1 Introduction

The general goal of the research proposed here is to develop a computer program (termed MOLGEN) to assist a molecular geneticist in planning laboratory experiments. It is to be an interactive system, drawing on both the expertise of the human geneticist and the expert knowledge stored in the data base. The system must be convenient and comfortable for the geneticist, and will need to be sophisticated and powerful from the point of view of computer science. The program will itself be an experiment in the development of intelligent systems.

The need for such a program is suggested by the recent growth of technology in molecular genetics. In the past three years many new site-specific restriction enzymes have been discovered (Nathans 1975). These enzymes have been applied to physical mapping of chromosomes, nucleotide sequence analysis of DNA, isolation of genes, and restructuring of DNA molecules. Such rapid growth extends throughout the field, with continuing advances in separative and physical techniques. With the many new procedures, there are possibilities for novel combinations of techniques which broaden the horizon of possible experiments. At the same time, it has become increasingly difficult for any one scientist to keep track of what is available, as well as the limitations and use of the newer techniques. Many "ingenious" discoveries, in fact, can be viewed as a judiciously selected combination of well established unit processes. Hence an intelligent computer system is likely to be pragmatically useful in this area.

An immediate application of the proposed system will be the simulation of recombination of random segments of bacterial DNA cut by restriction endonucleases. In experiments currently underway in Professor Lederberg's laboratory, geneticists would like to have quantitative predictions of reaction component concentrations at any particular time, given initial concentration of Bacillus subtilis DNA and site-specific restriction enzymes. The conception of the enzyme simulation program (see part III) in the MOLGEN system was directed precisely toward this kind of need. Part of this effort will involve the creation of a knowledge base of restriction enzymes with associated kinetic data, recognition site information, and characterization of availability and degree of purity. Convenient access to a comprehensive source of this information would itself be of considerable value.

A subproject which should also prove very useful to the geneticist will be the summarization of all empirically established sequences of DNA and RNA, along with restriction enzyme site specificity. Such information might play an important role in the current effort of Professor Lederberg's laboratory in dissecting the genetic structure of Bacillus subtilis. Recognition sites in a wide variety of naturally occuring nucleic acid species has been summarized in a preliminary way by Sobell (Sobell 1973).

An important factor in selecting this domain was the availability of a great many tangible problems covering a wide range of

difficulty. This means that it will be possible to develop the system incrementally, and still be able to perform useful tasks even at the outset.

That branch of computer science known as artificial intelligence (AI) has already developed techniques for computer-assisted problem solving in the experimental sciences. Examples of successful projects are: the DENDRAL project at Stanford in the field of mass spectrometry; the organic synthesis systems developed by Wipke, Corey, and others; the MYCIN project at Stanford in the field of chemotherapeutic advice; and Pople and Myer's DIALOG system for medical diagnosis (Pople 1975). The joint effort of computer scientists and experts in the experimental sciences has allowed the creation of excellent computer systems. These systems are presently being used to facilitate research in the experimental sciences. The ideas and technology behind the projects are now available for application to an experimental planning program for molecular genetics.

There appear to be two factors common to those task domains to which computer-assisted problem solving techniques have been applied successfully:

1) the development of systematic and logical methods of analysis of problems and application of tools by the experimental scientists themselves,
2) the possibility of solving problems in the domain using a strongly constrained and consistent body of knowledge.

In the current state of the art, heuristic problen solving programs containing a few thousand facts and heuristics about a task domain can be built and used effectively. The existence of these two factors in a domain suggests that the computer scientist can build an intellectual bridge into the domain in a period of time that might be measured in months rather than years.

Some problems of experimental molecular genetics currently share both of these factors. For example, the growth of knowledge about restriction enzymes has been important to genetics because of their many applications to manipulation and separation of DNA structures. Despite the large number of enzymes, each can be represented by a small set of descriptors. This makes the task of representing the knowledge far more limited than it might first appear.

One major issue of artificial intelligence that relates to this research is how to plan experiments involving the choice of several steps in a large space of possibilities. This includes such fundamental points as when to design an entire experimental plan first and then fill in details later (top-down), and when to design a complete step at a time (bottom-up). A second major issue will be the development and management of large knowledge bases. The MOLGEN system will need extensive files of knowledge about DNA structures, enzymes, separative techniques, and physical processes. For example some of this knowledge will involve mathematical models of kinetics; some will be detailed sequences of base pairs; some will be "fuzzy" and inexact

descriptions of a particular process or structure. The knowledge bases must be designed to allow as much of the data as possible to be handled in a uniform manner. A third major computer science problem will be the creation of a simulation program which can simulate the action of enzymes on DNA structures. Many of the current techniques developed for simulation in other domains may prove useful here. The MOLGEN system will also have to deal with cases of incomplete and inexact knowledge in the planning and the simulation stages. Molecular genetics, as a task domain, offers problems which are limited and structured enough to be feasible with available techniques, as well as those which are challenging enough to advance the art.

The overall purpose, then, of providing computer assistance in the planning process is to make it possible for an investigator to explore a wide range of possible experiments. Calculations of the effectiveness of computer assisted planning can only be approximate, but the following is offered for a rough estimate of the potential. Professor Lederberg has suggested that the knowledge actively used in the planning of experiments in molecular genetics could be encoded as approximately a thousand computer "rules". (This fits confortably in the range used by current knowledge-based intelligent systems.) It has also been observed that most experiments in molecular genetics involve on the order of seven levels of formal decision once the context of a particular goal has been set. We believe that fairly crude heuristics can reduce the likely choices to perhaps ten. Ten major alternatives at

each level of planning yields a total "search space" of about ten million alternatives. A search space of this size fits within the limits of current AI programs but still demands a definition of more powerful search-reduction heuristics for high performance. The purpose of these estimates is not so much to give a true measure of program performance, as to illustrate the notion that the number of possible experiments is indeed much greater than could actually be performed. Prof. Lederberg has suggested that a number of important and perhaps novel experiments are not being considered because of the difficulties in keeping current in a rapidly expanding discipline. Meeting this need is precisely the goal of the MOLGEN project.

2 Background and Related Work

2.1 Relationship to Other Research in Artificial Intelligence

There are two primary reasons for proposing the MOLGEN project:

(1) to develop a high performance computer program augmenting human intelligence in important research areas in molecular genetics,

(2) to provide a context for testing and extending ideas in the design of intelligent systems.

This section will discuss the existing scientific and technological base and some proposed extensions and show the relationships between MOLGEN and other AI programs.

2.1.1 Planning and Heuristic Search

The problem of planning an experiment fits into the paradigm of heuristic search, a well established framework for computer based problem solving. Problems in this framework can be represented as a tree of subproblems, with solutions existing at unknown depths along unknown paths. Judgmental rules and procedures, called heuristics, are applied to direct the search. When we speak of the expertise or good judgment of an expert, we are often referring to the heuristics he has developed to search effectively.

There are typically a number of alternative procedures to choose from at each step in the course of an experiment in molecular

genetics. These alternate steps can be viewed as a family of possible experiments. This family may then be represented as a tree structure so that the planning of a particular experiment is equivalent to choosing a path in the tree. Each node represents a certain "state" in the experiment where the physical entities being used by the experimenter are in a particular condition and certain quantities are known. For example, the physical entities may be a mixture of oligonucleotides whose terminal base sequences are known. The activity in the planning process centers around selecting an operation to carry the experiment to the next state. These operations are chosen from the program's knowledge base and are naturally expressed as transformations from one state to another. For example, in the the context of molecular genetics, a chromatography step could be represented as a transformation which "transforms" a mixture (initial state) into its separated component parts (final state). Each particular type of chromatography would be more specifically described by the parameters of the physical entities which are active in the separation. For example, gel electrophoresis would be described in terms of its ability to separate oligonucleotides according to length. Part of the knowledge embedded in the transformation would be a rule for estimating the effectiveness of gel electrophoresis on any particular mixture of oligonucleotides for the various gels which are available. Finally, a completed plan is a sequence of transformations which when applied successively to the initial state of the experiment transform it to the desired goal or final state.

This model of problem solving based on the selection of a path according to heuristic knowledge is often called the heuristic search paradigm. It has been used as a model for a variety of problems since such early projects in AI as the "Logic Theory Machine" (1956) by Newell and Simon. One dimension of progress in AI has been the development of techniques for the creation of plans within the heuristic search paradigm.

2.1.2 Strategies for Planning

Several algorithms have been developed to assist in choosing a "minimum cost" path in a heuristic search tree (Nilsson 1971), where an estimating function is available which can measure how "close" any intermediate point in the experiment is to a final state. In order to guarantee that the algorithms will find a minimal path, the estimating function must never overestimate the distance to the goal. For complex problems, a practical difficulty continues to appear in many contexts. Simply stated, it is sometimes best to retreat from a goal in order to get closer to it. In mathematical theorems, this arises in those cases where it is easier to solve a more general theoren than a specific one. In organic synthesis, it is sometimes better to build up a rather complicated structure which seems "farther" from a target compound than some current step in the synthesis, but from which a direct and often elegant reaction will transform the complicated structure almost directly to the desired product.

In complex scientific domains, the distance estimating

functions are almost impossible to find. Even if such a metric can be found, it is unlikely we can guarantee that it never "overestimates" the distance to a goal state. Nevertheless, in special cases, quantitative distance measures can sometimes be found when discrimination among the products of the alternative transformations is statistically testable. Otherwise, planning programs must rely on domain specific or even goal specific knowledge to guide the selection of the next step.

Some more general techniques for planning have been developed for robot problem solvers. For example, the STRIPS system (Fikes 1972), in the course of building long plans, attempts to assemble short sequences of operators which may later prove useful. The idea is that it is more efficient to assess the effect of the smaller plans and reuse them than to generate them again from scratch later. The algorithms include methods for disentangling the local effects of that piece of a plan from its immediate context and predicting its effect in a new context.

Another planning idea of general application has been developed in Sacerdoti's work on hierarchical planning (Sacerdoti 1973). This can be seen as a formalization of the common sense notion of planning major steps first, and attending to details later. In our terms, it means recognizing that some preconditions for use of a transformation are more important than others. This ordering allows the most important criteria to be considered first. In contrast to earlier methods for generating plans, which proceeded a step at a time toward a

goal, hierarchical planning involves sketching out a plan in several passes of refinement of detail. This approach is applicable where it is possible to indicate an ordering for the criteria used in choosing the next step of a plan.

Our examination of several experiments in molecular genetics suggests that this idea has a great deal of relevance. For example, in dealing with enzyme data, it is far more important for most purposes that an enzyme be applicable, i.e. that it have the desired effect on a substrate, than that its most active pH be in a certain range. In terms of planning, this suggests that "applicability" is a more important piece of planning information than is "pH range". Hence, the adjustment of pH in an experiment is a detail which can be considered during a later pass in the planning process. This ordering of importance need not be assumed constant for all experiments, but may be considered as part of the planning heuristics. The application and development of these planning issues will be part of the interesting computer science questions to be investigated in the MOLGEN project.

The problem of experiment planning can be compared to the automatic generation of computer programs. The planner's task is to find a sequence of instructions that will transform the input into the desired output. However, the tools that the experiment planning program will work with are conceptually simpler and the range of possible "programs" or experiments is more restricted than those utilized in the general problem. Although there are many unit

processes, the planning will be facilitated by needing little or no recursion or looping, limited "data types", few steps, and in addition will have a detailed knowledge of the subject matter. Thus the heuristics necessary to plan experiments are more manageable.

2.1.3 Acquisition of Knowledge

One trend in the development of intelligent systems has been an increase in the size of the knowledge base used by the programs. Domain specific knowledge is used to guide and hence speed their search. The process by which a system acquires this knowledge is one of the central design issues for an intelligent system. The DENDRAL programs, for instance, which are used in the analysis of mass spectra of organic compounds (Buchanan 1969), acquired their chemical and mass spectral knowledge by a painstaking process in which an AI scientist worked with a chemist, eliciting from the chemist the theories, facts, rules, and heuristics applicable to reasoning from mass spectra. This same approach has been used successfully in protein structure determination using x-ray crystallography and in diagnosis of glaucoma eye disease (Kulikowski 1973).

More recent versions of the DENDRAL program have special built-in editor programs for acquiring knowledge and data from the chemical domain. For example, a chemical structure editor is used in entering fragments of chemical structures, and other editors specialize in acquiring chemical structure constraints and fragmentation rules. Another approach, typified by the SECS program for organic chemical

synthesis (Wipke 1975), involved use of a special language (ALCHEM) for describing organic reactions. For each of the areas of organic synthesis expertise, a chemist specialist was responsible for encoding in ALCHEM the particulars about the useful reactions in his specialty. The idea was that a user of the SECS program could then draw upon the combined knowledge from all the experts who had participated. This process not only contributed to the growth of the knowledge base, but also resulted in changes and additions to the ALCHEM language which made it both more powerful and more convenient.

The MYCIN program, used for diagnosis and treatment planning of infectious disease using antibiotics (Shortliffe 1973, 1975) uses a more elaborate method for the acquisition of knowledge. In this system, knowledge is embedded in a collection of some 300 decision rules. New rules can be added via a rule acquisition program. This program accepts rules in a stylized form of English, converts them to an internal form, and then prints them out again in English to indicate its understanding of the rule. The system also has a model of its knowledge, in the form of abstracted descriptions of classes of rules. It uses these both to help interpret the rule the physician has typed in, and as a check on its content. If, for example, the new rule differs from most others in that class, the physician is warned, and can then alter the rule if he wishes.

One of the more interesting components of the MYCIN program is its explanation system. This concept was introduced earlier in

Winograd's SHRDLU program (Winograd 1972), and refers to the ability of a program to explain its activities to its user. For example, MYCIN may ask the physician a particular question about the results of a culture. If the physician wants to know why the question was asked, he can ask the system. It will then examine its reasoning, and indicate the purpose of the question by displaying the relevant rule. The same rules which guide the consultation are thus being used to explain the program's behavior.

2.1.4 Applications to MOLGEN

It is our goal in the development of the MOLGEN system to draw upon and extend some of the ideas and techniques mentioned above. With regard to planning, we will be experimenting with ways to relate hierarchical notions of planning to the concepts of molecular genetics. Furthermore, we will investigate techniques to make the MOLGEN programs handle the diversity of goals in molecular genetics experiments. Note that when a physician starts to use MYCIN, the program does not have to ask him what he wants to do. The physician always wants to diagnose a disease and prescribe treatment. For MOLGEN, however, the experimental goals are as diverse as the experiments in molecular genetics.

We expect MOLGEN to draw upon the same knowledge base in its efforts to work on a variety of experimental planning problems . In spite of the diversity of experiments in molecular genetics, we have categorized three main types of experiments as follows:

1. Given a structural hypothesis, plan experiments and

procedures which would verify the hypothesis. (This is an example of structural analysis.)

2. Given starting and target DNA structures, plan experiments and procedures which will transform the initial structure to the final structure. (This is an example of structural synthesis.)
3. Given an initial DNA structure, hypothesized final structure, and a set of steps in an experiment to be carried out, verify the probable correctness of the final structure or show any other possible final structures. (This is an example of a 'proof' checker.)

An initial goal towards meeting the first two objectives is the development of a planning program which will allow a user to "sketch" out an experiment, and allow MOLGEN to fill in the details . An example showing how this might proceed is given in section 3 of this proposal.

A second area for developing on the themes mentioned above is in the area of knowledge acquisition and representation. The main extension to previous ideas has to do with the handling of a diversity of data types. MOLGEN must deal with a broad range of types of knowledge -- including DNA structures, enzymes, chromatographic techniques and various physical processes. For each of these types of data, MOLGEN is expected to assist the user in entering and modifying the data and knowledge base. This should be done in a form convenient to the user's view of each particular type and yet at the same time in

a way which keeps the program's diverse collection of knowledge and data types from becoming too complex to be manageable. Similarly, the explanation system must be implemented in a way that makes the reasoning of the program accessible to the user, through all levels of the knowledge base. A successful treatment for a broad spectrum of diverse knowledge will be a major design goal for the MOLGEN project.

2.2 Related Work in Computers Applied to Genetic Biochemistry

We propose to provide in this section a summary limited to work directly related to the simulation and planning aspects of the MOLGEN project.

Some of the most advanced prior work in simulation has been done in the area of enzyme kinetics. The basic approach has been to set up the differential equations describing the chemical reactions , and then solve those equations on either a digital (Chance and Shephard 1969, Bates 1973, and Garfinkel 1968) or analog (Chance 1967) computer system. It is interesting that attempts have been made to facilitate use of the systems by working geneticists, either by means of interactive display mechanisms (Bates 1973), or an English-like language for describing the chemical systems (Garfinkel 1968). We intend to expand upon these user-oriented techniques during our system development.

We also know of a system (Green 1970) which uses matrix

representations of the steady-state equations describing enzymes to permit rapid calculation of the distribution of all enzymatic forms at any given moment. We feel that the accurate modelling of this steady-state behavior will prove vital to our enzymatic simulation system, although we still have to investigate the general applicability of the matrix representation technique for multi-enzyme systems.

Crothers (1968, 1971) has developed several computer programs to simulate various aspects of the process of de/renaturation. The detailed mathematical models available on conformational changes (Bloomfield 1974) will facilitate the computer simulation of these processes.

The problem of representing the often "fuzzy" knowledge about DNA structures has been considered in a model for the heterogeneity of base composition (Elton 1974). The point is that the exact composition is rarely known for more than a few hundred bases. Elton's model classifies the DNA structures into "segments" with different underlying base compositions, each segment categorized by an estimate of length and G-C content. Our representation model for DNA structures will allow such classification of portions of DNA, as well as many other factors like the distribution of important features (nicks, gaps, etc.) along parts of the DNA strands.

Finally, we find of direct relevance to our design of experimental plans the use by Khorana's group (Powers 1975) of computer programs to help plan the detailed steps involved in gene syntheses.

The genes are synthesized in small segments and then joined by some chemical or enzymatic means. The program, using knowledge of the chemical environment, determines optimal segment lengths and order of joining. This will be one of many experimental paradigms we hope the MOLGEN system will encompass.

2.3 Supporting Research at the Stanford Heuristic Programming Project

The Heuristic Programming Project at Stanford, of which the MOLGEN research will be a part, is one of the few AI research groups that emphasizes development of intelligent systems for use in the sciences. This section discusses some of these intelligent systems which are being developed and actively used. These interdisciplinary projects serve to illustrate the range of experience that the MOLGEN research can draw on.

2.3.1 The DENDRAL Project

The Heuristic DENDRAL project was started by Professors Lederberg and Feigenbaum in 1965, and has diversified and matured considerably since that time. The DENDRAL program (Buchanan 1975 and Appendix A) is an application of artificial intelligence to biomedical molecular structure determination problems. The program is designed to explain empirical data, in particular, to interpret experimental data from a mass spectrometer, an analytical instrument used in organic chemistry. It uses heuristic search techniques to explore plausible

candidate explanations of the empirical data in much the same way that other artificial intelligence programs explore spaces of hypotheses.

A major contribution has been demonstrating that a large amount of scientific knowledge can be managed and used by a complex problem-solving program. Another contribution has been showing that AI techniques can be successfully applied to scientific problems.

The DENDRAL molecular structure building program, CONGEN, is currently used by chemists as a tool for building molecular structures from structural pieces which the chemist has inferred from whatever data are available. The DENDRAL PLANNER (Smith 1973) uses a large amount of knowledge specific to mass spectrometry to infer structural constraints directly from empirical data. The DENDRAL PREDICTOR is a simulation model of the mass spectrometer that allows the program to make testable predictions of candidate hypotheses. META-DENDRAL is a program for discovering new inference rules in the field of mass spectrometry.

From the start, major emphasis has been placed on avoiding the "outsider-insider" distinction by having trained chemists work on the programs and make them useful to working scientists. This emphasis is in contrast to other research on "toy" problems conducted without reference to the needs of working scientists. Since May 1974, the project has moved to the interactive computing environment of the NIH-funded SUMEX-AIM facility (Carhart 1975). Because of this, many scientists outside this university have been able to use the DENDRAL

computer programs to further their own research. These programs are currently receiving heavy use from local users and outside users who are investigating mass spectrometry problems for a variety of different compound classes.

2.3.2 The MYCIN Project

The MYCIN project (Davis 1975) at Stanford is another interdisciplinary research group including members of the Heuristic Programming Project, whose goal is to apply artificial intelligence reasoning programs to a scientific problem. The objective of this MYCIN research is to develop a computer-based system capable of using the judgmental knowledge of experts to assist physicians in selecting appropriate antimicrobial therapy for patients with infectious diseases. The work concentrates initially on the use of antimicrobial agents in the treatment of bacteremias.

The current MYCIN program utilizes data available from the microbiology and clinical chemistry laboratories, plus the physician's response to computer-generated questions, to provide physicians with consultative advice on antimicrobial therapy. Therapy selection takes into account both the patient's infections and the range of possible identities of the organisms causing these infections. One contribution of the MYCIN system has been developing general techniques for explaining its reasoning steps to permit the program to explain all of its actions and reasoning, including, for example, the deduction of the identities of pathologic agents. Another important part of the MYCIN

system is a rule-acquisition program for computer acquisition of judgmental knowledge about those concepts which the program uses in making deductions. This permits experts in the field of infectious disease therapy to teach the system those therapeutic decision rules which they use in their clinical practice. Making rules, exploring MYCIN's reasoning, or adding new knowledge to the system is done in a stylized subset of English.

2.3.3 Protein Structure Modeling Project

The Protein crystallography project involves scientists at the University of California at San Diego as well as members of the Heuristic Programming Project. The general objective of the project is to apply problem solving techniques in artificial intelligence to the "phase problem" of x-ray crystallography, in order to determine the three-dimensional structures of proteins. Specifically, the task is to develop a computational system which can infer the tertiary structure of a protein molecule in the absence of phase information normally obtained from multiple isomorphous replacement procedures.

In the context of artificial intelligence, project objectives center around knowledge acquisition and program organization. The knowledge acquisition task involves formalizing the knowledge and heuristics used by expert protein crystallographers to infer the tertiary structure of proteins from x-ray crystallographic data, and to formalize this expertise in appropriate data structures and heuristic procedures. The program organization plans are variations on the

HEARSAY (Reddy 1973) model of cooperating "expert" modules in a program.

3 Detailed Plans for Work During the Proposal Period

3.1 Introduction

This section presents our overall view for the molecular genetics system as well as a detailed list of specific goals for the first year covered by this proposal. The system will be composed of three major, interacting parts: an experiment planning system, an enzymatic action simulation program, and a collection of knowledge bases containing the rules and heuristics of molecular genetics.

The experiment planning program will collect information about a problem from the user, select an appropriate methodology (information retrieval, simulation, hierarchical planning, or some combination) and then work interactively with him to solve the problem. Some questions can be answered by information retrieval from a knowledge base, such as a question about the enzymes that function under certain pH or salt concentrations. On the other hand, if a geneticist asked for a prediction of the ratio of linear to circular DNA in a test tube after 10 minutes application of ligase, then a straightforward simulation would be in order. In verifying that a given set of steps in an experiment produces the final result, a combination of simulation and retrieval would be used by the planning program. In more difficult, higher order problems, such as structural analysis or synthesis, hierarchical planning is needed, perhaps involving both retrieval and simulation as well.

A simulation program will provide detailed modelling of the action of enzymes on DNA structures. Our goal is a program which will be able to produce accurate enough results to answer questions like the one posed above, i.e. what will happen after application of ligase for 10 minutes, 2 hours, etc.

The knowledge bases will be composed of collections of the rules and heuristics used by geneticists, as well as facts about enzymes, experimental methods, and physical processes like renaturation. They must allow access in retrieval, simulation, and planning modes, so provision for the representation of many types of knowledge must be supplied.

Along with the major system components discussed above, certain themes will remain dominant during all phases of system design. Primary consideration will be given to making the system an easy and natural tool for the geneticist to use. DNA structure entry, editing, and display will be by way of an interactive program with numerous user-oriented features. Explanation facilities (in the model of the MYCIN system) will be provided whenever possible, and all knowledge bases will be made easily extendable and modifiable. Building the trust of the user is considered necessary to the continued development of the system. The best way to provide for this will be to insure that it is used by geneticists, and that a strong interaction between computer scientists and genetics experts continues.

The remainder of this part of our proposal will discuss in

detail our plan for system development in the first year. Briefly summarized, our four major goals are:

1. Begin the construction of a rudimentary experiment planning program and an associated planning knowledge base.
2. Build a simulation program which will apply basic enzymatic actions to DNA structures and summarize the resulting structures.
3. Design the knowledge base containing rules and heuristics used by the geneticists, enzyme knowledge, and knowledge of experimental methods and physical processes.
4. Complete the construction of an interactive DNA structure entry and editting system.