Form R-396 (7-85)

Standard Title Page -- Report on State Project

Report No.	Report Date	No. Pages	Type Report: Final	Project No. : 9507
VHTRC 86-R31	April 1986	19	Period Covered: July 1985 - April 1986	Contract No.:

Title and Subtitle

Expert Systems as Applied to Bridges and Pavements -- An Overview

Author(s)

William Zuk

Performing Organization Name and Address

Virginia Highway and Transportation Research Council
Box 3817, University Station
Charlottesville, Virginia 22903-0817

Sponsoring Agencies' Names and Addresses

Va. Dept. of Highways & Transp.	University of Virginia
1221 E. Broad Street	Charlottesville
Richmond, Virginia 23219	Virginia 22903

Key Words

Bridges
Computers
Experts
Expert Systems
Knowledge Base
Knowledge Engineer
Inference Engine
Pavements

Supplementary Notes

Abstract

Expert systems is a rapidly emerging new application of computers to aid decision makers in solving problems. This report gives an overview of what expert systems are and of what use they may be to a transportation department. The focus of the applications is in the general area of bridges and pavements. Further studies that eventually will lead to a working expert system are anticipated.

FINAL REPORT

EXPERT SYSTEMS AS APPLIED TO BRIDGES AND PAVEMENTS -- AN OVERVIEW

by

William Zuk
Faculty Research Scientist

Virginia Highway & Transportation Research Council
(A Cooperative Organization Sponsored Jointly by the Virginia
Department of Highways & Transportation and the
University of Virginia)

Charlottesville, Virginia

April 1986
VHTRC 86-R31

The research reported here, because of the scope, does not fall within the purview of any currently established advisory committee. However, the research was supported by the Council administration in connection with the conceptual research mission.

ABSTRACT

Expert systems is a rapidly emerging new application of computers to aid decision makers in solving problems. This report gives an overview of what expert systems are and of what use they may be to a transportation department. The focus of the applications is in the general area of bridges and pavements. Further studies that eventually will lead to a working expert system are anticipated,

EXPERT SYSTEMS AS APPLIED TO BRIDGES AND PAVEMENTS -- AN OVERVIEW

by

William Zuk
Faculty Research Scientist

INTRODUCTION

This report proposes to answer two questions. The first is, What are expert systems?, and the second is, Of what use are expert systems to transportation departments? Specific applications will be cited, particularly applications regarding bridges and pavements.

The study described is intended to be the first part of a four-part project. The end product of the four parts is expected to be a working model of an expert system useful to an operational branch of a transportation department. Bridges and pavements are initially targeted because of the principal investigator's familiarity with these subjects. However, in future years, expert systems should find other applications in transportation departments.

WHAT ARE EXPERT SYSTEMS?

Research into expert systems started in the mid-1960s, and since then research and its accompanying literature have grown at a rapid rate. Conferences and short courses on the subject currently abound. Numerous institutions and private companies have been created to provide services in this area. The Bibliography at the end of this report lists but a fraction of all known material related to expert systems.

In principle, expert systems are quite simple, but in detail they are extremely complex. In a simple explanation, expert systems are computerized programs dealing with complex situations that require solutions based on comprehensive knowledge of a subject. The name "expert systems" derives from the concept that such computerized programs should perform much as a human expert would in making decisions. Because the core of the program deals with broad knowledge of a subject rather than with hard facts and figures, it is sometimes referred to as "Knowledge Based Systems," or KBS. To that extent, the advanced methodology of programming used is considered a form of "Artificial Intelligence" (AI).

For many readers, this explanation may be sufficient; however, for those who wish to know more about what experts systems are and how to go about developing and using them, further explanations are given. Although this subject is replete with its own jargon, an attempt will be made to discuss it in ordinary English insofar as possible.

The beginning point in building an expert system is to isolate a suitable generic problem that needs solution. The problem should be a recurring, complex one involving many factors (technical or nontechnical) some of which may be unknown or have to be assumed. The resolution of the problem could be in many alternative forms. The problem should be such that a satisfactory resolution would require the input of one or more people possessing extensive knowledge and experience in the subject area. Although not a direct requirement, the complexity of the problem must be matched with available computer capabilities, such as storage capacity and speed of operation. If personal microcomputers are to be used, obviously less complex problems can be handled than with mainframe super computers. Finally, the time and expense of developing and maintaining the computerized expert system should eventually prove cost-, time-, or performance-effective from an operational point of view. Unfortunately, due to the newness of expert system development, hard data on this last requirement are lacking.

One of the more successful expert systems programs developed is called EMYCIN and is used in the diagnosis of certain human diseases. Another is called PROSPECTOR and is used for the location of mineral deposits. Both have good track records as compared to human experts although they are not intended to totally replace human experts. Later in this report, a number of problems in the transportation field suitable for expert system development will be listed.

Following the isolation of a suitable problem for programming, a good working knowledge of the subject must be acquired (referred to as "knowledge acquisition"). Sources of information include relevant documents, literature, and, most importantly, personal knowledge held by experts on the subject. Depending upon how complex the subject is the number of references and experts to be consulted, and the familiarity that the expert system builder has with the subject, the time to acquire the necessary knowledge may be from a few days to many months.

Researching documents and literature is a well-established procedure, but the interviewing of human experts requires special skills and methodologies. People with these skills are called "knowledge engineers." Quite often, experts themselves are not fully conscious as to how they make decisions, as many factors are drawn from a deep reservoir of subconscious knowledge, commonly referred to as experience, hunches, intuition or gut feeling. Since the free time of most experts is very limited, the knowledge engineer should acquaint himself with the subject matter as much as possible through literature before interviewing. Questions can then be more direct and probing. A tape recording of all interviews is highly desirable so that information is correctly collected for later analysis. Questionnaires filled out in the absence of a knowledge engineer are of little use, as an active dialog is invariably required to bring to light all rational and nonrational factors in the decision-making process. Obtaining the consent of an expert to be interviewed in depth can be difficult, not only because of free-time constraints, but also because of the expert's concern for the "brain-drain" of his expertise. Various forms of inducement "to tell all" may have to be used, such as an appeal to the expert's unique qualifications, the good of an organization, obligation for the training of young people in the field, the importance of the project to the profession, and personal compensation enhancement in some way.

Ideally, the interview should be conducted with minimal interruption from the outside so as not to cause the expert to loose his train of thought. Once the initial problem is posed, all paths of the decision process should be pursued, whether the process is firm or fuzzy, rational or nonrational. It is naive to assume that all decisions are based on sound logic. (Ways of dealing with fuzzy or imprecise knowledge will be dealt with later in this report in connection with the design of the computer program.) In the course of the interview it is obviously important to elicit as many applicable rules, reasons, or clearly definable procedures as possible so as to eventually build a workable expert system.

To be reasonably certain that the expert system is complete and devoid of any controversy, a number of experts on the same subject should be interviewed. There are some differences of opinion among even the best of experts. Such differences have to be taken into account in the organization of the computer program, as by listing alternatives supported by statements of consequences or probabilities of success. (It is clear from this situation that ultimately, even with a good expert system, it is a human who is responsible for the final decision, not a computer.)

Following the important step of acquiring the knowledge necessary in the decision-making process, a basic structure of the computer program has to be formulated. The nature of the structure depends on the characteristics of the problem and the manner in which decisions can or must be made.

As an example, consider the normal decision-making steps of a human in something as simple as buying a new pair of shoes. Figure 1 shows the basic outline of the process.

In a more complete outline, numerous factors would have to be listed in each of the boxes although the mind may not actually factor them out separately. For instance, in the consideration of "Need,"

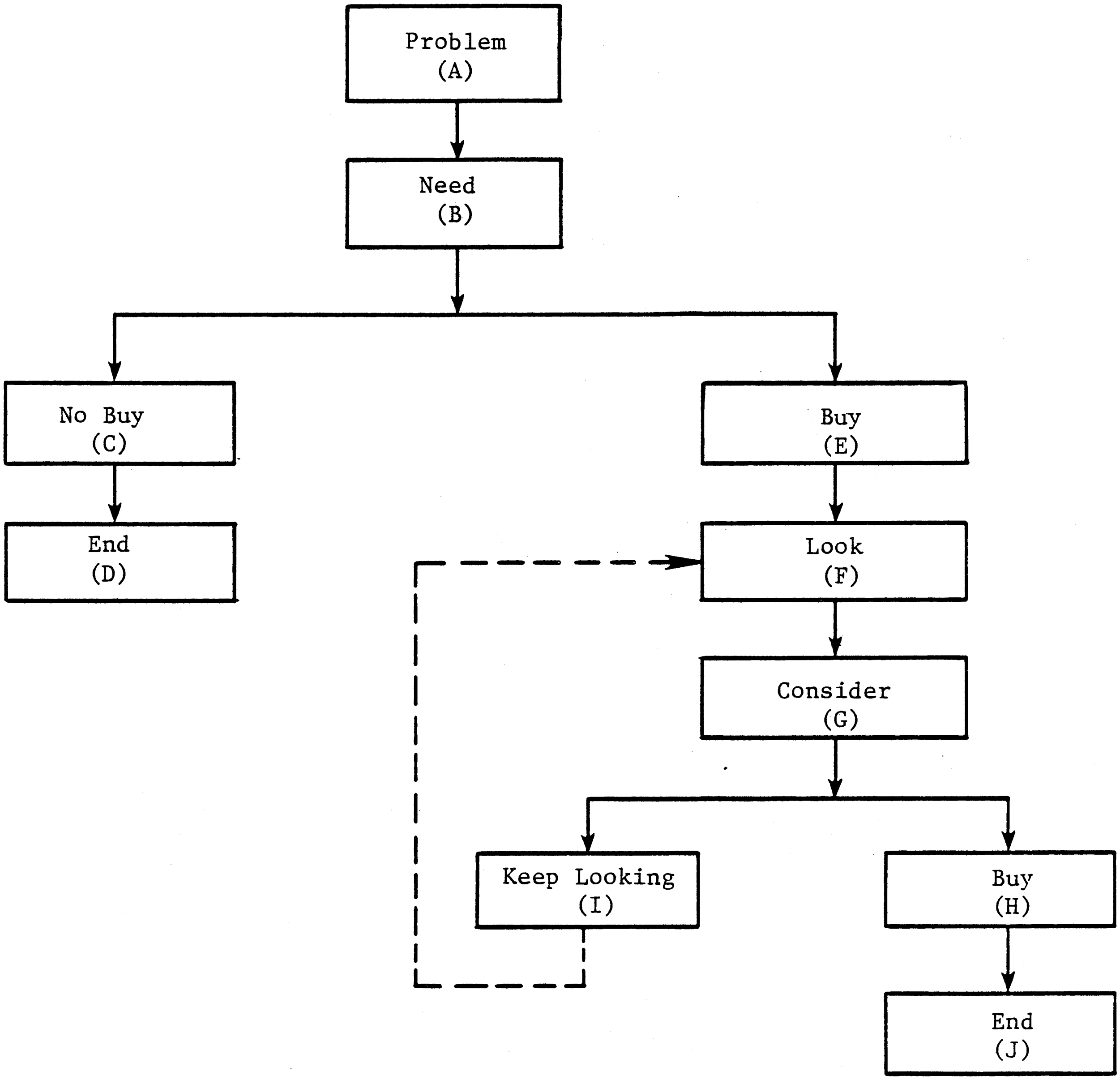

Figure 1. Framework for a natural decision-making process.

(A) Purchase of a new pair of shoes
(B) Consideration of need, availability of funds, etc.
(C) Decision not to buy
(D) End of problem
(E) Decision to buy
(F) Look at some available new shoes
(G) Consideration of size, color, style, price, etc.
(H) Decision to buy a specific pair of shoes
(I) Keep looking for the proper pair of shoes
(J) End of problem

subheadings as need based on the old shoes being worn or damaged, need for shoes of a special color or style, and need for shoes for a special purpose (as for jogging) would be listed. In many cases, the need is not clear-cut and rather subjective, in that different people will judge the perceived need quite differently, based on a mix of rational and nonrational factors. Similar types of subclassifications could be made in regard to the factors in all of the other boxes.

Figure 2 shows the same shoe problem from the generally accepted KBS or computerized point of view. Note that what the mind does naturally and holistically, the KBS requires first a knowledge engineer, then a separation of the problem into two basic parts: a "knowledge base" and an "inference engine." Stored in the knowledge base are data about the shoes in the store (size, color, price, make, etc.), information about the customer (foot size, color and price preference, etc.) as well as all the rules one uses in selecting the "right shoe" (comfortableness of fit, attractiveness, quality, affordability, etc.). Although the rules should be as precise as possible, they need not be exact, but may be somewhat "fuzzy." Rules are generally expressed in "If-Then" relationships. For example, if the shoe is the correct size, then see if it is the right color. Many such rules generally are needed. Sometimes the knowledge base is subdivided into the categories of "data" and "rules" for clarity.

The "inference engine" represents the general control process from which a conclusion can be drawn or inferred from the more specific information and rules in the "knowledge base." The two basic control strategies used are called "forward chaining" and "backwards chaining." A combination of the two is also possible. Continuing with the shoe illustration, in the forward chaining process, the salesperson (or in the case of a KBS, the computer) would ask the customer (or computer user) a series of questions concerning shoe size, color, style, price, etc. The salesperson (or computer) would then systematically scan the stock and select a pair of shoes that meets all the criteria. In the backwards chaining process, the salesperson (or computer) asks no initial questions, but selects a popular or even random pair of shoes from the stock. At this point, the size, color, style, price, etc., are evaluated. If the pair is not judged satisfactory, another pair is selected, with the process continuing until a satisfactory pair is found.

It is clear that a combination of the two, sometimes referred to as "sideways chaining," can work as well if not better in many situations where only a few critical questions are asked initially, then a trial shoe is selected. If the shoe is not satisfactory, the reason for the dissatisfaction is determined so as to select another shoe closer to that desired. The combination method often yields a solution faster and more directly than either method used separately.

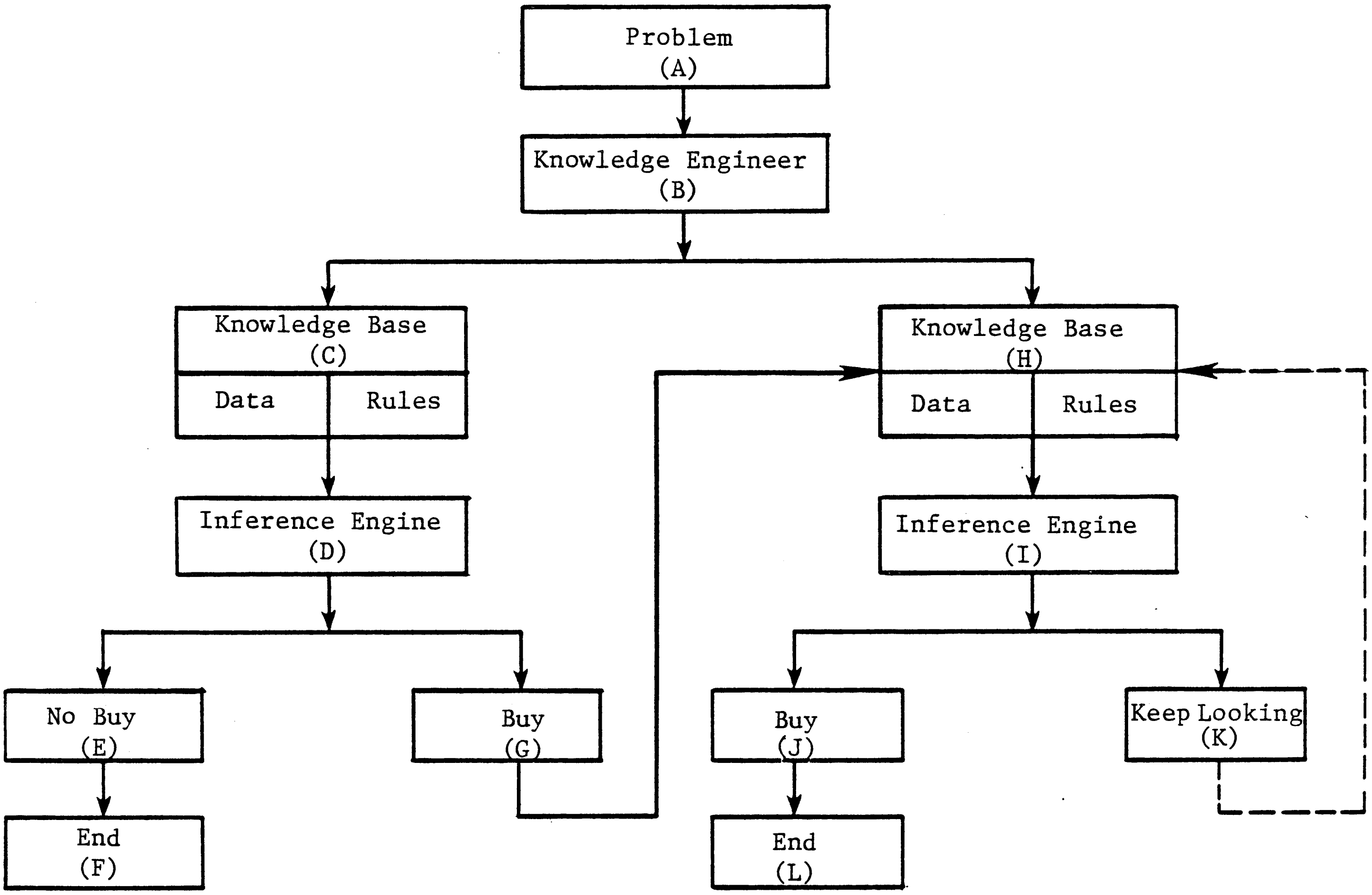

Figure 2. Framework for a knowledge-based system.

(A) Purchase of a new pair of shoes
(B) Person skilled in collecting and organizing knowledge
(C) Knowledge related to whether a new pair of shoes should be purchased
(D) Control mechanism to reach a conclusion
(E) Decision not to buy
(F) End of problem
(G) Decion to buy
(H) Knowledge related to some available shoes and method of selection
(I) Control mechanism to reach a conclusion
(J) Decision to buy a specific pair of shoes
(K) Keep looking for the proper pair of shoes
(L) End of problem

It may be that some existing expert system program is already built with the necessary structure, and it could be used as a shell to incorporate the knowledge applicable to the new problem. A list of most of the established programs is given in the reference literature, although many new ones are under development. However, if a program is to be developed from the beginning, the following features should be incorporated.

1. Input/output (I/O) statements, instructions, questions, and the like should be in plain English insofar as possible. The difficulty lies in the fact that the storage and processing of large quantities of plain English require a computer with an extraordinarily big memory. In an effort to reduce storage and increase processing speed, specialized or symbolic languages are often used, such as LISP and PROLOG, although more standardized languages as BASIC, FORTRAN, or PASCAL can be used.

2. The program should be designed to arrive at the most probably correct solution in a relatively direct and quick way. If such a solution is not deemed adequate or proper by the user, then other solutions should be examined.

3. The program should be able to explain its own reasoning process when asked, much as one would ask a human expert to explain a point and expect an answer.

4. Some method should be available in the program to deal with uncertainties in the knowledge base. It is desirable also for the system to ask for information omitted by the user, but at the same time deal with the omitted information if it is unavailable.

5. The program should be able to detect and report conflicts, inconsistencies, or deviant data.

6. All information, including that for previously solved problems, should be capable of being restored and retrieved. (In future expert systems, heuristics or the ability to learn from previously solved problems will also become a required feature.)

7. The program should be adaptable enough to update or change information easily. Particularly in the development stage, the program should allow for changes based on user comments.

8. Preferably, the software should operate on commonly available personal microcomputers so as to make it useful to as wide a group of people as possible.

One of the most troublesome aspects of designing a computerized expert system is that of dealing with uncertainties or omissions in the knowledge base. Various theories of "fuzzy logic" have been proposed to deal with this aspect. The use of statistics is one way to arrive at probabilities or degrees of certainty. Bayes' Theorem is one of the commonly used statistical tools, although it is rare that enough data are available on the subject to make even statistics reliabably exact. The use of numerical weighting factors on different components in the system is another method available to deal with uncertainties. As weighting factors must usually be done judgmentally by humans, this method too is less than exact, but it is better than nothing. A third method used is that of making comparisons. A degree of quantification in judging is possible by associating comparisons on some simple numeric scale as 0-10. In any of the three methods, if a specific number cannot be assigned to the uncertainty, upper and lower limits can be used to at least narrow the degree of uncertainty.

The knowledge engineer, working with one or more computer programmers, should be able to produce a reasonably good first try working program for an expert system for the selected problem. Assistance, if needed, can be obtained from published materials, short courses, similar working programs, and even commercially available expert system generator software. It is unlikely that a highly refined program would be produced on the first try; so testing, debugging, and refining the software are to be expected. A good way to do this is to have a cooperative expert on the subject try it out on some case studies. His comments should be noted for possible incorporation into an improved program. Comments from some potential users who may not be experts are also desirable. At this point, the program should be altered accordingly. Expert system programs, however, must never be thought of as totally finished products, as provision should be made for ongoing improvements and updating.

OF WHAT USE ARE EXPERT SYSTEMS?

Although many thousands of uses of expert systems are conceivable, the discussion here will generally be limited to those applicable to bridge and pavement problems. The listing given later contains known expert systems developed or under development in the areas under consideration; however, there are probably others being developed that are unknown to the writer, as the subject is expanding rapidly.

In the early 1970s the MYCIN Expert System Program was developed for use by the medical profession in the diagnosis and treatment of infectious diseases. The knowledge base and the inference engine (using backward chaining) are separately arranged. By substituting structural engineering knowledge for medical knowledge, the MYCIN program was converted by Stanford University in 1978 to a knowledge based consultant for structural analysis. Called SACON (Structural Analysis Consultant), it recommends the best analysis strategy for the application of the complex structural finite element technique computer program called MARC. SACON, in effect, "interviews" the user about his structure, collecting information that will allow it to infer an appropriate analysis strategy for numerical simulation. A more detailed explanation, along with several examples, is given in the publication cited in the Bibliography under the authorship of Bennett et al. An extension of SACON under study is the integration of SACON and MARC into a single closed loop system so that such information as the stress distribution and deformation pattern of a complex structure can be obtained directly.

Developed at Duke University is an expert system called BDES (Bridge Design Expert System) that is capable of aiding an engineer in the design of bridge superstructures. At present the BDES considers only conventional superstructures of short to medium span, although it allows for either steel or prestressed concrete girders. The user inputs basic data concerning the bridge length, width, skew, number of lanes, loading, materials properties, etc. The system then takes over in a series of steps and determines whether to use steel or concrete, the sizes and spacing of the various components, etc., using the forward chaining control strategy. The recommendations regarding the final design are based on the least weight of material. The program provides graphic as well as numerical output. A more complete explanation of BDES is contained in the reference in the Bibliography under the authors Welch and Biswas.

At Carnegie-Mellon University, a similar expert system program called HIRISE has been written by M. Maher and S. Fenves for the preliminary design of simple structural systems for buildings.

Under development at the National Bureau of Standards is an expert system to give recommendations on the selection of constitutents for durable concrete. Called DURCON, the program will account for the four major concrete deterioration problems; namely, freeze-thaw, sulfate attack, corrosion of reinforcing steel, and cement-aggregate reactions. The knowledge base incorporates much of the "Guide to Durable Concrete," American Concrete Institute, <u>Publication 201.2R</u>, 1977, along with heuristic information obtained by experts in the field. The inference engine uses forward chaining and the computer language is in PASCAL.

In operation on an IBM personal computer, the system asks a series of questions and allows for the user to request an explanation of a question. The output is in the form of recommendations concerning the cement, aggregate, water-cement ratio, air content, etc. Further information is cited in the publication listed in the Bibliography under Clifton et al.

The University of California, through a grant from the National Science Foundation, is developing an expert system to recommend a repair or construction strategy for dealing with pavement distress. It will attempt to simulate the decision-making process of road repair experts. The program, designed to operate on a microcomputer, will question the user about such things as the nature of defects, the amount of traffic, and climate. The output will be the recommendation of one of thirteen available repair techniques. It differs from the existing pavement management program PAVER developed by the Corps of Engineers in that it is intended for use by relatively inexperienced road maintenance crews rather than by professional managers. It is believed that the same program can be adapted to other highway problems such as traffic congestion and traffic light timing.

GEOTOX is an expert system now being tested to assess potential toxic waste sites. Developed by Lehigh University, the program asks the user about the site's characteristics, such as geological formations, subsurface soil characteristics, sinkholes, hydrology, and climate fluctuations. If information from the user is lacking, the system recommends studies that should be made on the site. The program has graphic capabilities that help users visualize the site. The final output is in the form of a recommendation concerning the site's potential to contaminate ground or surface water and recommendations on control or containment strategies.

Based on research at Stanford University, an expert system named HYDRO helps engineers to forecast such things as flood frequency and magnitude in a river basin. Typical of other expert system programs, it asks a series of questions about the site and climatic conditions. The output is in the form of answers qualified by the degree of certainty.

There is little doubt that in the next few years hundreds of expert system programs will be written for application in civil engineering. Areas as interpreting field conditions, diagnosing failures, monitoring performance, planning projects, design, education, and research are replete with ill-defined problems amenable to solutions by knowledge based systems. Several meetings were held with engineers in the Virginia Department of Highways and Transportation to define some of the more pressing bridge and pavement problems that might be addressed by expert systems. The criteria described earlier in this report were used to formulate the list of problems that follows.

Bridges

1. Optimization of a multiple span bridge layout, considering such factors as terrain, span lengths, materials, etc.

2. Determination of a bridge type and configuration where there are a number of special obstructions, as intersecting roads, immovable structures, or environmental concerns.

3. Solution to the best design and method of construction where a highway bridge is involved with a railroad line or bridge.

4. How best to schedule construction of a particular bridge, considering the maintenance of traffic, use of detours, etc.

5. How best to sequence construction of bridges where several are to be built close to one another.

6. Determination of the best procedure for coordinating design and construction matters with people outside the Bridge Division and the VDH&T (including the public).

7. How best to deal with emergency situations, as with the sudden collapse of a bridge.

8. How to decide when to repair, rehabilitate, modify, or replace an old bridge.

9. Establishment of a knowledge base on the interpretation of the AASHTO specifications.

10. How to deal with nonrational decisions, such as those involving esthetics.

Pavements

1. Determination as to when a bituminous or concrete pavement should be resurfaced with an overlay.

2. How best to correct a type of distress in a pavement (as cracking, rutting, etc.)

3. When to repair or rehabilitate a pavement.

4. Choice of the best alternative in rehabilitating a pavement (as resurfacing, reconstruction, widening, etc.).

5. How best to handle traffic while repairing or rehabilitating.
6. Procedures for emergency repairs (as for damage by a flood).
7. Setting of priorities for snow removal.
8. Procedures to follow to correct for accidents attributed to pavement conditions (as slipperiness).
9. When and where to cut grass, brush, etc., along pavements.
10. Allocation of funds for maintenance of pavements.

However, despite the virtues of expert systems, there are some limitations. Although expert systems are intended to simulate the decision-making process of human experts, computers at this time cannot truly function as holistically as can human minds. The following are some examples of what expert systems cannot do.

1. Restructure and reorganize knowledge.
2. Break rules when the situation demands.
3. Function as quasiexperts when the limits of the programmed knowledge in an area is reached.
4. Make intuitive judgements.
5. Check the validity of conclusions independently.
6. Deal with self-observed mannerisms and emotions of humans.

The bottom line for any working knowledge based system is its real effectiveness, considering such factors as development costs, cost of needed hardware, time or money expenditures incurred in problem execution, completeness or accuracy of solutions, and shortage of human experts. Simple problems, problems requiring exact solutions, and problems that are to be solved only once are not good candidates for expert systems.

FUTURE STUDIES

Anticipating the continuation of this study, the writer (with the concurrence of others in the Department) has selected Item No. 8 in the list of bridge problems given previously for further development as a case study of the potential value of expert systems. The problem of bridge repair or replacement not only meets all the criteria for an expert system, but is one of particular importance at this time when bridge repair is of growing importance in the state and in the nation.

The eventual benefits of having a working knowledge based system for bridge repair, as well as many other problems, are manifold. They are summarized as follows:

1. Rapid decisions pertaining to complex situations could be made, as the computer would do much of the evaluating of alternative solutions.

2. The likelihood of repeating mistakes would be reduced through the stored knowledge in the computer.

3. Aided by the information stored, the training of new or replacement personnel will be facilitated.

4. In some situations, fewer and less senior people than are now required could function as replacements, thus providing cost savings.

5. The expert knowledge of senior people who retire would be preserved for reference.

The case study will be conducted in three phases; namely, knowledge acquisition, software development, and testing. Separate working plans will be prepared for each phase.

BIBLIOGRAPHY

"AI to Aid War on Potholes," Engineering News Record, December 12, 1985, p. 31.

"An Overview of Expert Systems," National Bureau of Standards, NBSIR 82-2505, May 1982.

"Artificial Intelligence," Engineering News Record, March 28, 1985, pp. 20-23.

Bennett, J., L. Creary, R. Englemore, and R. Melosh, "SACON: A Knowledge Based Consultant for Structural Analysis," Stanford University Computer Science Department, Report No. STAN-CS-78-699, 1978.

Bennett, J., and R. Englemore, "SACON: A Knowledge Based Consultant for Structural Analysis," Proceedings, Sixth International Joint Conference on AI, Tokyo, 1979, pp. 47-49.

Clifton, J., B. Oltikar, and S. Johnson, "Development of Durcon, An Expert System for Durable Concrete," National Bureau of Standards, NBSIR 85-3186, July 1985.

Feigenbaum, E., Expert Systems in the 1980s, Pergamon Press, 1981.

Fenves, S., M. Maher, and D. Sriram, "Knowledge Based Expert Systems in Civil Engineering," Computing in Civil Engineering, Proceedings, Third Conference, ASCE, 1984, pp. 248-255.

Fenves, S., M. Mahar, and D. Sriram, "Expert Systems, C. E. Potential," Civil Engineering/ASCE, October 1984, pp. 44-47.

"First Expert System will Assess Toxic Waste Sites," Civil Engineering/ ASCE, December 1985, p. 14.

Foley, M., "Expert System Shells Boost AI Market," High Technology, March 1985, p. 21.

Forsyth, R., Expert Systems: Principles and Case Studies, Chapman and Hall Publication, 1984.

Forsyth, R., Expert Systems, Chapman and Hall Publication, 1984.

Hayes-Roth, F., D. Waterman, and D. Levat, Building Expert Systems, Addison-Wesley Publication, 1983.

Hendrickson, C. T., D. R. Rehabk, and S. J. Fenes, "Expert Systems in Transportation Systems Engineering," Carnegie-Mellon University, Technical Report, 1985.

Kinncan, P., "Software Tools Speed Expert System Development," *High Technology*, March 1985, pp. 16-20.

Knowledge Based Systems and AI, Course 390 Manuals, Integrated C Computer Systems Publication, 1985.

Lansdown, J., "Expert Systems: Their Impact on the Construction Industry," John Lansdown and RIBA Conference Fund, 1982.

Michaelsen, Michie & Boulanger, "The Technology of Expert Systems," *Byte*, April 1985, pp. 303-312.

Naylor, C., *Building Your Own Expert System*, Wiley & Sons, 1985.

Negoita, C. V., *Expert Systems and Fuzzy Systems*, Addison-Wesley Publication, 1985.

Thompson, B. A., and W. A. Thompson, "Inside an Expert System," *Byte*, April 1985, pp. 315-330.

Waterman, D., *A Guide to Expert Systems*, Addison-Wesley Publication, 1984

Weiss and Kullkowski, *A Practical Guide to Designing Expert Systems*, Rowman & Allanheld Publication, 1984.

Welch, J., and M. Biswas, "Application of Expert Systems in the Design of Bridges," Transportation Research Center Report, Duke University, 1986.