J. Comp. Phys.* **24**, 96–107.

Crippen, G. and Havel, T. (1990), Global energy minimization by rotational energy embedding, *J. Chem. Inf. Comp. Sci.* **30**, 222–227.

Davidor, Y. (1989), Analogous crossover, in *Proceedings of the Third International Conference on Genetic Algorithms*, Morgan Kaufmann, San Mateo, CA.

Davis, L. (1989), Adapting operator probabilities in genetic algorithms, in *Proceedings of the Third International Conference on Genetic Algorithms*, Morgan Kaufmann, San Mateo, SA.

Davis, T. and Principe, J. (1991), A simulated annealing like convergence theorem for the simple genetic algorithm, in *Proceedings of the Fourth International Conference on Genetic Algorithms*, Morgan Kaufmann, San Mateo, CA.

Dudek, J. and Scheraga, H. (1990), Protein structure prediction using a combination of sequence homology and global energy minimization I. Global energy minimization of surface loops, *J. Comp. Chem.* **11**, 121–151.

Eisenberg, D. and McLachlan, A. D. (1986), Solvation energy in protein folding and binding, *Nature* **319**, 199–203.

Fasman, G. (1989), Development of the prediction of protein structure, in *Prediction of Protein Structure and the Principles of Protein Conformation*, Plenum Press, New York.

Ferrin, T. E. (1988), The MIDAS display system, *J. Mol. Graphics* **6**, 13–27.

Fogarty, T. (1989), Varying the probability of mutation in the genetic algorithm, in *Proceedings of the Third International Conference on Genetic Algorithms*, Morgan Kaufmann, San Mateo, CA.

Garnier, J., Osguthorpe, D., and Robson, B. (1978), *J. Mol. Biol.* **120**, 97–120.

Gibrat, J., Garnier, J., and Robson, B. (1987), Further developments of secondary structure prediction using information theory: New parameters and consideration of residue pairs, *J. Mol. Biol.* **198**, 425–443.

Goldberg, D. (1987), Genetic algorithms with sharing for multimodal function optimization, in *Genetic Algorithms and Their Applications: Proceedings of the Second International Conference on Genetic Algorithms*, Lawrence Erlbaum, Hillsdale, NJ.

Goldberg, D. (1989), *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison-Wesley, San Mateo, CA.

Grefenstette, J. and Baker, J. (1989), How genetic algorithms work: a critical look at implicit parallelism, in *Proceedings of the Third International Conference on Genetic Algorithms*, Morgan Kaufmann, San Mateo, CA.

Holland, J. (1975), *Adaptation in Natural and Artificial Systems*, University of Michigan Press, Ann Arbor, MI.

Holley, L. and Karplus, M. (1989), Protein secondary structure prediction with a neural network, *PNAS* **86**, 152–156.

Jaenicke, R. (1991), Protein folding: Local structures, domains, subunits, and assemblies, *Biochemistry* **30**(13), 3147–3161.

Janikow, C. and Michalewicz, Z. (1991), An experimental comparison of binary and floating point representations in genetic algorithms, in *Proceedings of the Fourth International Conference on Genetic Algorithms*, Morgan Kaufmann, San Mateo, CA.

Khechinashvili, N. (1990), Thermodynamic properties of globular proteins and the principle of stabilization of their native structure, *Biochimica et Biophysica Acta.* **1040**, 346–354.

Kneller, D., Cohen, F., and Langridge, R. (1990), Improvements in protein secondary structure prediction by an enhanced neural network, *J. Mol. Biol.* **214**, 171–182.

Lambert, M. and Scheraga, H. (1989a), Pattern recognition in the prediction of protein structure. I.

Tripeptide conformational probabilities calculated from the amino acid sequence, *J. Comp. Chem.* **10**, 770–797.

Lambert, M. and Scheraga, H. (1989b), Pattern recognition in the prediction of protein structure. II. Chain conformation from a probability-directed search procedure, *J. Comp. Chem.* **10**, 798–816.

Lambert, M. and Scheraga, H. (1989c), Pattern recognition in the prediction of protein structure. III. An importance-sampling minimization procedure, *J. Comp. Chem.* **10**, 817–831.

Levitt, M. (1983), Protein folding by restrained energy minimization and molecular dynamics, *J. Mol. Biol.* **170**, 723–764.

Li, Z. and Scheraga, H. (1987), Monte Carlo-minimization approach to the multiple-minima problem in protein folding, *PNAS* **84**, 6611–6615.

Li, Z. and Scheraga, H. (1988), Monte Carlo recursion evaluation of free energy, *J. Phys. Chem.* **92**, 2633–2636.

Lipton, M. and Still, W. (1988), The multiple minimum problem in molecular modeling. Tree searching internal coordinate conformation space, *J. Comp. Chem.* **4**, 343–355.

Lucasius, C. B. and Kateman, G. (1989), Application of genetic algorithms in chemometrics, in *Proceedings of the Third International Conference on Genetic Algorithms*, Morgan Kaufmann, San Mateo, CA.

Lucasius, C. B., Blommers, M. J. J., Buydens, L. M. C., and Kateman, G. (1990), A genetic algorithm for conformational analysis of DNA, in *Handbook of Genetic Algorithms*, Van Nostrand Reinhold, New York.

Marqusee, S., Robbins, V. H., and Baldwin, R. L. (1989), Unusually stable helix formation in short alanine-based peptides, *PNAS* **86**, 5286–5290.

Momany, F., McGuire, R., Burgess, A., and Scheraga, H. (1975), Energy parameters in polypeptides. VII. Geometric parameters, partial atomic charges, nonbonded interactions, hydrogen bond interactions, and intrinsic torsional potentials for the naturally occurring amino acids, *J. Phys. Chem.* **79**(22), 2361–2381.

Montana, D. and Davis, L. (1989), Training feedforward neural networks using genetic algorithms, in *Proceedings of the Third International Joint Conference on Artificial Intelligence*, Morgan Kaufmann, San Mateo, CA.

Mühlenbein, H. (1989), Parallel genetic algorithms, population genetics, and combinatorial optimization, in *Proceedings of the Third International Conference on Genetic Algorithms*, Morgan Kaufmann, San Mateo, CA.

Ooi, T., Oobatake, M., Nemethy, G., and Scheraga, H. A. (1987), Accessible surface areas as a measure of the thermodynamic parameters of hydration of peptides, *PNAS* **84**, 3086–3090.

Piela, L. and Scheraga, H. (1989), The multiple minima problem in the conformational analysis of molecules. Deformation of the potential energy hypersurface by the diffusion equation method, *J. Phys. Chem.* **93**, 3339–3346.

Purisima, E. and Scheraga, H. (1987), An approach to the multiple-minima problem in protein folding by relaxing dimensionality: Tests on enkephalin, *J. Mol. Biol.* **196**, 697–709.

Qian, N. and Sejnowski, T. (1988), Predicting the secondary structure of globular proteins using neural network models, *J. Mol. Biol.* **202**, 865–884.

Radcliffe, N. and Wilson, G. (1990), Natural solutions give their best, *New Scientist* **126**, 47–50.

Richards, F. (1991), The Protein Folding Problem, *Sci. Am.* **264**, 54–63.

Richardson, J. S. (1981), The anatomy and taxonomy of protein structure, *Adv. Protein Chem.* **34**, 167–284.

Richardson, J. S. and Richardson, D. C. (1990), The origami of proteins, in *Protein Folding: Deciphering the Second Half of the Genetic Code*, AAAS Press, Washington, D.C.

Ripoli, D. R., Vasquez, M., and Scheraga, H. A. (1991), The electrostatically driven monte carlo method: Application to conformational analysis of decaglycine, *Biopolymers* **31**, 319–330.

Schaffer, J. and Morishima, A. (1987), An adaptive crossover distribution mechanism for genetic algorithms, in *Genetic Algorithms and Their Applications: Proceedings of the Second International Conference on Genetic Algorithms*, Lawrence Erlbaum, Hillsdale, NJ.

Sirag, D. and Weisser, P. (1987), Toward a unified thermodynamic genetic operator, in *Genetic Algorithms and Their Applications: Proceedings of the Second International Conference on Genetic Algorithms*, Lawrence Erlbaum, Hillsdale, NJ.

Skolnick, J. and Kolinski, A. (1990), Simulations of the folding of a globular protein, *Science* **250**, 1121–1125.

Stolorz, P., Lapedes, A., and Xia, Y. (1991), Predicting protein secondary structure using neural net and statistical methods, to appear in *J. Mol. Biol.*

Tanese, R. (1989), Distributed genetic algorithms, in *Proceedings of the Third International Conference on Genetic Algorithms*, Morgan Kaufmann, San Mateo, CA.

Vajda, S. and Delisi, C. (1990), Determining minimum energy conformations of polypeptides by dynamic programming, *Biopolymers* **29**, 1755–1772.

Vasquez, M., Nemethy, G., and Scheraga, H. A. (1983), Computed conformational states of the 20 naturally occurring amino acid residues and of the prototype residue a-aminobutyric acid, *Macromolecules* **16**, 1043–1049.

Walbridge, C. (1989), Genetic algorithms: What computers can learn from Darwin, *Technology Review* **92**, 46–53.

Wayner, P. (1991), Genetic algorithms, *Byte* **16**, Jan., 361–364.

Weiner, S., Kollmann, P., Nguyen, D., and Case, D. (1986), An all atom force field for simulations of proteins and nucleic acids, *J. Comp. Chem.* **7**(2), 230–252.

Whitley, D. and Kauth, J. (1988), Sampling long schemata in genetic algorithms, Tech Report CS-88-105, Computer Science Dept., Colorado State Univ.

Whitley, D. and Hanson, T. (1989a), The GENITOR algorithm and selective pressure: why rank-based allocation of reproductive trials is best, in *Proceedings of the Third International Conference on Genetic Algorithms*, Morgan Kaufmann, San Mateo, CA.

Whitley, D. and Hanson, T. (1989b), Optimizing neural nets using faster, more accurate genetic search, in *Proceedings of the Third International Conference on Genetic Algorithms*, Morgan Kaufmann, San Mateo, CA.

Whitley, D., Starkweather, T., and Bogart, C. (1990a), Genetic algorithms and neural networks: optimizing connections and connectivity, *Parallel Computing* **13**, 347–361.

Whitley, D. and Starkweather, T. (1990b), GENITOR II: a distributed genetic algorithm, *J. Exp. Theor. Artif. Intell.* **2**, 189–214.

Wilson, S. and Cui, W. (1990), Applications of simulated annealing to peptides, *Biopolymers* **29**, 225–235.