AI Magazine Volume 27 Number 3 (2006) (© AAAI)

TEXTAL

Crystallographic Protein Model Building Using AI and Pattern Recognition

Kreshna Gopal, Tod D. Romo, Erik W. McKee, Reetal Pai, Jacob N. Smith, James C. Sacchettini, and Thomas R. Ioerger

■ TEXTAL is a computer program that automatically-interprets electron density maps to determine the atomic structures of proteins through X-ray crystallography. Electron density maps are traditionally interpreted by visually fitting atoms into density patterns. This manual process can be time-consuming and error prone, even for expert crystallographers. Noise in the data and limited resolution make map interpretation challenging. To automate the process, TEXTAL employs a variety of AI and pattern-recognition techniques that emulate the decision-making processes of domain experts. In this article, we discuss the various ways AI technology is used in TEXTAL, including neural networks, case-based reasoning, nearest neighbor learning and linear discriminant analysis. The AI and pattern-recognition approaches have proven to be effective for building protein models even with medium resolution data. TEXTAL is a successfully deployed application; it is being used in more than 100 crystallography labs from 20 countries.

Proteins are large and complex macromolecules that are essential to the chemical processes in living systems. For example, enzymes are proteins that are responsible for catalyzing the thousands of metabolic reactions in the living cell. Proteins also play signaling, regulatory, transport, immune-response, and mechanical roles in cells. Proteins are made up of amino acids (also called *residues*) that are linked through covalent chemical linkages known as *peptide* bonds—the amino acids form a linear polymeric structure called a *polypeptide chain*. Typically proteins contain 100 to 1,000 amino acids, arranged in a specific order for a given protein. The average number of residues in natural proteins is about 300. There are 20 unique amino acids that are commonly found in nature.

Knowledge of a protein structure is essential to understanding how the protein functions, its role in diseases, and how drugs (for example, inhibitors) can be designed. In the past few years, the genomic sequence databases have grown phenomenally. The entire genomes of the human plus those of many other organisms are now known and accessible. Keeping up the protein structure determination rate with this growth of genomic information has become a major challenge. In fact, the ratio of solved crystal structures to the number of discovered proteins is about 0.15 (Tsigelny 2002). The *structural genomics* initiative (Burley et al. 1999) is a worldwide effort aimed at solving protein structures in a high-throughput mode, primarily by X-ray crystallography and nuclear magnetic resonance (NMR) spectroscopy methods.

X-ray crystallography is the most widely used technique to accurately determine the structure of proteins and other macromolecules. It is based on the fact that X rays are diffracted by crystals due to the regular spacing of molecules in the crystal lattice. X rays are scattered by the electrons around atoms, and this scattering results in diffraction patterns.

Crystallographic structure determination involves many steps: first the protein has to be

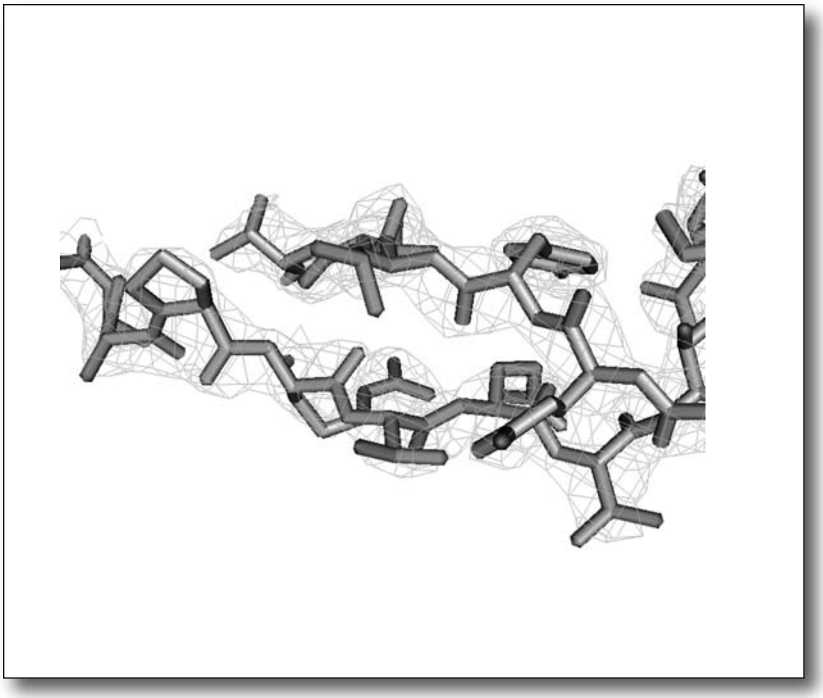

Figure 1. Example of Electron Density Around a Fragment of a Protein Structure.

The fragment shown consists of two strands of a β-sheet in exocytosis-sensitive phosphoprotein (PDB ID: 1KFQ). The electron density map has been calculated from the solved structure at 2.8 Å. This image was generated with PyMOL (W. L. DeLano, www.pymol.org).

isolated, purified, and crystallized. After crystallization, X rays are shone through the crystal and diffraction data (intensities of diffraction spots) are collected. The diffraction pattern can in principle be used to reconstruct a map of the electron density around the molecule by inverse Fourier transform, although phases for the structure factors (Fourier coefficients) have to be estimated. This is necessary because the diffraction spots contain information only about the amplitudes of diffracted waves; the phase information, which is also required for calculating the map, is lost. Approximate phase information can be obtained by a variety of experimental techniques, including multiwavelength anomalous diffraction, multiple isomorphous replacement, and molecular replacement (McRee 1999).

The sample of diffraction spots at which intensities can be collected is limited, which constrains the degree to which atoms can be distinguished from one another. This imposes limits on the *resolution* of the map, measured in Å (or Angstrom, where 1 Å = 10^{-10} m). The resolution is determined by a variety of experimental factors. At 4 Å, the backbone may appear connected, but side chains might not be very distinguishable. At 3 Å, it may be possible to discriminate a few residues. In 2 Å maps, all residues usually appear quite distinct, and at 1 Å, we can even see the density around individual atoms. In the majority of cases, data can be collected only at medium resolution. Thus, the major focus (and challenge) of automated map interpretation is for the 2–3 Å resolution range.

Automated Electron Density Map Interpretation

The final step in protein crystallography is model building, or determining coordinates of atoms from an electron density map. Model building is typically a two-stage process. First, the path of the polypeptide chain through the density is determined. Then, amino acids are fitted into the density map; each amino acid has several rotational degrees of freedom and can adopt various conformations. The directionality of the chain must also be determined. Computer graphics programs are widely used to visualize and manipulate the model as well as the density in three dimensions. Contoured meshes are used to portray the density at various levels of detail (figure 1). Fitting of maps is a decision-making process that must take into account factors like the quality of the electron density, stereochemistry of amino acids, recognition of secondary structures, and so on.

Once a preliminary structure has been built, it can often be used to obtain better phase information and generate an improved map, which can then be reinterpreted. This process can go through many cycles, and it may take weeks or sometimes months of effort for an expert crystallographer to produce a refined structure, even with the help of molecular three-dimensional visualization programs. The difficulty of manual structure determination depends on factors like the size of the structure, resolution of the data, the complexity of the molecular packing, and so on. There can be many sources of errors and noise, which distort the electron density map, making interpretation difficult. There is also a subjective component to model building—decisions of an expert are often based on what seems most reasonable in specific situations, based on background knowledge and experience.

Various tools and techniques have been proposed for automated protein model building: treating model building and phase refinement as one unified procedure using free atom insertion in ARP/wARP (Perrakis, Morris, and Lamzin 1999), fitting α-helices and β-sheets, followed by local sequence assignment and extension through loops in MAID (Levitt 2001), template matching and iterative fragment extension in Resolve (Terwilliger 2002), expert systems (Terry 1983), molecular scene analysis (Leherte et al. 1997), using templates from the Protein Data Bank (Jones, Zou, and Cowtan 1991), template convolution and other FFT-based approaches (Kleywegt and Jones 1997), and so on. Many of these approaches require user-intervention or work well only with high quality data. TEXTAL, however, has been designed to be fully automated and to work with medium-quality data (around 2.8 Å resolution). Most maps are, in fact, noisy and fall in the low to medium resolution category due to difficulties in protein crystallization and other limitations of the data-collection methods.

The Architecture of the TEXTAL System

TEXTAL combines both AI and non-AI techniques to address the various facets of the complex problem of automated electron density map interpretation. It takes a real-space pattern-recognition approach to model building. TEXTAL has been designed to be robust to noise and has been optimized for medium resolution X-ray diffraction data (in the 2.4 to 3.0 Å range).

TEXTAL tries to mimic the typical strategy employed by human crystallographers when they interpret electron density maps. It adopts a divide-and-conquer, multistage approach to the problem. First, it builds a set of chains of *C*α atoms representing the backbone. (*C*α atoms are the connection points along the backbone where the side chains are attached). This is done by predicting the positions of the *C*α atoms that lie along the backbone trace (medial axis of the density contours) and connecting them to form chains of *C*αs. This is followed up by fitting side chains into the density based on pattern recognition of the local density around the *C*α atoms. A case-based reasoning approach is used for fitting side chains, where we match the density regions around *C*αs with instances in a database of regions, retrieve corresponding residue structures (that is, atomic coordinates) that best fit the density, and concatenate them to build a complete structure.

TEXTAL is modular, and different components can be used independently or in various possible combinations. Figure 2 shows the major subsystems of TEXTAL.

Findmol identifies a contiguous biological molecular unit of the protein in an electron density map (McKee et al. 2005). The boundaries of the repeating asymmetric unit often cut the molecule into multiple fragments, which makes map interpretation difficult. Findmol can identify a contiguous region of density in which to build by using a combination of clustering and symmetry operations.

C-alpha pattern-recognition algorithm (CAPRA) models the backbone (or main chain) of the protein. It takes an electron density map as input, and outputs a file in the protein data bank (PDB) format containing a set of *C*α chains representing the true backbone as accurately as possible. CAPRA itself is made up of several modules, as described in figure 2.

Lookup is used to build local models of side chains attached to each *C*α atom using case-based reasoning and nearest neighbor learning. Lookup takes spherical regions of density (of 5 Å radius) around the *C*α atoms determined in CAPRA and retrieves their best matches from a database of solved cases. The known structures of the matches are used to model the side chains in a piecewise manner.

Postprocessing routines refine the initial model built by Lookup. There are two main routines in this subsystem: (1) *sequence alignment*, where the sequence of residues in the initial model produced by Lookup is aligned with the known sequence of amino acids of the protein, based on a dynamic programming approach proposed by Smith and Waterman (1981). This enables another round of Lookup to make corrections in the amino acid identities initially determined; (2) *real-space refinement*, where slight adjustments in the positions of atoms are made to better fit the density (Diamond 1971).

Uses of AI Technology

Crystallographers rely heavily on expert knowledge to make decisions at many steps in map interpretation. To automate this process, intelligent methods are needed. AI and pattern-recognition approaches are well suited to address the various challenges involved. Furthermore, databases of solutions (previously solved structures) are available that can be exploited to help solve new structures.

We now describe the specific ways in which AI techniques are used in TEXTAL. We emphasize that many of the AI techniques developed are novel and potentially applicable to many other difficult problems, especially those that share common challenges with TEXTAL: noisy and high-dimensional data, recognition of patterns in three dimensions, computationally costly retrieval from large databases, expensive domain expertise, and so on.

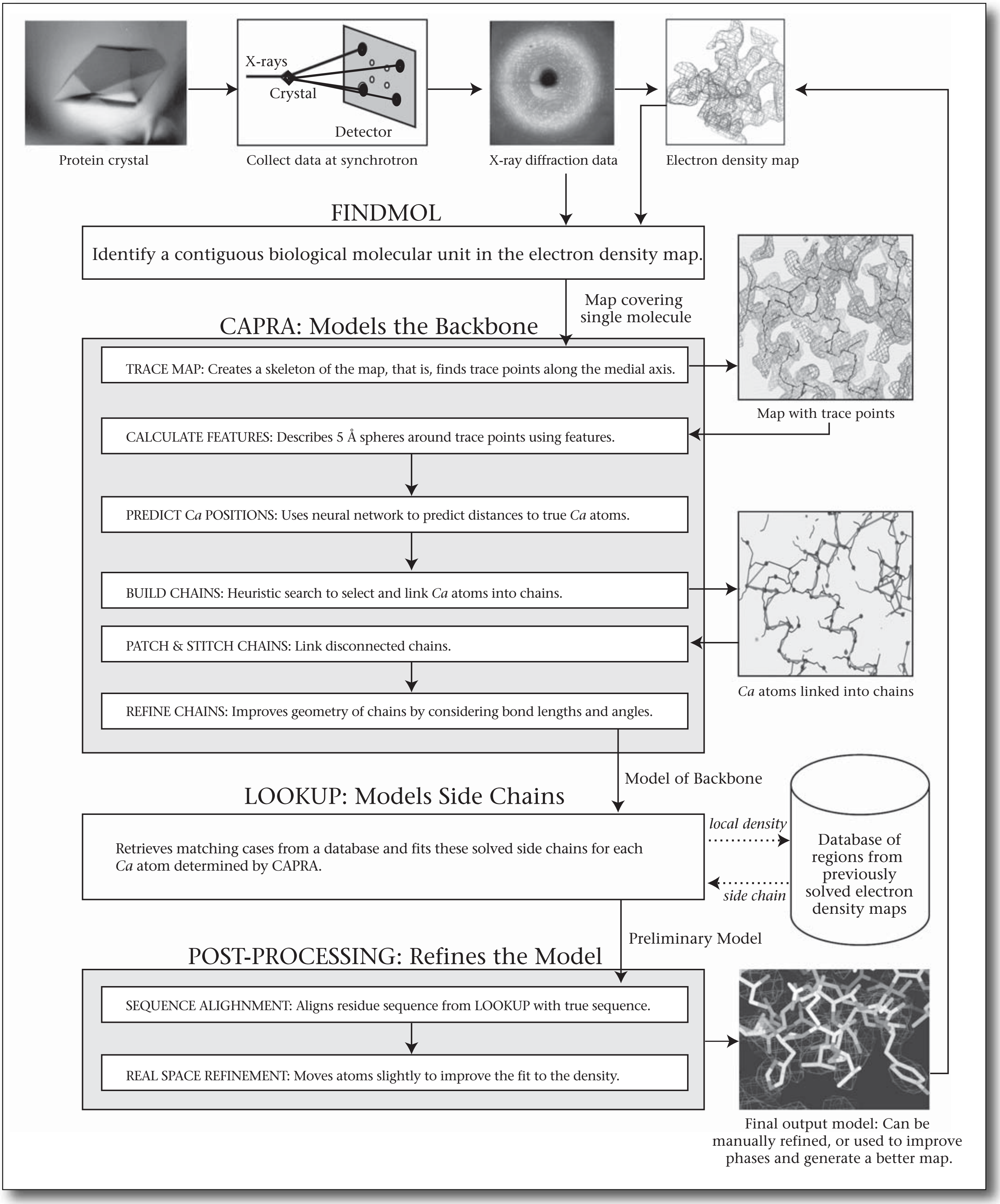
X-rays
Crystal
Detector
Protein crystal
Collect data at synchrotron
X-ray diffraction data
Electron density map
FINDMOL
Identify a contiguous biological molecular unit in the electron density map.
Map covering single molecule
CAPRA: Models the Backbone
TRACE MAP: Creates a skeleton of the map, that is, finds trace points along the medial axis.
Map with trace points
CALCULATE FEATURES: Describes 5 Å spheres around trace points using features.
PREDICT Ca POSITIONS: Uses neural network to predict distances to true Ca atoms.
BUILD CHAINS: Heuristic search to select and link Ca atoms into chains.
PATCH & STITCH CHAINS: Link disconnected chains.
Ca atoms linked into chains
REFINE CHAINS: Improves geometry of chains by considering bond lengths and angles.
Model of Backbone
LOOKUP: Models Side Chains
Retrieves matching cases from a database and fits these solved side chains for each Ca atom determined by CAPRA.
local density
side chain
Database of regions from previously solved electron density maps
Preliminary Model
POST-PROCESSING: Refines the Model
SEQUENCE ALIGHNMENT: Aligns residue sequence from LOOKUP with true sequence.
REAL SPACE REFINEMENT: Moves atoms slightly to improve the fit to the density.
Final output model: Can be manually refined, or used to improve phases and generate a better map.

Figure 2. Architecture of the TEXTAL System.

Shown are the main subsystems: Findmol, CAPRA, Lookup, and Postprocessing.

Extraction of Rotation-Invariant Features that Characterize Density Patterns

The fundamental pattern-recognition approach in TEXTAL is based on extracting numeric features that attempt to capture relevant information about local electron density for various purposes (such as identifying *C*α atoms, comparing side chains, or detecting disulfide bridges). The features were derived manually based on knowledge about crystallography and have the important property of being rotation invariant (since the regions that we want to compare can occur in any three-dimensional orientation). Nineteen features have been defined in TEXTAL (Ioerger and Sacchettini 2003). They can be categorized into four classes that capture different types of information about density patterns: statistical features related to the electron density distribution, information on symmetry (as defined by the distance from the center of the region to its center of mass), moments of inertia (and their ratios), and features that try to represent the geometric shape of the region. These 19 features can be calculated at different radii. For instance, to model side chains, we calculate the features over spheres of size 3, 4, 5, and 6 Å; this is necessary since amino acids vary in shape and size, and each feature captures slightly different information for different sizes. In the following sections, we show how these features are used in various ways for pattern recognition.

Neural Network to Predict *C*α Coordinates

To determine the three-dimensional coordinates of *C*α atoms, TEXTAL uses a traditional feed-forward neural network to predict the distance of various candidate positions (along the density trace) to the nearest true *C*α, and selects the ones that are predicted to be closest (Ioerger and Sacchettini 2002). The objective of the neural network is to learn the relationship between characteristics of electron density patterns around a coordinate and its proximity to *C*α atoms. We use the 19 features (defined at 3 and 4 Å) to characterize the local density; these features are input to the network, which uses one layer of 20 hidden units with sigmoid thresholds and outputs the predicted distance to a true *C*α atom. The network is trained with a set of coordinates in maps of solved proteins with known distances to true *C*αs, and the network weights are optimized using backpropagation (Hinton 1989).

Heuristic Search to Build Chains

An AI-based approach is also used to link the *C*α atoms (as predicted by the neural network described above) into backbone chains. The primary criterion is based on connectivity in the density map, although there are often many alternative branches, creating ambiguity. Linking *C*α atoms into chains is a combinatorial search problem; whenever possible, an exhaustive search is done to create a solution that maximizes chain length. When a complete search becomes intractable, TEXTAL uses a heuristic function to guide the search for the best way to connect *C*α atoms, based on criteria that favor better adherence to stereo-chemical constraints and secondary structures. These heuristics and decision criteria try to capture the type of reasoning that experienced crystallographers employ, such as following apparent α-helices and β-strands. It should be emphasized that automation of this process is particularly challenging because noisy data, such as breaks in backbone connectivity or close contacts between side chains, can be easily misleading. A more thorough discussion of the methods used to build the backbone can be found in Ioerger and Sacchettini (2002).

Case-Based Reasoning to Connect Broken Chains

This is a backbone improvement step that follows the initial construction of the backbone chains. It attempts to further connect different chains, especially in regions of weak density, for example, where the backbone makes a loop. A case-based reasoning approach is employed to "stitch" chains together. Regions of the structure that probably should have been connected (typically at close extremities of different chains) are identified, and a database of protein structure fragments (constructed from approximately 100 PDB files) is searched to find the most plausible fragment that could connect them. The case matching is done by superposing all chain fragments (of 7 to 11 consecutive *C*α atoms) from the database with the region under consideration and computing the root mean square deviation. If the deviation is small enough, and the electron density in the region is adequately high, then stitching is justified, which may entail adding new *C*α atoms, guided by the retrieved case. This approach is necessary to deal with noise in typical real-world diffraction datasets.

Case-Based Reasoning to Model Side Chains

After the backbone is built, the density is fitted with side chains around the estimated *C*α positions. We adopt a case-based reasoning approach that, in essence, makes the following query: have we seen a region with a similar pattern of density in a previously solved map? Given a spherical region with a radius of 5 Å centered on each of the Cα atoms, a database of about 50,000 previously solved regions (constructed from maps of approximately 200 proteins) is searched to identify the best match. This involves recognition of the unknown patterns of density by comparison to known cases. The comparison can be made by a density correlation metric based on how well the density distribution of the two regions superimpose over each other. However, this objective similarity metric involves computing the optimal superposition between two three-dimensional regions for a large number of rotational conformations; this makes the metric expensive and we cannot afford to run it on the whole database. Thus, we use an inexpensive and approximate feature-based measure of similarity to select *k* cases (400, for instance); the selected cases are then examined by the more expensive density correlation metric to make the final choice.

There are two noteworthy issues related to this approach: (1) a fast and effective similarity metric has to be defined to do the filtering, such that as many good matches as possible are caught in the top *k* cases filtered. In Gopal et al. (2004b), we compare various similarity measures and argue that probabilistic and statistical measures outperform geometric ones (like those based on Manhattan or Euclidean distance); (2) the choice of *k* is important since it influences the performance, both in terms of computational cost and quality of retrievals. In Gopal et al. (2004a), we empirically and theoretically analyze the choice of a suitable value for *k* and provide a model to predict *k* based on a loss function that represents the ability of approximate measures of similarity to rank good matches (according to the objective metric.

The two-stage method for case retrieval has been previously proposed, in different flavors and application domains. For example, in Forbus, Gentner, and Law (2001), MAC/FAC (for "many are called but few are chosen") is proposed as a general strategy for efficient, similarity-based retrieval. Other similar applications include a feature-based recognition of residue environments in proteins (Mooney et al. 2005) and information retrieval (Jones et al. 2000).

Feature Weighting for Pattern Recognition

Features can be noisy, and their relative contributions to the description of local density patterns can vary. Irrelevant features can have a large (negative) impact on pattern recognition (Almuallim and Dietterich 1994). Thus, we use a novel feature-weighting algorithm called Slider to assign weights to features to reflect their relevance in comparing regions of density patterns. Slider can be described as a *filter* approach to feature weighting (John, Kohavi, and Pfleger 1994). It uses a greedy, heuristic search method that tries to optimize the rank of matches relative to mismatches. Slider adjusts weights incrementally, such that, for a given set of regions, known matching regions are ranked better than known mismatching ones (Gopal et al. 2005). In each iteration, we consider only those weights at which matches and mismatches switch as nearer neighbors to query instances; these weights can be efficiently computed by solving linear equations. The classification accuracy is more likely to change at these particular weights, thereby making the search fast and effective. This approach is a more efficient and informed way of finding the weight values that are most promising candidates for update, thereby circumventing the intractability of exhaustive search over all possible weight vectors.

Linear Discriminant Analysis to Detect Disulfide Bridges

A pattern-recognition approach is also used to automatically detect disulfide bridges in electron density maps (Ioerger 2005). A *disulfide bridge* is a covalent bond between the sulfur atoms of two cysteine residues from different parts of the polypeptide chain. The residues with disulfide bridges can be located anywhere in the chain, and this cross-linking contributes to the stability of the protein. Disulfide bridges occur in roughly one out of every four proteins; localizing them in an electron density map can facilitate model building, especially since the presence of a disulfide bridge reveals the position of cysteine residues.

Disulfide bridges are detected by the following method: First, local spherical regions in the electron density map are characterized by 19 numeric features calculated at four different radii (the same features used for building side chains). Then a linear discriminant model is applied to estimate resemblance of the local density pattern to a disulfide bridge, based on a training set with known disulfide and nondisulfide examples. The training cases are used to determine the parameters of the linear discriminant. In particular, the Fisher linear discriminant model is used to optimally maximize class separation, while minimizing variance within each class. This classification method projects the high-dimensional data onto an optimal line in feature-space, along which classification is performed, using a single threshold to distinguish between the two classes.

Quality of Models Built by TEXTAL

The payoff of TEXTAL is mostly in terms of time saved to solve a structure. While a crystallographer may spend several days and sometimes weeks of painstaking effort to interpret a single map, TEXTAL produces a solution in a couple of hours, without human intervention. Even if the model produced by TEXTAL is only partially accurate, it provides a reasonable initial solution, which can be manually refined by the crystallographer to produce a more accurate and complete model.

The quality of output produced by TEXTAL depends on the size and complexity of the structure and the quality of the data. TEXTAL and its subsystems have been designed to work for a wide variety of proteins, of different sizes, with different structural components. TEXTAL usually outputs a reasonable model even with average-quality data (that is, around 3 Å resolution.) Typically CAPRA builds about 80 to 90 percent of the backbone, with less than 1 Å root mean square distance error. (For perspective, the average distance between consecutive $C\alpha$ atoms in proteins is 3.8 Å). TEXTAL usually predicts more than 50 percent of the side chains with the correct identity. In cases where TEXTAL cannot find the exact amino acid, it typically places one that is structurally similar to the correct one. The model produced by TEXTAL can be manually improved, or used to generate better phase information and create a better electron density map, which can be fed back into TEXTAL for subsequent model building. For an average-sized protein (300 residues), TEXTAL's processing time is about 2 hours. Figures 3a and 3b show examples of models built by TEXTAL from experimental data and compare them to the true structures. For a more detailed discussion on the performance of TEXTAL, refer to Ioerger and Sacchettini (2002) and Ioerger and Sacchettini (2003).

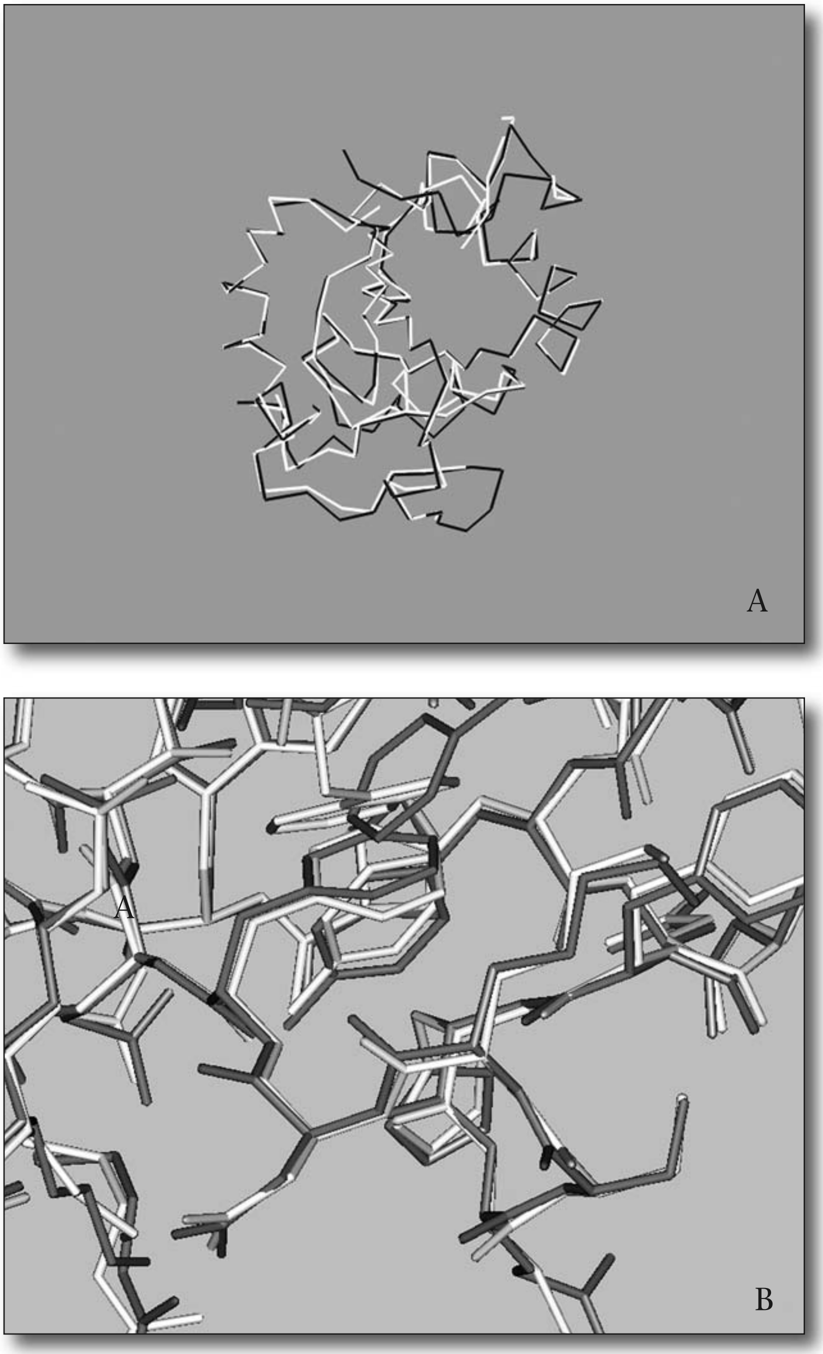

Figure 3. Model of Tryparedoxin-I (PDB ID: 1QK8), A Monomer of 147 Residues.

The manually built and refined model is shown in gray; the model built by TEXTAL is shown in white. TEXTAL builds 96 percent of the structure. Figure 3a (top) shows how TEXTAL builds the α-helices, β-sheets and loops fairly accurately. TEXTAL correctly identifies 93 percent of the residues; the placement of residues is also accurate, as shown in figure 3b (bottom). The root mean square error over all atoms is 0.859 Å. These images were generated with PyMOL (W. L. DeLano, pymol.org).

Development and Deployment

The TEXTAL project was initiated in 1998 as a collaboration among researchers at Texas A&M University. Twenty scientists, students, and programmers (from both the computer science and biochemistry and biophysics departments) have been involved in the project over the years. The TEXTAL software is about 100,000 lines of C / C++, Python, and Perl code. We use

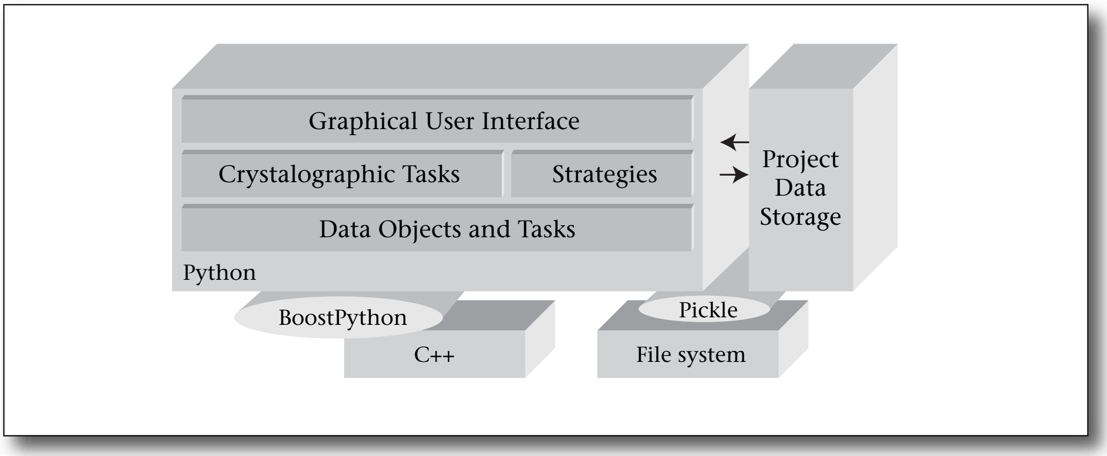

Figure 4. Architecture of the PHENIX System.

Subversion, or SVN[1] for concurrent version control to coordinate the development of TEXTAL. SVN enables tracking of code updates, allows developers to access the latest version of the code from anywhere, and allows multiple developers to work simultaneously on the same code in safety. TEXTAL can be used through a variety of interfaces, from a web-based service (WebTex), to a command-line interface and Python scripting, to a graphical front-end (PHENIX GUI).

Deployment through WebTex

WebTex is a web-based interface to TEXTAL that allows users to upload data, which is processed on our local server.[2] The first version of WebTex was made available to the public in June 2002. Users have to register online for an account and upload their diffraction data or electron density maps, specify options through a simple interface, and submit their jobs. These are processed on our server (an SGI Origin 2000), and the results are automatically emailed to the users. Typically it takes a couple of hours to run TEXTAL on a medium-sized protein, and around 10 minutes to run CAPRA by itself. Users can also monitor online and in real time the progress of their jobs and view and download all the data files related to each of their runs. In September 2005, we launched a new version of WebTex, which included many new features, such as building a model directly from structure factors (instead of requiring users to prepare an electron density map, as in the previous version). Also, users can run the following modules of TEXTAL independently: (1) identify a contiguous biological molecular unit of the protein in an electron density map (by the Findmol program); (2) run CAPRA to find chains of $C\alpha$ atoms that represent the backbone of the molecule; (3) build the complete model, that is, determine and refine the main chain as well as the side chains, and (4) given a trace of $C\alpha$ atoms and a file of structure factors (or an electron density map) as inputs, determine and refine the side chains.

WebTex is freely available to all users. TEXTAL being computationally intensive (Lookup, in particular), restrictions are imposed on the size of maps that can be uploaded and on the number of concurrent jobs that can be submitted. Another practical consideration is our obligation to maintain confidentiality of users' data. During the period from June 2002 to May 2006, WebTex has been used by about 150 users from 80 institutions (both academic and industrial) in 20 countries.

Deployment through PHENIX

PHENIX (Python-based hierarchical environment for integrated xtallography) is a comprehensive software package for automated X-ray crystal structure determination, developed as a collaboration among multiple research groups in the crystallographic computing community (Adams et al. 2004).[3] The PHENIX software provides a variety of algorithms to process collected diffraction data into a refined molecular model, and to facilitate structure solution for both the novice and expert crystallographer. The architecture of the PHENIX system is depicted in figure 4. The Python scripting language[4] provides the backbone of the system. The Boost.Python library is used to integrate C++ code into Python. On top of this, the data objects, crystallographic tasks, strategies (or network of tasks), and finally a graphical user interface are constructed. The project data storage makes use of the pickle mechanism in Python to store data on the file system.

The main components and developers of the PHENIX system include the following:

CCTBX: The Computational Crystallography Toolbox[5] provides a suite of programs for high-throughput structure determination, including routines for tasks such as substructure search and reciprocal-space refinement, implemented at the Lawrence Berkeley National Laboratory.

Phaser: This is a program for phasing macromolecular crystal structures by molecular replacement using maximum likelihood methods, developed at the University of Cambridge.[6]

Solve and Resolve: These systems have been developed at Los Alamos National Laboratory.[7] Solve aims at automated crystallographic structure solution through MAD and MIR phasing, and Resolve performs statistical density modification, local pattern matching, automated model building, and prime-and-switch minimum-bias phasing.

TEXTAL: The automated electron density map interpretation component, developed at Texas A&M University.[8]

The first alpha test version of PHENIX was released in July 2003, mainly to collaborators and selected test users. Thirteen more releases have been made since July 2003; the beta release was released in July 2006. The software is available for commonly used computing platforms: Redhat Linux, HP Tru64, SGI Irix 6.5, and Windows. PHENIX is an ambitious large-scale project that is expected to have a significant impact in the field of protein crystallography. The main payoff is the availability of a wide and comprehensive range of high-throughput crystallography tools in an integrated computational environment. Re-searchers benefit substantially from the ease and flexibility to link various crystallographic tasks together, without having to switch among a multitude of crystallographic data-processing programs and file formats.

Deployment through Binary Distributions

In September 2004, Linux and OSX versions of TEXTAL were made available for download from our website[9] and on CD-ROM. TEXTAL site licenses can be procured from our website. License keys (based on MAC addresses of target machines) are automatically generated and e-mailed to applicants. The distributions of TEXTAL provide more flexibility to the user as compared to WebTex; it allows TEXTAL modules to be invoked from the command line with a variety of options. Another major advantage of the binary distribution is that it allows users to maintain confidentiality of proprietary X-ray diffraction data for highly sensitive projects by running TEXTAL on their own machines.

Conclusion

TEXTAL is an excellent illustration of effective integration of AI technology with other tools to solve a real, difficult, and significant problem in an interdisciplinary fashion. In this article, we have emphasized the importance and challenges of high-throughput protein crystallography in structural genomics, and the contribution of automated protein model-building systems like TEXTAL. We described a variety of AI and pattern-recognition techniques that were employed to address the various facets of this complex problem: a neural network, heuristic search, case-based reasoning, nearest neighbor learning, linear discriminant analysis, and feature weighting. This effort has required addressing a number of practical issues (such as a speed versus accuracy tradeoff, distributed development, multiple interfaces, and so on), as well as a commitment to learning detailed domain knowledge about crystallography and protein structure in order to interact with users, understand common practice, develop effective features, understand sources of noise, and adapt AI algorithms in an intelligent way customized to this domain. We argue that many of the AI issues dealt with, and techniques developed, can be applied in other domains typified by the need to recognize visual patterns (especially in three dimensions), noisy inputs, extensive domain knowledge, large-scale databases, and computationally costly case matching and retrieval.

TEXTAL is continuously being enhanced—existing modules are being improved and new features are being added. Recent developments include (1) an optimization method for side chain identification and fitting based on the *Nelder-Meade Simplex* algorithm; (2) identification of *noncrystallographic symmetry* through pattern recognition in real-space (that is, electron density), which can be used to improve phases; and (3) the use of *selenium* sites from seleno-methionine MAD experiments to enhance side chain building and sequence alignment through identification of methionine residues. We are also exploring the use of similar pattern-recognition techniques to identify and build nucleic acids (RNA and DNA) and other macromolecules in electron density.

Acknowledgments

This work was supported by the National Institutes of Health (Grant PO1-GM63210) and the Robert A. Welch Foundation (Grant A-0015).

Notes

[1] subversion.tigris.org.

[2] textal.tamu.edu.

[3] www.phenix-online.org.

[4] www.python.org.

[5] cctbx.sourceforge.net.

[6] www-ructmed.cimr.ac.uk/phaser/index.html.

[7] www.lanl.gov.

[8] textal.tamu.edu:12321.

[9] textal.tamu.edu:12321.

References

Adams, P. D.; Gopal, K.; Grosse-Kunstleve, R. W.; Hung, L. W.; Ioerger, T. R.; McCoy, A. J.; Moriarty, N. W.; Pai, R.; Read, R. J.; Romo, T. D.; Sacchettini, J. C.; Sauter, N. K.; Storoni, L. C.; and Terwilliger, T. C. 2004. Recent Developments in the PHENIX Software for Automated Crystallographic Structure Determination. *Journal of Synchrotron Radiation* 11(1): 53–55.

Almuallim, H., and Dietterich, T. G. 1994. Learning Boolean Concepts in the Presence of Many Irrelevant Features. *Artificial Intelligence* 69(1–2): 279–305.

Burley, S. K.; Almo, S. C.; Bonanno, J. B.; Capel, M.; Chance, M. R.; Gaasterland, T.; Lin, D.; Sali, A.; Studier, W.; and Swaminathian, S. 1999. Structural Genomics: Beyond the Human Genome Project. *Nature Genetics* 23(10): 151–157.

Diamond, R. 1971. A Real-Space Refinement Procedure for Proteins. *Acta Crystallographica Section A* 27: 436–452.

Forbus, K.; Gentner, D.; and Law, K. 2001. MAC/FAC: A Model of Similarity-Based Retrieval. *Cognitive Science* 19(2): 141–205.

Gopal, K.; Romo, T. D; Sacchettini, J. C; and Ioerger, T. R. 2004a. Efficient Retrieval of Electron Density Patterns for Modeling Proteins by X-ray Crystallography. In *Proceedings of the International Conference on Machine Learning and Applications*, 380–387. Piscataway, NJ: Institute of Electrical and Electronics Engineers.

Gopal, K.; Romo, T. D; Sacchettini, J. C; and Ioerger, T. R; 2004b. Evaluation of Geometric & Probabilistic Measures of Similarity to Retrieve Electron Density Patterns for Protein Structure Determination. In *Proceedings of the International Conference on Artificial Intelligence*, 427–432. Irvine, CA: CSREA Press.

Gopal, K.; Romo, T. D; Sacchettini, J. C; and Ioerger, T. R. 2005. Determining Relevant Features to Recognize Electron Density Patterns in X-Ray Protein Crystallography. *Journal of Bioinformatics & Computational Biology* 3(3): 645–676.

Hinton, G. E. 1989. Connectionist Learning Procedures. *Artificial Intelligence* 40(1): 185–234.

Ioerger, T. R. 2005. Automated Detection of Disulfide Bridges in Electron Density Maps using Linear Discriminant Analysis. *Journal of Applied Crystallography* 38(1): 121–125.

Ioerger, T. R., and Sacchettini, J. C. 2002. Automatic Modeling of Protein Backbones in Electron Density Maps via Prediction of Cα Coordinates. *Acta Crystallographica Section D* 5: 2043–2054.

Ioerger, T. R., and Sacchettini, J. C. 2003. The TEXTAL System: Artificial Intelligence Techniques for Automated Protein Model-Building. In *Methods in Enzymology* volume 374, eds. R. M. Sweet and C. W. Carter, 244–270. Boston: Academic Press.

John, G.; Kohavi, R.; and Pfleger, K. 1994. Irrelevant Features and the Subset Selection Problem. In *Proceedings of the 11th International Conference on Machine Learning*, 121–129. San Francisco: Morgan Kaufmann.

Jones, K. S.; Walker, S.; and Robertson, S. E. 2000. A Probabilistic Model of Information Retrieval: Development and Comparative Experiments. *Information Processing and Management* 36(6): 779–840.

Jones, T. A.; Zou, J. Y.; and Cowtan, S. W. 1991. Improved Methods for Building Models in Electron Density Maps and the Location of Errors in These Models. *Acta Crystallographica Section A* 47: 110–119.

Kleywegt, G. J., and Jones, T. A. 1997. Template Convolution to Enhance or Detect Structural Features in Macromolecular Electron Density Maps. *Acta Crystallographica Section D* 53: 179–185.

Leherte, L.; Glasgow, J. I.; Fortier, S.; Baxter, K.; and Steeg, E. 1997. Analysis of Three-Dimensional Protein Images. *Journal of Artificial Intelligence Research (JAIR)* 7: 125–159.

Levitt, D. G. 2001. A New Software Routine That Automates the Fitting of Protein X-Ray Crystallographic Electron Density Maps. *Acta Crystallographica Section D* 57: 1013–1019.

McKee, E. W.; Kanbi, L. D.; Childs, K. C.; Grosse-Kuntsleve, R. W.; Adams, P. D.; Sacchettini, J. C.; and Ioerger, T. R. 2005. FINDMOL: Automated Identification of Macromolecules in Electron Density Maps. *Acta Crystallographica Section D* 61: 1514–1520.

McRee, D. E. 1999. *Practical Protein Crystallography*. San Diego, CA: Academic Press.

Mooney, S. D.; Liang, M. H.; DeConde, R.; and Altman. R. B. 2005. Structural Characterization of Proteins Using Residue Environments. *Proteins: Structure, Function and Bioinformatics* 61(4): 741–747.

Perrakis, A.; Morris, R.; and Lamzin, V. 1999. Automated Protein Model-Building Combined with Iterative Structure Refinement. *Nature Structural Biology* 6: 458–463.

Smith, T. F., and Waterman, M. S. 1981. Identification of Common Molecular Subsequences. *Journal of Molecular Biology* 147(1): 195–197.

Terry, A. 1983. The CRYSALIS Project: Hierarchical Control of Production Systems, Technical Report HPP-83-19, Stanford University, Stanford, CA.

Terwilliger, T. C. 2002. Automated Main-Chain Model-Building by Template-Matching and Iterative Fragment Extension. *Acta Crystallographica Section D* 59: 34–44.

Tsigelny, I. F. (ed.). 2002. *Protein Structure Determination: Bioinformatic Approach*. La Jolla, CA: International University Line.

Kreshna Gopal is a doctoral student in the Department of Computer Science at Texas A&M University. He received a B.Tech. in computer science and engineering from the Indian Institute of Technology, Kanpur, and an M.S. in computer science at Texas A&M University as a Fulbright fellow. His research interests are in the areas of artificial intelligence and bioinformatics. His email address is kgopal@cs.tamu.edu.

Tod Romo received a B.S. in mathematics and a B.S. in biology from Trinity University and received a Ph.D. from Rice University in biochemistry and cell biology. He has worked as part of the W. M. Keck Center for Computational Biology at Rice and the Texas A&M Center for Structural Biology at the Institute for Biosciences and Technology.

Erik McKee received his B.S. in biochemistry and M.S. in chemistry from Texas A&M University. He is currently a doctoral student in Texas A&M University's Department of Computer Science.

Reetal Pai is a doctoral student at Texas A&M University's Department of Computer Science. Her research interests are in the design and application of AI and pattern-recognition algorithms, particularly in the field of computational biology.

Jacob Smith received a B.S. in mathematics from the University of Texas at Austin. He is currently working on a doctorate, with a focus in computational crystallography, at Texas A&M University's Department of Computer Science.

James Sacchettini is a professor in the Department of Biochemistry and Biophysics with a joint appointment in the Department of Chemistry at Texas A&M University. He is also the Wolfe-Welch chair in sciences at Texas A&M University, the director of the Center for Structural Biology, and a member of the faculty of the Institute of Biosciences and Technology in Houston, Texas. Sacchettini's research has primarily focused on the design and synthesis of novel compounds, which are being tested as drug candidates against tuberculosis and malaria worldwide.

Thomas Ioerger is an associate professor in the Department of Computer Science at Texas A&M University. He received a B.S. from Pennsylvania State University in the area of molecular and cellular biology, and an M.S. and Ph.D. in computer science from the University of Illinois. His research interests are in the areas of AI, multiagent systems, machine learning, and bioinformatics.