Research in Progress

MIT Artificial Intelligence Laboratory

The MIT AI Laboratory has a long tradition of research in most aspects of Artificial Intelligence. Currently, the major foci include computer vision, manipulation, learning, English-language understanding, VLSI design, expert engineering problem solving, common-sense reasoning, computer architecture, distributed problem solving, models of human memory, programmer apprentices, and human education.

Understanding Visual Images

Professor Berthold K. P. Horn and his students have studied intensively the image irradiance equation and its applications. The reflectance and albedo map representations have been introduced to make surface orientation, illumination geometry, and surface reflectivity explicit. Recent work has centered on modelling the effects of the atmosphere which distort intensity values and make classification of terrain and related computations using the albedo map inaccurate. Horn and Brian Schunk have recently proposed a mechanism for determining optical flow which is remarkably resilient. William Silver has completed an implementation and elaboration of the idea of photometric stereo introduced earlier in the group by Dr. Robert Woodham.

Natural Vision

Professor David Marr and his colleagues Drs. Eric Grimson, Keith Nishihara, Shimon Ullman, and Kent Stevens, together with Ellen Hildreth, have made considerable progress on a computational theory of human vision. The approach gives key emphasis to a succession of explicit descriptions at varying levels of visual processing, including the zero-crossing map, the primal and 2½D sketches, and the so-called Spasar 3D representation. Recent work has centered on directional selectivity, evidence for a fifth, smaller channel for early processing, the Marr-Hildreth theory of edge detection, a model of the retina, a computational theory of stereopsis and its implementation, surface reconstruction by filling in from a sample of values in the 2½D sketch, texture gradients, surface contours, and a study of the sufficiency of the zero-crossing map by attempting to extend Logan's theorem to two dimensions. Recently, Dr. Mike Brady has joined the Laboratory and has initiated a study of the psychology of reading.

Manipulation

Dr. John Hollerbach has developed a theory of human handwriting and shown that it accounts for much of the available empirical data. He has also shown that the Lagrangian formulation of the equations of motion of a 6 degree of freedom manipulator can be formulated as recurrence relations leading to an exact solution in linear time. Other work has concerned the control theory necessary to make manipulators move quickly, an analysis of compliant motion, and a theory of spatial planning by Lozano-Perez. We have constructed a tendon finger and touch sensor, a cartesian arm, and have a tendon shoulder under construction. Berthold Horn and his students have further developed the configuration state model for manipulator control.

Over the past two years we have started a program of research into the development of VLSI systems. Professor Gerald Sussman, John Holloway, Tom Knight, Howie Shrobe, and others have developed a number of systems, including a single ship SCHEME processor. We have TTL implementations of a number of our vision algorithms, and we are in the process of switching some of them to VLSI.

Expert Problem Solving and VLSI Design

Traditional automated synthesis techniques for circuit design are restricted to small classes of circuit functions for which mathematical methods exist. Professor Gerald J. Sussman and his group have developed computer-aided design tools that can be of much broader assistance. The work has developed the idea of analysis by propagation of constraints. Guy L. Steele developed a language to support such programming, Johan de Kleer studied causal and teleological reasoning in the recognition of circuit function from schematics, and Howie Shrobe has worked on constraint satisfaction and the development of an interactive knowledge-based system for substantially supporting VLSI design. Jon Doyle has studied belief revision via truth maintenance and non-monotonic logics, as well as self-conscious adaptive deliberate reasoning programs. Richard Waters, Charles Rich, and Howie Shrobe have developed the idea of a programmer's apprentice.

Learning and Language

Professor Patrick H. Winston has studied systems that reason by analogy. During the past year, he has worked out a theory and implemented a system. Professor William Martin has continued his work on computer understanding of natural language. Robert Berwick has developed a theory and system that can learn a substantial part of the syntax of English, based on a theory of syntactic recognition developed in the laboratory by Mitchell Marcus. Other work on Natural Language understanding has centered on determination of focus during discourse and response generation.

Personal Machines and Computing Concepts

Intelligent information processing places enormous demands on computing resources, and the computational facilities required to effectively support research in Artificial Intelligence are equally large. About five years ago it became clear to Richard Greenblatt and his associates that our computing needs would be best served by powerful personal computers which execute LISP extremely efficiently and which maintain the level of system support. Currently we have eleven LISP machines and plan a further nine over the next couple of years. All our machines are interconnected through an eight-megabit packet switched network called the CHAOSNET.

On the theoretical side, Professor Randall Davis has developed a theory of distributed problem solving which he calls the Contract Net, in which goals are broadcast to sub-contractors, who then bid for them based on their current work load, the description of the contract, and their abilities. Professor Carl Hewitt and his colleagues have developed an architecture for large scale distributed systems which they call the Apiary, and which is based on the migration of processes to free processors to optimize throughput and achieve load balancing. They have introduced a descriptive formalism called OMEGA, which contributes to many of the issues of current concern in knowlege representation, and they have applied it to describe the various structured entities such as messages in a study of cooperating sequential programming.

Basic Theory

Professor Marvin Minsky has worked on a theory of human thinking, which likens the mind to a society of agents and attempts to combine a number of insights from psychoanalytic, developmental, and cognitive theories of mind. Further work by Richard Greenblatt and Dr. Lucia Vaina develops the idea of thread memory.

Information Sciences Institute
University of Southern California

ISI is an off-campus research center in the University of Southern California's School of Engineering. The Institute engages in a broad set of research and application oriented projects in the computer sciences. These projects range from advanced research efforts aimed at producing new concepts, through development of prototype systems, to operation of a major Arpanet computer facility. Except for the Computer Services group, these projects range from one up to ten people. ISI is composed of about 130 full time people plus a number of part-time graduate research assistants and, except for a small operation in San Diego, is entirely housed in facilities at Marina del Rey, California. ISI currently operates about a half-dozen Tenex and TOPS-20 systems as network facilities. Part of these facilities support in-house research; the remainder supports research at other ARPA-supported sites.

AI-Related Projects

Specification Formulation, Testing, and Implementation

Robert Balzer, Lee Erman, Martin Feather, Neil Goldman, Philip London, and David Wile

This is an area of three interrelated projects aimed at facilitating the creation, testing, and optimized implementation

of program specifications. The first effort, called Specification Acquistion From Experts (SAFE), is directed at helping people create unambiguous, consistent, and complete formal program specifications through informal description. While end users are quite capable of providing informal process-oriented descriptions of the task being automated, formalisms of any kind provide major impediments. The informal descriptions are characterized by partial, rather than complete, constructs. The system uses a knowledge base of program well-formedness rules to disambiguate and complete the informal natural language descriptions. An early version of the system has successfully converted several small informal specifications into formal specifications (Balzer 78). Attention is now focused on handling large specifications through incremental formalization; to support this, a generalized AI architecture for knowledge-based systems, based on the Hearsay-II blackboard model, has been designed and implemented (Balzer 80a).

The second project (Balzer 80b) seeks to provide a methodology guaranteed to produce only valid implementations of the formal specifications by involving the computer in the implementation process to maintain consistency of the evolving implementation with the original formal specification. This is accomplished by limiting the activity of programmers to the choice of optimizations to be utilized. These optimizations are embodied in a catalog of transformations, applied by the system in response to the programmer's choices, after verifying that they are validly applicable.

The third project seeks to ensure that a formal specification matches the user's intent, by enabling its behavior to be investigated systematically through symbolic execution. The focus of this project will be in explaining the simulated behavior meaningfully.

Designed for use by all of these projects is a new abstract specification language, called GIST (Goldman 79).

Cooperative Interactive Systems

David Wilczynski, Robert Lingard, and William Mark

Current interactive systems must be made more habitable for a wide variety of users, especially those having little experience with computing. We are currently building the Consul (Mark 80) system to explore methods for allowing natural interaction—natural-language requests, explanation of system facilities, user-understandable error handling—between users and a set of online tools (for text manipulation, message handling, etc.) Consul works by mapping user requests into appropriate system actions as dictated by a detailed model of system capabilities and requirements. This model is based on built-in tool-independent knowledge, but is tailored to individual tools as part of the tool building process. Our research therefore includes not only modelling and mapping methods that generalize across tools, but also a programming environment for tool builders that integrates their tools into Consul's knowledge base.

Knowledge Delivery in Multiparagraph Text

William Mann and James Moore

This project is developing new methods for autonomous text composition by machine, with the focus at the larger-than-sentence level. The project is producing a series of knowledge delivery systems. The first of these (Mann 79a), which was called KDS (Knowledge Delivery System), was developed as a testbed for methods of text composition from computer-internal knowledge representations such as semantic nets. Innovations in KDS included a new expressive paradigm that does not rely on dividing the semantic net into sentence-sized pieces; a problem-solving text organizer that delegates part of its work to other processes; explicit principles for ordering and grouping concepts for expression; rules for aggregating small expressions into larger, more natural ones; and a novel text improvement methods based on a rule-competition-oriented variant of the classic hill-climbing algorithm. The ideas in KDS were tested on two principal domains: description of the structure and use of a computer message system and description of a multi-actor, multi-contingency policy on how to respond to indications of fire in a computer center.

A second system, called PENMAN, is currently being developed to explore the problems of creating a portable knowledge delivery facility which is useful in multiple knowledge domains. PENMAN will seek to deliver knowledge (in English) from inside a system which was not designed to have a knowledge delivery component.

Computer Comprehension of Natural Language

William Mann and James Moore

Despite its great social significance, human communication remains a complex, poorly understood process. Particularly obscure is comprehension, the process by which a sequence of words effects the cognitive state of the hearer. Human-computer interaction could be made easier and more effective if computers could comprehend strings of symbols in more human-like ways. Our prior research has created a model of human communication as a kind of goal-pursuit activity (Mann 79b). The model represents four principal levels of knowledge in communication: goals being pursued, speech acts, and linguistic symbols used to convey all of these. This project is building an experimental system called DCS (Dialogue Comprehension System), which will read and follow transcripts of actual human dialogues. In particular, it will contain processes that can recognize the establishment, pursuit, and satisfaction of individual goals, including representation of state changes in the communicating individuals.

InterLisp Developement

Mel Pirtle, David Dyer, and William Rizzi

This effort is developing and will maintain a portable, large address-space InterLisp implementation. The first version is currently being done for the VAX computer.

Distributed Sensor Nets-DSN

Danny Cohen, Jeff Barnett, Iris Kameny, and Yechiam Yemini

The objective of this project is to develop a methodology for defining DSN components and their organization that is both sensor and scenario independent. The three main DSN problem areas that are currently being addressed are: how to combine different kinds of data into higher level knowledge and assign appropriate levels of confidence to the knowledge; the development of sophisticated, goal-oriented communication protocols; and the design of a closely coupled user-DSN interface to carry out DSN command, control, communication, and intelligence functions.

A software testbed has been designed and is currently being implemented to test and evaluate different DSN configurations. A major goal is to discover how to incrementally automate some of the command, control, communication, and intelligence functions of the user as soon as they can be understood and specified. This should allow the user to spend increasingly larger portions of his time on more complex tasks and decision making, and should indicate the appropriate distributed process structure and communication needs of the system.

Selected References

(1)   (Balzer 78) Balzer, R., N. Goldman, and D. Wile, "Informality in Program Specifications," IEEE Trans. Software Eng. SE-4, (2), March 1978.

(2)   (Balzer 80a) Balzer, R., L. D. Erman, P. London, and C. Williams, "Hearsay-III: A Domain-Independent Framework for Expert Systems," in 1st National Artificial Intelligence Conf., Palo Alto, CA, August 1980. (To appear)

(3)   (Balzer 80b) Balzer, R., N., "Transformational Implementation: An Example," IEEE Trans. Software Eng., November 1980. (Also appeared as Technical Report, USC/Information Sciences Institute, RR-79-79).

(4)   (Goldman 79) Goldman, N. M. and D. S. Wile, "A Relational Data Base Foundation for Process Specifications, " in Chen, P. P. (ed.), Int. Conf. on Entity-Relationship Approach to Systems Analysis and Design, pp. 413-432, December 1979.

(5)   (Mann 79a) Mann, W. C., and J. A. Moore, Computer as Author—Results and Prospects, USC/Information Sciences Institute, Technical Report RR-79-82, 1979.

(6)   (Mark 80) Mark, W., "Rule-Based Inference in Large Knowledge Bases," in 1st National Artificial Intelligence Conf., Palo Alto, CA, August 1980. (To appear)

Other Projects

Program Verification
Susan Gerhar, Ralph London, Ray Bates, Rod Erickson, David Thompson, and David Wile
Study of methods to prove logical correctness of programs.

Formal Specification of Systems
Steve Crocker, Richard Gillmann, Vittal Kini, Leo Marcus, David Martin, Wiliam Overman, and Dono van-Mierop
Three interrelated projects aimed at understanding how to specify and verify computer systems and languages.

VLSI Implementation System
Danny Cohen, Vance Tyree, Ron Ayres, Leroy Richardson, and George Lewicki.
The process of decoupling the design process from the idiosyncracies of the various fabrication facilities, the information management of Multi-Project Chip runs, and high-level design methodology.

Wide Band Communication
Steve Casner, William Brackenridge, Danny Cohen, Randy Cole, and Eric Mader
Exploration of satellite-based wide-band packet computer communication, both the technology needed to support it and of opportunities for its use.

Command and Control Graphics
Richard Bisbey, Dennis Hollingworth, and Ben Britt
Development of a device-independent graphics system and graphics-oriented command and control applications programs.

Internet Concepts Research
Jon Postel, Danny Cohen, Carl Sunshine, and Greg Finn
Exploration of aspects of protocols for the interconnection of computer communication networks, specifically the design and prototype implementation of an internetwork multimedia computer message system and the design and analysis of internetwork host and gateway protocols.

QPRIM
Lou Gallenson and Joel Goldberg
Building a facility for producing computer emulators, with emphasis on debugging target programs.

Application Downloading
Robert Balzer, Al Cooperband, Martin Feather, Philip London, and Dave Wile
Work to improve utility of processing terminals by permitting the division of responsibility between host and terminal to be made after application is designed, with the partitioning and synchronization to be achieved via designer-

directed machine transformations of a checked-out and debugged single-machine implementation.

Packet Radio Terminal System Evaluation
    Tom Ellis and Steve Saunders
    Work intended to result in a demonstration-level portable terminal to test and evaluate various solutions to the issues raised by extreme portability in the packet-radio environment.

The Stanford Heuristic Programming Project: Goals and Activities

by the Staff of the Heuristic Programming Project

The Heuristic Programming Project (HPP) of the Stanford University Computer Science Department is a laboratory of about fifty people—faculty, staff, and graduate students—whose main goals are these:

> ...to model, and thereby to gain a deep understanding of, the nature of scientific reasoning processes in various types of scientific problems, and various areas of science and medicine;
> ...as part of the methodology, and as a coordinate activity, to construct "Expert Systems"—programs that achieve high levels of performance on tasks that normally require significant human expertise for their solution; the HPP therefore has a natural applications orientation.

The HPP was started by Professor Edward A. Feigenbaum and Professor Joshua Lederberg (now President, Rockefeller University) as the DENDRAL project in 1965. Professor Bruce Buchanan joined shortly thereafter, and is Co-Principal Investigator of the HPP.

For its computing facilities, the HPP uses the Stanford-based SUMEX-AIM National Resource for Applications of AI to Medicine and Biology (a pair of DEC KI-10s and a DEC 2020); and the SU-SCORE machine (a DEC 2060).

HPP research includes basic AI research, applications-oriented research, and development of community tools useful for building expert system. The programs developed can be used as problem solving assistants or tutors, but also serve as excellent vehicles for research on representation and control of diverse forms of knowledge.

The main issues of building expert systems are coincident with general issues in AI. We build working programs that demonstrate the feasibility of our ideas within well defined limits. By investigating the nature of expert reasoning within computer programs, such processes are "demystified." Ultimately, the construction of such programs becomes itself a well-understood technical craft.

The foundation of each of the projects described in this article is expert knowledge: its acquisition from practitioners, its accommodation into the existing knowledge bases, its explanation, and its use to solve problems. Continued work on these topics provides new techniques and mechanisms for the design and construction of knowledge bases, its explanation, and its use to solve problems. Continued work on these topics provides new techniques and mechanisms for the design and construction of knowledge-based programs; experience gained from the actual construction of these systems then feeds back both (a) evaluative information of the ideas' utility and (b) reports of quite specific problems and the ways in which they have been overcome, which may suggest some more general method to be tried in other programs.

One of our long-range goals is to isolate AI techniques that are general, to determine the conditions for their use and to build up a knowledge base about AI techniques themselves. Under support from ARPA, NIH/NLM, ONR, NSF, and industry, the HPP conducts research on five key scientific problems of AI, as well as a host of subsidiary issues:

1. *Knowledge Representation*—How shall the knowledge necessary for expert-level performance be represented for computer use? How can one achieve flexibility in adding and changing knowledge in the continuous development of a knowledge base? Are there uniform representations for the diverse kinds of specialized knowledge needed in all domains?
2. *Knowledge Utilization*—What designs are available for the inference procedure to be used by an expert system? How can the control structure be simple enough to be understandable and yet sophisticated enough for high performance? How can strategy knowledge be used effectively?
3. *Knowledge Acquisition*—How can the model of expertise in a field of work be systematically acquired for computer use? If it is true that the power of an expert system is primarily a funciton of the quality and completeness of the knowledge base, then this is the critical "bottleneck" problem of expert systems research.
4. *Explanation*—How can the knowledge base and the line of reasoning used in solving a particular problem be explained to users? What constitutes an acceptable explanation for each class of users?
5. *Tool Construction*—What kinds of software packages can be constructed that will facilitate the implementation of expert systems, not only by the research community but also by various user communities?

Artificial Intelligence is largely an empirical science. We explore questions such as these by designing and building programs that incorporate plausible answers. Then we try to determine the strengths and weaknesses of the answers by experimenting with perturbations of the systems and extrapolations of them into new problem areas. The test of success in this endeavor is whether the next generation of system builders finds the questions relevant and the answers applicable to reduce the effort of building complex reasoning programs.

Thus, the work of the HPP can be described in three ways: by the Expert Systems it writes, by the software tools it creates, or by the basic research issues that motivate the

research. For this article we have chosen to describe the programs and tools, and weave the basic issues through the discussion where possible.

Domains in Science

Dendral

The DENDRAL Project is the most mature of the HPP projects, and its programs are the most widely used.

Originally the Heuristic DENDRAL program was designed to help organic chemists determine the molecular structure of unknown compounds, given a list of the atoms in the molecule, and data about how the molecule breaks up when it is bombarded with energy in a mass spectrometer. This can be a difficult task even for expert chemists, one reason why it was chosen to demonstrate a theoretical point which recurs in all HPP work, namely, that in order to solve difficult problems with the facility of an expert, a program must have the knowledge of an expert. In the case of DENDRAL, this is knowledge about how molecules, or substructures of molecules, behave in a mass spectrometer. The knowledge is represented by modular situation-action rules. The situation describes a particular configuration of atoms, and the action describes how the configuration is fragmented in the spectrometer.

Dendral became an ideal testbed for exploring the power of multiple sources of knowledge when it was extended to include data from 13C NMR spectrometry. In general, DENDRAL was able to dramatically restrict the number of candidate structures it generated, given both the mass spectrum and the 13C NMR spectrum of the unknown molecule.

Over the years, DENDRAL has been generalized to handle all ringed structures, more recently stereoisomers, and a very wide variety of constraints, under user control.

DENDRAL is used daily by a large community of researchers, at universities and in industry, in the United States and abroad.

Meta-DENDRAL

The Meta-DENDRAL program automatically discovers rules which describe the fragmentation behavior of molecules in a mass spectrometer. Fragmentation rules were provided for DENDRAL by consulting experts, but in Meta-DENDRAL the emphasis is on automatic induction of the rules. Two theoretical issues motivated the Meta-DENDRAL effort. First, the induction of rules from data is essentially a task of theory formation, which is such a fundamental process in science that its automation, in any domain, has great intrinsic interest. Second, it was obvious after DENDRAL that extraction of domain knowledge from experts was, and would continue to be, a bottleneck in the construction of applications-oriented programs. Automatic induction of knowledge is an alternative knowledge-acquisition strategy.

Meta-DENDRAL infers fragmentation rules from three sources of information. It has a simple theory of general fragmentation processes; it knows, for example, that double and triple bonds do not break. It is given the molecular structures of a "family" of molecules, and also the mass spectrum for each molecule in the family. Meta-DENDRAL proposes fragmentation rules to account for the data in each mass spectrum, and then it generalizes some of these rules, specializes others, and disconfirms some entirely in light of the rules proposed for other molecules in the family.

The rules that Meta-DENDRAL has discovered have been judged as excellent by experts in organic chemistry. They have been published in a major chemical journal. More recently, Meta-DENDRAL has been extended to induce rules of 13C NMR spectrometry. Several results of this effort have also been published.

Crysalis

An area of practical and theoretical interest to workers in AI and protein crystallography is the interpretation of three-dimensional electron density maps. These maps provide graphic information about the electron density distribution in a protein molecule, but they do not reveal the positions of the individual atoms in the molecule. The goal is to develop a complete model of the molecule. However, the plan-generate-test paradigm on DENDRAL is not adequate because the complexity of protein molecules prohibits the generation of candidate structures. In CRYSALIS an alternative paradigm has been developed that involves the integration of a number of sources of knowledge about protein chemistry. Each knowledge source has access to a "blackboard," on which it writes its hypotheses about the molecule. Individual knowledge sources make hypotheses at different levels of abstraction. For example, one proposes the identities of the atoms in a part of the molecule, another proposes sidechains of atoms which branch off of the skeleton of the molecule. Problem solving in this environment is called "opportunistic" because each knowledge source uses information as it becomes available on the blackboard to support or disconfirm its own hypotheses. It follows the strategy of human protein crystallographers who build models incrementally, starting from the most "obvious" regions of the electron density map and working out to unknown regions.

MOLGEN

MOLGEN is a collaborative project with scientists at the Stanford Medical School. MOLGEN's task is to provide intelligent advice to a molecular geneticist on the planning of experiments involving the manipulation of DNA. The geneticist has various kinds of laboratory techniques available for changing DNA material (cuts, joins, insertions, deletions,

and so on); techniques for determining the biological consequences of the changes; various instruments for measuring effects; various chemical methods for inducing, facilitating, or inhibiting change; and many other tools. MOLGEN offers planning assistance in organizing and sequencing such tools to accomplish an experimental goal.

There are actually two MOLGEN systems that design experiments in rather different ways. One is based on the observation that human scientists rarely plan experiments from scratch, but instead start with an abstract or "skeletal" plan which contains the entire experimental design in outline form. The major design task is to instantiate the details of the plan by finding tools that will work best in the given problem environment. This system depends heavily upon large amounts of domain specific knowledge, especially upon good heuristics for choosing among alternatives for plan-step instantiations.

The other approach used in MOLGEN to design experiments involves the interactions between steps in a plan as it is being formulated. A method called "constraint posting" is used to make the interactions between steps explicit. Constraints are dynamically formulated and propagated during hierarchical planning and are used to coordinate the design of sub-parts of an overall plan.

In MOLGEN the problem of integration of many diverse sources of knowledge is central since the essence of the experiment planning process is the successful merging of biological, genetic, chemical, topological, and instrument knowledge. The problem of representing processes is also brought into focus since the expert's knowledge of experimental strategies must also be represented and put to use.

Medical Domains and Consultation Systems

VM (Ventilator Management)

A ventilator is a piece of medical equipment that assists a patient with his breathing. The task of the VM program is to provide real-time advice to clinicians about patients undergoing mechanical ventilation. For VM the ultimate therapeutic goal is to remove the patient from the ventilator, and VM gives advice on ventilator adjustments that expedite this goal. VM use is being explored at the intensive care unit of the Pacific Medical Center, which has an on-line computer-based patient monitoring system that automatically obtains measurements of approximately thirty physiological parameters. VM is intended as an extension of this system that will (1) provide a summary of the patient status easily understood by the clinician, (2) recognize untoward events in the patient/machine system and provide suggestions for corrective action, (3) give advice on adjustment of the mechanical ventilator based on an assessment of the patient status and therapeutic goals, (4) detect possible measurement errors, and, (5) maintain a set of patient-specific expectations and goals for future evaluation.

One interesting aspect of the VM program is that it works with multiple streams of data that are sent by the monitoring system over time. Interpreting these data requires a time-based perspective of the patient's status. For example, a recommendation that a patient is ready for a change in therapy is made after VM analyzes both current and recent past physiological statuses. Another important task for the program is to distinguish between real data and noise. The patient monitoring situation is inherently noisy, because a sensor that monitors a particular physiological parameter picks up signals from other sources, either in the patient's body or in the environment. VM "ignores" gross fluctuations in its data (e.g., those arising from turning the patient), and is able to use the multiple sources of data to eliminate some other noise.

The VM program is currently in a research and development phase.

RX

The goal of the RX project is to reliably extract knowledge about the evolution and treatment of chronic diseases from patient records that are stored in a computer database. The RX program is intended to facilitate semi-automatic knowledge acquisition, by helping the user formulate and test hypotheses about an extant database of records. The database is a subset of the American Rheumatism Association Medical Information System (ARAMIS), which contains the records of about 6000 patients. Such a database contains the answers to many yet-unasked research questions; the patient records can be partitioned into statistical samples, and tests can be run on them. For example, a simple question about the effect of gender on some aspect of a disease or treatment could be answered by dividing the relevant patient-records into males and females, and running the appropriate statistical tests on the dependent variable associated with the aspect of the disease or treatment of interest. RX is designed to help research clinicians answer these sorts of questions. It has a knowldege base containing 1) detailed medical knowledge, 2) knowledge of statistical procedures, and 3) knowledge of the contents and structure of the database on which it will be used. When it is completed, RX will use its knowledge to formulate a plan for testing the user's hypothesis, then execute this plan, and finally design and execute new statistical experiments based on the previous results.

Consultation Systems

A number of consultation systems have been developed at the HPP. The first of these was the MYCIN program, which diagnoses meningitis and blood infections. MYCIN interacts with a physician by asking a series of questions, and then suggests the organism responsible for the infection and recommends appropriate antibiotics. In conjunction with the TEIRESIAS program, MYCIN is able to explain why it asked a given question, that is, what the answer to the question will contribute to the diagnosis. Furthermore, during the course of

a consultation or after a diagnosis has been made, MYCIN can explain how it arrived at a conclusion. In the areas of meningitis and blood infections, MYCIN has achieved levels of performance equal to nationally recognized experts.

In addition to an explanation facility, the TEIRESIAS program provides a consultation approach to knowledge acquisition. In the event that an expert user of MYCIN disagrees with a diagnosis or line of reasoning, the TEIRESIAS program will help him modify MYCIN's knowledge. It leads him systematically back through the line-of-reasoning that produced the diagnosis to the point at which he indicates the analysis went awry, and interacts with him to modify rules or acquire new rules. Lastly, it reruns the consultation to test the solution and gain the expert's concurrence.

Care was taken in the design of MYCIN to keep its knowledge about meningitis and blood infections separate from its backward-chaining control structure. The EMYCIN ("Essential" MYCIN) package derived from this effort. It has no domain knowledge, but it provides the user with a representation for such knowldege, and with a control structure for manipulating it.

PUFF, a computer program for the interpretation of standard laboratory measures of pulmonary function, was implemented in EMYCIN. About 50 quantitative parameters are calculated from measurement of lung volumes, flow rates, and diffusion capacity. PUFF uses these measurements, as well as the patient's history, the diagnosis of the referring physician, and its knowledge of three types of pulmonary diseases to produce a report and diagnosis of the patient's condition. The PUFF system is now in routine use at the Pacific Medical Center.

A non-medical consultation program implemented in EMYCIN is SACON. The objectives of this project were to demonstrate the generality of EMYCIN, and to develop and partially implement a practical "automated consultant" for structural engineers. The consultation is intended to provide advice regarding the use of a structural analysis program called MARC, which offers a large (and to the novice, bewildering) choice of analysis methods to model the structure of interest. The user of MARC typically knows what he wants it to do, e.g. examine the behavior of a structure under expected loading conditions, but he does not know how to set up MARC to do it. SACON recommends an analysis strategy which is used to direct the MARC user in the choice of specific input data, e.g. numerical methods and material properties. The performance of the SACON program matches that of a human consultant for the limited domain of structural analysis problems that was initially selected.

The most recent consultation program at the HPP is the oncology protocol management system, call ONCOCIN. It is an interactive system to be used by oncology faculty and fellows in the Stanford outpatient oncology clinic. A consultation program has been implemented that gives high-quality advice on chemotherapy for patients with Hodgkin's and non-Hodgkin's Lymphoma. One important design issue was how to represent the time course of the disease; this is an interesting problem for many medical, and also non-medical, applications. ONCOCIN uses parts of EMYCIN, but other

representations and control schemes have also been developed in response to the special needs of the project.

Education as a Domain

GUIDON

The major objective of the GUIDON project is to design and build a Computer-Aided Instruction program that uses (a) knowledge of a domain and (b) knowledge of tutoring strategies, while keeping the two distinct. The aim is to develop methods for teaching the expert knowledge in extant knowledge bases, like MYCIN's knowledge about infectious diseases. In fact, the initial version of GUIDON uses MYCIN's knowledge base. GUIDON is designed to be a supplemental tool for problem-solving practice in technical domains. Students have a background that prepares them for solving problems in a mixed-initiative dialogue with GUIDON. Particular problems are presented to the student, and the tutoring strategies are used to help him solve the problems. GUIDON's knowledge of the task domain, e.g. infectious diseases, is used to prepare understandable presentations for the student and model his approach to specific problems. Although GUIDON currently operates as a tutorial for MYCIN, it is extendable to any MYCIN-like consultation system.

Domains of Engineering

The Computer Diagnostician Project
(In collaboration with IBM)

The purpose of this project is to explore the use of causal, structural, and teleological models of computer systems in diagnosing computer system faults. The major components of the research are the development of an adequate "machine-definition" language for encoding such models and the identification of general diagnostic techniques. Each model must include not only a description of the system's "anatomy" and "physiology" but also an argument as to how this structure realizes the system's function in terms of general principles of design and everyday analogies. The advantages of this approach over a MYCIN-like system of device and malfunction specific rules are economy, the ability to diagnose a wider range of faults, and superior explanation and knowledge acquisition capabilities. The practical goal of the research is an automated diagnostician for a contemporary computer system like the IBM 4331. Other intended byproducts of the research include a program for assisting human field engineers, a training facility for field engineers based on the device models, and feedback on hardware design.

The VLSI Design Aids Project

The overall objective of the VLSI design aids project is the

development of intelligent, high-performance computer tools to assist in the design of very large scale integrated circuits. The technology of integrated circuits will soon permit between 10^5 and 10^7 gates to be in a single circuit; the problem of designing circuits of this size is very considerable. The tools developed in this project will be integrated into the design automation system that is being developed by the Stanford Center for Integrated Systems. The initial focus of the project is on design problems at the layout level.

The project is currently developing a heuristic layout program (HLP), which is a technology independent, knowldege-based program to automatically generate a geometric (mask-level) layout of an integrated circuit, given an abstract description of the circuit (e.g., a hierarchical symbolic layout of the circuit.) The program is intended to serve both as a layout tool, and as a test bed for experimentation with various layout heuristics. HLP follows the hierarchical planning-with-constraints scheme developed as part of the MOLGEN project.

Interpretation of Multi-Sensor Data
(In collaboration with Schlumberger-Doll Research)

The wire-line services provided by Schlumberger involve lowering a variety of instruments into bore-holes to take physical measurements, to record them, and to intepret them. The recorded measurements are called logs, and can provide a wide range of information. Usually, the measurements are taken to determine the type, quantity, and location of hydrocarbons. Currently, the interpretation of the data is done by a combination of human interpreters and a variety of signal processing algorithms.

Our interest is to provide better tools for the log interpreters and to research the possibility of developing a program that can itself interpret logs. Such a program needs to combine symbolic reasoning with traditional, numeric, signal processing algorithms. It would also need knowledge about the signal characteristics and about geology that relates to the kinds of information expected in the signals. Finally, it would need to perform analyses by integrating information from many different measurements as well as symbolic data. Some aspects of the problem are similar to the problem of understanding speech, with noisy data and many sources of diverse, errorful knowledge. This problem is also similar to the VM (Ventilator Manager) problem described above, complementary streams of information to counterbalance the noise in the data. The initial research will be in determining, representing, and using relevant geological information. This appears to be the critical backbone of the program.

Tools for Knowledge Engineers

AGE

In the past decade a number of methods have been developed at the Heuristic Programming Project for representing and using expert knowledge in computer programs to achieve expert performance. The AGE package is an attempt to consolidate these methods into a single program, which will be used to build more knowledge-based, expert programs. The aim is to speed up the process of building these programs by packaging common AI software tools so that they need not be reprogrammed for every problem. A long-range objective is to implement methods and techniques discovered at AI laboratories.

There is an emphasis in the AGE project on keeping AGE simple to use. The eventual goal is that experts in various fields should be able to use AGE to build knowledge-based systems, without the assistance of a knowledge engineering specialist.

Units Package

The Units Package is an interactive language for knowledge representation and acquisition. It represents symbolic information in a frame-structured semantic network. The Units Package was originally developed as part of the MOLGEN project to provide a uniform method of representing the variety of types of information required for genetics experiments.

The Units Package itself contains very little inferential machinery. In the MOLGEN application two quite different approaches to controlling the inferential processes of experiment planning have been developed. The Units Package does provide a knowledge base representation language, pattern matchers, a knowledge base editor, access function, and a message-passing protocol for activating attached procedures. Each application builds up its program starting with these, sometimes augmented by a knowledge base from another application.

EMYCIN

The EMYCIN ("Essential" MYCIN) project is an attempt to provide a framework for building consultation programs in various domains. It uses the domain-independent components of the MYCIN system, notably the production rule mechanism and backward-chaining control structure. Then for each particular consultation domain the system builder supplies the rules and parameters of that domain to produce a functioning program. Work on the EMYCIN project is devoted to providing a useful environment for the new system builder, with emphasis on speeding the acquisition and debugging of the knowledge of the new domain.

To date, EMYCIN has been successfully applied in the domains of pulmonary function (PUFF), structural analysis (SACON), and blood clotting disorders (CLOT). A skilled user of EMYCIN can, with the help of a domain expert, bring up a small consultation system very quickly, often in less than one week.

RLL

Implementing a complex AI task often begins by designing and building a suitable representation language, in which the primitives of the domain are easy to express and manipulate.

The representation language used in one domain is seldom borrowed and adapted to another, because the facilities that were assets for one task become limitations elsewhere. For this reason, most such languages are built from scratch. The goal of the RLL effort is to reduce the amount of time expended in building a representation language, by providing a Representation Language Language, that is, a language that provided the user with the components of many representation languages, and with the ability to integrate them. RLL contains a large library of "representational pieces," for example, the mode of inheritance used by the Examples link of the Units package, or the A-Kind-Of type of slot used in the MIT Frames Representation Language, FRL. A novice user can easily design a language simply by picking an amalgamation of pieces; RLL is responsible for meshing them together into a coherent and working whole. A more advanced user can exploit RLL's mechanisms for designing new parts, for example, a new mode of inheritance, or a new type of format for a slot. He could then use these in the system he is building. The use of high level operators (e.g. "Create-New-Inheritance-Like(Instance-Of)") will greatly lessen the time required to construct the desired language.

The RLL system is an ongoing effort. It has recently been used in two small, internal tasks, an adventure game simulation and an exploration into a more complete self-description of various of its parts, using lower level primitives.

Handbook of Artificial Intelligence

The Handbook of Artificial Intelligence is a comprehensive overview of important ideas, techniques, and systems that have been developed during the last twenty five years of AI research. It contains about 200 short articles that are written at a level appropriate for students interested in AI, and for non-specialist engineers and scientists. In addition, overview articles discuss sub-areas of AI and the ideas and issues that characterize these sub-areas. Some of the overview articles are among the best critical discussions available of activity in the field.

Volumes 1 and 2 (of three) of the Handbook will be published early in 1981.

Expert Systems Research: Why Do It? What Is Its Significance?

Artificial Intelligence is the computer science of symbolic representations of knowledge and symbolic inference. There is a certain inevitability to this branch of computer science and its application, in particular, to medicine and biosciences. The cost of computers will fall drastically during the coming two decades. As it does, many more of the practitioners of the world's professions will be persuaded to turn to economical automatic information processing for assistance in managing the increasing complexity of their daily tasks. They will find, in most of computer science, help only for those of their problems that have a mathematical or statistical core, or are of a routine data-processing nature. But such problems will be rare, except in engineering and physical science. In medicine,

biology, management—indeed in most of the world's work—the daily tasks are those requiring symbolic reasoning with detailed professional knowledge. The computers that will act as "intelligent assistants" for these professionals must be endowed with such reasoning capabilities and knowledge. The researchers of the Heuristic Programming Project believe that their basic and applied work in Artificial Intelligence is aimed at this inevitable development.

Artificial Intelligence Center
SRI International
Menlo Park, CA 94025

Peter Hart, Director
Earl Sacerdoti, Associate Director
Charles Untulis, Assistant Director

The objective of our program of research is to develop capabilities in computers for intelligent behavior in complex situations, and to understand and extend the methods and the principles underlying such performances. We conduct a variety of projects aimed at increasing the ability of computer-based systems to solve problems, communicate with people, and perceive and interact with the physical world.

The following are our major areas of activity:

* Planning and Problem-Solving
* Image Processing and Vision
* Natural Language Processing
* Knowledge-Based Expert Systems
* Industrial Automation and Robotics
* Distributed Data Management
* Automatic Program Synthesis

Our strategy for research is to establish a nucleus of long-term projects in these core areas of artificial intelligence and to fit related activities into these larger contexts. Underlying these projects is a set of basic concepts and techniques that provide the mechanisms for integration: representation and modeling; inference and commonsense reasoning; heuristic search procedures and control structures.

Brief descriptions of the core areas will illustrate the scope of the program; the names of staff members involved are listed after each section.

Planning and Problem Solving—This work focuses on commonsense reasoning about actions and their effects. Systems have been developed that plan and execute action sequences for a mobile robot, that interactively guide the assembly and disassembly of electromechanical equipment by an apprentice technician, and that plan and execute retrieval strategies from large, distributed data bases. Efforts are under way to develop plans that can be executed by multiple effectors in parallel, that can be modified interactively by human decision-makers, and that will make it possible for problem-solving to be carried out by multiple independent processors.

(Barbara Grosz, Robert Moore, Nils Nilsson, William Park, Ann Robinson, Earl Sacerdoti, David Wilkins)

Image Processing and Computer Vision—This work covers a broad range of activities, from research into the fundamental principles of vision (in animals or machines) to applications in remote sensing, cartography, medicine, and advanced automation. Continuing basic research investigates the initial stages of visual systems, so that intrinsic features of points in a scene (namely, incident illumination, reflecting characteristics, and surface shape) may be determined. Systems are also being developed to act as assistants to a cartographer, aiding him in image calibration, mensuration, tracing of features, and interpretation.
(Gerald Agin, Harry Barrow, Robert Bolles, Ted Brain, David Falconer, Martin Fischler, Thomas Garvey, Gerald Gleason, Gregory Jirak, David Kashtan, Lynn Quam, Martin Tenenbaum, Andrew Witkin, Helen Wolf)

Natural-Language Processing—This work has spanned the full range of language phenomena, encompassing acoustics, phonology, prosodics, syntax, semantics, discourse structure, and the context of communication. It entails the integration of knowledge, beliefs, goals, and plans, as well as the use of deductive inference procedures and commonsense reasoning. Systems have been developed both for basic research on understandng natural-language dialogues and for providing practical natural-language interfaces to a variety of computer software systems.
(Barbara Grosz, Norman Haas, Gary Hendrix, Jerry Hobbs, Kurt Konolige, William Lewis, Robert Moore, Ann Robinson, Jane Robinson, Earl Sacerdoti, Don Walker)

Industrial Automation and Robotics—This work explores and develops general-purpose, cost-effective techniques and hardware/software modules for computer-controlled systems of manipulators (especially robots), sensors (especially visual), and other components that are flexible, adaptable, and easily trained to perform material handling, inspection, and assembly tasks in discrete-part manufacturing.
(Stephen Barnard, Robert Bolles, Gerald Gleason, John Hill, Jan Kremers, Dennis McGhie, David Nitzan, William Park, Randall Smith, Antony Sword)

Knowledge-Based Expert Systems—This work combines substantive knowledge about a particular field of research with procedures for acquiring and managing that knowledge to construct systems that can be used by specialists in the course of their professional work. Procedures for intelligent acquisition and management utilize rule-based representations of judgmental knowledge, partitioned semantic networks to store factual knowledge, subjective Bayesian probability theory to propagate measures of belief, and augmented transition network grammars for language processing.
(Richard Duda, John Gasching, Peter Hart, Kurt Konolige, Reve Reboh, Michael Wilber)

Distributed Data Management—This work applies artificial intelligence techniques to the management of information stored in large, distributed data bases. It incorporates techniques from commonsense reasoning, automatic programming, and natural-language understanding to create systems for interpreting data base queries posed in English, translating these queries into programs that can run in the languages of the data base management systems in which the desired information is stored, and monitoring the execution of those programs over a computer network.
(Barbara Grosz, Norman Haas, Gary Hendrix, Kurt Konolige, Robert Moore, Earl Sacerdoti, Daniel Sagalowicz)

Automatic Program Synthesis—This work focuses on basic issues entailed in generating computer algorithms automatically from a specification of the purpose of a given program. It draws upon and supports related work in the Center on theorem-proving, automatic planning, and programming languages for artificial intelligence. Systems have been developed for program generation based on both theorem-proving and program transformation techniques.
(Richard Waldinger)

Resources and Facilities

The Center has a wide range of advanced computer capabilities, including a DEC KL-10 time-shared computer that provides the main facility for program development. On-line access through CRT and printing terminals is available to all personnel, in support as well as staff funcitons. The computer is accessible over both the ARPANET and TYMNET communication networks.

The Center also has extensive laboratory facilities for research in vision and industrial automation. The vision laboratory has DeAnze and Ramtek color displays controlled by a VAX 11-780 with a high-speed link to the KL-10 via a shared disk channel. The industrial automation laboratory includes a Unimate 2000B manipulator with an LSI-11 microcomputer for performing coordinate transformations, an assembly station consisting of an x-y table controlled by a PDP 11-40, an SRI-developed Vision Module, an Autoplace manipulator, and a Unimate Puma. A testbed is being developed that will contain several robots equipped with vision and tactile sensing; it will coordinate subassembly stations feeding into a final-assembly operation. There is also a laboratory for research on visually guided arc welding with a Cincinnati Milacron T3 controlled by a PDP 11-34.

Other equipment includes a multicolor krypton ion laser rangefinder controlled by an LSI-11 microcomputer, a Threshold Technology VIP-100 word recognition system, and a Federal Screw Works VOTRAX ML-I Multi-Lingual Voice system for synthesizing speech.

Through an internal SRI computer network, the Center also has access to the SRI DEC PDP-11/45 UNIX time-sharing system, a B6700, and a CDC 6400.