Protein Model Representation and Construction

M. Sullivan[1] J. Glasgow[1] E. Steeg[1] L. Leherte[2] S. Fortier[1]

Abstract

Crystallographic studies play a major role in current efforts towards protein structure determination. However, despite recent advances in computational tools for molecular modeling and graphics, the task of constructing a protein model from crystallographic data remains complex and time-consuming, requiring extensive expert intervention. This paper describes an approach to automating the process of model construction, where a model is represented as an annotated trace (or partial trace) of the three-dimensional backbone of the structure. Potential models are generated using an evolutionary algorithm, which incorporates multiple fitness functions tailored to different structural levels in the protein. Preliminary experimental results, which demonstrate the viability of the approach, are reported.

Introduction

A fundamental goal of research in molecular biology is to understand protein structure. Protein crystallography is currently at the forefront of methods for determining the three-dimensional conformation of a protein, yet it remains labor intensive and relies on an expert's ability to construct, evaluate and refine potential models for the structure. A protein model represents a hypothesis of the tertiary structure of a protein; a good model is one which makes sense (in terms of our knowledge of the chemistry, biology and physics of the molecule) and is consistent with the experimental data. Currently, building a protein model is a trial-and-error process, which is assisted by the use of computer graphics for tracing the polypeptide chains and modeling the side chains, and for viewing and improving the resulting model (Jones *et al.* 1991). Errors in the initial, and subsequent, models may be corrected using a refinement process, which involves modifying the model to minimize the difference between the experimentally observed data and the data calculated using a hypothetical crystal containing the model. It has been proposed that the process of protein model building could be improved through the development of computational tools (Branden & Jones February 1990). This paper reports on such a tool for model construction, which will be incorporated in a fully automated system for protein structure determination from crystallographic data.

An approach to molecular scene analysis has previously been proposed (Fortier *et al.* 1993; Glasgow, Fortier, & Allen 1993) where a scene model is generated using a topological analysis of the protein image data (Leherte *et al.* 1994). Although initial results suggest that this approach is promising, it relies on a single model which may not correspond to the optimal trace of the protein backbone for the given data. The research described in this paper addresses the shortcomings of the previous approach by proposing a technique that generates and evaluates multiple possible protein models using an evolutionary computation methodology. In this approach, mutation operators are applied to build structural models using data derived from a topological analysis of a protein image. A novel aspect of the research is that multiple fitness functions are used to evaluate potential models on the basis of criteria applicable to different lengths of substructures. This approach can be incorporated into a heuristic search strategy that will determine a path from an initial uninterpreted protein image to a fully-interpreted model.

The goal of the research described in this paper is to design an approach to protein model construction and to implement it in a comprehensive computational system for molecular scene analysis. At medium resolution($\sim$ 3 Å), we define a protein model as a path (or subpath) through a graph consisting of critical

[1]Departments of Computing and Information Science and Chemistry, Queen's University, Kingston, Canada K7L 3N6, {sullivan,glasgow,steeg,fortier}@qucis.queensu.ca

[2]Laboratoire de Physico-Chimie Informatique, Facultés Universitaires Notre-Dame de la Paix, Namur, Belgium, leherte@scf.fundp.ac.be

point nodes, corresponding to amino acid residues, and weighted edges, corresponding to potential polypeptide bonds. A model also specifies a set of environments, which describe properties of the individual critical point nodes on the graph corresponding to amino acid residue classes.

Model Construction

In this section we describe an intelligent system for generating protein models from a critical point graph. A model corresponds to a trace of the graph corresponding to a potential backbone for the protein. It is not our goal to find the "best" model for the structure, only to determine a set of reasonable – in terms of our knowledge of the molecule and physical/chemical constraints – models which can then be put through a more rigorous evaluation process to find the best candidates to participate in the next iteration of image refinement and generation.

An *evolutionary programming* (Fogel 1995) approach was developed to generate potential backbone traces for a protein. This approach involves taking a *population* (a set of potential traces) and probabilistically selecting the "fittest" traces with respect to a given evaluation function. The chosen traces are then modified and placed in the next generation for a population. Successive generations of populations lead to new and expanded protein traces being created and examined. A *growth model* is utilized to track the backbone trace through the graph. The basic tenet of this growth methodology is to apply transformational (intelligent mutation) operators to incrementally extend and develop members of a population of traces. We incorporate three such operations in our system: 1) an *add* mutation extends a trace by adding an edge to the path; 2) a *delete* mutation removes an edge from the end of a path; and 3) a *split* mutation probablistically removes an internal edge resulting in two new subtraces. Figure 1 illustrates the application of these three operations to a trace in a given population.

Our evolutionary system is novel in the sense that it evaluates potential models at varying structural levels of the protein. We divide our traces into multiple classes and use specialized fitness criteria for each class as specified in Table 1. Note that individual criteria not only address different lengths of traces, but also different structural levels: Class 1 focuses on local distance and angle criteria among residues; Class 2 looks at secondary structure conformations; and Class 3 examines super-secondary structure interactions.

Different fitness functions are incorporated into the system using a variation of the *island model* (Davis 1991). Instead of maintaining a single population of

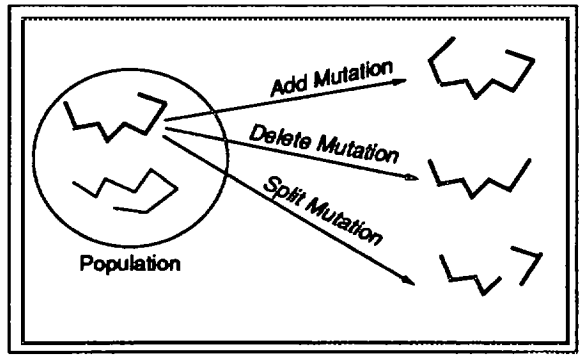

Figure 1: Mutation operations for evolutionary approach to trace generation.

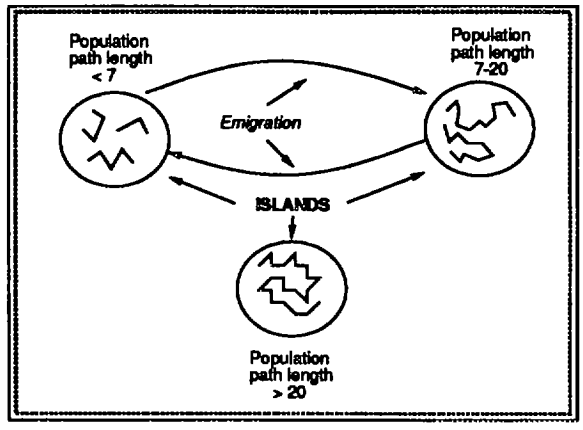

Figure 2: Abstract view of the evolutionary program using a three-islands model.

traces, the system considers multiple populations, one for each class of traces. Individual populations are isolated and do not communicate with other populations except through the process of emigration: when a population generates a trace that is outside its bounds, then the trace is moved to the appropriate population. For example, if the population illustrated in Figure 1 was restricted to traces of length 7 to 20, then the traces resulting from the split mutation could emigrate to a population of smaller paths. Thus, the populations can be viewed abstractly as a group of islands (see Figure 2) where paths may emigrate from one island to another as a result of intelligent mutation.

The basic algorithm for the evolutionary program consists of the following steps:

- Build the initial population of traces by randomly selecting single edges from the critical point graph.
- Repeat until the new population is full:
  - Retrieve a trace from the population using a

Table 1: Fitness function criteria for different island classes.

CLASS	TRACE LENGTH	FITNESS FUNCTION CRITERIA
1	1 – 6	Graph edge weights and simple bond angles
2	7 – 20	Bond and torsion angles, residue distances
3	21 – 30	Super-secondary structure

tournament[1] selection technique.

- Perform mutation in a probabilistic fashion.
- Add the trace to the new population.

- Iteratively process the multiple populations (using intelligent mutation operations) until a desired result or stopping criteria is achieved.

Test Results

The evolutionary algorithm for generating traces was implemented and tested using two islands. Following, we describe the fitness functions for each of these islands and experimental results of applying the algorithm to critical point graphs constructed from crystallographic data.

Island 1: Traces of length 1 to 6 are ranked using a fitness function based on edge weights (from the critical point graph) and bond angles. In order to determine the preferred ranges for values, experimental data were acquired and a discrete binned distribution (histogram) was calculated for both the weights and angles.

A *reward/punishment* technique was used to calculate the fitness value for a trace. Rewards (positive values) or punishments (negative values) are awarded based on the region the weights and angles fell into for a given trace. The fitness value for a trace T is simply a sum of the reward/punishment function (RPF_1) applied to the weight of each edge (W_e) and each angle (θ) in the trace:

$$fitness(T) = \sum_{e \in T} RPF_1(W_e) + \sum_{\theta \in T} RPF_1(\theta)$$

The fitness function for island 1 was tested on proteins Phospholipase A2 ($f389$) and Ribonuclease H ($2rn2$). Testing was performed for an island size of 100 and a single run consisted of 20 generations.[2]

The algorithm was run 40 times for protein $2rn2$ and 100 times for protein $f389$ and the fittest member of the population selected at the end of each run. Table 2 illustrates the results of these runs. For $2rn2$, 21 out of the 40 runs produced correct traces that spanned a portion of the protein backbone. The remaining 19 runs produced traces that either omitted a single residue from the backbone or contained a repeat of a residue. That is, traces of the form < 12_13_14_15_17_18 > (residue 16 was skipped) or < 12_13_13_14_15_16 > (residue 13 was represented as two distinct critical points) were generated. The results for protein $f389$ were not quite as positive: 36 out of 100 of the traces were totally correct. Runs for this protein also produced several traces (8 in total) that contained disulfide bridge connections. Errors also occurred when a trace reaches the end of the backbone, but continues to add edges.

Table 2: Results of testing fitness function for island 1.

Length	4	5	6	7
Correct		5	10	6
1 Duplicate		4	4	
1 Jump	1	2	2	3
2 Duplicates				2

(a) results for $2rn2$

Length	3	4	5	6	7
Correct	2	3	14	10	7
1 Duplicate		4	5	9	8
1 Jump			3	2	
2 Duplicates		1	1	4	1
1 Dup/1 Jump			1	5	6
2 Dup/1 jump			2	1	3

(b) results for $f389$

On the whole, the results for island 1 were promising. All traces with infeasible inter-residue distances or angles were eliminated from consideration. The incorrect traces that remained in the population were ones that could not be discarded based on local criteria alone (e.g., connectivity through critical points corresponding to side chains). The important result

[1] A tournament selection involves repeatedly randomly choosing some number n of individuals from the population and retaining the fittest individual for the intermediate population.

[2] Initially tests were carried out with longer runs. However, little improvement was found after 20 generations.

is that for both proteins we produced multiple good traces of length 7 that could be exported to the second island.

Island 2: The fitness function for traces of length 7 to 20 is based on a Bayesian model of the distributions of simple angles, torsion angles and inter-residue distances in known secondary structure classes across a set of diverse protein backbone structures.

First, a finite mixture model of Gaussians (for distance data) and Von Mises circular distributions (for angle data) was learned using a modified version of the SNOB minimal message length (MML) classifier (Wallace & Dowe 1994). The training data consisted of over 10,000 examples of traces of lengths 4 and 7 calculated directly from Protein Data Bank structure files, from generated electron density maps with noise added, and from experimental maps preprocessed using Orcrit. Currently, the trained secondary structure recognition module is used only to estimate the likelihood that a trace is indeed characteristic of a valid protein backbone. Ongoing research involves considering a set of modules trained for recognition of helix, sheet, turn, and coil classes, so that higher-order information patterns (e.g., "runs" of helix or sheet) can be rewarded or penalized as appropriate. As in island 1, the current fitness function for island 2 is a sum, in this case over the segments s of length 4 of a trace:

$$fitness(T) = \sum_{s \in T} RPF_2(s),$$

where $RPF_2(s)$ is a nonlinear reward/penalty function that imposes a heavy penalty on segments showing low likelihood of fitting a real protein backbone structure and produces graduated rewards for segments showing higher likelihood values.

The fitness function for island 2 was tested on proteins $f389$ and $2rn2$. The results generated were mixed. 30 runs were performed, each resulting in a trace of length 8 - 20. The shorter traces (length 8-10) were found to be totally correct. Although longer traces contained correct subtraces, they often had jumps (incorrect bonds) between correct models of the backbone. One reason for this is that although the angles for the incorrect portions made chemical sense, the distances deviated from the norm. This suggests that a greater emphasis should be placed on distance criteria. Also note, that although island 2 introduced occasional errors, it often managed to improve on traces that were passed on by island 1.

The evolutionary approach to model construction allows us to derive potential traces of a protein backbone using both local and global evaluation criteria. Our models also contain other valuable information, in the form of environments, which can be used to associate critical points with individual residues in the sequence (Baxter *et al.* 1996).

Currently, our fitness functions for model generation only consider a model in terms of its trace. Future research will involve investigating the use of environment information in the fitness criteria. As well, we plan to incorporate additional islands, and corresponding fitness functions, to take into consideration further global constraints (such as super-secondary structure preferences) in the evolutionary system. Our ultimate goal is to integrate the model construction module with other processes (image generation, model evaluation and image refinement) that are being developed for molecular scene analysis.

References

Baxter, K.; Steeg, E.; Lathrop, R.; Glasgow, J.; and Fortier, S. 1996. From electron density and sequence to structure: Integrating protein image analysis and threading for structure determination. In *Proceedings of the 4th International Conference on Intelligent Systems for Molecular Biology.* AAAI/MIT Press.

Branden, C., and Jones, T. February 1990. Between objectivity and subjectivity. *Nature* 343:687–689.

Davis, L., ed. 1991. *Handbook of Genetic Algorithms.* Van Nostrand Reinhold, New York.

Fogel, D. 1995. *Evolutionary computation: Toward a new philosophy of machine intelligence.* IEEE Press, New York.

Fortier, S.; Castleden, I.; Glasgow, J.; Conklin, D.; Walmsley, C.; Leherte, L.; and Allen, F. 1993. Molecular scene analysis: The integration of direct methods and artificial intelligence strategies for solving protein crystal structures. *Acta Crystallographica* D49:168–178.

Glasgow, J.; Fortier, S.; and Allen, F. 1993. Molecular scene analysis: crystal structure determination through imagery. In Hunter, L., ed., *Artificial Intelligence and Molecular Biology.* AAAI Press. 433–458.

Jones, T.; Zou, J.; Cowan, S.; and Kjeldgaard, M. 1991. Improved methods for building protein models in electron-density maps and the location of errors in those models. *Acta Crystallographica* A47:110–119.

Leherte, L.; Fortier, S.; Glasgow, J.; and Allen, F. 1994. Molecular scene analysis: A topological approach to the automated interpretation of protein electron density maps. *Acta Crystallographica D* D50:155–166.

Wallace, C. S., and Dowe, D. 1994. Intrinsic classification by MML - the Snob program. In *Proc. 7th Australian Joint Conference on Artificial Intelligence*, 37–44.