A Simple Approach to Protein Structure Prediction Using Genetic Algorithms

Katie Braden
Electrical Engineering Department
Stanford University
Stanford, CA 94305
kbraden@stanford.edu

Abstract: We develop a novel protein structure prediction technique using genetic algorithms. This implementation improves upon past work by Unger and Moult[1], wherein amino acid residues were modeled with a single characteristic, hydrophobicity, and protein structures were assumed to be two-dimensional. We model proteins as three-dimensional chains of residues with hydrophobicity, side-chain size, and charge characteristics. Employing this model, we use the Genesis Genetic Algorithm to optimize protein structure. This protein structure prediction technique is tested with two short proteins and demonstrated to yield optimized structure.

Introduction:

Chemists, physicists, biologists, and computer scientists have attempted to develop a realistic framework for predicting protein topology for nearly four decades[2]. Many protein structure prediction frameworks, from quantum physical principle-based analyses[3] to environmental variable-based approaches[4], have been proposed. However, none of these has successfully predicted the structure of a wide variety of types and sizes of proteins accurately. We have developed a simple approach to short-protein structure prediction, employing Genetic Algorithms (GAs) for optimization, that improves upon the method proposed by Unger and Moult in 1993[1]. While Unger and Moult rely upon a two-dimensional protein model in which residues have only a single binary characteristic, we model proteins in three-dimensional integer space, and residues with characteristics of hydrophobicity, charge, and side-chain size. As a result, the model we present predicts protein structure more realistically and accurately than Unger and Moult's model, while maintaining computational simplicity and speed not present in more complex models. Our model is accurate and useful as a first-order structure prediction tool.

Background:

Traditionally, researchers developing protein folding algorithms have attempted to predict the protein folding pathway in order to determine structure. In predicting folding pathways, we examine patterns of residues in other, similar proteins and build new proteins from given sequences with regard to these characteristic patterns. Pathway methods have not successfully predicted a wide-variety of proteins accurately.

Therefore, researchers have recently begun to develop statistical perspectives upon protein folding, deriving multi-dimensional landscapes of folding probability for entire proteins, as well as common protein sub-structures[2]. These statistical frameworks take into account (1) local characteristics, such as side-chain size and charge, and (2) global characteristics, such as hydrophobicity, bond structure, interaction strength, and probabilities of given conformations. These conformational probabilities are calculated from similar protein or substructure training sets. In addition, environmental variables such as temperature and the presence of a chemical can be taken into account in these landscapes. Landscape analysis represents a shift from previous efforts, which sought to optimize energetic properties of the folded protein and failed to incorporate knowledge that proteins do not always fold in energy-optimized conformations.

GAs are especially well-suited to solving landscape-type analysis and optimization problems[1]. GAs are different from other search methods in that they use probabilistic rules to optimize solutions. That is, fitness measures are supplied by the user, and assignment of these values to a population if potential solutions generates a statistically ranked population. More fit individuals are recombined an mutated to form a new population generation, yielding a more statistically fit population at large. Gradually, statistically more fit individuals are produced until an individual that is maximally fit emerges. This correlates nicely with the probabilistic landscape approach to protein structure prediction. Well-optimized sub-units in a population are repeatedly retained and re-combined, based upon fitness characteristics specified by the user, to find a most-fit individual. If the user provides a realistic model of protein and residue characteristics and their contributions to structure fitness, GAs are likely to predict the correct structure. For further background on the basic function of Genetic Algorithms, please refer to Grefenstette[5] and Goldberg[6]. Development of a simple model for local and global protein characteristics and their contribution to fitness of the short-protein structure is the subject of our analysis.

Highly complex models of protein structure prediction using GAs have been implemented[7-9]. These models take into account the three residue characteristics that our model does. In addition, they consider the presence and strength of hydrogen bonding interactions, disulfide bridges, and dihedral bond angles. Further, while we stipulate only a single site in each crossover event, multiple crossover sites are used in complex models. These more complex methods first attempt to predict the appropriate conformation of sub-units within the protein chain, drawing upon residue characteristics and information from highly conserved regions from similar proteins. They then fold these subunits together. While accurate, these complex GA-based structure prediction methods require prior knowledge of the protein structure and are slow. For the purposes of small-protein folding prediction, we demonstrate that computational complexity can be greatly reduced with our model, while approximate structure is still well predicted.

Implementation Background:

This simple model for protein structure prediction seeks an optimal spatial conformation for a given protein with static characteristics. The initial population includes a linear conformation and a set of random conformations. After fitness of each conformation is assessed, crossover, reproduction and mutation operate on the population. Throughout the analysis, the protein remains constant; that is, there is no mutation of the characteristics or ordering of the residues in the protein. We have chosen to use the Genesis Genetic Algorithm (GGA) for optimization of protein structure[5]. This user-friendly GA is well documented, which will enable others to replicate our method easily. In addition, it allows analysis based upon gene structure, where genes are handled as integer values. This simplifies development of the fitness evaluation procedure significantly.

Optimization Method:

In preparing our GA-based approach to optimizing protein structure, we follow the four principles of preparation outlined in Koza[10]. We developed a representation scheme, a method of fitness measurement, optimized the major parameters of the GA run, and chose termination criteria.

Protein Spatial Representation

The protein under study was located in a three-dimensional integer space. We developed a binary string based representation for protein structure and then exploited the gene-based representation scheme in Genesis. The binary representation consists of groups of five bits, arranged in a string, where each group codes for one residue. Thus, the binary representation for a protein under study is L*5 bits long where L is the number of residues in the protein. A group of 5 bits is decoded to an integer between 0 and 31. The 32 different codings each represent a potential location at which a residue can be found with respect to the previous residue in the chain. This representation ensures that each residue moves to a point that is no more than one integer away from the previous residue along each of the three cardinal axes. All moves are made from the perspective of the user, which is constant throughout the analysis. Once the protein structure has been derived, it can be graphed on coordinate axes which can be rotated to provide different perspectives. The residue representation is summarized in table 1 and illustrated in figure 1.

Note that moves along the cardinal directions, where the residue location differs with respect to the old residue location by one integer value along only one axis, are twice as likely to be chosen in the initial population as other moves. However, after evolution has progressed, this affect will be negligible.

Binary String	Gene	Translation	Binary String	Gene	Translation
00000	0	UP	10000	16	DOWN-AWAY
00001	1	RIGHT	10001	17	DOWN-TOWARD
00010	2	LEFT	10010	18	DOWN-RIGHT-TOWARD
00011	3	DOWN	10011	19	DOWN-RIGHT-AWAY
00100	4	AWAY	10100	20	DOWN-LEFT-TOWARD
00101	5	TOWARD	10101	21	DOWN-LEFT-AWAY
00110	6	UP-RIGHT	10110	22	RIGHT-TOWARD
00111	7	UP-LEFT	10111	23	RIGHT-AWAY
01000	8	UP-TOWARD	11000	24	LEFT-TOWARD
01001	9	UP-AWAY	11001	25	LEFT-AWAY
01010	10	UP-RIGHT-TOWARD	11010	26	UP
01011	11	UP-RIGHT-AWAY	11011	27	RIGHT
01100	12	UP-LEFT-TOWARD	11100	28	LEFT
01101	13	UP-LEFT-AWAY	11101	29	DOWN
01110	14	DOWN-RIGHT	11110	30	AWAY
01111	15	DOWN-LEFT	11111	31	TOWARD

Table 1. Binary and gene representation of each residue in the protein structure. Translation directions are from the user's perspective, which remains constant throughout the analysis.

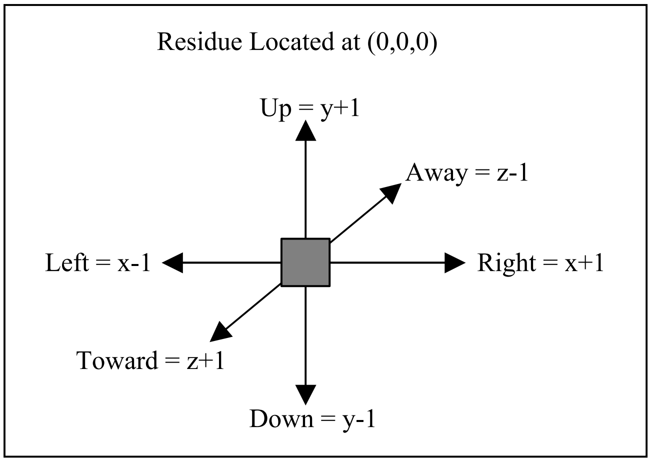

Figure 1. Definition of moves, from a residue located at (0,0,0). These are referred to in the translations in table 1.

Protein Characteristic Representation

Protein characteristics remain constant throughout structure optimization. They are encoded as binary strings of length N, in which element (i) of the string holds the characteristic of amino acid residue (i) in the chain. In this representation, three characteristics are encoded in four strings. These strings represent hydrophobicity, side-chain size, and hydrophilic positive and negative charge. The presence of a characteristic is encoded as 1 and absence as 0. Small side-chain size is encoded as 1. Table 2 summarizes the characteristics of each amino acid.

Amino Acid	Abbreviation	Hydrophobic	Small Side Chain	+ Charge	- Charge
Glycine	G	Y	Y		
Alanine	A	Y	Y		
Valine	V	Y	Y		
Leucine	L	Y	Y		
Isoleucine	I	Y	Y		
Methionine	M	Y			
Phenylalanine	F	Y			
Tryptophan	W	Y			
Proline	P	Y			
Serine	S		Y		
Threonine	T		Y		
Cysteine	C		Y		
Tyrosine	Y				
Asparagine	N				
Glutamine	Q				
Aspartic Acid	D				Y
Glutamic Acid	E				Y
Lysine	K			Y	
Arginine	R			Y	
Histidine	H			Y	

Table 2. Characteristics of amino acid residues. 'Y' denotes that the residue has the characteristic.

Fitness Measure

The raw search space of a given protein containing L resides is large: roughly 32^L. To save computational time in evaluating the fitness of a given protein structure, we first attempt to eliminate all non-viable protein structures. Structures are laid out on a three-dimensional integer grid, and if more than one residue is present at a single coordinate location, they are classified as non-viable and assigned a fitness of 0. We next assign value to the relative spatial configurations of residues with various combinations of the characteristics listed above. In so doing, we make the following assumptions, which are valid in the case of most proteins that reside in normal pH, aqueous environments.

- Hydrophobic regions of the protein tend to localize to the center of the structure, surrounded by hydrophilic regions when possible.
- Small side-chain residues fit more easily at the center of a tightly folded structure whereas larger side-chain residues tend to be located at the perimeter of such folds.
- Positively and negatively charged residues attract and so tend to be adjacent.
- Positively charged residues repel one another so tend to be located as far apart as possible. The same is true of negatively charged residues.

In quantifying these assumptions to measure fitness, we consider pairs of residues that are not connected in the protein chain, but do not assess connected residue pairs. This prevents rewarding or penalizing adjacencies that are not directly controlled by the algorithm. We found it best to consider pairs adjacent in only the cardinal directions to reward linearity above kinkiness.

We optimized the values assigned to the presence of the conformations listed above in order to obtain an end result that closely matched experimental protein conformation. The fitness measures listed in table 3 are the result of this optimization. Table 3 assumes a comparison between two non-connected residues, k and n, that are adjacent in one of the cardinal directions.

Fitness Evaluation	Fitness Value	Explanation
hydrophobicity[k] = 1 and hydrophobicity[n] = 1	+6	Hydrophobics conglomerate and locate at the center of folds.

hydrophobicity[k] = 1 and hydrophobicity[n] = 0
hydrophobicity[k] = 0 and hydrophobicity[n] = 1
	+2	Combined with above, ensures that hydrophilics are pushed outside.

SmallSize[k] = 1 and SmallSize[n] = 1	+4	Small side-chain residues fit together in tight folds more easily.

SmallSize[k] = 1 and SmallSize[n] = 0
SmallSize[k] = 0 and SmallSize[n] = 1
	+2	Combined with above, ensures that large side-chains are pushed outside.

PositiveCharge[k] = 1 and NegativeCharge[n] = 1
NegativeCharge[k] = 1 and PositiveCharge[k] = 1
	+2, exp.	Opposite charges attract.

PositiveCharge[k] = 1 and PositiveCharge[n] = 1
NegativeCharge[k] = 1 and NegativeCharge[n] = 1
	-2, exp.	Like charges repel.

Table 3. Summary of fitness measures comparing two non-connected adjacent residues k and n.

Experimental Parameters

We optimized the experimental parameters for this GA protein structure prediction technique based upon comparison of the GA structure result to NMR experimental structure for two short-length proteins. These randomly chosen test-set proteins were 1ACW, a toxin found in scorpions, and 1ALG, an Hgr inhibitor. We will refer to these as Scorp and Hgr. The characteristics of these proteins are summarized in table 4.

Protein	HGR	SCORP
Sequence	QGLGCDEMLQGFAVAV KMGATKAD	VSCEDCPEHCSTQKAQAKCD NDKCVCEPI
Hydrophobicity String	0 1 1 1 0 0 0 1 1 0 1 1 1 1 1 1 0 1 1 1 0 0 1 0	1 0 0 0 0 0 1 0 0 0 0 0 0 0 1 0 1 0 0 0 0 0 0 0 1 0 0 1 1
SmallSize String	0 1 1 1 1 0 0 0 1 0 1 0 1 1 1 1 0 0 1 1 1 0 1 0	1 1 1 0 0 1 0 0 0 1 1 1 0 0 1 0 1 0 1 0 0 0 0 1 1 1 0 0 1
PositiveCharge String	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 1 0 0	0 0 0 0 0 0 0 0 1 0 0 0 0 1 0 0 0 1 0 0 0 0 1 0 0 0 0 0 0
NegativeCharge String	0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1	0 0 0 1 1 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 1 0 1 0 0 0 0 1 0 0

Table 4. Summary of characteristics of the test-set with which the experimental parameters and fitness value assignments were optimized.

We varied four experimental parameters: initial population size, number of generations run, crossover rate, and mutation rate, to achieve the best result in the smallest number of trials. The simplified spatial model that we used, three-dimensional integer space, made a direct quantitative comparison between the NMR structure model of the protein and the GA-optimized model difficult. Therefore, the efficacy of the fitness measures was judged by careful visual comparison of the GA structure result to the NMR experimental structure, and the resulting optimal fitness measures are listed in table 3. In choosing the fitness measures, we aimed to achieve a structure most similar to the NMR experimental structure. These fitness measures did not necessarily yield the most tightly folded structure.

The initial population size was set to 1000 to optimize variety in the initial population, encouraging faster convergence, while minimizing required computation time. We optimized the combination of mutation rate and crossover rate to yield the best terminal fitness in the shortest number of generations possible. The optimal values were found to be 0.73 for the crossover rate and 0.005 for the mutation rate.

Terminal Criteria
The terminal result was identified when fitness of the best individual in the population ceased to increase with increasing numbers of generations. At least two orders of magnitude of generations without changing fitness were observed to confirm the terminal result.

Tableau
The experimental characteristics are summarized in the Genetic Algorithm Tableau[10] presented in Table 5.

Objective:	To find the structural conformation that most closely resembles the experimental conformation.
Representation Scheme:	L genes for a protein consisting of L residues: each gene takes one of 32 values and each value represents the location of the residue with respect to the preceding residue.
Fitness Cases:	Evaluated based upon non-connected residues adjacent in the cardinal directions. Characteristics of the protein are stored in static arrays and used to evaluate fitness. Please see table 3 for details.
Raw Fitness:	Sum of positive contributions of adjacent hydrophobic, small side-chain, or opposite charge residues and negative contributions of adjacent like-charge residues. Fitness is maximized.
Parameters:	Population Size M = 1000 Generations G vary based upon protein length Crossover Rate = 0.73 Mutation Rate = 0.005 Necessary Genesis Options: aCefLM
Termination Criteria:	The best individual is identified when the best fitness in the population ceases to change over the span of two orders of magnitude of generations.
Result Designation:	The most fit individual in the population, derived from the checkpoint file dumped from the last generation of the run.

Table 5. Tableau summarizing the experimental method for the simple GA-based protein structure optimization technique.

Analysis Method:

In order to analyze the results of the Genesis optimization, we developed a simple protein structure viewer in C and Matlab. This viewer consists of a C program to translate the binary representation of a protein structure into a set of coordinates in three-dimensional integer space and a Matlab program to graph these coordinates and the chain connecting them. In order to compare the GA structure to the experimentally determined structure for the test-set proteins, we spliced code from PDB files containing NMR structural coordinates. In each case, we chose the first structural model given in the PDB file. The simple protein viewer is available upon request.

Discussion of Results:

The GA-based protein structure prediction technique yielded good results for the test-set proteins. As discussed above, it is difficult to quantify the quality of the prediction given the simplified coordinate system employed. However, visual comparison of the predicted results to experimental results suggests that the simple GA-based protein structure prediction technique works accurately and quickly. The optimal conformation of Hgr illustrated in figure 2(b) has fitness of 232 and is achieved at generation 800. The optimal conformation of Scorp illustrated in figure 2(e) has fitness of 156 and is achieved at generation 3500. These results are compared to NMR structures in figures 2(c) and (f).

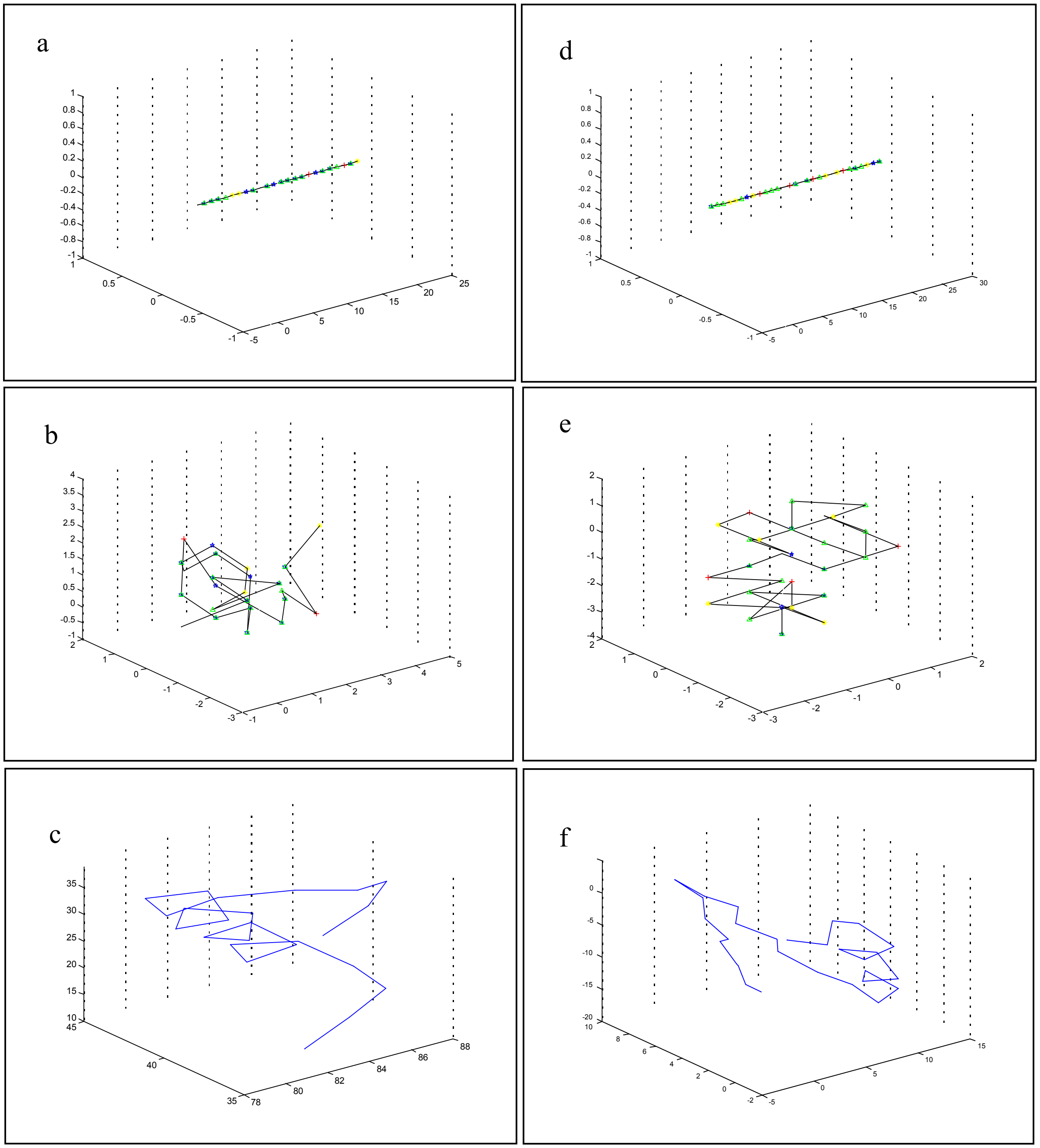

Figure 2. Results of GA-based protein structure prediction technique and comparison to NMR experimental structure:

(a) Hgr initial linear conformation. *(d) Scorp initial linear conformation.*
(b) Hgr folded by GA. *(e) Scorp folded by GA.*
(c) Hgr NMR structure. *(f) Scorp NMR structure.*

Figures (a) through (c) illustrate the analysis of Hgr and figures (d) through (f) illustrate the analysis of Scorp. Figure (b) peers down the helix that is viewed from the side in figure (c). Figures (b) and (c) illustrate a structure that consists of a helical segment surrounded by two tails. This similarity between the predicted and experimental conformations of Hgr in figures 2(b,c) suggests that the protein structure viewer works. Figure (e) shows a helix leading into a conglomerated tail region. This helical region is parallel to that in figure (f) but, in contrast to the GA prediction in figure (e), the tail in figure (f) is extended. This case illustrates a shortcoming of the prediction technique: it has a higher tendency than nature to fold into conglomerate regions.

The simple protein viewer uses color to represent the presence of each of the fitness characteristics at each residue location in the protein. In color versions of figures (b) and (e), we can see that hydrophobic, small-side-chain residues are localized to the inside of the protein folds. In addition, we note that like-charge interactions are minimized and opposite-charge interactions are maximized. All of these results suggests that the simple GA-based protein structure prediction technique is effective as a first-order analysis.

It is interesting to examine intermediate stages during the protein structure optimization. We find that clear schema emerge early in the process and are conserved in the best-of-generation individuals until optimal conformation is reached. For example, the best of generation individual from generation 30 of the Hgr analysis has fitness of 170 and is illustrated in figure 3. It is already clear that the protein will fold into a helical conformation, though optimization occurs over 770 additional generations.

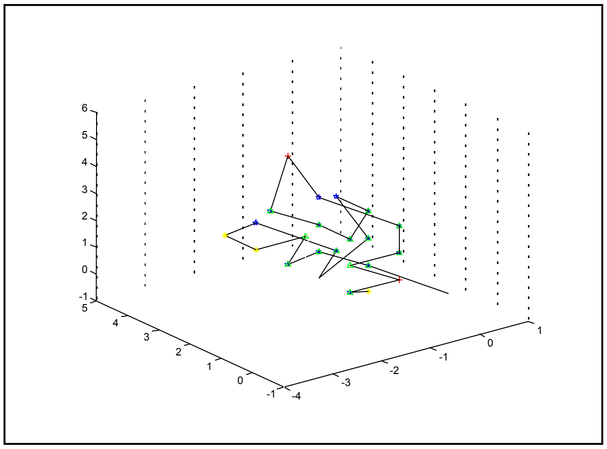

Figure 3. Intermediate conformation of Hgr, at generation 30 out of 800 total. This structure has fitness of 170 compared to the terminal fitness of 232, and the helical conformation is already forming.

Suggestions for Future Work:

The fitness measure presented here is easily expanded to new residue characteristics. For example, the next implementation might take the hydrogen bonding tendencies of each residue into account. The method is also easily extended to calculate substructure characteristics and incorporate them into the overall structural analysis. For example, segments containing structures that represent helices might be more highly conserved through crossovers. In addition, local environmental variables could be introduced. For example, the residues of a known transmembrane protein segment could be isolated from the rest of the analysis in order to prevent spurious conglomeration around the hydrophobic transmembrane region. Certainly, a more complex three dimensional space could also be implemented to improve the accuracy of the model. Clearly, the expansion possibilities associated with this simple technique are very promising, and there is much room future work in this area.

In preparation for future work, it will be important to implement a quantitative measure for comparison of GA-predicted structure to experimental structure. We suggest that a simple measure might first recalculate coordinated of the experimental structure to make bond distances constant. The measure might then orient the first two residues of the predicted and experimental structure at the same place on the coordinate axes in order to align them and normalize bond distances. The measure could then calculate the average difference between the locations of corresponding residues in the two structures. We also suggest development of a more robust protein viewer to aid in visual analysis.

Conclusion:

We have developed a simple GA-based method for prediction of short-protein structure. By modeling proteins in three-dimensional integer space, and residues with characteristics of hydrophobicity, charge, and side-chain size, this method improves upon that presented by Unger and Moult[1]. This technique is simple, fast, and the only prior information required to use this simulator is the protein sequence, and the amino acid characteristics. As a result, this model is a very attractive mechanism for making first-order approximations about protein structure.

Acknowledgements:

Thanks to Darren Lewis for advice regarding conceptualization and programming.

Works Cited:

[1] R. Unger and J. Moult. "Genetic Algorithms for Protein Folding Simulations," *Journal of Molecular Biology*. 231, 75-81. 1993.

[2] J. E. Shea and C. L. Brooks. "From Folding Theories to Folding Proteins: A Review and Assessment of Simulation Studies of Protein Folding and Unfolding," *Annual Review of Physical Chemistry*. 52, 499-535. 2001.

[3] J. B. Foresman and A. Frisch. *Exploring Chemistry with Electronic Structure Methods, 2nd Ed*. Gaussian, Inc., Pittsburgh, USA. 2000.

[4] R. Unger and J. Moult. "Local Interactions Dominate Folding in a Simple Protein Model," *Journal of Molecular Biology*. 259, 998-994. 1996.

[5] J. J. Grefenstette. *A User's Guide to Genesis, Version 5.0*. October, 1990.

[6] D. E. Goldberg. *Genetic Algorithms*. Addison-Wesley, San Francicso, USA. 1989.

[7] T. Dandekat and P. Argos. "Folding the Main-Chain of Small Proteins with the Genetic Algorithm," *Journal of Molecular Biology*. 236, 844-861. 1994.

[8] J. T. Pedersen and J. Moult. "Ab Initio Protein Folding Simulations with Genetic Algorithms: Simulations of the Complete Sequence of Small Proteins," Proteins – Structure Function and Genetics. S1, 179-184. 1997.

[9] J. T. Pedersen and J. Moult. "Protein Folding Simulations with Genetic Algorithms and a Detailed Molecular Description," Journal of Molecular Biology. 269, 240-259. 1997.

[10] J. R. Koza. Course Notes for Genetic Algorithms and Genetic Programming. Spring, 2002.