PHILOSOPHICAL TRANSACTIONS OF THE ROYAL SOCIETY B BIOLOGICAL SCIENCES

Artificial Intelligence in Molecular Biology: A Review and Assessment [and Discussion]

C. J. Rawlings, J. P. Fox, E. A. Thompson and B. Robson

Phil. Trans. R. Soc. Lond. B 1994 **344**, 353-363
doi: 10.1098/rstb.1994.0074

Artificial intelligence in molecular biology: a review and assessment

C. J. RAWLINGS[1] AND J. P. FOX[2]

[1]*Biomedical Informatics Unit and* [2]*Advanced Computation Laboratory, Imperial Cancer Research Fund, P.O. Box 123, Lincoln's Inn Fields, London WC2A 3PX, U.K.*

SUMMARY

Over the past ten years, molecular biologists and computer scientists have experimented with various computational methods developed in artificial intelligence (AI). AI research has yielded a number of novel technologies, which are typified by an emphasis on symbolic (non-numerical) programming methods aimed at problems which are not amenable to classical algorithmic solutions. Prominent examples include knowledge-based and expert systems, qualitative simulation and artificial neural networks and other automated learning techniques. These methods have been applied to problems in data analysis, construction of advanced databases and modelling of biological systems. Practical results are now being obtained, notably in the recognition of active genes in genomic sequences, the assembly of physical and genetic maps and protein structure prediction. This paper outlines the principal methods, surveys the findings to date, and identifies the promising trends and current limitations.

1. INTRODUCTION

Artificial intelligence (AI) and molecular biology are emerging as a distinctive interdisciplinary subject, with a rapidly growing research community (Hunter 1993; Hunter *et al.* 1993). In this paper we review some of the more prominent recent developments and consider a range of AI techniques and applications. We generally confine the discussion to areas where there are sufficient results to draw some conclusions, if provisional ones.

It is now generally accepted that modern molecular biology research needs many different types of software to support the management, analysis and interpretation of data. It is therefore not surprising that the inherent complexities of the subject would attract AI practitioners and result in the application of AI methods to molecular biological problems. The motivation for many AI researchers is the hope that their technologies can provide a fresh outlook on many of the hard scientific problems that are facing molecular biology today and that the result will be a new generation of intelligent molecular biology software systems.

A difficulty with the term 'artificial intelligence' is that it means different things to different people. The field is in fact developing on a number of fronts, not all of which are of much immediate interest to molecular biologists. Much AI research is aimed at developing theories of animal and human reasoning, planning, learning, vision, hearing and natural language, and the field also has a strong tradition of engineering and mathematics. For example, AI has acquired prominence because of its development of certain novel technologies, like 'expert systems' and 'neural networks'. Their value, as in other engineering fields, is primarily that they seem to be capable of many useful tasks. In contrast, other areas of AI are developing as a formal discipline. For instance, one important range of developments in AI focuses on the development of powerful theories of reasoning, grounded in philosophy and mathematical logic. In between these three extremes of natural science, mathematics and engineering are other subfields of AI which are concerned with developing new types of computer language, developing theories of robotics and other autonomous systems, and with more philosophical but potentially important questions about the nature of information, knowledge and computation.

We have selected five mainstream AI research themes where the software tools have matured sufficiently to be put to practical use in the general field of molecular biology. These themes are the development of knowledge-based systems, the use of symbolic (qualitative) as distinct from numerical (quantitative) computing methods, the automatic learning of new concepts from data (machine learning), the automatic processing and understanding of natural language and methods for searching for plausible solutions to large, complex problems. These technologies are being applied in the area of protein structure to interpret NMR spectra for determining three dimensional structure and to learn new protein folding rules and apply them in protein structure prediction. In the area of molecular genetics, they are being used to create integrated knowledge bases with encyclopaedic coverage of molecular genetics, to

Phil. Trans. R. Soc. Lond. B (1994) **344,** 353–363
Printed in Great Britain

simulate biochemical metabolism and the life cycles of viruses, to identify higher-order structures in nucleic acid sequences, to learn new rules for recognizing active genes in genomic sequence data and also to build novel genetic map construction programs.

There is a great diversity among the different AI techniques being applied to molecular biological problems. Some clear themes and areas of success are now emerging which we highlight in the final section.

2. ADVANCED DATABASES AND KNOWLEDGE BASES

An obvious feature of molecular biology is its capacity to generate prodigious quantities of data. Furthermore, the human genome project is making the management and interpretation of molecular genetic data an acute problem for modern biology. However, notwithstanding the daunting scale of the problem, the difficulties of mere storage and retrieval of information are likely to be satisfactorily addressed by advances in conventional computer and database technology. The most important factors will probably be the combined effects of the increasing performance/cost ratio of computer hardware, improved software technologies able to exploit parallel computer hardware and advanced data networking.

What added benefits might AI have to offer to users of scientific data, particularly those in molecular biology? The short answer is that where conventional databases have been primarily concerned with supporting efficient storage and retrieval, AI's 'knowledge-based systems' have emphasized support for interpretation of the information stored in a database.

GENESIS (Friedland *et al.* 1982) was the first serious attempt to build an integrated knowledge base genetic engineering system. It was intended for use in laboratory data management and experiment planning. GENESIS provided a substantial database of DNA sequence data, extended with documentation and derived data (such as restriction maps). The whole knowledge base was organized into chunks of related information, or 'frames', representing natural biological concepts and their interrelationships. For example, a fragment of the GENESIS knowledge base dealing with genes is shown in figure 1. Each name in this hierarchy refers to a frame, a data-structure which defines the attributes which characterize each class of genes (e.g. globin-genes). The details of specific genes associated with a gene class are stored in 'instance frames' at the bottom of the hierarchy.

animal genes
  protein-coding genes
    contractile-protein genes
    globin genes
    heat-shock genes
    histone genes
plant genes ... etc.

Figure 1. A short extract from the GENESIS knowledge base showing the hierarchical organization of frames representing classes of biological entities. Detailed information, such as the data on specific genes associated with a gene class are stored in 'instance frames' at the bottom of the hierarchy.

Organizing a knowledge base as a frame hierarchy is intuitively natural from a biologist's point of view, but it also offers a number of technical benefits as well. The most important of these is that of automatic 'inheritance' of knowledge about concepts over concept classes. The frame-based representation and the inheritance mechanisms used are closely related to those used in object-oriented database systems (Cattell 1991). Where frame-based systems differ is the use of higher level programming languages such as LISP, Prolog or special rule-based reasoning systems. The GENESIS system was supplied with a rule-based programming language, for example. This language, GENGLISH, permitted the knowledge base designer to attach data manipulation rules to particular concepts in the hierarchy. Operators in these rules permitted the user to simulate the activity of key enzymes used as reagents in genetic engineering (see later).

These and other techniques have proved to have lasting value in the development of molecular biology applications of knowledge bases. For example, Yoshida *et al.* (1992) are developing LUCY, a 'human genome encyclopaedia', which is intended to provide a uniform structure for integrating a range of public and private laboratory databases (currently focused on chromosome 21) but extracts from several public databases such as the Genome Data Base and GenBank have been successfully integrated into the knowledge base, together with over 40 genetic, cytogenetic, restriction and long-range physical maps. Like GENESIS, the LUCY system organizes its data using frame techniques and provides specialized languages appropriate for the representation of molecular biological data.

Unlike GENESIS, LUCY does not have a special-purpose inference language but a language called Prolog (Clocksin & Mellish 1981). Prolog (for Programming in Logic) is a language developed as a result of AI research on mathematical logic and is widely used for logical reasoning and general programming. Logic programming technology was developed in Europe, and European workers were quick to investigate its value as a knowledge representation and knowledge base query language (e.g. Lyall *et al.* 1984; Rawlings *et al.* 1985) and some have argued that it has advantages over conventional programming techniques for use in molecular biology (e.g. Barton & Rawlings 1990). The Argonne laboratories have also put considerable effort into using logic programming techniques in developing integrated knowledge bases to support research in human (Hagstrom *et al.* 1992) and *E. coli* genetics (Baehr *et al.* 1992). A number of studies have looked at the combination of logic programming with frame and object-based data representations. Gray *et al.* (1990) apply this combination of techniques to protein structure analysis, and conclude that the two methods are complementary.

Other logic-based knowledge-based systems include the GeneSys system (Overton *et al.* 1990) which explores issues in the automation of biosequence

PHILOSOPHICAL TRANSACTIONS OF THE ROYAL SOCIETY B BIOLOGICAL SCIENCES

analysis, notably with respect to the structure–function relationship in gene expression and the PAPAIN system (Clark *et al.* 1990) which is designed to automate the analysis of protein sequences with the aim of providing assistance in protein structure prediction.

3. QUALITATIVE MODELLING AND SIMULATION

Closely associated with the methods for building knowledge-based management systems are those for building simulations of biological processes. Because many aspects of molecular biology are not amenable to rigorous mathematical treatment, qualitative approaches for modelling biochemical processes have been developed. The MOLGEN project (Stefik 1981, Friedland & Iwasaki 1985) was one of the first experiments in the qualitative simulation of molecular biological processes. The motivation behind MOLGEN was to plan genetic engineering experiments automatically and in order to do this it was necessary to simulate the activity of the reagents and processes of molecular genetics, e.g. restriction endonucleases, DNA ligases, transcription, translation, etc. The ideas of automated experiment planning have also been developed by Carhart *et al.* (1988) and Jiang *et al.* (1990).

In the MOLGEN-II project (Friedland & Kedes 1985) the emphasis shifted to developing programs that could (re)discover scientific hypotheses and in particular the reasoning involved in the elucidation of attenuation as a method of control of gene expression. This research has required the development of techniques for representing and reasoning about biological processes and experimental techniques. An important outcome of MOLGEN-II (Karp 1993) is thus a qualitative model and simulation of gene expression in the trp operon. The approaches and technologies developed to support qualitative molecular biological simulations used in the MOLGEN project have also influenced simulations of genetic regulation in bacteriophage l (Meyers & Friedland 1984) and DNA metabolism (Brutlag *et al.* 1991). As fundamental biological processes, gene regulation and expression (Weld 1984; Koton 1985) have also been studied in some detail and in some cases models have also been developed of the life cycle of simple organisms such as the lambda bacteriophage (Meyers & Friedland 1984) and the human immunodeficiency virus (Koile & Overton 1989).

Many people (e.g. Karp 1992) have observed that for knowledge-based systems to make a significant impact on molecular biology, they should have a basis in the 'common-sense biochemistry' as taught to every under-graduate biologist. The EcoCYC project (Karp & Riley 1993) is proposing to do this in a comprehensive way for the genetics and biochemistry of the bacteria *E. coli*. Others are concentrating on methods for representing the detailed aspects of cellular metabolism, in particular Michael Mavrovouniotis and co-workers as reviewed in Mavrovouniotis (1993*a*) and further developed in Mavrovouniotis (1993*b*) and Kazic (1993).

4. MACHINE LEARNING

Ever since scientists began to collect and store large collections of data on computers they have been fascinated by the idea that it might be possible to develop techniques for 'discovering' patterns in the data that would be hard for them to find unaided. The classical approach, of course, is for a scientist to closely direct a computer to search for patterns in a database, to confirm specific suspicions or hypotheses by appropriate methods. But there is also a growing body of work on techniques whereby the computer searches for correlations and formulates hypotheses without the guidance of a human investigator. This work ranges from attempts to find statistical regularities automatically (e.g. Blum 1982) to highly ambitious projects aimed at showing that a computer can formulate scientific conjectures and theories without human intervention (e.g. Lenat 1983).

The value of unsupervized 'discovery' methods is controversial. However, in a limited form, techniques for automatically finding regularities in large collections of data are promising to be useful. In machine learning the computer is presented with examples of different data patterns of interest and it then searches for collections of features which will discriminate between the different pattern categories or seeks to identify generic features among them. A number of methods have been developed, the most prominent of which are symbolic induction and neural networks. The former methods generate explicit rules of the form 'if . . . then . . .', whereas the latter use a quantitative weighting technique. We shall concentrate on the latter here, as Sternberg *et al.* discuss rule induction methods in detail elsewhere in this volume.

Neural networks were developed in the 1950s by researchers interested in modelling the brain mechanisms involved in perception. They developed artificial networks in hardware which could learn to recognize patterns, presented as sets of features, by progressively increasing the numerical weight associated with features which are typically present in specific categories of pattern, and decreasing the weight attached to features which are typically absent. Such a network can be trained to distinguish the different categories by providing feedback indicating whether an example is or is not a member of a particular category. Perceptrons, as the early networks were called, subsequently fell out of favour because of demonstrations by Minsky & Papert (1969) that they could not learn to discriminate some important types of pattern, although perceptron-like networks were used by Stormo *et al.* (1982) to recognize translation initiation sites. Recent technical advances, however, have overcome enough of these difficulties that neural network software can be useful in many practical pattern recognition applications.

Among the first applications in molecular biology was the prediction of the secondary structure of globular proteins. Qian & Sejnowski (1988) trained a network by presenting it with amino acid sequences whose secondary structure is known, and tested its ability to correctly classify new sequences which were

PHILOSOPHICAL TRANSACTIONS OF THE ROYAL SOCIETY B BIOLOGICAL SCIENCES

non-homologous with the training set. The initial successful recognition rate with randomly assigned weights was at the chance level of 33% but during training this rose to an average of 64.3% for the three states, a performance level that was superior to other methods available at the time. The reader is directed to a number of recent reviews that cover the use of artificial neural networks for predicting structure and function in both protein and DNA sequences. These include Hirst & Sternberg (1992) and Presnell & Cohen (1993) as well as chapters by Steeg and Holbrook *et al.* in Hunter (1993).

The use of neural networks to detect errors in biological sequences was suggested by Brunak *et al.* (1991). They trained a network to recognize mRNA splicing signals in 33 human genes from the EMBL data library, during which they noticed that some sequences appeared to 'disturb' the learning because the network weights did not stabilize on a specific signal assignment. Subsequent investigation revealed discrepancies from the original papers for three genes due to misprints and other errors of interpretation. A similar study of 241 sequences from GenBank revealed nine new errors. Brunak *et al.* argue that neural networks could be used as 'computerized proof readers', or gatekeepers, to detect possible errors before accepting data into a database.

A more recent development in the use of neural networks in molecular biology is to combine conventional and machine learning techniques. Mural *et al.* (1992) have examined the use of neural networks in finding protein coding regions in DNA sequences. Computer-based recognition of DNA features can be difficult; statistical analysis can help but in many cases the consensus sequence is insufficient to specify the feature of interest. They argued that results from 'knowledge-free' methods, i.e. those which simply use the sequence (figure 2*a*), are encouraging but the networks are large (particularly for complex features such as protein coding regions) and training requires large amounts of supercomputer time. They therefore take a 'knowledge-based' approach, in which they preprocess the sequence statistically, to first identify potentially meaningful biological signals (figure 2*b*), labelling it along its length with seven different measures. These different labellings are then used as input to the network, rather than the base sequence. Using this method they located 90% (71/79) of exons of more than 100 bases and correctly classified 96% (16592/17576) of 100 base sequences coming from coding/non-coding sequences. Of the 1113 test windows classified as coding 92% were correct (8% false positives). Applied to an anonymous 58 kb sequence of human DNA in Huntington's region the method located a number of potential exons clustered as to suggest several genes.

Craven & Shavlik (1993) have revised Mural's approach in order to interpret prokaryotic DNA sequence data, specifically to locate coding regions in genomic sequences and detect frameshift errors. They compared the performance of a network using different representations of the sequences (as bases, codons and other features). They also compared the

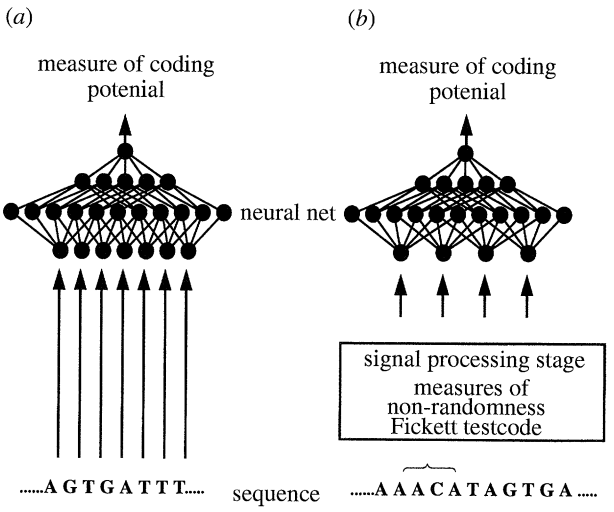


Figure 2. Artificial neural networks can be used to recognize coding regions in genomic sequence. (*a*) Early methods used a 'knowledge-free' approach, but more recent approaches (*b*) have used networks to learn the best combination of different statistical or pattern-based methods developed for gene identification (see text).

technique with conventional approaches, predicting that neural networks should outperform simple statistical methods because they do not make the assumption that the statistics of neighbouring elements in the sequence are independent. They found that the best conventional techniques yielded 87.2% correct assignment of subsequences as coding/non-coding. The best neural network resulted in 87.45% correct assignment, but when statistical techniques were used to define the features of coding regions the neural network learned to make 89.15% correct assignments.

Although the idea that computers can contribute to the development of scientific theories is controversial, Shavlik *et al.* (1992) argue that it has some potential in molecular biology. They propose a scheme whereby a theory is expressed as a set of 'if . . . then . . .' rules, such as a theory which relates sequence data to signals. These rules can be translated into an equivalent network topology in software which can be trained on examples in the usual way. The trained network can then be translated back into a set of rules, yielding a more refined theory. They did two experiments using this scheme, in recognizing *E. coli* promoter sequences, and compared the results with other machine learning and conventional consensus techniques. They found the method to be both superior to other methods and, unlike normal neural networks, the results are in the form of explicit rules which are intelligible to biologists and are more informative than a consensus sequence.

5. LINGUISTIC METHODS

The motivation behind linguistic approaches to molecular sequence analysis is to understand the structure of genetic sequences as languages. These studies are predicated on the widely held assumption

PHILOSOPHICAL TRANSACTIONS OF THE ROYAL SOCIETY B BIOLOGICAL SCIENCES

that genetic sequences are similar to natural languages and that as written languages rely on the sequence of letters (and punctuation) to convey meaning, so genetic sequences carry biological meaning (descriptions of structure and function) in the linear order of bases or amino acids. A long term goal of computational genetic linguists is to develop accurate automatic methods to identify the biological meaning of the features encoded in molecular sequences.

The formal methods used to describe the syntax of languages are grammars. The more complex the language, the greater is the need for representational flexibility in the grammar. David Searls (1993) provides an excellent overview of language theory, the different types of grammatical systems and the classification of genetic grammars. The practical value of formalizing a language as a grammar is that it facilitates the construction of a program that can parse sentences from that language. The most useful outcome of a parser is the parse tree and figure 3 shows how a genetic 'sentence' can be decomposed into other genetic 'phrases' or elements described by a genetic grammar.

The simplest languages can be described by 'regular expressions'. Although simple, many genetic sequence patterns (motifs) associated with biological functions can be expressed using regular expressions. Nevertheless, most sequence pattern matching programs (e.g. QUEST, Abarbanel *et al.* 1984; ARIADNE, Lathrop *et al.* 1987) use a pattern language that has been extended beyond pure regular expressions in order to accommodate some of the less regular features of biological sequences. Therefore, although extended regular languages can capture many of the sequentially local features in genetic sequence motifs (e.g. PROSITE, Bairoch 1991) and have resulted in some important practical programs, many aspects of nucleic acid and protein structure and function are not encoded in local sequence motifs, but are a consequence of local spatial interactions mediated by long range and higher-order sequence relationships.

To extend the range of biological meanings that can be recognized by linguistic methods, it is necessary to consider more complex languages. In a study of the linguistic classification of genetic grammars (Searls 1993) the starting point is a definite clause grammar (DCG) for representing genetic structures. DCGs have a close association with the Horn Clause Logic employed by the logic programming language Prolog (Pereira & Warren 1980) and most Prolog systems are able to interpret and transform a DCG into an executable Prolog program which can be used as a top-down parser for the language described by the DCG.

Prolog support for DCGs is made flexible by the expedient of allowing grammar rules to be augmented with native Prolog code. In what Searls names a string variable grammar (SVG) further extensions to the DCG take advantage of the logic programming paradigm to provide features necessary to describe higher-order interactions among genetic sequences and give SVG properties necessary to describe some of the context-sensitive features of nucleic acids (Searls & Liebowitz 1990) such as nonlinear features found in RNA pseudoknots and other secondary structures formed as a result of internal base pairing.

A practical demonstration of the use of these techniques is the processing of gene structure information incorporated in the GenBank database. In a paper in which they use a grammatical model of eukaryotic gene structure and in which they focus on

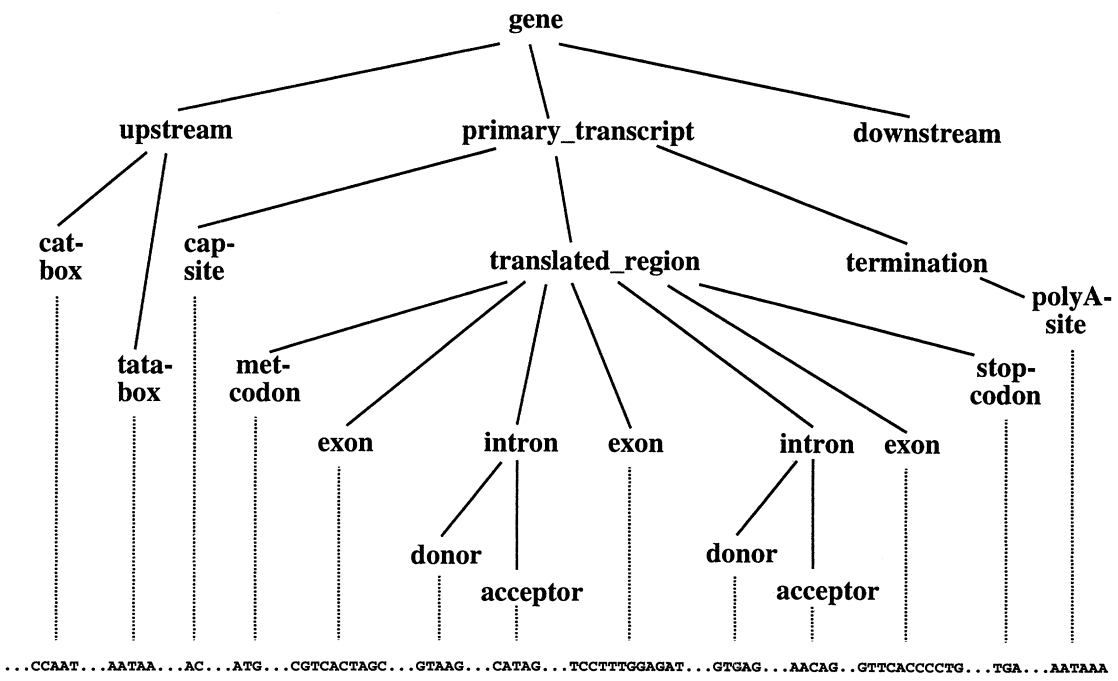


Figure 3. A schematic representation of a parse tree for a gene grammar showing how the parser breaks down the structure into component parts. Adapted with permission from Searls (1993).

globin genes, Aaronson *et al.* (1993) were able to reveal a significant number of errors in the database. These included a small number of incorrectly specified features, but more significantly, a larger number of genetic elements that were missing completely from the GenBank feature table. By exploiting the gene grammar structure and the ability to reason by analogy among α-globin sequences with a high degree of sequence homology, it was possible to propose the existence of 30% more introns and 40% more exons than were listed in the feature tables. As well as providing a new method for data quality control in GenBank, the linguistic approach suggests a possible method for dealing with the important problem of consistency and completeness of the GenBank feature tables: a particular issue for older entries.

Nucleotide and amino acid sequence databases are important primary data resources, but higher order databases – such as catalogues of conserved sequence motifs with known biological properties – are becoming increasingly important. Perhaps the most well known protein sequence motif collection is the PROSITE database (Bairoch 1991). In PROSITE, and other similar databases, extended regular expressions are used to represent the sites and sequence patterns. The coverage of the pattern description language to all the required patterns is an important factor in the usefulness of the database and affects the potential for the database to grow to cover more sophisticated (i.e. long-range) patterns as they are discovered.

In recent releases of PROSITE there have been a number of patterns that cannot be completely described by the PROSITE pattern language. Helgeson & Sibbald (1993) have addressed these problems using a formal linguistic approach and have developed the PALM pattern language. PALM has many features in common with an SVG but is focused on the requirements of representing protein sequence patterns: in particular frequentistic patterns using an operator which counts occurrences of particular patterns in a sequence segment.

6. CONSTRAINT-BASED SYSTEMS

Many problems of interpreting data from molecular biology experiments have combinatorial complexity (or worse) and computationally efficient methods are important if competing interpretations are to be evaluated in reasonable time. It can also be argued that many aspects of scientific reasoning can be naturally characterized as the search for consistent interpretations among data and the hypotheses that constrain the possible solutions. Constraint-based systems provide a solution to both these considerations.

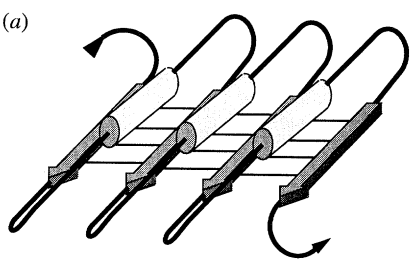
The constraint-based approach represents the dependencies among all the objects in a problem as constraints and uses a problem solver that will 'prune' illegal solutions and their consequents (constraint propagation) when a constraint is violated (Kumar 1992).

The first ever example of an AI application in the molecular sciences, the DENDRAL system for interpreting mass spectrometer data (Lindsay *et al.* 1980) used a 'generate and test' approach, which can be considered as the most straightforward constraint-based method. The determination of protein structure from nuclear magnetic resonance (NMR) spectra has also been addressed using constraint-based problem solving techniques. These and other AI approaches to protein structure determination from NMR spectra are reviewed in Edwards *et al.* (1993).

RNA structure prediction is another problem with combinatorial complexity that has been the subject of constraint-based approaches. In Heuze (1989) a constraint-based implementation of an established combinatorial method for predicting RNA secondary structure (Gouy 1989) uses parallel constraint logic programming (see later). Major *et al.* (1991) use a hybrid approach in their MC-SYM system for predicting the three-dimensional structure of RNA. MC-SYM combines the use of a symbolic programming language (the functional language Miranda) to implement a constraint satisfaction algorithm which prepares a partial model of the RNA structure for a numerical energy minimization package (CHARMm) which then computes the detailed energetic and conformational constraints on RNA folding.

Protein topology prediction, i.e. hypothesizing the most plausible spatial organization of secondary structure elements has a known combinatorial complexity (figure 4) for all-β and α/β proteins (Clark *et al.* 1991). By constructing this problem as a constraint satisfaction problem and using protein folding rules as constraints, Clark *et al.* were able to test the accuracy and coverage of a number of protein folding rules published in the scientific literature. This was possible because the representation of constraints and the query language for their database of protein topolo-

(*a*)



(*b*)

no. of strands	2	6	10
no. of topologies	3	87 480	3.57×10^{10}

Figure 4. (*a*) The prediction of protein topology in proteins containing all β-sheet or alternating α/β structures involves determining the most plausible order and orientation for the β-strands (arrows) in the sheet. This problem has combinatorial complexity (*b*) with the number of possible topologies (t) rising with the number of β-strands (n) according to the function: $t = n!(3n - 1)/2$.

PHILOSOPHICAL TRANSACTIONS OF THE ROYAL SOCIETY B BIOLOGICAL SCIENCES

gical structures (Rawlings *et al.* 1985) were rules written in the Prolog language.

More recently, by analysis of a protein topology database and the use of inductive learning methods (see Sternberg *et al.*, this symposium), Clark *et al.* (1993) have extended the number of constraints on protein folding in α/β proteins and re-implemented their method in the parallel constraint logic programming (CLP) system, ElipSys. CLP is a recent development in AI languages, extending the features of logic programming languages, such as Prolog, with special purpose problem solving techniques from operations research and algebraic constraint propagation (Van Hentenryk 1991). Clark *et al.* (1993) demonstrated that CLP is very well suited to programming these types of problems, both from the point of view of providing a concise and comprehensible representation language, and from that of computational efficiency. They showed that they could achieve considerable performance gains (approximately 60-fold) over the original Prolog implementation. Furthermore, because the ElipSys system is also a parallel logic language, it was possible to further increase performance on a parallel computer system as an almost linear function of the number of the processing elements available.

A major problem in the use of a logic representation of protein folding rules is providing a means to deal with uncertain or partial constraints. Using in-built optimization operators in the ElipSys system Clark *et al.* (1993) developed an uncertainty management scheme using a scoring scheme which uses the number of times a constraint (rule) was found to be true (or false) in the protein topology database. By using this scheme, the most plausible topological structure(s) were those that violated the fewest or weakest constraints.

Protein topology prediction is a good example of knowledge-intensive constraint satisfaction where there is a relatively large number of high level constraints that can be used to prune the hypothesis space. At the other extreme are data-intensive problems where there are few general rules but many individual constraints coming from experimental data. The assembly of ordered maps of genetic markers from measures of distance between the markers would generally be characterized as data-intensive. The assembly of restriction maps from the size of fragments from single and double restriction endonuclease digests of DNA was identified early on as amenable to AI techniques and in particular constraint-based search (Stefik 1978, 1981).

In the CPROP program (Letovsky & Berlyn 1992) the construction of a genetic linkage map is dealt with as a constraint satisfaction problem. All the order information is represented as constraints and CPROP uses rules for combining local or partially ordered maps into larger maps. As regions are combined, new distance constraints are created and a constraint propagation phase is initiated to ensure that the derived information is kept consistent.

In recent work by Doursenot *et al.* (1993), the ElipSys parallel CLP system was used to program a constraint-based approach to assembling long-range physical genetic maps from hybridization fingerprinting methods (Lehrach *et al.* 1990). Their CME program combines heuristic data reduction methods and constraint satisfaction to derive the optimal ordering of DNA markers. By using CME, Doursenot *et al.* were able to show the power of introducing different constraints to prune the hypothesis space and reduce program execution times. They also showed that CME was able to build maps that were as good (by the same criteria) as those generated originally (Mott *et al.* 1993). Furthermore it could be demonstrated, for the first time, that for the given data the original map was optimal.

7. DISCUSSION AND ASSESSMENT

In the early days of applied AI in molecular biology (and many other areas), there was great optimism that the leverage provided by AI technology would overcome some of the practical difficulties of using complex, and sometimes unwieldy AI programming environments. Furthermore, there was a general belief that the development of cost-effective symbolic computing hardware that could efficiently run AI programs written in languages such as Lisp and Prolog would ensure that such systems would eventually become widely accepted. For many reasons, however, this was not what happened and increasingly the applied AI community has become more pragmatic in its approach, so that the emergence of hybrids of AI and conventional software techniques are now common. This trend is evident in AI applications in molecular biology and perhaps most clearly seen in the area of advanced knowledge-based systems.

One of the developments enabling AI methods to be applied to molecular biology problems is the convergence of AI and database technologies, which is leading to a new class of databases called 'deductive databases'. These systems have the ability to manage large-scale data together with the 'intelligence' arising from logical reasoning and other inference capabilities. These deductive and other knowledge-based systems are being made the basis for building what are often referred to as encyclopaedic systems (e.g. Karp 1992; Yoshida *et al.* 1992) that bring together not just the data from a wide variety of sources into a common framework, but also simulations of biochemical processes, experiment planning, data interpretation and other functions. If these projects are successful, then they will greatly ease the problems encountered by many scientists when trying to assemble information from the many different molecular biology and genetics databases.

A major difficulty in earlier large-scale knowledge-based projects such as GENESIS (Friedland & Kedes 1985) was the huge amount of biological detail that had to be encoded before significant results were possible. The rapidly changing nature of AI software technology and the absence of a consensus on programming environments has also meant that much work has had to be repeated. A challenge to

PHILOSOPHICAL TRANSACTIONS OF THE ROYAL SOCIETY B BIOLOGICAL SCIENCES

the new projects involved in this research will be to ensure that as well as addressing the scientific issues (computer science and biological science), the necessary procedures are put in place to enable these digital encyclopaedia to be used by any computational biologist: not just those that are prepared to program in Lisp or Prolog.

Early applications of AI in molecular biology drew heavily on the capabilities of the Lisp programming language. Although Lisp is still popular in many centres, particularly in the U.S.A., there is a trend developing for the adoption of logic programming languages, such as Prolog, for molecular biological applications. One advantage that Prolog brings is its close relationship with a relational style of programming and the facility with which it is possible to link to relational database systems. The Prolog language also provides the structures necessary to build higher-level representations such as object-oriented (Gray *et al.* 1990) and frame-based representations (e.g. Overton *et al.* 1990; Yoshida *et al.* 1992). The most active users of Prolog are principally the knowledge-based systems development community (e.g. Clark *et al.* 1990; Hagstrom *et al.* 1992) but Prolog is also the language of choice for genetic grammar and linguistics research (Searls 1993) where the close relationship between definite clause grammars and Prolog is a clear advantage.

The appealing notion that computer programs can be built that can hypothesize new relationships (e.g. rules for predicting protein structure from amino acid sequence) by learning them directly from molecular biological data has yielded mixed results. The most popular and successful approach for machine learning of molecular biological concepts has been artificial neural networks. However, not all commentators are wholeheartedly optimistic about their future (e.g. Hirst & Sternberg 1992). One problem that this research area has helped to bring into focus, and that pervades computational molecular biology is selecting the correct training and test data sets and having accepted criteria for success. There is an urgent need for scientists in computational molecular biology to establish a well-documented and easily accessible set of reference data that can be used to compare competing methodologies.

Machine learning research in molecular biology has shown two further areas where convergence between technologies are yielding clear results. In the first, Shavlik *et al.* (1992) have developed a method for automatically extracting a rule-based 'comprehensible' description of what has been learned from a trained artificial neural network. This means that in future, the rules 'learned' by a neural network will be more easy to both understand and incorporate into knowledge-based systems. The second example of hybrid or converging technologies is the development of combinations of signal processing (e.g. statistical methods) and machine learning techniques. This is how the GRAIL system (Mural *et al.* 1992) predicts protein coding regions in genomic DNA and using a similar approach Craven & Shavlik (1993) showed that the neural network learned how to combine the results from the best pattern-based and statistical gene recognition programs to achieve a better result than any of them did individually.

An important goal for researchers who cross the boundaries of machine learning and natural language understanding is to be able to learn a grammar from examples of the language. Clearly, the possibility of learning detailed genetic grammars directly from molecular sequences must be a very long-term goal. However, the use of a combination of machine learning and genetic grammars to identify undocumented genetic elements in the GenBank database (Aaronson *et al.* 1993) is an important result for grammar induction techniques that complements the continued developments of more sophisticated grammars for representing biological structure and function in sequence data.

For AI techniques to be fully integrated into molecular biology computing, it is important that they be able to build systems that are efficient enough to reason with raw data and assist directly with interpretation of scientific data. Constraint-based systems have many advantages when it comes to solving large scale application problems and an important recent development in constraint handling technology has been the convergence of logic programming, operations research and other problem solving techniques in constraint logic programming languages. Several recent examples of the application of CLP, including protein topology prediction (Clark *et al.* 1993), prediction of RNA structure (Heuze 1989) and the assembly of large-scale physical genetic maps (Doursenot *et al.* 1993), have demonstrated that this technology is sufficiently general and powerful to address a wide range of problems efficiently. Ongoing developments in CLP technology will link CLP languages with deductive databases to produce constrained deductive databases which should be very well suited to many more problems in molecular biology.

To conclude, this paper has drawn on a selection of recent research and publications to demonstrate the breadth of techniques and problems that have been addressed in AI and molecular biology. There are now signs that, through the adoption of an increasingly pragmatic approach and the integration with more traditional software technologies, some important practical results are being achieved and that AI is now making a contribution to computational molecular biology.

We thank Dominic Clark, Catherine Hearne and Simon Parsons for their comments on earlier drafts of this paper.

REFERENCES

Aaronson, J.S., Haas, J. & Overton, G.C. 1993 Knowledge discovery in GenBank. In *Proceedings of First International Conference on Intelligent Systems for Molecular Biology* (ed. L. Hunter, D. Searls & J. Shavlik), pp. 3–11. Washington: AAAI Press.

Abarbanel, R.M., Wieneke, P.R., Jaffe, D.A. & Brutlag, D.L. 1984 Rapid searches for complex patterns in biological molecules. *Nucl. Acids Res.* **12**, 263–280.

PHILOSOPHICAL TRANSACTIONS OF THE ROYAL SOCIETY B BIOLOGICAL SCIENCES

Baehr, A., Dunham, G., Ginsburg, A., Hagstrom, R. *et al.* 1992 An integrated database to support research on *Escherichia coli*. *Tech. Rep. ANL-92/1*. Argonne National Laboratory.

Bairoch, A. 1991 PROSITE: a dictionary of sites and patterns in proteins. *Nucl. Acids Res.* **19**, 2241–2245.

Barton, G.J. & Rawlings, C.J. 1990 A PROLOG approach to analysing protein structure. *Tetrahedron Comput. Meth.* **3**, 739–756.

Blum, R.L. 1982 Discovery, confirmation and incorporation of causal relationships from a large time-oriented clinical database: the RX project. *Comput. Biomed. Res.* **15**, 164–187.

Brunak, S., Engelbrecht, J. & Knudsen, S. 1991 Neural network detects errors in the asssignment of mRNA splice sites. *Nucl. Acids Res.* **18**, 4797–4801.

Brutlag, D.G., Galper, A.R. & Millis, D.H. 1991 Knowledge-based simulation of DNA metabolism: prediction of enzyme action. *Comput. Applic. Biosci.* **7**, 9–19.

Carhart, R.E., Cash, H.D. & Moore, J.F. 1988 StrateGene: object-oriented programming for molecular biology. *Comput. Applic. Biosci.* **4**, 3–9.

Cattell, R.G.G. 1991 *Object data management: object-oriented and extended relational database systems*. Reading, Massachusetts: Addison-Wesley.

Clark, D.A., Rawlings, C.J., Barton, G.J. & Archer, I. 1990 Knowledge-based orchestration of protein sequence analysis and knowledge aquisition for protein structure prediction. *Proceedings: AAAI Spring Symposium 1990*, 28–32.

Clark, D.A., Rawlings, C.J., Shirazi, J., Veron, A. & Reeve, M. 1993 Protein topology prediction through parallel constraint logic programming. In *Proceedings of First International Conference on Intelligent Systems for Molecular Biology* (ed. L. Hunter, D. Searls & J. Shavlik), pp. 83–91. Washington: AAAI Press.

Clark, D.A., Shirazi, J. & Rawlings, C.J. 1991 Protein topology prediction through constraint-based search and the evaluation of topological folding rules. *Protein Engng* **4**, 751–761.

Clocksin, W.F. & Mellish, C.S. 1981 *Programming in Prolog*. Berlin: Springer-Verlag.

Craven, M.W. & Shavlik, J.W. 1993 Learning to predict reading frames in *E. coli* sequences. In *Proceedings of 26th Hawaii International Conference on Systems Science–Biotechnology* (ed. L. Hunter), pp. 773–782. IEEE Computer Society.

Doursenot, S., Clark, D.A., Rawlings, C.J. & Veron, A. 1993 Contig mapping using ElipSys. In *Proceedings of AI and the Genome Workshop, 13th International Joint Conference on Artificial Intelligence*. Chambery, France.

Edwards, P., Sleeman, D., Roberts, G.C.K. & Yun-Lian, L. 1993 An AI approach to the interpretation of the NMR spectra of proteins. In *AI and molecular biology* (ed. L. Hunter), pp. 396–432. AAAI Press.

Friedland, P. & Kedes, L.H. 1985 Discovering the secrets of DNA. *Communs ACM* **28**, 1164–1186.

Friedland, P. & Iwasaki, Y. 1985 The concept and implementation of skeletal plans. *J. Automated Reason.* **1**, 161–208.

Friedland, P., Kedes, L., Brutlag, D.L., Iwasaki, Y. & Bach, R. 1982 GENESIS: a knowledge based genetic engineering simulation system for representation of genetic data and experiment planning. *Nucl. Acids Res.* **10**, 323–340.

Gouy, M. 1989 Secondary structure prediction of RNA. In *Nucleic acid and protein sequence analysis. Practical approach* (ed. M. J. Bishop & C. J. Rawlings), pp. 259–284. Oxford: IRL Press.

Gray, P.M.D., Paton, N.W., Kemp, G.J.L. & Fothergill, J.E.F. 1990 An object-oriented database for protein structure analysis. *Protein Engng* **3**, 235–243.

Hagstrom, R., Michaels, G.S., Overbeek, R. *et al.* 1992 GenoGraphics for Openwindows. *Tech. Rep. ANL-92/11*. Argonne National Laboratory.

Helgeson, C. & Sibbald, P.R. 1993 PALM – a pattern language for molecular biology. In *Proceedings of First International Conference on Intelligent Systems for Molecular Biology* (ed. L. Hunter, D. Searls & J. Shavlik), pp. 172–180. Washington: AAAI Press.

Heuze, P. 1989 RS2P RNA secondary structure prediction in ElipSys. *Tech. Rep. ElipSys/10*. European Computer-Industry Research Centre, Ababellastrasse 17, D8000, Munich 81, Germany.

Hirst, J. & Sternberg, M.J.E. 1992 Prediction of the structural and functional features of protein and nucleic acid sequences by artificial neural networks. *Biochemistry* **31**, 7211–7218.

Hunter, L. 1993 *Artificial intelligence in molecular biology*. California: AAAI Press/The MIT Press.

Hunter, L., Searls, D. & Shavlik, J. 1993 *Proceedings of First International Conference on Intelligent Systems for Molecular Biology*. Menlo Park, California: AAAI Press.

Jiang, K., Zheng, J., Higgins, S.B. *et al.* 1990 A knowledge-based experimental design system for nucleic acid engineering. *Comput. Applic. Biosci.* **6**, 205–212.

Karp, P. 1992 A large knowledge-base of bacterial genes and metabolism. In *Proceedings of AAAI Workshop on Communicating Scientific and Technical Thinking*, pp. 133–137. Washington: AAAI Press.

Karp, P. 1993 A qualitative biochemistry and its application to the regulation of the tryptophan operon. In *AI and molecular biology* (ed. L. Hunter), pp. 289–323. Washington: AAAI Press.

Karp, P. & Riley, M. 1993 Representations of metabolic knowledge. In *Proceedings of First International Conference on Intelligent Systems for Molecular Biology* (ed. L. Hunter, D. Searls & J. Shavlik), pp. 207–215. Washington: AAAI Press.

Kazic, T. 1993 Representation, reasoning and the intermediary metabolism of *Eschericia coli*. In *Proceedings of 26th Hawaii International Conference on Systems Science–Biotechnology* (ed. L. Hunter), pp. 853–862. IEEE Computer Society.

Koile, K. & Overton, G.C. 1989 A qualitative model for gene expression. In *Proceedings of the 1989 Summer Computer Simulation Conference*. Society for Computer Simulation.

Koton, P.A. 1985 Towards a problem solving system for molecular genetics. MIT Laboratory of Computer Science; *Technical Report MIT/LCS/TR-338*.

Kumar, V. 1992 Algorithms for constraint satisfaction problems: a survey. *AI Mag.* **13**, 32–44.

Lathrop, R., Webster, T.A. & Smith, T.F. 1987 ARIADNE: Pattern-directed inference and hierarchical abstraction in protein structure. *Communs ACM*. **30**, 909–921.

Lehrach, H., Drmanac, R., Hoheisel, J. *et al.* 1990 Hybridization fingerprinting in genome mapping and sequencing. *Genome Analysis*, vol. 1 (*Genetic and physical mapping*), pp. 39–81. Cold Spring Harbour Laboratory Press.

Lenat, D.B. 1983 The role of heuristics in learning by discovery: three case studies. In *Machine learning: an artificial intelligence approach* (ed. R. S. Michalski, J. G. Carbonell & T. M. Mitchell), pp 243–306. Palo Alto, California: Tioga Press.

Letovsky, S. & Berlyn, M.B. 1992 CPROP: A rule-based program for constructing genetic maps. *Genomics* **12**, 435–446.

Lindsay, R.K., Buchanan, B.G., Feigenbaum, E.A. & Lederberg, J. 1980 *Applications of artificial intelligence for organic chemistry: the DENDRAL project*. New York: McGraw-Hill.

PHILOSOPHICAL TRANSACTIONS OF THE ROYAL SOCIETY B BIOLOGICAL SCIENCES

Lyall, A., Hammond, P., Brough, D. & Glover, D. 1984 BIOLOG–a DNA sequence analysis system in Prolog. *Nucl. Acids Res.* **12**, 633–642.

Major, F., Turcotte, M., Gautheret, D., Lapalme, G. & Cedergren, R. 1991 The combination of symbolic and numerical computation for three-dimensional modeling of RNA. *Science, Wash.* **253**, 1255–1260.

Mavrovouniotis, M.L. 1993*a* Identification of qualitatively feasible metabolic pathways. In *AI and molecular biology* (ed. L. Hunter), pp. 325–364. AAAI Press.

Mavrovouniotis, M.L. 1993*b* Identification of localized and distributed bottlenecks in metabolic pathways. In *Proceedings of First International Conference on Intelligent Systems for Molecular Biology* (ed. L. Hunter, D. Searls & J. Shavlik), pp. 275–283. Washington: AAAI Press.

Meyers, S. & Friedland, P. 1984 Knowledge-based simulation of genetic regulation in bacteriophage lambda. *Nucl. Acids Res.* **12**, 1–9.

Minsky, M. & Papert, S. 1969 *Perceptrons*. Cambridge, Massachusetts: MIT Press.

Mott, R., Grigoriev, A., Maier, E., Hoheisel, J. & Lehrach, H. 1993 Algorithms and software tools for ordering clone libraries: applications to the mapping of the genome of *Schizosaccharomyces pombe*. *Nucl. Acids Res.* **21**, 1965–1974.

Mural, R.J., Einstein, J.R., Guan, X., Mann, R.C. & Uberbacher, E.C. 1992 An artificial intelligence approach to DNA sequence feature recognition. *Trends Biotechnol.* **10**, 66–69.

Overton, G.C., Koile, K. & Pastor, J. 1990 GeneSys: a knowledge management system for molecular biology. In *Computers and DNA SFI Studies in the sciences of complexity* (ed. G. Bell & T. Marr), pp. 213–239. Addison-Wesley.

Pereira, F.C.N. & Warren, D.H.D. 1980 Definite clause grammars for language analysis. *Artif. Intell.* **13**, 231–278.

Presnell, S.R. & Cohen, F.E. 1993 Artificial neural networks for pattern recognition in biochemical sequences. *A. Rev. Biophys. Biomol. Struct.* **22**, 283–298.

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

Rawlings, C.J., Taylor, W.R., Nyakairu, J., Fox, J. & Sternberg, M.J.E. 1985 Reasoning about protein topology using the logic programming language PROLOG. *J. molec. Graph.* **3**, 151–157.

Searls, D.B. & Liebowitz, S. 1990 Logic grammars as a vehicle for syntactic pattern recognition. In *Proceedings of Workshop on Syntactic and Structural Pattern Recognition*, pp. 402–422. International Association for Pattern Recognition.

Searls, D.B. 1993 The computational linguistics of biological sequences. In *Artificial intelligence in molecular biology* (ed. L. Hunter), pp. 47–120. California: AAAI Press.

Shavlik, J.W., Towell, G.G. & Noordewier, M.O. 1992 Using neural networks to refine existing biological knowledge. *Int. J. Genome Res.* **1**, 81–107.

Stefik, M. 1978 Inferring DNA structures from segmentation data. *Artif. Intell.* **11**, 85–114.

Stefik, M. 1981 Planning with Constraints [MOLGEN: Part 1]. *Artif. Intell.* **16**, 111–140.

Stormo, G.D., Schneider, T.D., Gold, L. & Ehrenfuecht, A. 1982 Use of the perceptron algorithm to distinguish translational initiation sites in *E. coli*. *Nucl. Acids Res.* **10**, 2997–3011.

Van Hentenryck, P. 1991 Constraint logic programing. *Knowledge Engng Rev.* **6**, 151–194.

Weld, D.S. 1984 Switching between discrete and continuous process models to predict genetic activity. MIT Artificial Intelligence Laboratory; *Tech. Rep. 7–93.*

Yoshida, K., Smith, C. Kazic, T. *et al.* 1992 Toward a human genome encyclopedia. In *Proceedings of the International Conference on Fifth Generation Computer Systems 1992* (ed. ICOT), pp. 307–319. ICOT.

Discussion

E. A. Thompson (*Department of Statistics, University of Washington, U.S.A.*). A statistician might call 'constraint based-problem solving', 'data-based model fitting', and there are probability network methods for finding solutions to optimization problems on complex spaces. In particular, Markov chain Monte Carlo methods applied on these probability networks can explore the very large potential solution spaces. Does Dr Rawlings have any comment on the relationship of these probabilistic methods to the deterministic optimization algorithms and/or neural nets?

C. J. Rawlings. There are several reasons why the CLP and other AI approaches are considered useful. Firstly, the AI and logic programming framework allows information in the broader sense to be employed in problem solving and thus a close coupling can be achieved between all relevant information, not just that part where quantitative data has been collected. Secondly, there are many problems, in molecular biology and elsewhere, where there is insufficient data for a statistical or mathematical analysis to be appropriate. Where such detailed data exists, there is a clear case that statistical methods should be employed. The CLP and AI approaches do not exclude mathematical and statistical approaches and there is an increasing trend towards building hybrid and cooperative systems that exploit the flexibility of AI programming environments with the precision of statistical and numerical analyses.

Deterministic information represented in AI programs is often used to make the problem-solving process more comprehensible to the user (an important factor for many applications where safety is a prime concern) and to help guide the search process and achieve an efficient implementation. It is not clear how such strategic and procedural knowledge could be usefully exploited by probabilistic methods. On the other hand, the flexible optimization methods embodied in CLP languages can subsume statistical methods or could be programmed to exploit a statistical approach in favour of other optimization methods such as branch and bound search. The approach we have adopted is to complement non-deterministic optimization techniques (e.g. simulated annealing) with deterministic methods embodied in CLP and we believe that this is a practical and productive way of building molecular biology software.

B. Robson (*Proteus International plc, Macclesfield, U.K.*). Although I am very much an artificial intelligence enthusiast, I am concerned that in many applications to molecular biology, it is a question of 'new lamps for old'. Many techniques already exist which are often

PHILOSOPHICAL TRANSACTIONS OF THE ROYAL SOCIETY B BIOLOGICAL SCIENCES

almost as good, occasionally better, and widely available. Almost all the rules in the PROSITE database, for example, can be coded (if sometimes less elegantly) using the 'regular expression' system widely available in Unix systems. Complex grammar systems are not always essential, even I suspect for abstract descriptions of the genome. Similarly, genetic algorithm approaches are in almost all respects comparable with the simplex method of Relder and Mead, at least for global searching as applied by us. Although I believe that in artificial intelligence the Emperor's New Clothes are *really* there, they are often, I suspect, old clothes cleaned up.

C. J. RAWLINGS. The scope of this question is very broad and there are several aspects that require an answer. The first is simply that there are many equally legitimate approaches to software development for research purposes, and each makes a distinctive contribution. The first approach, which could be categorized as being bottom-up, is that typified by much contemporary molecular biology software. A specialist employs a traditional programming language (e.g. Fortran or C) to painstakingly hand craft a program that solves a particular problem. Such a program can make a significant contribution to science, but its range of applications is generally narrow. The AI approach takes a more top-down approach which exploits general problem-solving methods (perhaps developed for an entirely different class of applications) and applies them to new problems. From the perspective of a scientific programmer who perhaps only considers the initial scientific result to be important, this could appear to be a re-invention.

Those of us that espouse AI methods are motivated by some of the other advantages that high-level programming environments bring, such as the possibility of re-using knowledge and information, generating solutions that can be expressed in terms that are comprehensible to the user (such as in rule-based expert systems) and the possibility that if a generic solution can be found, related problems can be solved without extensive re-programming. Furthermore, understanding a particular problem as an instance of a more general class has other advantages since the theoretical or computational framework developed for the general problem can sometimes provide new scientific insights for the specific one.

Contrary to what is implied in the question, the application of methods from natural language and grammars to the recognition of sequence motifs is a good example of the advantages of taking a broader view. Although it is true that Unix regular expressions can code some of the PROSITE patterns, a significant and increasing number cannot. The research into more advanced grammars to represent complex patterns and to build parsers that can recognize them is attempting to anticipate increasing problems in this area. Furthermore, because the grammatical systems are so general the same family of grammars can also be made to recognize much more complex genetic structures (e.g. RNA pseudoknots) and be re-used in projects that are investigating the use of machine learning techniques in DNA sequence analysis.

PHILOSOPHICAL TRANSACTIONS OF THE ROYAL SOCIETY B BIOLOGICAL SCIENCES