Application of Divide and Conquer Extended Genetic Algorithm to tertiary protein structure of chymotrypsin inhibitor-2

doi:10.1533/abbi.2006.0038

A. Alfaro[1], M. Doan[2], J. Finke[3], M. Galdes[4] and M. Zohdy[5]

[1]*Mechanical Engineering Department, Oakland University, Rochester, Rochester, MI, USA*
[2]*Statistics and Biology Department, Human Biology Department, Michigan State University, East Lansing, MI, USA*
[3]*Chemistry Department, Oakland University, Rochester, MI, USA*
[4]*Materials Science and Engineering Department, Michigan State University, East Lansing, MI, USA*
[5]*Electrical and Computer Engineering Department, Oakland University, Rochester, MI, USA*

Abstract: Determining the method by which a protein thermodynamically folds and unfolds in three-dimension is one of the most complex and least understood problems in modern biochemistry. Misfolded proteins have been recently linked to diseases including Amyotrophic Lateral Sclerosis and Alzheimer's disease. Because of the large number of parameters involved in defining the tertiary structure of proteins, based on free energy global minimisation, we have developed a new Divide and Conquer (DAC) Extended Genetic Algorithm. The approach was applied to explore and verify the energy landscape of protein chymotrypsin inhibitor-2.

Key words: Protein folding, Genetic Algorithm, Genetic Algorithm for protein folding, protein folding of chymotrypsin inhibitor-2, protein folding of chymotrypsin inhibitor-2 using Genetic Algorithms.

INTRODUCTION

At this time, the folding and unfolding processes as well as intermediate stages are not completely available for most proteins (Yap and Cosic 1999). Therefore, it is very important that we choose a relatively simple protein that is very well known so that it can be used as a model to test our Genetic Algorithm (GA). Chymotrypsin inhibitor-2 (CI-2) is one of the select few proteins with a known folding–unfolding process and structure (Jackson and Fersht 1991). CI-2 is a serine protease inhibitor found in barley seeds. CI-2 has been found to have one of the simplest known folding pathways partly due to the fact that it lacks disulfide bonds. It folds in a two-step procedure, with no intermediate stages present to complicate the necessary computations; hence, deficiency of these states allows computer simulations to run smoothly due to a reduced number of free energy minima. At the transition state, it is believed that the central α-helix and the β-sheets have been folded, while the ends of the protein remain unfolded.

The optimisation capabilities of our GA are used to help us minimise the free energy of intramolecular interactions within the protein. We have chosen to use three energy representations of these interactions that are directly linked to tertiary structure of proteins. The interactions are dihedral angles, Lenard Jones (LJ) energies of attraction and the effects of entropy and temperature on free energy. Each of these interactions is represented mathematically by known formula and minimised by our GA. The reason for minimisation is due to the fact that proteins fold at their lowest state of free energy (Ogura *et al*. 2003). Figure 1 illustrates the change in free energy as a protein folds and unfolds. An increase in free energy will cause proteins to unfold. However, the large amount of parameters needed to specify free energy is one of the main difficulties encountered when attempting to apply the GA to the protein CI-2.

A significant number of optimisation programs using GAs have been previously used to predict the secondary and tertiary structures of proteins. These programs mainly focus on a particular characteristic of proteins such as dihedral energy interactions and the hydrophobic–hydrophilic properties of amino acids. Some GAs use a pre-produced library of possible tertiary structures as the building blocks, which are then mutated and recombined to make a whole

Corresponding Author:
M. Zohdy
Electrical & Systems Engineering Department
148 DHE, Oakland University, Rochester, MI, 48309, USA
Tel: +1-248-370-2234; Fax: +1-248-370-4463
Email: zohdyma@oakland.edu

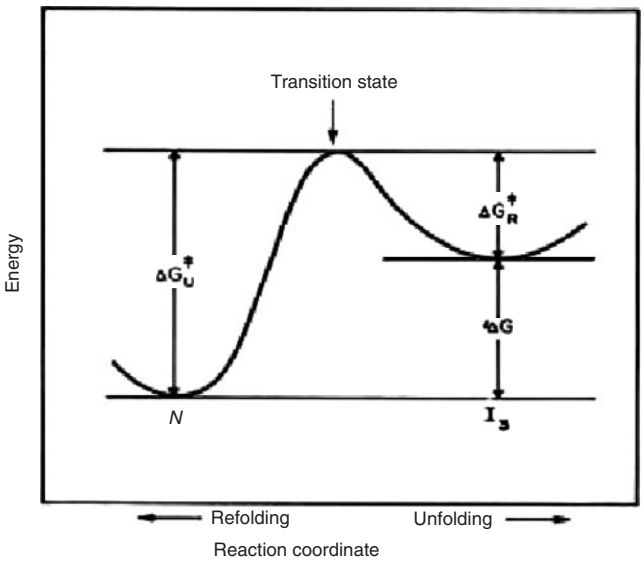

Figure 1 Illustration of typical change in free energy as a protein folds and unfolds.

new structure that is evaluated by a fitness function. In addition, different methods of fitness, recombination, mutation and reproduction have been implemented in our algorithm (Hoque *et al*. 2005; Song *et al*. 2005; Hiroyasu *et al*. 2003).

GENETIC ALGORITHM OPTIMISATION

A *GA* is a guided random search technique that is used to obtain solutions to global optimisation problems. This technique is bioinspired by steps in the natural science of genetics. The search begins with a random population of suitable individuals, each encoded by a chromosome of strings, trees, binary or other codes (Day *et al*. 2003). A fitness function must be defined in the program so that the quality of each possible individual solution can be determined. This allows the GA to move away from less fit solutions and towards the most optimal ones (Wang *et al*. 1994). Once the fitness of each individual is determined, the selection process begins. To create members for the next generation, individuals from the current generation are selected on the basis of their fitness. Individuals with higher fitness usually have a higher probability of being selected; thus, ensuring that solutions do not get stuck in a local optimum (Lin *et al*. 2005). To keep the population of possible solutions diverse, GAs implement recombination of a selected random pair of individuals. A crossover point is randomly chosen, and the two individuals are recombined by swapping segments of their codes to form two new members for the next generation. This process continues until the next generation has a full population of individuals. After recombination, mutation of some of the weak members occurs with small probability, ensuring that the GA explores a wider fitness landscape. The process of mutation involves a random change of genes in the chromosomes of some of the individuals. The GA continues to select, determine fitness, recombine and mutate each generation of individuals until a global solution is found.

These traits make the GA a very effective optimisation technique for free energies involved in protein folding. The ability to work with a population of simultaneous possible solutions instead of just one solution makes the GA an effective tool to determine three-dimensional protein structure.

Extended Genetic Algorithm

The Extended Genetic Algorithm (EGA) is a modification of the basic GA to serve as a more suitable tool for protein folding optimisation. In this algorithm, multiple and swapping mutations as well as multiple recombination are used and all the parameters involved are annealed. *Annealed parameters* simply mean that the rates of recombination and mutation gradually diminish as each generation of individuals is created. At the beginning of a run, the algorithm must perform a wide and thorough search of the free energy landscape being optimised, so that it can locate all the minima and then converge to the global minimum. Once the algorithm starts to converge towards the global minimum, there is no more need for so much recombination and mutation. So we reduce the rates at which this happens to reduce the computational time and get a more accurate convergence. The code is also designed in such a way that certain elite individuals who are chosen by the algorithm can be reproduced into the next generation.

Divide and Conquer Extended Genetic Algorithm

Unlike previous work using the GA, we are developing an algorithm that will take into account all significant energy interactions such as dihedral angles using explicit objective fitness, LJ forces (attractive forces) using implicit

 doi:10.1533/abbi.2006.0038

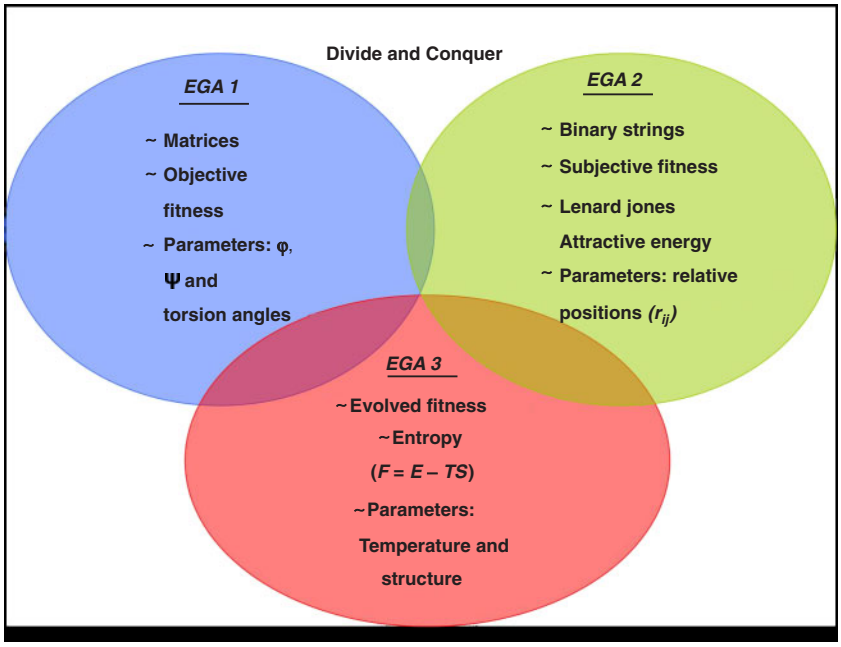

Figure 2 Portrayal of the DAC-EGA and its three components.

subjective fitness and repulsive forces. Objective fitness judges individuals in the GA using solely an explicit mathematical formula. Subjective fitness involves a mathematical formula, but, in addition, the individuals are examined by experts who determine which individuals are fit according to their judgement. We also account for temperature and entropy in the algorithm by co-evolving the GA free energy fitness. To deal with the complexity of the problem and the large number of parameters, the concept of divide and conquer has been applied to our algorithm. This concept subdivides the protein-folding problem into simpler related problems. The Divide and Conquer Extended Genetic Algorithm (DAC-EGA) divides the problem into three different modules, each of which uses its own GA and deals with one of the three significant energy interactions we take into account. Figure 2 portrays the DAC-EGA, displaying the details of our divisions. The first module minimises the energy interactions of dihedral angles, using matrices to encode individuals representing torsion angles as parameters and an objective fitness function. The second module minimises the LJ energy of attraction, using binary strings to encode individuals, denoting relative positions and a subjective fitness function. The final module deals with the effects that entropy and temperature have on energy. All of these modules use reproduction, recombination and mutation of individuals in order to search the free energy landscape. The modules are all annealed so that the rates of recombination and mutation are decreased since their impact is not as crucial when the solutions converge to the global minimum. In the future, improving the efficiency and accuracy of the DAC-EGA will facilitate the study and understanding of the folding and unfolding processes of unknown proteins.

COMPUTATIONAL ANALYSIS

Algorithm parameters

DAC-EGA includes new annealed rates of reproduction, recombination and mutation. These rates were based on the exponential and Poisson equations. The exponential equation is inspired by the relationship between the temperature and the hardness of a metal being annealed (Fig. 3). Similar relationships were implemented for the number of generations and rate of the operators in the EGA and DAC-EGA. As the number of generations increase and the population moves closer to the solution, there is less need for recombination and mutation. In the case of the reproduction operator, the inverse of this relationship was used. As the population approaches the solution, it is most beneficial to reproduce a larger percentage of the fittest individuals so that they will converge easily. As mentioned earlier, recombination and mutation were also used. The number of tested individuals in the population was 100, and we used them to successfully optimise a function of approximately 12 variables that represent 12 amino acids from a small segment of the protein.

PRELIMINARY COMPUTATIONAL RESULTS

It was found that the number of members in the initial population directly affects the performance of the algorithm. A small initial population does not provide an extensive search space, so it increases the chance of being limited to a local optimum. In contrast, increasing the number of individuals in the initial population expands the search space that greatly improves the chances of finding a global optimum.

 doi:10.1533/abbi.2006.0038

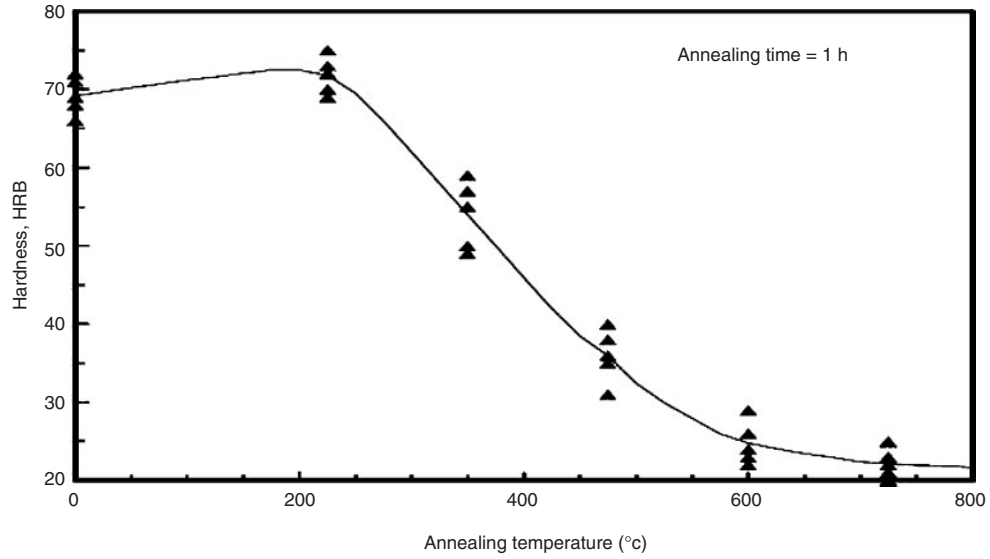

Figure 3 A plot that demonstrates the concept of annealing by plotting the relationship between hardness and the temperature of a metal as it is heated.

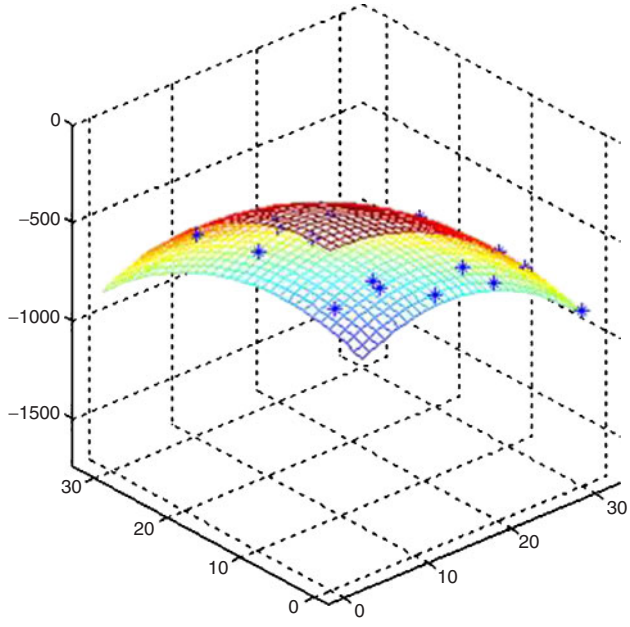

Figure 4 Initial population scattered randomly on a surface.

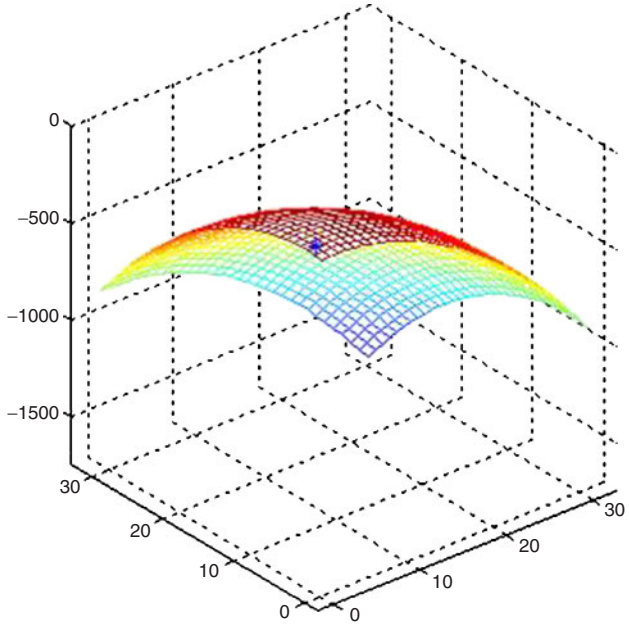

Figure 5 After 50 iterations, the population has successfully converged to the global maximum.

The EGA effectiveness was demonstrated by optimising sets of simple functions of n variables. For the purpose of visual representation, we show our data for the optimisation of two variables as illustrated in Figure 4. The initial population is scattered near the surface to be maximised.

As the EGA performs a finite number of iterations, the population starts to converge towards the global optimum. Figure 5 shows the correctly converged population after 50 iterations.

DAC-EGA CI-2 results

The DAC-EGA was found to be most effective when the attached optimisation results were obtained via a customised MATLAB program. It begins by generating a suitable binary-coded population. During the selection phase, the program chooses the elite members of each generation, objectively or subjectively, on the basis of their fitness rank and then recombines them with other randomly chosen common members. The EGA modifies the above by incorporating iteration-dependent annealed rates. Also at the extended recombination phase, the elite members combine with three or more randomly chosen individuals in order to promote improvement for the next generation. This process repeated itself 1,000 times, and showed that it takes enough generations to correctly converge the individuals to the global minimum.

 doi:10.1533/abbi.2006.0038

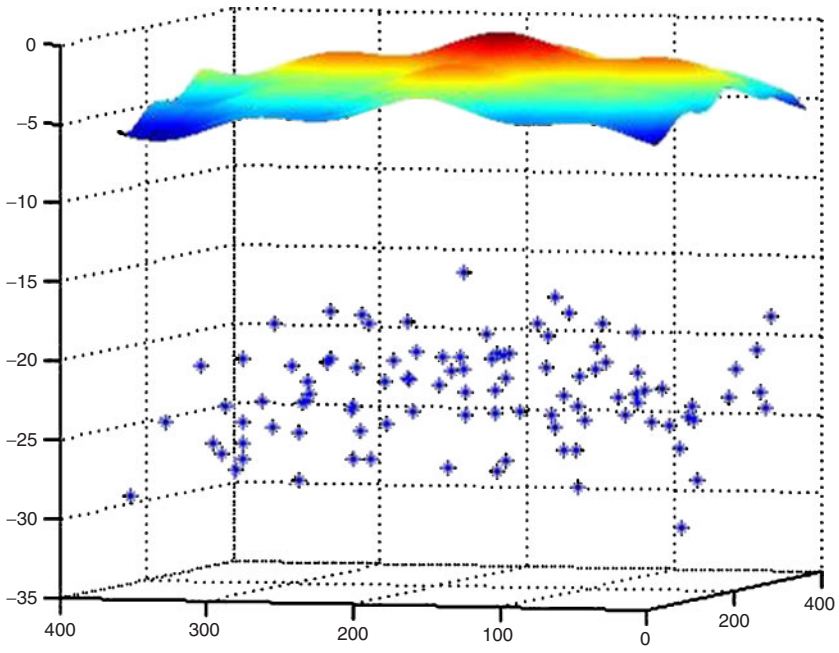

Figure 6 Initial population (CI-2 dihedral angles) scattered in three-dimensional space.

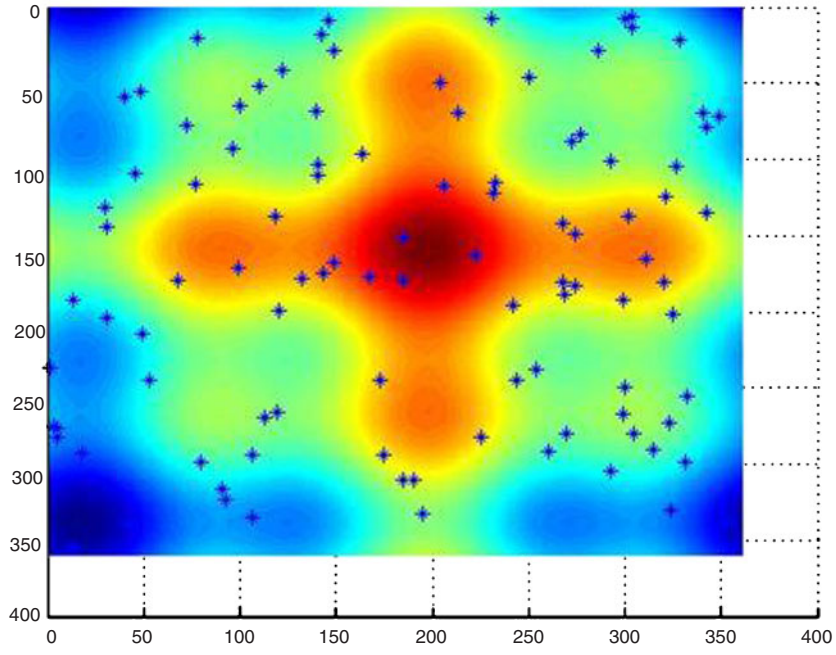

Figure 7 Initial population (CI-2 dihedral angles) as seen from the top view.

Dihedral angles

Dihedral angles are angles made by a group of molecules relative to one another in three-dimensional space. In this module, we used a formula that relates free energy to the dihedral and torsion angles of molecules. In our function, φ represents the variable (angle) we are trying to minimise and φ_0 the average angle for the variable that has been determined experimentally. When minimising this energy function along with 12 other variables like we did using our GA, we obtain one possible value for each of those variables. The obtained value represents the angle of the corresponding variable that yields the lowest free energy in the presence of the other variables. That is how this module measures fitness objectively, by solely using the formula. These data will ultimately map into a structure in combination with the other two modules. The DAC-EGA successfully minimised the dihedral energy function (Equation (1)) for the first 12 torsion angles of CI-2. This was validated through comparison of our results with the

 doi:10.1533/abbi.2006.0038

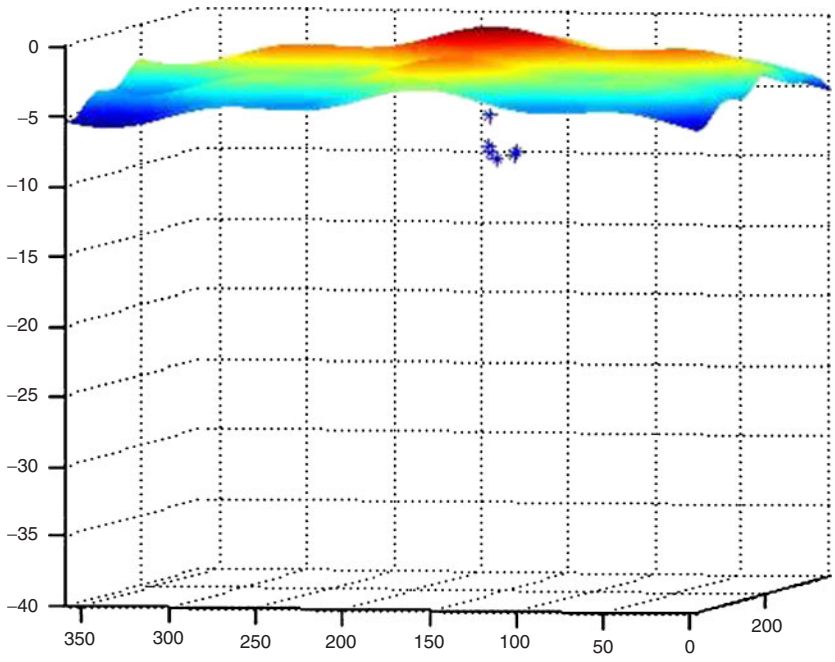

Figure 8 Final population (CI-2 dihedral angles) correctly converged to the global optimum.

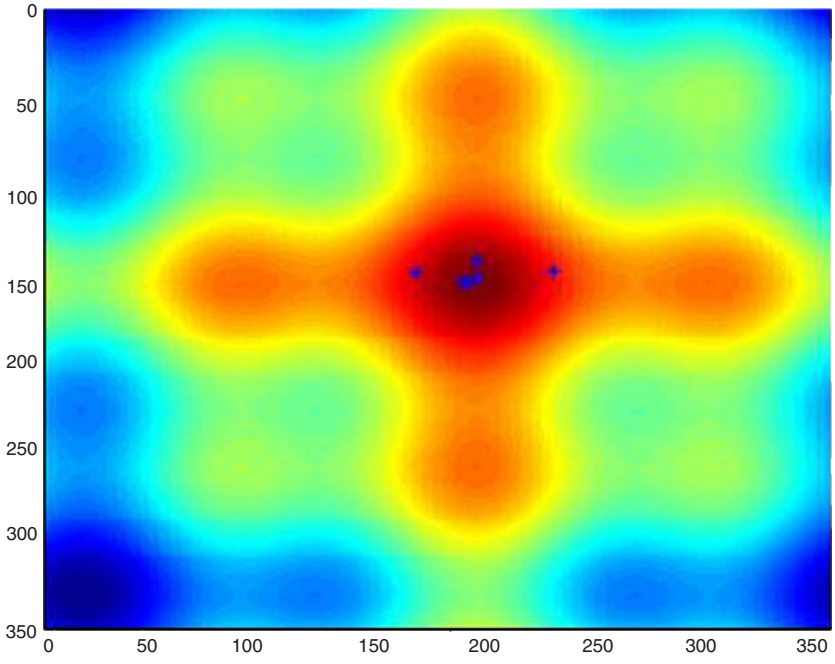

Figure 9 Top view (CI-2 dihedral angles) of the correctly converged population.

already known (experimentally determined) data for CI-2. Figures 6–9 show the energy surface for the first two variables (dihedral angles of CI-2) and how the initial population of the algorithm converges after it runs through 1,000 iterations (Nakashima *et al*. 2005).

$$E_{\text{dihedral}} = \sum_{\text{dihedrals}} [\varepsilon_{\varphi 1}[1 - \cos(\varphi - \varphi_0)] + \varepsilon_{\varphi 2}[1 - \cos(3(\varphi - \varphi_0))]]. \quad (1)$$

Lenard Jones

The second module deals with *LJ energies*. These are the energies of attraction between different molecules in the protein chain. This algorithm deals with LJ energies of attraction by using the formula as in Equation (2). This formula works with relative distances (in Å) between atoms. In this formula, σ represents the distance for a certain variable that will give a minimum amount of energy in the presence of the other variables, and r represents the average

 doi:10.1533/abbi.2006.0038

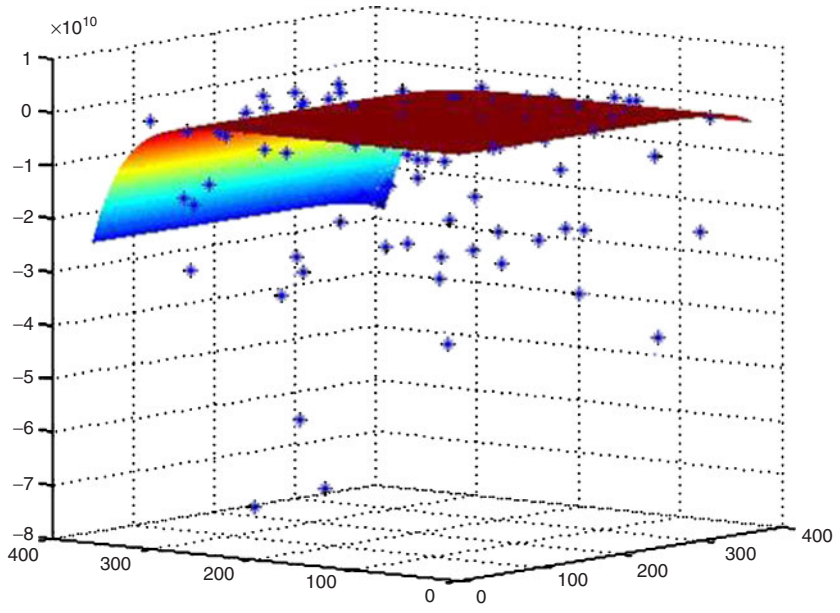

Figure 10 Initial populations (CI-2 Lenard Jones) scattered in three-dimensional space.

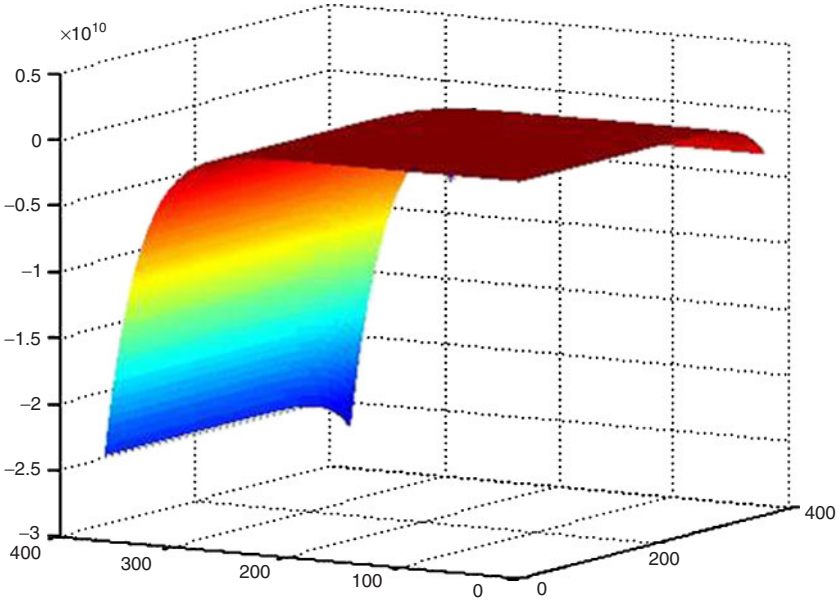

Figure 11 Final population (CI-2 Lenard Jones) correctly converged to the global optimum.

distance between two molecules in a chain (0.5 nm), which we have kept constant in the function. This module focuses on the minimisation of the relative positions between certain amino acids that are attracted to each other. We use the LJ function as a fitness function as well as the judgment of an expert who could help us determine which solutions are fit and which are not. The graphs (Figs. 10 and 11) show the LJ energy function for the first two amino acids to which attractive energy was applied. The obtained solutions were in agreement with experimental data, which show that this information and the information obtained above could be combined and interpreted to develop a three-dimensional structure of the protein.

$$E_{\mathrm{LJ}} - \sum_{|i-j|>3} \varepsilon_{\mathrm{LJ}} \left[5 \left(\frac{\sigma_{ij}}{r_{ij}} \right)^{12} - 6 \left(\frac{\sigma_{ij}}{r_{ij}} \right)^{10} \right]. \qquad (2)$$

Entropy

In the third module, we took entropy into account, and examined the effect it had on the free energy of the protein. Entropy and temperature affect the free energy of a protein as described by the following relation:

$$G = E - TS,$$

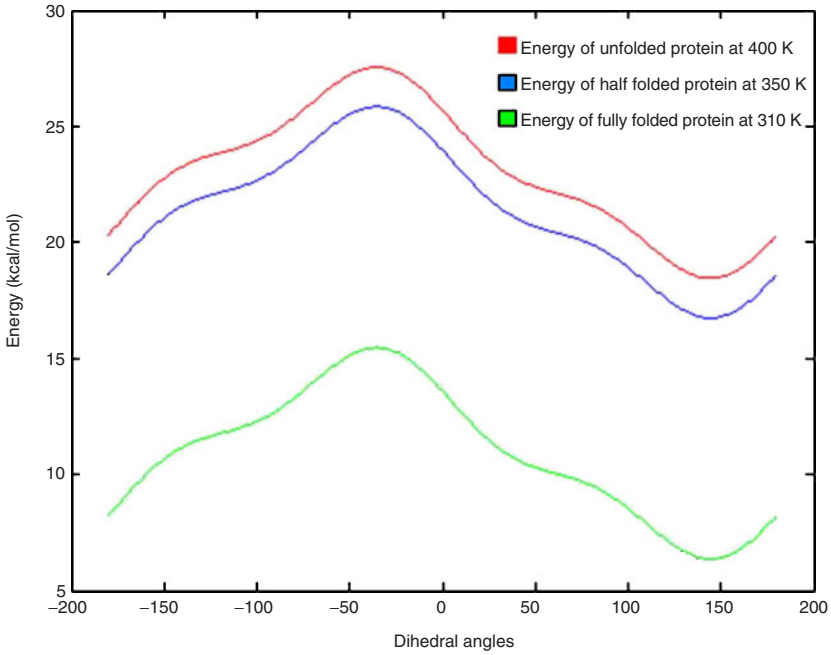

Figure 12 This plot illustrates that exposing the protein to high temperatures will increase the entropy of the protein, as well as free energy, and the protein will denature as a result.

where G indicates free energy; E, total energy; T, temperature; and S, entropy. Our experiments used an evolved formula that follows the basic principles of protein folding to represent entropy (S).

$$S = Q\left(\frac{Q}{Q_{\max^2}}\right),$$

where $Q_{\max}$ indicates maximum number of contacts possible (amino acids at a distance of $\leq$0.5-nm apart); Q, number of contacts at a particular state.

Experimentally, it has been determined that CI-2 is half-folded at a temperature of 350 K. The graph in Figure 12 shows the effect that temperature has on the entropy, and the decrease in free energy.

When this protein is exposed to high temperatures, it denatures, and therefore has no native contacts ($S = 0$); this means that TS will not affect the free energy of the protein. This module uses a co-evolved fitness function, which means that the function itself changes depending on the conditions of the protein. As the temperature decreases, the protein starts to fold. As a result, the value of S starts to increase, and this in turn decreases the free energy of the protein because the TS component of the equation is being subtracted from the total energy of the protein as shown in Figure 12.

CONCLUSIONS

The DAC-EGA proved to be a very effective tool for the global optimisation of multiple energy parameters in the three-dimensional structure of CI-2 protein. The solution concept of divide and conquer simplified our problem greatly because this approach greatly reduced the number of parameters taken into account simultaneously. This was augmented by integrating new concepts such as annealing, multiple and swapping mutations, multiple recombination and reproduction, as well as objective, subjective and evolved fitness descriptions. DAC-EGA was successful in realistic large multivariable protein structure optimisation based on objective, subjective and evolved free energy. Our GA optimises sets of variables simultaneously but it can be further expanded, which will be very promising, especially when applied towards larger proteins. The structure of CI-2 has already been determined, so we verified our results to be correct within an acceptable error margin. The number of variables in the algorithm can be increased to however as many needed as possible, hence it can be applied to proteins with large structures. The DAC-EGA can be used as an optimisation tool to extract important information such as optimal angles and relative distances between molecules in a folded protein. Correctly interpreting this information will allow us to map the tertiary structure of important proteins that are not well known today, as well as link various diseases to protein mis-folding (Buj and Sundarraj 2005).

ACKNOWLEDGMENTS

The authors thank the readers and revisers of their article as well as the NIH and NSF for their supplied grant, 0609152, and support.

REFERENCES

Bui TN, Sundarraj G. 2005. An efficient genetic algorithm for predicting protein tertiary structures in the 2D HP model. In Genetic and Evolutionary Computation Conference, p. 385–92.

Day RO, Lamont GB, Pachter R. 2003. Protein structure prediction by applying an evolutionary algorithm. In International Parallel and Distributed Processing Symposium, San Miguel Regla, Hidalgo, Mexico, August 25–28, 2006, p. 8.

Hiroyasu T, Miki M, Iwahashi T, *et al.* 2003. Dual individual distributed genetic algorithm for minimizing the energy of protein tertiary structure. In SICE 2003 Annual Conference, vol. 3, p. 2756–61.

Hoque MdT, Chetty M, Dooley LS. 2005. A new guided genetic algorithm for 2D hydrophobic-hydrophilic model to predict protein folding. In 2005 IEEE Congress on Evolutionary Computation, vol. 1, p. 259–66.

Jackson SE, Fersht AR. 1991. Folding of chymotrypsin inhibitor 2. Part 1: Evidence for a two-state transition. *Biochemistry*, 30:10428–35.

Lin H-N, Wu K-P, Chang J-M, *et al.* 2005. GANA – A genetic algorithm for NMR backbone resonance assignment. In IEEE Computational Systems Bioinformatics Conference, p. 218–9.

Nakashima N, Matsubara A, Ono I, *et al.* 2005. A genetic algorithm taking account of substructures for NMR three-dimensional protein structure determination. In 2005 IEEE Congress on Evolutionary Computation, vol. 2, p. 1761–8.

Ogura S, Aoi K, Hiroyasu T, *et al.* 2003. Energy minimization of protein tertiary structures by local search algorithm and parallel simulated annealing using genetic crossover. In 2003 Congress on Evolutionary Computation, vol. 3, p. 1933–40.

Song J, Cheng J, Zheng T, *et al.* 2005. A novel genetic algorithm for HP model protein folding. In Sixth International Conference on Parallel and Distributed Computing, Applications and Technologies, p. 935–7.

Wang L-H, Kao C-Y, Ouh-Young M, *et al.* 1994. Using an annealing genetic algorithm to solve global energy minimization problem in molecular binding. In Sixth International Conference on Tools with Artificial Intelligence, p. 404–10.

Yap A, Cosic I. 1999. Application of genetic algorithm for predicting tertiary structures of peptide chains. *Eng Med Biol.* In Proceedings of the First Joint BMES/EMBS Conference. 1999 IEEEEngineering in Medicine and Biology 21st Annual Conference and the 1999 Annual Fall Meeting of the Biomedical Engineering Society, vol. 2, p. 1214.

 doi:10.1533/abbi.2006.0038

