PUBLICATIONS (continued)

Levinthal, E., Green, W., Jones, K., Tucker, R. "Processing the Viking Lander Camera Data." Journal of Geophysical Research, vol 82, no. 28, Sept. 1977.

Tucker, Robert B. "More on the Viking Mission." Keyboard, 1977/2 pp 1-4, 1977 (Hewlett-Packard).

Tucker, Robert B. "Viking Lander Imaging Investigation Picture Catalog of Primary Mission Experiment Data Record." NASA Reference Publication 1007, 568 pp, 1978.

BIOGRAPHICAL SKETCH

(Give the following information for all professional personnel listed on page 3, beginning with the Principal Investigator. Use continuation pages and follow the same general format for each person.)

NAME	TITLE	BIRTHDATE (Mo., Day, Yr.)
VEIZADES, Nicholas	R&D Engineer Instrumentation Research Labs.	August 25, 1932
PLACE OF BIRTH (City, State, Country)	**PRESENT NATIONALITY** (If non-U.S. citizen, indicate kind of visa and expiration date)	**SEX**
Larissa, Greece	U.S. Citizen	☒ Male ☐ Female

EDUCATION *(Begin with baccalaureate training and include postdoctoral)*

INSTITUTION AND LOCATION	DEGREE	YEAR CONFERRED	SCIENTIFIC FIELD
City College of San Francisco, California (1954-55)			
University of California, Berkeley	B.S.	1958	Electrical Engineering
Stanford University	M.S.	1961	Engineering Science

HONORS

MAJOR RESEARCH INTEREST	ROLE IN PROPOSED PROJECT
Electronic circuit design	Electronics Engineer

RESEARCH SUPPORT *(See instructions)*

Grant No.	Title of Project	Funding: Current Year	Funding: Project Period	% of Effort	Grant Agency
RR-00612	Resource Related Research - Computers and Chemistry (DENDRAL)	$221,255 (5/80-4/81)	$641,419 (5/80-4/83)	5	NIH

RESEARCH AND/OR PROFESSIONAL EXPERIENCE *(Starting with present position, list training and experience relevant to area of project. List all or most representative publications. Do not exceed 3 pages for each individual.)*

1962 - present — Electronics Engineer, Department of Genetics, Stanford University School of Medicine:

- 1978 - present — SUMEX Computer Project
- 1962 - 1978 — Instrumentation Research Laboratories

1961 - 1962 — Project Engineer, Fairchild Semiconductor (Instrumentation), Division of Fairchild Instrument and Camera Company, Palo Alto, California

1958 - 1961 — Senior Engineer, Link Division, General Precision, Inc., Palo Alto, California

PUBLICATIONS (none)

BIOGRAPHICAL SKETCH

(Give the following information for all professional personnel listed on page 3, beginning with the Principal Investigator. Use continuation pages and follow the same general format for each person.)

NAME	TITLE	BIRTHDATE (Mo., Day, Yr.)
YEAGER, William J.	System Programmer	June 16, 1940

PLACE OF BIRTH (City, State, Country)	PRESENT NATIONALITY (If non-U.S. citizen, indicate kind of visa and expiration date)	SEX
San Francisco, California, U.S.A.	U.S. Citizen	[X] Male [ ] Female

EDUCATION (Begin with baccalaureate training and include postdoctoral)

INSTITUTION AND LOCATION	DEGREE	YEAR CONFERRED	SCIENTIFIC FIELD
University of California, Berkeley	B.A.	1964	Mathematics
California State University, San Jose	M.A.	1967	Mathematics
University of Washington, Seattle Doctoral studies (1969-70)	None	--	Mathematics

HONORS

MAJOR RESEARCH INTEREST	ROLE IN PROPOSED PROJECT
Network communications	System Programmer

RESEARCH SUPPORT *(See instructions)*

RESEARCH AND/OR PROFESSIONAL EXPERIENCE *(Starting with present position, list training and experience relevant to area of project. List all or most representative publications. Do not exceed 3 pages for each individual.)*

1978 - present	System Programmer, SUMEX Computer Project, Department of Genetics, Stanford University School of Medicine
1975 - 1978	Scientific Programmer, Instrumentation Research Laboratories, Department of Genetics, Stanford University School of Medicine
1971 - 1975	Programmer, Bendix Field Engineering, Moffett Field, California
1970 - 1971	Programmer, WELLSCO Data Corp., San Francisco, California
1968 - 1969	Mathematics Instructor, Gavilan Jr. College, Gilroy, California
1967 - 1968	Mathematics Instructor, California Western Univ., San Diego
1966 - 1967	Mathematician/Programmer, Applied Physics Laboratory, Seattle, Washington
1966	Systems Representative, Burroughs Corp., San Jose, California

PUBLICATIONS

Smith, D.H., Achenbach, M., Yeager, W.J., Anderson, P.J., Fitch, W.L., Rindfleisch, T.: Quantitative Comparison of Combined Gas Chromatographic/Mass Spectrometic Profiles of Complex Mixtures. Anal. Chem,, 49, 1623, 1977.

Smith, D.H., Rindfleisch, T.C. and Yeager, W.J.: Exchange of Comments: Analysis of Complex Volatile Mixtures by a Combined Gas Chromatography-Mass Spectrometry System. Anal. Chem., 50, 1585,1978.

9 Collaborative Project Reports

The following subsections report on the AIM community of projects and "pilot" efforts including local and national users of the SUMEX-AIM facility at Stanford and those using the Rutgers-AIM facility (these are annotated with "[Rutgers-AIM]"). In addition to these detailed progress reports, we have included briefer summary abstracts of the fully authorized projects in Appendix A on page 331.

The collaborative project reports and comments are the result of a solicitation for contributions sent to each of the project Principal Investigators requesting the following information:

I. SUMMARY OF RESEARCH PROGRAM
- A. Project rationale
- B. Medical relevance and collaboration
- C. Highlights of research progress
  - --Accomplishments this past year
  - --Research in progress
- D. List of relevant publications
- E. Funding support (see details below)

II. INTERACTIONS WITH THE SUMEX-AIM RESOURCE
- A. Medical collaborations and program dissemination via SUMEX
- B. Sharing and interactions with other SUMEX-AIM projects (via computing facilities, workshops, personal contacts, etc.)
- C. Critique of resource management (community facilitation, computer services, communications services, capacity, etc.)

III. RESEARCH PLANS (8/80-7/86)
- A. Project goals and plans
  - --Near-term
  - --Long-range (8/81 forward)
- B. Justification and requirements for continued SUMEX use (This section will be of special importance to the study section and council review of the SUMEX-AIM renewal application)
- C. Needs and plans for other computing resources, beyond SUMEX-AIM
- D. Recommendations for future community and resource development

We believe that the reports of the individual projects speak for themselves as rationales for participation; in any case the reports are recorded as submitted and are the responsibility of the indicated project leaders.

9.1 Stanford Projects

The following group of projects is formally approved for access to the Stanford aliquot of the SUMEX-AIM resource. Their access is based on review by the Stanford Advisory Group and approval by Professor Feigenbaum as Principal Investigator.

9.1.1 AGE - Attempt to Generalize

AGE - Attempt to Generalize

H. Penny Nii and Edward A. Feigenbaum
Computer Science Department
Stanford University

ABSTRACT: Isolate inference, control, and representation techniques from previous knowledge-based programs; reprogram them for domain independence; write an interface that will help a user understand what the package offers and how to use the modules; and make the package available to other members of the AIM community and labs doing knowledge-based programs development, and the general scientific community.

I. SUMMARY OF RESEARCH PROGRAM

Project Rationale

The general goal of the AGE project is to demystify and make explicit the art of knowledge engineering. It is an attempt to formulate the knowledge that knowledge engineers use in constructing knowledge-based programs and put it at the disposal of others in the form of a software laboratory.

The design and implementation of the AGE program is based primarily on the experience gained in building knowledge-based programs at the Stanford Heuristic Programming Project in the last decade. The programs that have been, or are being, built are: DENDRAL, meta-DENDRAL, MYCIN, HASP, AM, MOLGEN, CRYSALIS [Feigenbaum 1977], and SACON [Bennett 1978]. Initially, the AGE program will embody artificial intelligence methods used in these programs. However, the long-range aspiration is to integrate methods and techniques developed at other AI laboratories. The final product is to be a collection of building-block programs combined with an "intelligent front-end" that will assist the user in constructing knowledge-based programs. It is hoped that AGE will speed up the process of building knowledge-based programs and facilitate the dissemination of AI techniques by: (1) packaging common AI software tools so that they need not be reprogrammed for every problem; and (2) helping people who are not knowledge engineering specialists write knowledge-based programs.

Medical Relevance and Collaboration

AGE is relevant to the SUMEX-AIM Community in two ways: as a vehicle for disseminating cumulated knowledge about the methodologies of knowledge engineering and as a tool for reducing the amount of time needed to develop knowledge-based programs.

(1). Dissemination of Knowledge: The primary strategy for conducting AI research at the Stanford Heuristic Programming Project is to build complex programs to solve carefully chosen problems and to allow the problems to condition the choice of scientific paths to be explored. The historical context in which this methodology arose and summaries of the programs that have been built over the last decade at HPP are discussed in [Feigenbaum 1977]. While the programs serve as case studies in building a field of "knowledge engineering," they also contribute to a cumulation of theory in representation and control paradigms and of methods in the construction of knowledge-based programs.

The cumulation and concomitant dissemination of theory occur through scientific papers. Over the past decade we have also cumulated and disseminated methodological knowledge. In Computer Science, one effective method of disseminating knowledge is in the form of software packages. Statistical packages, though not related to AI, are one such example of software packages containing cumulated knowledge. AGE is an attempt to make yesterday's "experimental technique" into tomorrow's "tool" in the field of knowledge engineering.

(2). Speeding up the Process of Building Knowledge-based Programs: Many of the programs built at HPP are intelligent agents to assist human problem solving in tasks of significance to medicine and biology (see separate sections for discussions of work and relevance). Without exception the programs were handcrafted. This process often takes many years, both for the AI scientists and for the experts in the field of collaboration.

AGE will reduce this time by providing a set of preprogrammed inference mechanisms and representational forms that can be used for a variety of tasks. Close collaboration is still necessary to provide the knowledge base, but the system design and programming time of the AI scientists can be significantly reduced. Since knowledge engineering is an empirical science, in which many programming experiments are conducted before programs suitable for a task are produced, reducing the programming and experimenting time would significantly reduce the time required to build knowledge-based programs.

Highlights of Research Summary

In addition to the framework for building programs based on the Blackboard model that was available last year, we have added the following additional tools:

1. Framework for building programs that use backward-chained production rules: Backward chaining of production rules is an inference generating mechanism that is used in the MYCIN program (and its offshoots). A simple framework has been implemented in AGE that can be used by itself (i.e. to write MYCIN-like programs) or as a part of a Blackboard based program.

2. Interface to the Units Package: There are kinds of knowledge for which the production rule representation is not suitable. We have augmented the rule-based representation in AGE with frame-like representation, as implemented in the Units package.

The Units data base can be used from the left-hand-sides of rules or can be modified by the right-hand-sides of rules. This combination, in addition to providing another representational form for the frameworks in AGE, provides inference mechanism for Units in the form of rules and other control mechanisms available in AGE.

Publications

Nii, H. Penny and Aiello, Nelleke, "AGE: a knowledge-based program for building knowledge-based programs," Proc. of IJCAI-6, pp. 645-655, vol. 2, 1979.

In addition, to acquaint a variety of users in the use of AGE, three documents are being prepared. They will be available July 1, 1980.

1. "Introduction of Knowledge Engineering, Blackboard Model, and AGE." A high level introduction to knowledge engineering and to the formulation of problems using the Blackboard model.

2. "The Joy of AGE-ing: A User's Guide to AGE-1." An introduction to the use of AGE-1 system.

3. "AGE Reference Manual." A detailed documentation.

II. INTERACTION WITH THE SUMEX-AIM RESOURCES

AGE availability

Currently AGE-1 is available to a limited number of groups on the PDP-10 at the SUMEX-AIM Computing Facility and on the PDP-20/60 at the SCORE Facility of the Computer Science Department. The current implementation is described briefly in a later section.

Dissemination

A three-day workshop was conducted on the week of March 4, 1980 for a limited number of people who had requested access to AGE. Without exception, the attendees represented organizations that wish to build knowledge-based programs, but could not do so because of lack of qualified staff. The aim of the workshop was to familiarize the user with AGE, and for each participant to implement a running program (even if a simple one) related to his own problem. The names of the organizations represented and brief descriptions of the problems for possible implementation on AGE are listed below:

Information Science Group, University of Missouri-Columbia

Interpretation of test results for determining the cause of blood coagulation problems in patient with excessive bleeding. If the interpretation problem can be successfully implemented, they will go on to implement a program that recommend anti-coagulation therapy.

Institute of Medical Electronics, University of Tokyo

Diagnosis of cardiovascular diseases using diverse data and knowledge, and therapy recommendation with re-evaluation diagnosis. In general, this group is interested in building programs that serve as research tools rather than as applied clinical tools.

Department of Psychology, University of Colorado

This groups is using the Blackboard framework in AGE to build a psychological model of prose comprehension. They have been using AGE for about one year.

Oak Ridge National Laboratory

Interpretation of physical signals--non-medical application.

Schlumberger-Doll Research Center

Interpretation of physical signals--non-medical application.

In the process of building AGE, we have used it to write some programs to serve as test programs. Three different versions of PUFF [Feigenbaum 1977; Kunz 1978]--one using the Event-driven control macro, one using the Expectation-driven control macro [Nii 1978], and another using backward-chained productions rules [Shortliffe 1977] were implemented. Since the domain-specific knowledge for PUFF already existed and was implemented in EMYCIN, each AGE version took about a week to bring up--time needed to reorganize the rules into KSs and to rewrite them in the AGE rule syntax. We have also tested a variety of small programs, including programs for cryptogram analysis, determining a bidding strategy for the game of hearts, and a graph traversal problem.

Profile of the Current AGE System

To correspond to the two general technical goals described earlier, AGE is being developed along two separate fronts: the development of tools and the development of "intelligent" user interface.

Currently Implemented Tools

The current AGE system provides the user with a set of preprogrammed modules called "components" or "building blocks". Using different combinations of these components, the user can build a variety of programs that display different problem-solving behavior. AGE also provides user interface modules that help the user in constructing and specifying the details of the components. A component is a collection of functions and variables that support conceptual entities in program form. For example, production rule, as a component, consists of: (1) a rule interpreter that support the syntactic and semantic description of production-rule representation as defined in AGE, and (2) various strategies for rule selection and execution.

The components in AGE have been carefully selected and modularly programmed to be useable in combinations. For those users not familiar enough to experiment with combining the components, AGE currently provides the user two predefined configuration of components--each configuration is called a "framework". One framework, called the Blackboard framework, is for building programs that are based on the Blackboard model [Lesser 77]. Blackboard model uses the concepts of a globally accessible data structure called a "blackboard", and independent sources of knowledge which cooperate to form hypotheses. The Blackboard model has been modified to allow flexibility in representation, selection, and utilization of knowledge. The other framework, called the Backchain framework, is for building programs that use backward-chained production rules as its primary mechanism of generating inferences.

The Front-End

To support the user in the selection, specification, and use of the components, AGE is currently organized around four major subsystems that interact in various ways. Around it is a system executive that allows the user access to the subsystems through menu selection. Figure 1. shows the general interrelationship among these subsystems.

The Browse and Design subsystems help to familiarize the user with AGE and to guide the user in the construction of user programs through the use of predefined frameworks. The third subsystem is a collection of interface modules that help the user specify the various components of the framework. The last subsystem is designed for testing and refining the user program. Each of the subsystem is described in more detail below:

BROWSE: The function of Browse subsystem is to guide the user in browsing through its textual knowledge base, called the MANUAL. The MANUAL contains (a) a general description of the building-block components on the conceptual level; (b) a description of the implementation of these concepts within AGE; (c) a description of how these components are used within the object program; (d) how they can be constructed by the user; and (e) various examples. The information in the MANUAL is organized to represent the conceptual hierarchy of the components and to represent the functional relationship among them.

DESIGN: The function of the DESIGN subsystem is to guide the user in the design and construction of his program through the use of predefined configuration of components, or framework. Each framework is defined in DESIGN-SCHEMA, a data structure in the form of AND/OR tree, that, on one hand, represents all the possible configuration of components within the framework; and, on the other hand, represents the decisions the user must make in order to design the details of the user program. Using this schema, the DESIGN subsystem guides the user from one design decision point to another. At each decision point, the user has access to the MANUAL and also to advice regarding design decisions at that point. An appropriate ACQUISITION module can be invoked from the DESIGN subsystem so that general design and implementation specifications can be accomplished simultaneously.

ACQUISITION: For each component that the user must specify, there is a corresponding acquisition module and editor that asks the user for task-specific information. The calling sequence of the acquisition module is guided by DESIGN-SCHEMA when the user is using the DESIGN subsystem. However, they can also be accessed directly from the system menu or Interlisp.

INTERPRETER: This subsystem contains several modules that help the user run and debug his program. The Check module checks for the completeness and correctness of the specification for an entire framework. The Interpreter executes the user program which can be executed with various tracing modes. AGE currently provides no special debugging tools beyond what is available in Interlisp.

EXPLANATION: AGE has enough information to replay its execution steps, and it has reasonable justifications for the actions within the various framework. However, AGE is totally ignorant of the user's task domain and has no means of conducting a dialogue about the task domain. A detailed history of the execution steps is available to the user. The HISTORYLIST can be used in a variety of ways, including the construction of explanations.

```
SYSTEM KNOWLEDGE       SUBSYSTEM          RESULT
                           |
+-----------+      +-----V------+
| MANUAL    |....>| BROWSE     |
+-----------+...   +-----+------+
              .          |
+-----------+  .   +-----V------+      +------------+
| DESIGN    |....>| DESIGN     |....>|USER SYSTEM |
| SCHEMA    |...   |            |      |   DESIGN   |
+-----------+  .   +-----+------+      +-----+------+
              .          |<..................|
+-----------+  .   +-----V------+      +------------+
|COMPONENTS |....>| ACQUISITION|....>|   USER     |
|           |      |            |      |  SYSTEM    |
+-----------+      +------------+      +-----+------+
                         |<.................|
                   +------V-----+
                   |INTERPRETER |.....> EXECUTION
                   +------|-----+       HISTORY LIST
```

Figure 1. AGE System Organization
(... = data flow; --- = control flow)

III. RESEARCH PLAN

Research Topics

The task of building a software laboratory for knowledge engineers is divided into two main sub-tasks:

1. The isolation of techniques used in knowledge-based programs: It has always been difficult to determine if a particular problem solving method used in a knowledge-based program is "special" to a particular domain or whether it generalizes easily to other domains. In existing knowledge-based programs, the domain specific knowledge and the manipulation of such knowledge using AI techniques are often so closely coupled that it is difficult to make use of the programs for other domains. One of our goals is to isolate the AI techniques that are general and determine precisely the conditions for their use.

2. Guiding the user in the initial application of these techniques: Once the various techniques are isolated and programmed for use, an intelligent agent is needed to guide the user in the application of these techniques. In AGE-1, we assume that the user understands AI techniques, knows what she wants to do, but does not understand how to use the AGE system to accomplish his task. A longer range interest involves helping the user determine what techniques are applicable to his task, i.e. it will assume that the user does not understand the necessary techniques of writing knowledge-based programs.

Research Plan

AGE-1 system is now complete, and will be released for general use on July 1. The research and development plan for AGE-2 include the following:

1. Improving the Front-end

Although the current Design subsystem provides specification functions that allow the user to interactively specify the knowledge of the domain and control structure, it does not (aside from simple advice) provide the user any help in the designing process. For example, AGE should be able to provide some heuristics on what kind of inference mechanisms and representation are appropriate for different kinds of problems. We have begun collecting knowledge-engineering heuristics, but much more work is needed in building a design aid that will be useful.

2. Adding More Tools

Our concept of a software laboratory is a facility by which the users are provided with a variety of preprogrammed components that can be combined into problem-solving frameworks--similar in spirit to designs of prefabricated houses. The user can augment and modify a framework to develop his own programs. We currently provide tools for developing programs that use the Blackboard framework and framework for backward-chained inference rules. We have also integrated the Units Package (described elsewhere) to be used within the Blackboard framework. Given the current set of components, other frameworks can, and need to be defined; i.e. other combinations of components that would be useful in solving a wide range of problems. Another inference mechanism, the heuristic search paradigm also need to be added.

3. Performance Test

Although various users have attempted to use the AGE system, it has not been tested for its power and flexibility. For the next three to five years, we will add to our task the development of an application problem complex enough to exercise the variety of components available in the current system.

Computing Resources and Management

I believe the computing and communication resources provide by the SUMEX Facility is one of the best in the country. The management is responsive to the needs of the research community and provides superb services. However, the system is getting to a point where no serious research and development is possible, because of the lack of computing cycles due to overcrowding. It is a compliment to the facility that there are so many users. On the other hand, our productivity has gone down in recent months, because of the heavy load on the system. It would appear that the situation will not improve on its own, since many of the projects that were small a few years ago are maturing into larger, more complex systems. Which is the way it should be. The environment in which the work is done also needs to grow. In short, without augmentation to the current computing power and storage space (which had never been generous), our ability to make research progress at SUMEX will be drastically curtailed.

9.1.2 AI Handbook Project

Handbook of Artificial Intelligence

E. A. Feigenbaum and A. Barr
Stanford Computer Science Department

I. SUMMARY OF RESEARCH PROGRAM

A. Technical Goals

The AI Handbook is a compendium of knowledge about the field of Artificial Intelligence. It is being compiled by students and investigators at several research facilities across the nation. The scope of the work is broad: Two hundred articles cover all of the important ideas, techniques, and systems developed during 20 years of research in AI. Each article, roughly four pages long, is a description written for non-AI specialists and students of AI. Additional articles serve as Overviews, which discuss the various approaches within a subfield, the issues, and the problems.

There is no comparable resource for AI researchers and other scientists who need access to descriptions of AI techniques like problem solving or parsing. The research literature in AI is not generally accessible to outsiders. And the elementary textbooks are not nearly broad enough in scope to be useful to a scientist working primarily in another discipline who wants to do something requiring knowledge of AI. Furthermore, we feel that some of the Overview articles are the best critical discussions available anywhere of activity in the field.

To indicate the scope of the Handbook, we have included an outline of the articles as an appendix to this report (see Appendix G on page 392).

B. Medical Relevance and Collaboration

The AI Handbook Project was undertaken as a core activity by SUMEX in the spirit of community building that is the fundamental concern of the facility. We feel that the organization and propagation of this kind of information to the AIM community, as well as to other fields where AI is being applied, is a valuable service that we are uniquely qualified to support.

C. Progress Summary

Because our objective is to develop a comprehensive and up-to-date survey of the field, our article-writing procedure is suitably involved. First drafts of Articles are reviewed by the staff and returned to the author (either an AI scientist or a student in the area). His final draft is then incorporated into a Chapter, which when completed is sent out for review to one or two experts in that particular area, to check for mistakes and omissions. After corrections and comments from our reviewers are incorporated by the staff, the manuscript is edited, and a final computer-prepared, photo-ready copy of the Chapter is generated.

We expect the Handbook to reach a size of approximately 1000 pages. Roughly two-thirds of this material will constitute Volume I of the Handbook, which will be going through the final stages of manuscript preparation in the Spring and Summer of 1980. The material in Volume I will cover AI research in Heuristic Search, Representation of Knowledge, AI Programming Languages, Natural Language Understanding, Speech Understanding, Automatic Programming, and Applications-oriented AI Research in Science, Mathematics, Medicine, and Education. Researchers at Stanford University, Rutgers University, SRI International, Xerox PARC, RAND Corporation, MIT, USC-ISI, Yale, and Carnegie-Mellon University have contributed material to the project.

D. List of Relevant Publications

Most of the chapters in Volume I of the AI Handbook have already appeared in preliminary form as Stanford Computer Science Technical Reports, authored by the respective chapter-editors:

HPP-79-12 (STAN-CS-79-726)
Ann Gardner. Search.

HPP-79-17 (STAN-CS-79-749)
William Clancey, James Bennett, and Paul Cohen.
Applications-oriented AI Research: Education.

HPP-79-21 (STAN-CS-79-754)
Anne Gardner, James Davidson, and Terry Winograd.
Natural Language Understanding.

HPP-79-22 (STAN-CS-79-756)
James S. Bennett, Bruce G. Buchanan, and Paul R. Cohen.
Applications-oriented AI Research: Science and Mathematics.

HPP-79-23 (STAN-CS-79-757)
Victor Ciesielski, James S. Bennett, and Paul R. Cohen.
Applications-oriented AI Research: Medicine.

HPP-79-24 (STAN-CS-79-758)
Robert Elschlager and Jorge Phillips. Automatic Programming.

HPP-80-3 (STAN-CS-80-793)
Avron Barr and James Davidson. Representation of Knowledge.

E. Funding Support Status

The Handbook Project is partially supported under the Heuristic Programming Project contract with the Advance Research Projects Agency of the DOD, contract number MDA 903-77-C-0322, E. A. Feigenbaum, Principle Investigator and under the core research activities of the SUMEX-AIM resource.

II. INTERACTIONS WITH SUMEX-AIM RESOURCE

A. Collaborations and medical use of programs via SUMEX

We have had a modest level of collaboration with a group of students and staff at the Rutgers resource, as well as occasional collaboration with individuals at other ARPA net sites.

B. Sharing and interactions with other SUMEX-AIM projects.

As described above, we have had moderate levels of interaction with other members of the SUMEX-AIM community, in the form of writing and reviewing Handbook material. During the development of this material, limited arrangements have been made for sharing the emerging text. As final manuscripts are produced, they will be made available to the SUMEX-AIM community both as on-line files and in the hardcopy, published edition.

C. Critique of Resource Management

Our requests of the SUMEX management and systems staff, requests for additional file space, directories, systems support, or program changes, have been answered promptly, courteously and competently, on every occasion.

III. RESEARCH PLANS (8/80 - 7/83)

A. Long Range Project Goals

The following is our tentative schedule for completion and publication of the AI Handbook:

Spring and Summer, 1980 - Volume I will go through final editing, computer typesetting, and printing.

Fall, 1980 through Spring, 1983 - Volume I will be published. Research for Volume II will be started and draft material will go through the external review process.

B. Justifications and requirements for continued SUMEX use

The AI Handbook Project is a good example of community collaboration using the SUMEX-AIM communication facilities to prepare, review, and disseminate this reference work on AI techniques. The Handbook articles currently exist as computer files at the SUMEX facility. All of our authors and reviewers have access to these files via the network facilities and use the document-editing and formatting programs available at SUMEX. This relatively small investment of resources will result in what we feel will be a seminal publication in the field of AI, of particular value to researchers, like those in the AIM community, who want quick access to AI ideas and techniques for application in other areas.

C. Your needs and plans for other computational resources

We will use document preparation programs at SUMEX and a xerographic output device at the Stanford Computer Science Department to produce the final copy of the AI Handbook.

D. Recommendations for future community and resource development

None.

9.1.3 DENDRAL Project

The DENDRAL Project
Resource-Related Research: Computers in Chemistry

Prof. Carl Djerassi
Department of Chemistry
Stanford University

I. Summary of Research Program

The DENDRAL Project is a resource-related research project. The resource to which it is related is SUMEX-AIM, which provides DENDRAL its sole computational resource for program development and dissemination to the biomedical community.

I.A. Project Rationale

The DENDRAL project is concerned with the application of state-of-the-art computational techniques to several aspects of structural chemistry. The overall goals of our research are to develop and apply computational techniques to the procedures of structural analysis of known and unknown organic compounds based on structural information obtained from physical and chemical methods and to place these techniques in the hands of a wide community of collaborators to help them solve questions of structure of important biomolecules. These techniques are embodied in interactive computer programs which place structural analysis under the complete control of the scientist working on his or her own structural problem. Thus, we stress the word assisted when we characterize our research effort as computer-assisted structure elucidation or analysis.

Our principal objective is to extend our existing techniques for computer assistance in the representation and manipulation of chemical structures along two complementary, interdigitated lines. We are developing a comprehensive, interactive system to assist scientists in all phases of structural analysis (SASES, or Semi-Automated Structure Elucidation System) from data interpretation through structure generation to data prediction. This system will act as a computer-based laboratory in which complex structural questions can be posed and answered quickly, thereby conserving time and sample. In a complementary effort we are extending our techniques from the current emphasis on topological, or constitutional, representations of structure to detailed treatment of conformational and configurational stereochemical aspects of structure.

By meeting our objectives we will fill in the "missing link" in computer assistance in structural analysis. Our capabilities for structural analysis based on the three-dimensional nature of molecules is an absolute necessity for relating structural characteristics of molecules to their observed biological, chemical or spectroscopic behavior. These capabilities will represent a quantum leap beyond our current techniques and open new vistas in applications of our programs, both of which will attract new applications among a broad community of structural chemists and biochemists who will have access to our techniques. This access depends entirely on our access to and the continued availability of SUMEX-AIM. These issues are discussed in detail in the subsequent section, Interactions with the SUMEX-AIM Resource.

The primary rationale for our research effort is that structure determination of unknown structures and the relationship of known structures to observed spectroscopic or biological activity are complex and time-consuming tasks. We know from past experience that computer programs can complement the biochemist's knowledge and reasoning power, thereby acting as valuable assistants in solving important biomedical problems. By meeting our objectives we feel strongly that our programs will become essential tools in the repertoire of techniques available to the structural biochemist.

Our research grant has recently been renewed for a three-year period beginning May 1, 1980. This renewal has come at a particularly opportune time in the development of computer aids to structure elucidation. We are beginning to push our techniques for spectral interpretation, structure generation (e.g., CONGEN) and spectral prediction to their limits within the confines of topological representations of molecular structure. Even so, these techniques are perceived to be of significant utility in the scientific community as evidenced by our workshops, the demand for the exportable version of CONGEN and the number of persons requesting collaborative or guest access to our programs at Stanford (see Interactions with the SUMEX-AIM Resource). In order to proceed further in providing to the community programs which are more generally applicable to biological structure problems and more easily accessible we must address squarely the limitations inherent in existing approaches and search for ways to solve them. Our major objectives are based on the following rationale.

None of our techniques (or the techniques of any other investigators) for computer-assisted structure elucidation of unknown molecular structures make full use of stereochemical information. As existing programs were being developed this limitation was less important. The first step in many structure determinations is to establish the constitution of the structure, or the topological structure, and that is what CONGEN, for example, was designed to accomplish. However, most spectroscopic behavior and certainly most biological activities of molecules are due to their three-dimensional nature. For example, some programs for prediction of the number of resonances observed in 13CMR spectra use the topological symmetry group of a molecule for prediction. However, in reality it is the symmetry group of the stereoisomer that must be used. This group reflects the usually lower symmetry of molecules possessing chiral centers and which generally exist in fewer than the total possible number of conformations. This will increase the number of carbon resonances observed over that predicted by the topological symmetry group alone. More generally, few of the techniques in the area of computer-assisted structure elucidation can be used in accurate prediction of structure/property relationships, whether the properties be spectral resonances or biological activities.