AI Magazine Volume 7 Number 2 (1986) (© AAAI)

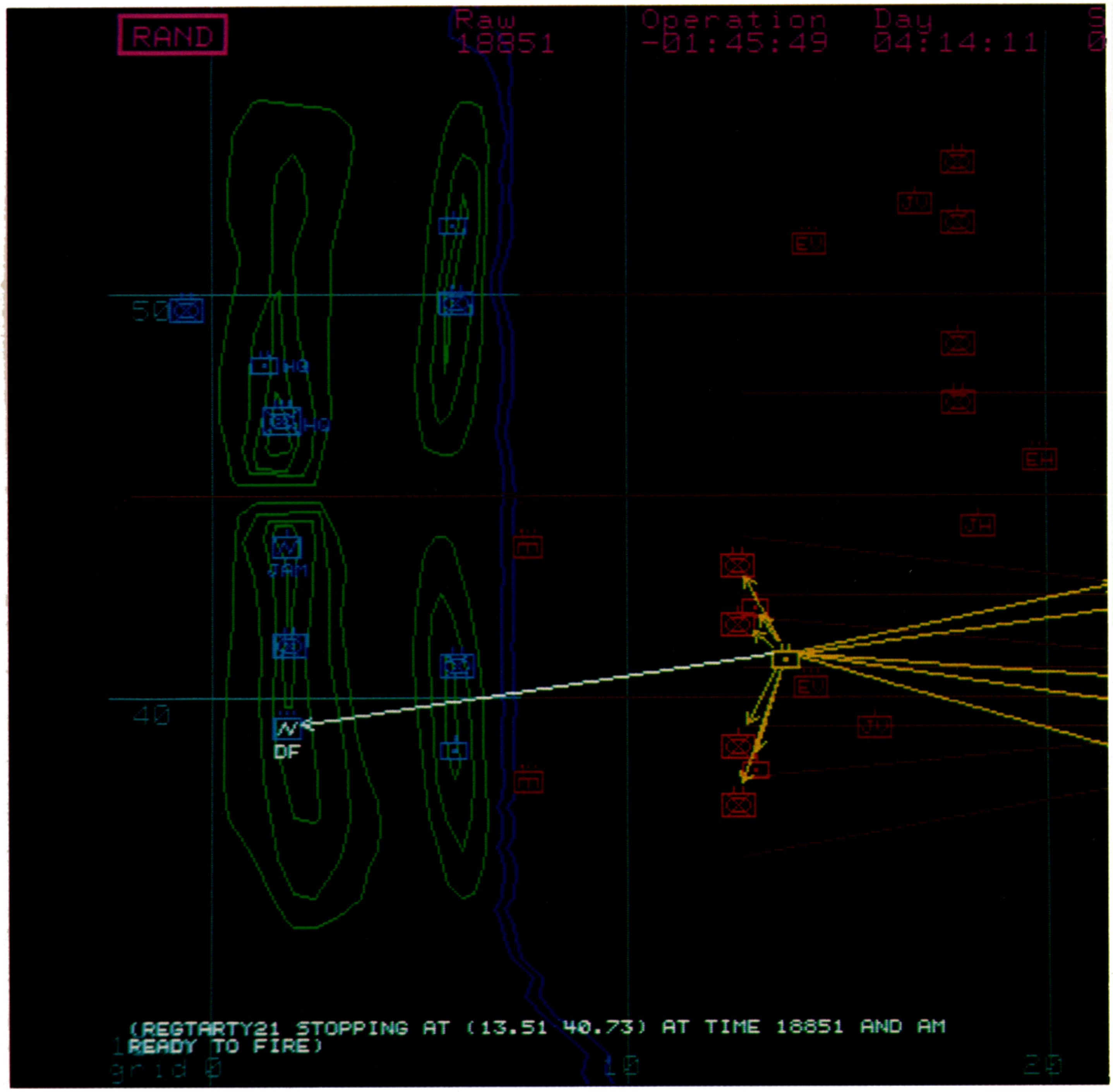
Artificial Intelligence: A Rand Perspective

Philip Klahr*

Inference Corporation,
5300 West Century Boulevard,
Los Angeles, California 90045

Donald A. Waterman

The Rand Corporation,
1700 Main Street, PO Box 2138,
Santa Monica, California 90406



*Formerly employed by The Rand Corporation.

The Early Years: The Beginnings of Artificial Intelligence

The Rand Corporation played a major role in the early development of artificial intelligence (AI). Of the twenty chapters in the first published book on AI, the 1963 *Computers and Thought* anthology by Feigenbaum and Feldman, six had been previously published as Rand research reports (Armer, 1962; Feigenbaum, 1961; Newell, Shaw & Simon, 1957, 1958; Newell & Simon, 1961a; Tonge, 1959). Much of this early work in AI was the result of the collaboration of two Rand employees, Allen Newell and Cliff Shaw, and a Rand consultant, Herbert Simon of the Carnegie Institute of Technology (later to become Carnegie-Mellon University).

Beginning in the mid-1950s, Newell, Shaw, and Simon's research on the logic theory machine, their chess playing program, and the general problem solver (GPS) defined much of the AI-related research during the first decade of AI. Their work encompassed research areas that are still prominent subfields of artificial intelligence: symbolic processing, heuristic search, problem solving, planning, learning, theorem proving, knowledge representation, and cognitive modeling. At Rand they left a legacy of publications that gave AI many of its building blocks and much of its momentum (Newell, 1954, 1960a, 1960b, 1962, 1963a, 1963b, 1963c; Newell & Shaw, 1957; Newell, Shaw, & Simon, 1956, 1957,1958, 1959a, 1959b, 1959c; Newell & Simon, 1956a, 1956b, 1959, 1961a, 1961b, 1961c; Newell, Simon, & Shaw, 1957; Shaw *et al.,* 1958; Simon, 1961, 1963).

It is important to note that this surge of AI activity at Rand did not take place in isolation. It occurred at a time and place when a host of fundamental notions about computer science and technology were being generated. In the 1950s Rand was involved in designing and

Abstract

This article presents a brief history of artificial intelligence research at the Rand Corporation. Rand has long been a leader in the field of AI, beginning with the seminal work of Newell, Shaw, and Simon some thirty years ago, and continues with recent work in expert systems and knowledge-based simulation. This article traces the major accomplishments in AI at Rand with particular emphasis on Rand's research during the past decade. The references highlight the major Rand documents on AI and related subjects

This article is a slightly revised version of the introduction to the book *Expert Systems: Techniques, Tools, and Applications,* by Philip Klahr and Donald A. Waterman (Reading, Massachusetts: Addison-Wesley, 1986).

building one of the first stored-program digital computers, the JOHNNIAC (see Figure 1) (Gruenberger, 1968);[1] George Dantzig and his associates were inventing linear programming (Dantzig, 1963); Les Ford and Ray Fulkerson were developing techniques for network flow analysis (Ford & Fulkerson, 1962); Richard Bellman was developing his ideas on dynamic programming (Bellman, 1953); Herman Kahn was advancing techniques for Monte Carlo simulation (Kahn, 1955); Lloyd Shapley was revolutionizing game theory (Shapley, 1951-1960); Stephen Kleene was advancing our understanding of finite automata (Kleene, 1951); Alfred Tarski was helping to define a theory of computation (Tarski, 1951); and James Culbertson (Culbertson, 1952, 1953) and Alton Householder (Householder, 1951a, 1951b) were investigating the relationship between neural nets, learning, and automata.[2]

Within this milieu, Newell, Shaw, and Simon were developing methods and directions for AI research. Perhaps equally important was their development of appropriate computational tools for AI programming. Using the notion of linked-list structures to represent symbolic information, Newell and his associates developed the first symbol-manipulating and list-processing languages, a series of IPL (Information-Processing Language) languages that culminated in IPL-V (Newell, 1963c; Newell & Tonge, 1960). In their 1963 paper (Bobrow & Raphael, 1963), Dan Bobrow and Bert Raphael (both of MIT at the time but also Rand consultants) included IPL-V as one of the earliest and most highly developed list-processing languages.

Because of Rand's unique computing environment and its close ties to the Carnegie Institute of Technology, several Carnegie graduate students were attracted to Rand, and several Ph.D. dissertations emerged including those of Fred Tonge (Tonge, 1959, 1960) and Ed Feigenbaum (Feigenbaum, 1959, 1961). During the early 1960s, Feigenbaum, in collaboration with Simon, continued to publish Rand reports (Feigenbaum, 1964; Feigenbaum & Simon, 1961a, 1961b, 1961c, 1962; Simon & Feigenbaum, 1964) describing his experiments with his verbal learning program EPAM. Even after completing his work at Carnegie, Feigenbaum remained a Rand consultant and was highly influential in Rand's research on expert systems and expert system languages that emerged in the early 1970s. Newell and Simon were also Rand consultants during the 1960s and 1970s. One of their associates, Don Waterman, joined Rand in the mid-1970s and brought much of their influence on the use of production systems to Rand's first work on expert systems.

[1] The JOHNNIAC, named after John von Neumann, a Rand consultant in the late 1940s and early 1950s, was in operation from 1953 to 1966. It was used extensively by Newell, Shaw, and Simon in their work on information processing

[2] Much of the credit for creating this intellectually stimulating environment belongs to John Davis Williams, who led Rand's Mathematics Department in the 1950s, and also to the United States Air Force for its generous sponsorship of a broad range of research activities.

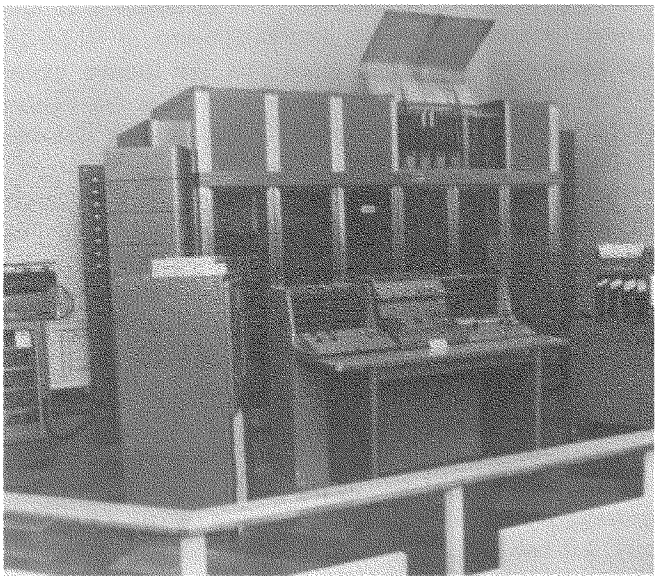
AI also had its share of controversy, however, at Rand and elsewhere. Given its quick rise to popularity and its ambitious predictions (Simon & Newell, 1958), AI soon had its critics, and one of the most prominent, Hubert Dreyfus, published his famous critique of AI (Dreyfus, 1965) while he was consulting at Rand. In addition, the early promise of automatic machine translation of text from one language to another (the emphasis at Rand was on translation from Russian to English) produced only modest systems, and the goal of fully automated machine translation was abandoned in the early 1960s.

The research in machine translation did, however, serve to elucidate the difficult problems of automated language understanding and translation. As a result, work in this area turned toward fundamental and generic issues of linguistic theory, and Rand engaged in over a decade of activity in computational linguistics. By 1967 Rand researchers had produced a wealth of literature (over 140 articles) on linguistic theory and research methods, computational techniques, the English and Russian languages, automatic content analysis, information retrieval, and psycholinguistics (Hays, Henisz-Dostert, & Rapp, 1967). In addition, David Hays produced one of the earliest textbooks on computational linguistics (Hays, 1967).

During the 1960s, Rand provided a center in which natural language researchers from all over the world could meet, communicate, and collaborate. Special seminar programs and summer symposia (for example, Kochen *et al.*, 1964) provided ample opportunities for researchers to exchange ideas and test theories. Work at Rand during this period included a number of developments: Martin Kay and his associates were working on the MIND system, which focused on research in morphology (Kay & Martins,1970), semantic networks (Kay & Su, 1970; Shapiro, 1971), and parsing (Kaplan, 1970, 1971; Kay, 1967); Jane Robinson was developing new syntactic analyzers (Robinson & Marks, 1965); Roger Levien and Bill Maron were developing the relational data file for information retrieval and question answering (Levien, 1969; Levien & Maron, 1965, 1966); Larry Kuhns was developing a sophisticated query language for database inference (Kuhns, 1967, 1970); and, in a somewhat different area, work was beginning on a new theory of "fuzzy sets" (Bellman, Kalaba, & Zadeh, 1964).

Human-Oriented Environments

Since its early involvement with the JOHNNIAC in 1953, Rand has continually worked on the development of human-oriented interfaces. Although much of this work has been outside the AI framework, the research has provided interactive computing environments that have made AI systems easier to design, implement, debug, and understand. Today, computational environments appropriate for AI systems comprise a prominent subfield of AI research.



JOHNNIAC, One of the First
Stored-Program Digital Computers
Figure 1

Major milestones in Rand's work on human-oriented environments are JOSS, the Rand Tablet, the Rand Video Graphic System, GRAIL, and BIOMOD. Cliff Shaw's JOSS (Shaw, 1964), one of the world's first true interactive computing systems, executed interpretively, had execution tracing facilities, could be used to solve mathematical problems with considerable precision, and had a clean, easy-to-learn syntax. These characteristics influenced a number of later programming systems and environments. JOSS has remained a key computing resource at Rand for over 20 years and is still used today.

The Rand Tablet (Davis & Ellis, 1964) was the first example of a two-dimensional writing surface for computer communication. A capacitative coupling between a stylus and a grid of wires embedded in the tablet provided sufficient accuracy to allow recognition of hand-printed characters approximately 1/4-inch high. Entire interactive computing systems, including GRAIL and BIOMOD, were based on use of the Rand Tablet. In fact, the entire GRAIL system was programmed with the tablet as the sole input device.

The Rand Video Graphic System (Uncapher, 1971) was one of the first interactive graphics systems to use raster-scan technology (see Figure 2). High-resolution video displays were driven from a rotating magnetic disk. At a time when the cost of providing individual self-contained systems was prohibitive, this rapid-response system gave many users simultaneous access to precision computer graphics.

The GRAIL system (Ellis, Heafner, & Sibley, 1969) and its successor BIOMOD (Groner *et al.,* 1971) pioneered the use of graphic displays and tablets as input-output devices for programming and modeling. GRAIL allowed the direct execution of programs, defined by a combination of flowcharts and coding forms, in which each box on a top-level flowchart could itself be defined in terms of another flowchart or (at the lowest level) a coding form. BIOMOD applied these interactive concepts to a major biomedical modeling system in which transfer equations and other forms of equations describing a dynamic system could be input directly on a displayed form. The results of a simulation defined by these equations could then be viewed graphically in terms of time-history plots of any of the variables (or combinations of them). More than 15 years ago these systems used a sophisticated real-time recognition algorithm for characters hand-printed on a tablet (Groner, 1966). In addition to these prototype and demonstration systems, a number of studies were conducted, principally by Barry Boehm, on the effects on user productivity of improvements in the speed and dependability of response time of interactive computer systems (Sevin, Boehm, & Watson, 1971).

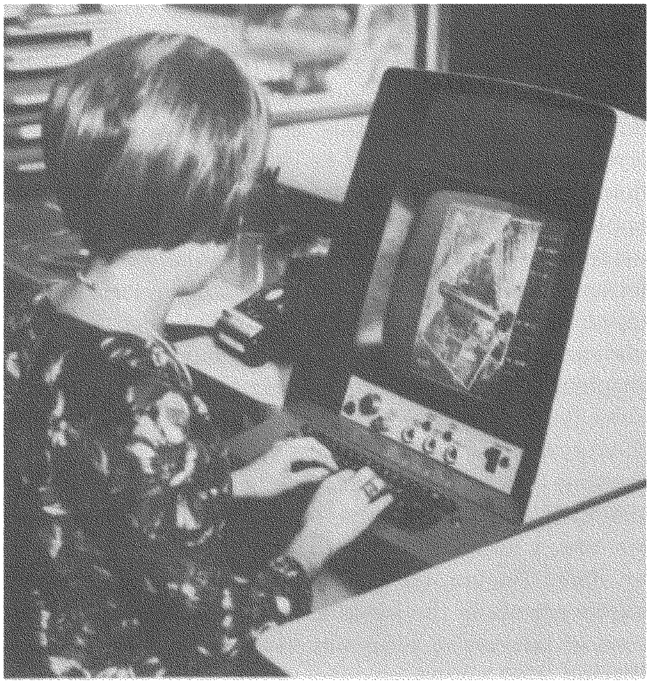
In 1972 several people involved in Rand's human-machine interface work, including Keith Uncapher, Tom Ellis, Bob Anderson, and Bob Balzer, who was just beginning to develop some of his ideas on program specification (Balzer, 1971), left Rand to form the University of Southern California's Information Sciences Institute. Anderson returned to Rand the following year and started Rand's work in an area that was to become known as expert systems.

Rand work on interactive systems in the late 1970s centered on developing UNIX-based tools, such as the Rand editor (Bilofsky, 1977) and the MH electronic message handler (Borden, Gaines, & Shapiro, 1979), and on research in interactive maps for dynamically displaying geographic information (Anderson & Shapiro, 1979). Map displays continued to play a prominent role in Rand research in the 1980s, particularly the color graphic displays used to dynamically depict geographic simulations.

Expert Systems

In the early 1970s, Bob Anderson and his associates began directing their attention toward providing aids for inexperienced computer users. The objective was to enable these users to exploit the power of computers, and even to program them, without having to become computer sophisticates. At the same time, Rand researchers were becoming increasingly interested in intelligent terminals and the possibility that such terminals might eventually be developed into powerful individualized computer workstations.

One of the initial goals of these researchers was to develop a simple, English-like language for computer users who were not programmers. Such a language, combined with intelligent terminals, could bring computers to a wide



The Rand Video Graphic System

Figure 2

range of potential users by providing an easy-to-use interactive environment in which to work.

The Rand effort was influenced by the work of Ed Feigenbaum (who retained a continuing link to Rand as a consultant) and his associates at Stanford University in the early 1970s, particularly in their use of rule-based models in the development of a system that became known as MYCIN. Anderson and his associates were particularly impressed with MYCIN's explanation facilities and its very readable English-like output. MYCIN's input, however, lacked this English-like quality because it had to be programmed in LISP, a language that was much too sophisticated for novice computer users. Therefore, Rand set out to build a language that allowed simple, English-like input as well as output.

This effort resulted in the RITA language (Anderson *et al.,* 1977; Anderson & Gillogly, 1976)[3] RITA was designed by Rand researchers as a language for developing intelligent interfaces to computer systems. RITA's unique English-like syntax could be read fairly easily by nonprogrammers, and its control mechanism gave RITA programs easy access to the local operating system. The language was used for developing not only interface programs but also expert systems, an application area to which RITA was not especially well suited. The problems that arose from attempts to develop expert systems in RITA (for example, slow execution speed and the limited expressiveness of the syntax) eventually led to the development of the Rule-Oriented System for Implementing Expertise (ROSIE) language.

RITA was used with some success as a tool for heuristic modeling in studies of international terrorism (Waterman & Jenkins, 1977). This work combined the talents of computer scientists and social scientists in the design of a knowledge-based expert system to aid terrorism researchers in their analysis of terrorist activities. The project precisely defined the basic concepts needed for a model of terrorism and used these concepts in implementing the expert system.

The success of RITA at Rand, combined with a growing interest in rule-based systems in the AI community, led two Rand researchers, Don Waterman and Rick Hayes-Roth, to organize a workshop in 1977 on rule-based pattern-directed systems. The papers presented at the workshop were later published in a book that essentially defined this emerging technology (Waterman & Hayes-Roth, 1978).

When Phil Klahr and Stan Rosenschein joined Rand's AI staff in 1978, rule-based systems became a major focus of Rand's AI research. Six Rand researchers specializing in rule-based systems gathered for an intensive two-day workshop to design the next-generation rule-based language. Although RITA had proven to be quite useful, the workshop participants wanted to develop a more sophisticated, more general-purpose, and even more English-like language specifically designed for building expert systems The result was the first design of ROSIE (Waterman *et al.,* 1979). Rand has since produced several versions of ROSIE, each intended to extend and improve the language (Fain *et al.,* 1981; Hayes-Roth *et al.,* 1981; Sowizral & Kipps, 1985).

ROSIE has been used in the development of several expert systems in a variety of application domains. In one application, Rand researchers developed the Legal Decision-making System (LDS), a prototype expert system to assist attorneys and claims adjusters in settling product liability cases (Waterman & Peterson, 1981) (see Figures 3 and 4) [4] This system enabled researchers to explore the feasibility of applying knowledge engineering techniques to the legal area. The work on legal reasoning, which initially focused on product liability in general, was later narrowed to the analysis and settlement of asbestos cases.

A second noteworthy application of ROSIE was in the area of military planning. A prototype expert system called TATR was developed to help targeteers select and prioritize airfields and target elements on these air-

[3]Most of Rand's work on expert systems, including RITA, ROSIE, and TATR, was supported by the Defense Advanced Research Projects Agency (DARPA) within its Information Processing Techniques Office.

[4]LDS was funded by Rand's Institute for Civil Justice

```
[RULE8:  COMPARATIVE NEGLIGENCE]
IF   the theory of strict-liability does apply to the plaintiff's loss
     and (the use of (the product) by the user is not negligent
         or the product's user is not the victim)
THEN assert the defendant is liable for the plaintiff's loss
     and the liability of the defendant is total
```

A ROSIE Rule from LDS for Determining Defendant Liability

Figure 3

fields (Callero, Waterman, & Kipps, 1984). The resulting program contained approximately 400 ROSIE rules.

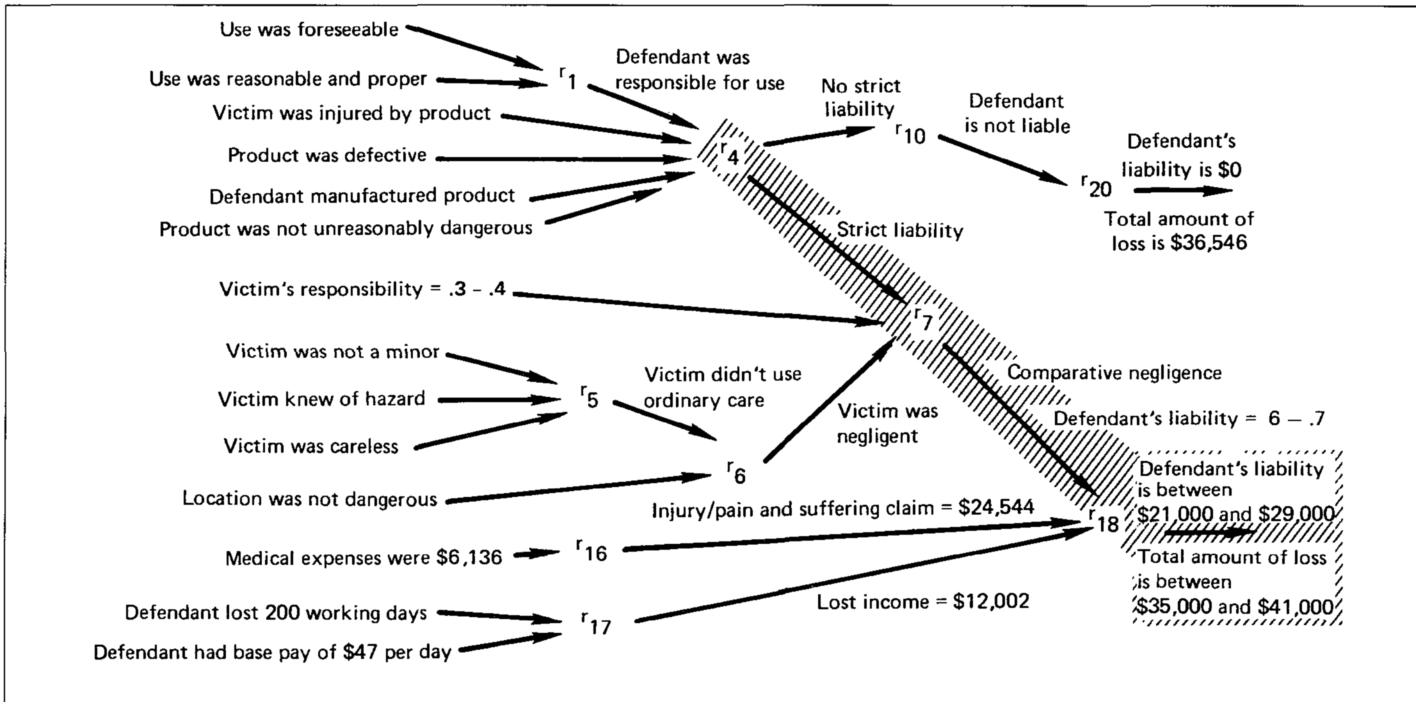
Another military application of ROSIE was also under way at Rand during the early 1980s. The Rand Strategy Assessment Center (RSAC) (Davis & Winnefeld, 1983), was designed to provide military strategists with a wargaming facility.[5] It combines a set of automated programs, or agents, with human teams to model superpower decisionmaking in conflict situations. RSAC used ROSIE to develop and implement the rule-based "scenario agent," a policy-level model of nonsuperpower behavior (Dewar, Schwabe, & McNaugher, 1982; Schwabe & Jamison, 1982). The scenario agent reacts to a current hypothetical military situation by determining if there is a threat to a nonsuperpower and, if so, specifying actions to take (sending messages, granting overflight rights, changing alignments, and so on). ROSIE also influenced RSAC's development of RAND-ABEL, a C preprocessor that facilitates the encoding of rules and decision tables in a C-based environment (Shapiro *et al.*, 1985a, 1985b). RSAC also incorporates other AI techniques, such as scripts and goal-directed search, in its operational framework (Steeb & Gillogly, 1983).

[5]RSAC has been supported by the Director of Net Assessment, Office of the Secretary of Defense

As expert system research grew at Rand and in the AI community, Rick Hayes-Roth, Don Waterman, and Doug Lenat (of Stanford University at the time but also a Rand consultant) organized a workshop in 1980 on rule-based systems and their application to the development of knowledge-based expert systems. This workshop produced the first comprehensive book on building expert systems (Hayes-Roth, Waterman, & Lenat, 1983), which includes a detailed comparison of expert-system-building languages (Waterman & Hayes-Roth, 1982). Expert systems quickly became a prominent subfield of AI research and has provided a new set of tools for application in government and industry. Research in expert systems continues to be a primary focus of Rand's AI research activity.

Knowledge-Based Simulation

Simulation has become a powerful mechanism for helping humans understand complex phenomena. Results produced by simulations have had substantial impact on a



Effect on LDS of Adding "Product Was Not Unreasonably Dangerous" to the Case Description

Figure 4

broad range of decisions in the military, government, and industry. Unfortunately, most simulations lack the utility needed for practical applications Simulations are costly to build, poorly organized, inadequately understood by users, difficult to modify, and poor in performance. Since the early 1960s, Rand has explored and developed techniques to make simulations more useful, understandable, modifiable, credible, and efficient.

Much of Rand's research in simulation methodology in the 1960s revolved around the development of the SIMSCRIPT language (Markowitz, Hausner, & Karr, 1962) and its successor, SIMSCRIPT II (Kiviat, Villanueva, & Markowitz, 1968). More recently, a research group headed by Phil Klahr has focused on applying AI and expert systems technology to simulation.[6] The goal has been to develop a research environment that helps users build and refine simulations with which to analyze and evaluate various outcomes. The primary result of this work has been ROSS, an English-like, object-oriented simulation language (McArthur & Klahr, 1982; McArthur, Klahr, & Narain, 1984). ROSS provides a programming environment in which users can conveniently design, test, and modify large knowledge-based simulations of complex mechanisms (see Figure 5).

Simulations written in ROSS are expert systems. They embody a human expert's knowledge of the objects that comprise the simulation domain. To build a ROSS simulation, it is necessary to specify the domain objects, their attributes, and their behavioral rules. ROSS has been used to design and build several military simulation systems, including a strategic air battle simulation called SWIRL (Klahr *et al.,* 1982) and a tactical ground-based combat simulation called TWIRL (Klahr *et al.*, 1984). The TWIRL system simulates a ground combat engagement between two opposing military forces (see opening illustration). It includes troop deployment, artillery firing, air interdiction, and electronic communication and jamming TWIRL was developed to experiment further with the ROSS language and to provide a prototype simulation that could be used to explore issues in electronic combat.

Computerized simulation can be a notoriously expensive tool, consuming huge amounts of computer time on powerful machines. Worse, a simulation can require many runs to adequately sample and explore the simulated system's behavior. However, the speed of almost all simulations can be dramatically increased by exploiting their inherent concurrency. In 1981, Henry Sowizral and Dave Jefferson (then at USC, but a full-time Rand consultant that summer) began investigating the use of parallel-processing in simulations. Their effort resulted in the design of a parallel-processing computer architecture called "Time Warp," which uses distributed simulation to significantly improve performance.[7] The Time Warp mechanism (Jefferson & Sowizral, 1982) accelerates a simulation's execution by removing the common restriction that simulated objects must always be synchronized in time.

```
(19)  (ask fighter-base when receiving
           (send >fighter guided by >gci to >penetrator)
       (~you schedule after !(~your scramble-delay) seconds
           tell ~the fighter chase ~the penetrator guided by ~the gci)
       (~you add ~the fighter to your list of fighters-scrambled)
       (~you remove ~the fighter from your list of fighters-available))
```

A ROSS Rule from the SWIRL
Strategic Air Battle Simulation System
Figure 5

Techniques for Expert System Development

To be effective, expert systems must contain a substantial amount of domain expertise organized for efficient search. In the late 1970s, Rand began to address issues of acquiring and effectively using such expertise. Research was conducted in a number of areas, such as acquiring knowledge by example, iteratively refining and expanding knowledge, devising efficient knowledge representations and structures for AI-based systems, and effectively using uncertain and inconsistent knowledge. During this time period, work in opportunistic planning made extensive use of research in knowledge representation and organizational techniques to develop a cognitive model of planning (Hayes-Roth & Hayes-Roth, 1978; Hayes-Roth *et al.,* 1979).[8]

Rand's exemplary programming project focused on acquiring knowledge by example The result of this work was the Exemplary Programming (EP) system (Faught *et al.,* 1980; Waterman *et al.,* 1980), which generates personalized computer programs from examples.[9] These programs can act as interfaces to complex computer systems or as intelligent assistants, freeing users from repeating detailed interactions with application programs. Writing such programs often cannot be justified because of the large number of programs needed, their personalized nature, and their fast-changing specifications. However, the EP methodology provides quick, easy, and inexpensive methods for creating individualized software of this type (see Figure 6).

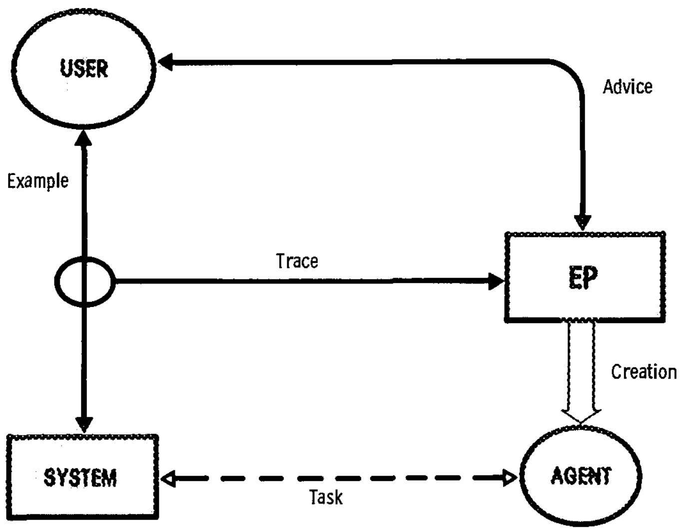
The acquisition, expansion, and refinement of knowledge was the focus of a Rand project on machine-aided knowledge acquisition (Hayes-Roth, Klahr, & Mostow, 1980a, 1980b; Mostow & Hayes-Roth, 1979). This project

[6]Rand's research on knowledge-based simulation, including work on ROSS, SWIRL, and TWIRL, has been supported by Rand's Project AIR FORCE

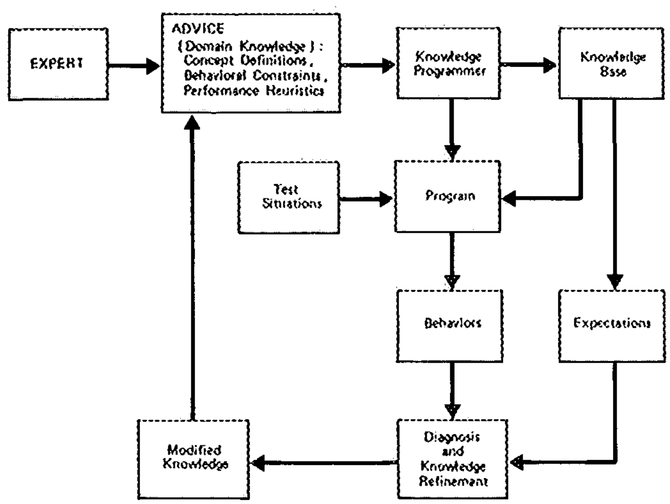
[7]Time Warp research has been supported by Rand's Project AIR FORCE, the System Development Foundation, and Rand corporate funds

[8]This work was funded by the Office of Naval Research

[9]EP was supported by DARPA and Rand corporate funds



The Exemplary Programming (EP)
Approach to Program Synthesis
Figure 6



Knowledge Programming and Refinement
Figure 7

addressed the transfer of expertise from humans to machines, as well as the functions of planning, debugging, knowledge refining, and autonomous machine learning (see Figure 7). The relative advantages of humans and machines in the building of knowledge-based systems were also considered, along with the issues of representing and structuring knowledge efficiently (Lenat, Hayes-Roth, & Klahr, 1979).[10]

[10]These efforts were supported by a grant from the National Science Foundation

A primary focus of Rand's research program has been the development of AI technology and its use in problem domains of practical importance. An important concern is the use of uncertain and inconsistent knowledge in expert systems. To study the effectiveness of inference methods under uncertainty, Ross Quinlan developed the INFERNO system (Quinlan, 1982).[11] This work documents the methods that today's expert systems use to grapple with inexact but valuable knowledge and suggests a new approach which avoids some of the problems.

Since 1979 Rand has conducted research in the area of distributed artificial intelligence (Wesson *et al.*, 1980; Thorndyke, McArthur, & Cammarata, 1981; Steeb *et al.*, 1981; Cammarata, McArthur, & Steeb, 1983; Steeb *et al.*, 1984).[12] Rand researchers have concentrated on developing and testing cooperative behaviors for air fleet control. Their work involves a network of cooperating expert systems and focuses on cooperative behaviors for plan generation, communication management, role negotiation, and data fusion

Summary

We have presented a brief history of AI research at The Rand Corporation beginning with the seminal work of Newell, Shaw, and Simon some thirty years ago and proceeding through recent work on expert systems and knowledge-based simulation. We have traced the major Rand accomplishements in AI with particular emphasis on Rand's research during the past decade. The extensive bibliography that follows highlights the major Rand publications on AI and related subjects

References

Anderson, R H , Gallegos, M , Gillogly, J J , Greenberg, R , & Villanueva, R (1977) RITA reference manual R-1808-ARPA, The Rand Corporation

Anderson, R H , & Gillogly, J J (1976) Rand intelligent terminal agent (RITA): Design philosophy R-1809-ARPA, The Rand Corporation

Anderson, R H , & Shapiro, N Z (1979) Design considerations for a computer-based interactive map display system R-2382-ARPA, The Rand Corporation

Armer, P (1962) Attitudes toward intelligent machines P-2114-1, The Rand Corporation Reprinted in E A Feigenbaum & J Feldman, (Eds), (1963) *Computers and thought* New York: McGraw-Hill, 389-405

Balzer, R M (1971) On the future of computer program specification and organization R-622-ARPA, The Rand Corporation

Bellman, R E (1953) An introduction to the theory of dynamic programming R-245, The Rand Corporation Expanded: (1956) *Dynamic Programming* R-295, The Rand Corporation and (1957) Princeton, N J : Princeton University Press

Bellman, R E , Kalaba, R E , & Zadeh, L A (1964) Abstraction and pattern classification RM-4307-PR, The Rand Corporation

Bilofsky, W (1977) The CRT text editor NED–Introduction and reference manual R-2176-ARPA, The Rand Corporation

[11]INFERNO was supported by Rand corporate funds

[12]Rand's research on distributed problem solving has been supported by the Information Processing Techniques Office of DARPA

Bobrow, D. G , & Raphael, B (1963) A comparison of list-processing computer languages RM-3842-PR, The Rand Corporation Also in (1964) *Communications of the ACM* 7:231-240 Reprinted in S. Rosen, (Ed), (1967) *Programming systems and languages* New York: McGraw-Hill, 490-511

Borden, B. S , Gaines, R S , & Shapiro, N. Z (1979) The MH message handling system: User's manual R-2367-AF, The Rand Corporation

Callero, M , Waterman, D. A , & Kipps, J. R (1984) TATR: A prototype expert system for tactical air targeting. R-3096-ARPA, The Rand Corporation

Cammarata, S , McArthur, D , & Steeb, R (1983) Strategies of cooperation in distributed problem solving N-2031-ARPA, The Rand Corporation Also in (1983) IJCAI 8:767-770

Culbertson, J T (1952) Hypothetical robots and the problem of neuroeconomy P-296, The Rand Corporation

Culbertson, J T (1953) Sense data in robots and organisms P-378, The Rand Corporation

Dantzig, G B (1963) *Linear programming and extensions* R-366-PR, The Rand Corporation and (1963) Princeton, N J : Princeton University Press Also see (1954-1958) series Notes on Linear Programming, The Rand Corporation

Davis, M R , & Ellis, T O (1964) The Rand Tablet: A man-machine graphical communication device RM-4122-ARPA, The Rand Corporation Also in (1964) Fall joint computer conference 26-1:325-331

Davis, M., Rosenschein, S , & Shapiro, N (1982) Prospects and problems for a general modeling methodology N-1801-RC, The Rand Corporation

Davis, P K , & Winnefeld, Y A (1983) The Rand strategy assessment center R-2945-DNA, The Rand Corporation

Dewar, J A., Schwabe, W , & McNaugher, T. L (1982) Scenario agent: A rule-based model of political behavior for use in strategic analysis N-1781-DNA, The Rand Corporation

Dreyfus, H L (1965) Alchemy and artificial intelligence P-3244, The Rand Corporation.

Ellis, T O., Heafner, J F , & Sibley, W L (1969) The GRAIL project: An experiment in man-machine communications RM-5999-ARPA, The Rand Corporation. See also (1969) The GRAIL Language and Operations RM-6001-ARPA, The Rand Corporation

Fain, J , Gorlin D , Hayes-Roth F , Rosenschein, S , Sowizral, H., & Waterman, D (1981) The ROSIE language reference manual N-1647-ARPA, The Rand Corporation

Faught, W S., Waterman, D A , Klahr P , Rosenschein, S J , Gorlin, D. M , & Tepper, S J (1980) EP-2: An exemplary programming system R-2411-ARPA, The Rand Corporation A shortened version appears in (1979) ACM National Conference, 135-142.

Feigenbaum, E A (1959) An information processing theory of verbal learning P-1817, The Rand Corporation

Feigenbaum, E A (1961) The simulation of verbal learning behavior P-2235, The Rand Corporation Also in (1961) Western joint computer conference 19:121-132 Reprinted in E A Feigenbaum & J Feldman, (Eds.), (1963) *Computers and thought* New York: McGraw-Hill, 297-309

Feigenbaum, E A (1964) Computer simulation of human behavior P-2905, The Rand Corporation.

Feigenbaum, E A , & Simon, H A (1961a) Forgetting in an association memory P-2311, The Rand Corporation. Also in (1961) ACM Conference 16:2C2-2C5

Feigenbaum, E. A., & Simon, H A (1961b) Performance of a reading task by an elementary perceiving and memorizing program P-2358, The Rand Corporation Also in (1963) *Behavioral Science* 8:72-76

Feigenbaum, E A , & Simon, H A (1961c) A theory of the serial position effect P-2375, The Rand Corporation Also in (1962) *British Journal of Psychology* 53:307-320

Feigenbaum, E. A., & Simon, H A (1962) Generalization of an elementary perceiving and memorizing machine P-2555, The Rand Corporation Also in C M. Popplewell, (Ed), (1963) *Information processing 1962, proceedings of IFIP congress.* Amsterdam: North Holland

Ford, L R., Jr , & Fulkerson, D R (1962) *Flows in networks* R-375-PR, The Rand Corporation and (1962) Princeton, N J : Princeton University Press Also see (1954-1958) series Notes on Linear Programming The Rand Corporation

Goldin, S E , & Klahr, P. (1981) *Learning and abstraction in simulation* IJCAI 7:212-215

Groner, G F (1966) Real-time recognition of handprinted text. RM-5016-ARPA, The Rand Corporation. Also in (1966) Fall joint computer conference, 591-601

Groner, G F , Clark, R L , Berman, R A., & DeLand, E C (1971) BIOMOD: An interactive computer graphics system for modeling R-617-NIH, The Rand Corporation.

Gruenberger, F. J (1968) The history of the JOHNNIAC RM-5654-PR, The Rand Corporation

Halliday, M A. K (1967) Some aspects of the thematic organization of the English clause RM-5224-PR, The Rand Coporation

Hayes-Roth, B , & Hayes-Roth, F (1978) Cognitive processes in planning R-2366-ONR, The Rand Corporation A revised version appears in (1979) A Cognitive Model of Planning, *Cognitive Science* 3:275-310

Hayes-Roth, B., Hayes-Roth, F , Rosenchein, S J , & Cammarata, S (1979) Modeling planning as incremental, opportunistic process N-1178-ONR, The Rand Corporation Also in (1979) IJCAI 6: 375-383

Hayes-Roth, F., Gorlin, D , Rosenschein, S , Sowizral, H , & Waterman, D (1981) Rationale and motivation for ROSIE N-1648-ARPA, The Rand Corporation

Hayes-Roth, F , Klahr, P , & Mostow, D J (1980a) Knowledge acquisition, knowledge programming, and knowledge refinement R-2540-NSF, The Rand Corporation

Hayes-Roth, F , Klahr, P , & Mostow, D J (1980b) Advice-taking and knowledge refinement: An iterative view of skill acquisition. P-6517, The Rand Corporation Also in J Anderson, (Ed), (1981) *Cognitive skills and their acquisition* Hillsdale, N J : Lawrence Erlbaum Associates, 231-253

Hayes-Roth, F , Waterman, D A , & Lenat, D B (Eds), (1983) *Building expert systems* Reading, Mass.: Addison-Wesley

Hays, D G (1967) *Introduction to computational linguistics* New York: American Elsevier

Hays, D G., Henisz-Dostert, B , & Rapp, M L (1967) Annotated bibliography of Rand publications in computational linguistics RM-3894-3, The Rand Corporation

Householder, A S (1951a) Neural nets for 'toad T1' RM-671, The Rand Corporation

Householder, A S (1951b) Some notes for simple Pavlovian learning RM-678, The Rand Corporation

Jefferson, D , & Sowizral, H (1982) Fast concurrent simulation using the Time Warp mechanism, Part I: Local control N-1906-AF, The Rand Corporation

Kahn, H. (1955) Use of different monte carlo sampling techniques. P-766, The Rand Corporation

Kaplan, R M (1970) The MIND system: A grammar-rule language RM-6265/1-PR, The Rand Corporation

Kaplan, R M (1971) Augmented transition networks as psychological models of sentence comprehension. P-4742, The Rand Corporation Also in (1972) *Artificial Intelligence* 3:77-100

Kay, M. (1967) Experiments with a powerful parser RM-5452-PR, The Rand Corporation

Kay, M , & Martins, G R (1970) The MIND system: The morphological-analysis program RM-6265/2-PR, The Rand Corporation.

Kay, M , & Su, S Y W. (1970) The MIND system: The structure of the semantic file. RM-6265/3-PR, The Rand Corporation

Kiviat, P J., Villanueva, R , & Markowitz, H M (1968) *The SIMSCRIPT II programming language* R-460-PR, The Rand Corporation and (1969) Englewood Cliffs, N J : Prentice-Hall

Klahr, P , Ellis, J , Giarla, W , Narain, S , Cesar, E , & Turner, S (1984) TWIRL: Tactical warfare in the ROSS language R-3158-AF, The Rand Corporation

Klahr, P , & Faught, W S , (1980) *Knowledge-Based Simulation* AAAI 1:181-183

Klahr, P , McArthur, D , Narain, S , & Best, E (1982) SWIRL: Simulating warfare in the ROSS language N-1885-AF, The Rand Corporation A shortened version appears as (1982) *SWIRL: An object-oriented air battle simulator* AAAI 2:331-334.

Kleene, S C (1951) Representation of events in nerve nets and finite automata RM-704, The Rand Corporation Also in C E. Shannon & J McCarthy, (Eds), (1956) *Annals of Mathematics Studies* 34, Princeton, N J : Princeton University Press, 3-41

Kochen, M , MacKay, D M , Maron, M. E , Scriven, M , & Uhr, L (1964) Computers and comprehension RM-4065-PR, The Rand Corporation

Kuhns, J L. (1967) Answering questions by computer: A logical study RM-5428-PR, The Rand Corporation

Kuhns, J L (1970) Interrogating a relational data file: Remarks on the admissibility of input queries R-511-PR, The Rand Corporation

Lenat, D B , Hayes-Roth, F , & Klahr, P (1979) Cognitive economy N-1185-NSF, The Rand Corporation A shortened version appears as (1979) *Cognitive economy in artificial intelligence systems* IJCAI 6:531-536

Levien, R E. (1969) Relational data file: Experience with a system for propositional data storage and inference execution RM-5947-PR, The Rand Corporation

Levien, R E , & Maron, M E (1965) Relational data file: A tool for mechanized inference execution and data retrieval RM-4793-PR, The Rand Corporation

Levien, R E , & Maron, M. E (1966) A computer system for inference execution and data retrieval RM-5085-PR, The Rand Corporation Also in (1967) *Communications of the ACM* 10:715-721

Markowitz, H M , Hausner, J C , & Karr, H W (1962) *SIMSCRIPT: A simulation programming language* RM-3310-PR, The Rand Corporation and (1963) Englewood Cliffs, N J : Prentice-Hall

McArthur, D , & Klahr, P., (1982) The ROSS language manual N-1854-AF, The Rand Corporation

McArthur, D , Klahr, P , & Narain, S (1984) ROSS: An object-oriented language for constructing simulations R-3160-AF, The Rand Corporation

Mostow, D J , & Hayes-Roth, F (1979) Machine-aided heuristic programming: A paradigm for knowledge engineering N-1007-NSF, The Rand Corporation A shortened version appears as (1979) *Operationalizing heuristics: Some AI methods for assisting AI programming* IJCAI 6:601-609

Narain, S , McArthur, D , & Klahr, P (1983) *Large-scale system development in several lisp environments* IJCAI 8:859-861

Newell, A (1954) The chess machine: An example of dealing with a complex task by adaptation P-620, The Rand Corporation Also in (1955) Western joint computer conference 7:101-108

Newell, A (1960a) On programming a highly parallel machine to be an intelligent technician P-1946, The Rand Corporation Also in (1960) Western joint computer conference 17:267-282

Newell, A (1960b) New areas of application of computers P-2142, The Rand Corporation Also in (1961) *Datamation* (January)

Newell, A (1962) Some problems of basic organization in problem-solving programs RM-3283-PR, The Rand Corporation Also in M C Yovits, G T Jacobi, & G D Goldstein, (Eds), (1962) Proceedings of the second conference on self-organizing systems Washington, D C : Spartan Books, 393-423.

Newell, A (1963a) Learning, generality and problem-solving RM-3285-1-PR, The Rand Corporation Also in C M Popplewell, (Ed), (1963) Information Processing 1962, Proceedings of IFIP Congress Amsterdam: North Holland, 407-412

Newell, A (1963b) A guide to the general problem-solver program GPS-2-2 RM-3337-PR, The Rand Corporation

Newell, A (Ed), (1963c) IPL-V programmers' reference manual RM-3739-RC, The Rand Corporation

Newell, A., & Shaw, J C (1957) Programming the logic theory machine P-954, The Rand Corporation Also in (1957) Western joint computer conference. 11:230-240

Newell, A , Shaw, J C , & Simon, H A (1956) Problem solving in humans and computers P-987, The Rand Corporation

Newell, A , Shaw, J C , & Simon, H A. (1957) Empirical explorations of the logic theory machine: A case study in heuristics P-951, The Rand Corporation Also in (1957) Western joint computer conference 11:218-230 Reprinted in E A Feigenbaum and J Feldman, (Eds), (1963) *Computers and thought* New York: McGraw-Hill, 109-133

Newell, A , Shaw, J C , & Simon, H A (1958) Chess-playing programs and the problem of complexity P-1319, The Rand Corporation Also in (1958) *IBM Journal of Research and Development* 2:320-335 Reprinted in E A Feigenbaum and J Feldman, (Eds), (1963) *Computers and thought* New York: McGraw-Hill, 39-70

Newell, A., Shaw, J C , & Simon, H A (1959a) The processes of creative thinking P-1320, The Rand Coporation Also in H E Gruber, G Terrell, & M Wertheimer, (Eds), (1962) *Contemporary approaches to creative thinking* New York: Atherton Press, 63-119

Newell, A , Shaw, J C , & Simon, H A (1959b) Report on a general problem-solving program P-1584, The Rand Corporation Also in UNESCO (1959) International conference on information processing, 256-264

Newell, A , Shaw, J C , & Simon, H A (1959c) A variety of intelligent learning in a general problem solver P-1742, The Rand Corporation Also in M C Yovits and S Cameron, (Eds), (1960) *Self-organizing systems* New York: Pergamon Press, 153-189

Newell, A , & Simon, H A (1956a) Current developments in complex information processing P-850, The Rand Corporation

Newell, A , & Simon, H A (1956b) The logic theory machine: A complex information processing system P-868, The Rand Corporation Also in (1956) *IRE Transactions on information theory* IT-2:61-79

Newell, A , & Simon, H A (1959) The simulation of human thought P-1734, The Rand Corporation Also in (1961) *Current Trends in Psychological Theory,* 152-179

Newell, A , & Simon, H A (1961a) GPS: A program that simulates human thought P-2257, The Rand Corporation Also in H Billing, (Ed), (1961) *Lernende automaten* Munich: R Oldenbourg KG, 109-124 Reprinted in E A Feigenbaum & J Feldman, (Eds), (1963) *Computers and thought* New York: McGraw-Hill, 279-293

Newell, A , & Simon, H A (1961b) Computer simulation of human thinking P-2276, The Rand Corporation Also in (1961) *Science* 134 (December 22):2011-2017

Newell, A , & Simon, H A (1961c) Computer simulation of human thinking and problem solving P-2312, The Rand Corporation Also in (1962) *IRE transactions on information theory* IT-8:94-133 Reprinted in (1961) *Computers and Automation* 10:(April), and in (1961) *Datamation* (June-July)

Newell, A , Simon, H A , & Shaw, J C (1957) Elements of a theory of human problem solving P-971, The Rand Corporation Also in (1958) *Psychological Review* 65:151-166

Newell, A , & Tonge, F M (1960) An introduction to information processing language V P-1929, The Rand Corporation Also in (1960) *Communications of the ACM* 3:205-211 Reprinted in S Rosen, (Ed.), (1967) *Programming systems and languages* New York: McGraw-Hill, 362-374

Quinlan, J R (1982) INFERNO: A cautious approach to uncertain inference N-1898-RC, The Rand Corporation Also in (1983) *The Computer Journal* 26:255-269

Robinson, J J , & Marks, S (1965) PARSE: A system for automatic syntactic analysis of English text, Parts I & II RM-4654-PR, The Rand Corporation

Schwabe, W , & Jamison, L M (1982) A rule-based policy-level model of nonsuperpower behavior in strategic conflicts. R-2962-DNA, The Rand Corporation

Sevin, M J, Boehm, B W, & Watson, R A (1971) A study of user behavior in problem solving with an interactive computer. R-513-NASA, The Rand Corporation

Shapiro, N Z., Hall, H E, Anderson, R H, & Lacasse, M (1985a) The RAND-ABEL programming language: History, rationale, and design R-3274-NA, The Rand Corporation

Shapiro, N Z, Hall, H. E, Anderson, R H, & Lacasse, M (1985b) The RAND-ABEL programming language: Reference manual N-2367-NA, The Rand Corporation.

Shapiro, S C. (1971) The MIND system: A data structure for semantic information processing R-837-PR, The Rand Corporation

Shapley, L S (1951-1960) Notes on the n-person game The Rand Corporation

Shaw, J C (1964) JOSS: A designer's view of an experimental on-line computing system P-2922, The Rand Corporation. Also in (1964) Fall joint computer conference 26-1:455-464

Shaw, J C, Newell, A., Simon, H A, & Ellis, T O (1958) A command structure for complex information processing P-1277, The Rand Corporation Also in (1958) Western joint computer conference 13:119-128

Simon, H A (1961) Modeling human mental processes P-2221, The Rand Corporation Also in (1963) Western joint computer conference 19:111-120

Simon, H A (1963) The heuristic compiler RM-3588-PR, The Rand Corporation Also in H A Simon and L Siklossy, (Eds), (1972) *Representation and meaning* Englewood Cliffs, N J.: Prentice-Hall, 9-43

Simon, H A, & Feigenbaum, E. A (1964) Studies in information processing theory: Similarity and familiarity in verbal learning. RM-3979-PR, The Rand Corporation

Simon, H. A, & Newell, A (1958) Heuristic problem solving: The next advance in operations research *Operations Research* 6 (January-February):1-10

Sowizral, H. A., & Kipps, J R (1985) ROSIE: A programming environment for expert systems R-3246-ARPA, The Rand Corporation

Steeb, R, Cammarata, S, Hayes-Roth, F A, Thorndyke, P W, & Wesson, R B (1981) Distributed intelligence for air fleet control R-2728-ARPA, The Rand Corporation

Steeb, R, & Gillogly, J (1983) Design for an advanced red agent for the Rand strategy assessment center R-2977-DNA, The Rand Corporation.

Steeb, R, McArthur, D, Cammarata, S, Narain, S, & Giarla, W (1984) Distributed problem solving for air fleet control: Framework and implementation N-2139-ARPA, The Rand Corporation

Tarski, A (1951) A decision method for elementary algebra and geometry R-109, The Rand Corporation

Thorndyke, P. W (1979) Heuristics for human knowledge acquisition from maps N-1193-ONR, The Rand Corporation Also in (1979) IJCAI 6:880-883

Thorndyke, P., McArthur, D, & Cammarata, S (1981) Autopilot: A distributed planner for air fleet control N-1731-ARPA, The Rand Corporation Also in (1981) IJCAI 7:171-177

Tonge, F M (1959) Summary of a heuristic line balancing procedure. P-1799, The Rand Corporation. Also in (1960) *Management Science* 7:21-42. Reprinted in E A Feigenbaum & J Feldman, (Eds.), (1963) *Computers and thought* New York: McGraw-Hill, 168-190

Tonge, F M (1960) *A heuristic program for assembly-line balancing.* P-1993, The Rand Corporation and (1961) Englewood Cliffs, N J: Prentice-Hall

Uncapher, K. (1971) The Rand video graphic system–an approach to a general user-computer graphic communication system R-753-ARPA, The Rand Corporation

Waterman, D A, Anderson, R H, Hayes-Roth, F, Klahr, P, Martins, G & Rosenschein, S J (1979) Design of a rule-oriented system for implementing expertise N-1158-1-ARPA, The Rand Corporation

Waterman, D. A, Faught, W S, Klahr, P, Rosenschein, S J, & Wesson, R (1980) Design issues for exemplary programming N-1484-RC, The Rand Corporation Also in A W Biermann, G Guiho, & Y Kodratoff, (Eds), (1984) *Automatic program construction techniques* New York: MacMillan, 433-460

Waterman, D A, & Hayes-Roth, F (Eds), (1978) *Pattern-directed inference systems* New York: Academic Press

Waterman, D A, & Hayes-Roth, F (1982) An investigation of tools for building expert systems R-2818-NSF, The Rand Corporation. Also in F Hayes-Roth, D A. Waterman, & D B Lenat, (Eds.), (1983) *Building expert systems* Reading, Mass: Addison-Wesley, 169-215

Waterman, D. A, & Jenkins, B (1977) Heuristic modeling using rule-based computer systems P-5811, The Rand Corporation Also in R H Kupperman and D M Trent, (Eds), (1979) *Terrorism: Threat, reality, response* Stanford, Calif: Hoover Institution Press, 285-324

Waterman, D A, & Peterson, M A (1981) Models of legal decisionmaking R-2717-ICJ The Rand Corporation A shortened version appears as (1980) *Rule-based models of legal expertise* AAAI 1:272-275

Wesson, R, Hayes-Roth, F, Burge, J, Stasz, C, & Sunshine, C (1980) Network structures for distributed situation assessment R-2560-ARPA, The Rand Corporation Also in (1981) *IEEE Journal of Systems, Man, and Cybernetics* SMC-11:5-23

Student Volunteers Needed For AAAI-86

AAAI-86 will be held August 10-15, 1986, in Philadelphia, Pennsylvania. Student volunteers are needed to help with local arrangements and staffing of the conference. To be eligible for a volunteer position, an individual must be an undergraduate or graduate student in any field at any college or university.

Student volunteers receive free registration at AAAI-86, as well as a copy of the conference proceedings and a "staff" T-shirt. Participating as a volunteer is a terrific way to meet students and researchers in AI and robotics from all over the world. (It's also a whole lot of fun!)

Student volunteers help with stuffing of registration packets, creating and distributing conference direction signs, preregistration and registration of attendees, staffing of science and engineering sessions, staffing of tutorials, and general organizational tasks. (This year student volunteers will *not* be asked to do any heavy labor or to man the exhibition areas.) Twelve hours of time are officially required from each volunteer, but in the general spirit of the group, many volunteers give more than the official minimum.

Some volunteers are needed for a day two weeks before the conference. Some are needed for a day three days before the conference begins. The majority of volunteers will be needed from 10-15 August.

If you are interested in participating in the AAAI-86 Student Volunteer Program, send your name, electronic mail address, telephone number(s), dates available, and student affiliation to: Linda Quarrie, Assoc. Engineer, Martin Marietta Denver Aerospace, PO Box 179, MS #0427, Denver, CO 80201 or arpanet: mmda@usc-isi.arpa with "attention: LJQ" in the subject header or call Linda at (303) 977-4142 or 977-1569 or 697-1834 for more information.