1145

TRANSPORTATION RESEARCH RECORD

Expert Systems for Transportation Applications

TRANSPORTATION RESEARCH BOARD
NATIONAL RESEARCH COUNCIL
WASHINGTON, D.C. 1987

Transportation Research Record 1145
Price $7.00
Editor: Elizabeth W. Kaplan
Typesetter and Layout: Joan G. Zubal

mode
1 highway transportation

subject areas
12 planning
21 facilities design
31 bituminous materials and mixes
55 traffic flow, capacity, and measurements

Transportation Research Board publications are available by ordering directly from TRB. They may also be obtained on a regular basis through organizational or individual affiliation with TRB; affiliates or library subscribers are eligible for substantial discounts. For further information, write to the Transportation Research Board, National Research Council, 2101 Constitution Avenue, N.W., Washington, D.C. 20418.

Printed in the United States of America

Library of Congress Cataloging-in-Publication Data

Expert systems for transportation applications.
p. cm. — (Transportation research record, 0361-1981 ; 1145)
ISBN 0-309-04522-3
1. Highway engineering—Data processing. 2. Expert systems
(Computer science) I. National Research Council (U.S.). Transportation
Research Board. II. Series.
TE7.H5 no. 1145
[TE153]
380.5 s—dc19 88-14836
[388.1'0028'5633] CIP

Sponsorship of Transportation Research Record 1145

GROUP 5—INTERGROUP RESOURCES AND ISSUES
William M. Spreitzer, General Motors Research Labs, chairman

Task Force on Expert Systems
Roswell A. Harris, University of Louisville, chairman
Domenick J. Billera, William Bowlby, Anselmo Osvaldo Braun, Edmond Chin-Ping Chang, Louis F. Cohn, Michael J. Demetsky, Jon D. Fricker, Geoffrey D. Gosling, Jerry J. Hajek, Chris T. Hendrickson, Mike Kaul, Olin W. Mintzer, Satish Mohan, Peter S. Parsonson, Michel Ray, Stephen G. Ritchie, Stephen Edwin Rowe, Alberto J. Santiago, Jerry B. Schneider, John R. Stone, George F. Swetnam, Hossein B. Takallou, James A. Wentworth, Marcus Ramsay Wigan

Stephen E. Blake, Transportation Research Board staff

The organizational units, officers, and members are as of December 31, 1986.

Transportation Research Record 1145

The **Transportation Research Record** series consists of collections of papers on a given subject. Most of the papers in a Transportation Research Record were originally prepared for presentation at a TRB Annual Meeting. All papers (both Annual Meeting papers and those submitted solely for publication) have been reviewed and accepted for publication by TRB's peer review process according to procedures approved by a Report Review Committee consisting of members of the National Academy of Sciences, the National Academy of Engineering, and the Institute of Medicine.

The views expressed in these papers are those of the authors and do not necessarily reflect those of the sponsoring committee, the Transportation Research Board, the National Research Council, or the sponsors of TRB activities.

Transportation Research Records are issued irregularly; approximately 50 are released each year. Each is classified according to the modes and subject areas dealt with in the individual papers it contains. TRB publications are available on direct order from TRB, or they may be obtained on a regular basis through organizational or individual affiliation with TRB. Affiliates or library subscribers are eligible for substantial discounts. For further information, write to the Transportation Research Board, National Research Council, 2101 Constitution Avenue, N.W., Washington, D.C. 20418.

Contents

Foreword

R. T. Lewis and W. F. Johnson review the case for innovation in transportation and discuss the objectives and role of Transport Canada in developing understanding of expert system technologies and their relationship to the transport sector in general and the department in particular. Areas of application of expert systems in research and development (R&D) programs are outlined, and the paper concludes with a description of Transport Canada's expert system R&D strategy and current R&D activities.

Edmond Chin-Ping Chang, in his first paper, discusses a prototype expert system to aid in the selection of computerized software packages currently being supported by the Federal Highway Administration. According to the author, the prototype expert system reviews and analyzes the information given by the user; evaluates it with various paths of reasoning; offers a conclusion; and, finally, suggests potential computer programs to specify and obtain the right traffic engineering software for the requirements of a particular application. Its design is based on information on various traffic engineering computer programs available from or being developed by the Federal Highway Administration.

Robert Shieng-I Tung and Jerry Schneider discuss how a knowledge-based expert system (KBES) approach can be used to solve the simple-mode (automobile), fixed-demand, discrete, multicriteria, equilibrium transportation network design problem. According to the authors, previous work on this problem has found that mathematical programming methods perform well on small networks with one objective. The authors state that a solution technique is needed that can be used on large networks that have multiple, conflicting criteria with different relative importance weights. The KBES approach discussed in this paper represents a new way to solve network design problems.

In his second paper, Edmond Chin-Ping Chang discusses expert systems for selecting left-turn phase treatment. Experience obtained from designing three prototype expert systems using artificial intelligence language and knowledge engineering tools is summarized.

J. J. Hajek, G. J. Chong, R. C. G. Haas, and W. A. Phang state that the principal objective of their paper is to show how knowledge-based expert system technology can be used to improve selection and planning of pavement maintenance and rehabilitation actions. The authors describe a knowledge-based computer program that can function like an expert when selecting and recommending one of these treatments: routing and sealing of cracks in cold areas. This computer program was named ROSE and is a part of a larger expert system being developed for selection and recommendation of all common preservation treatments.

Donald A. Bryson, Jr., and John R. Stone discuss a prototype expert system that recommends geometric modifications to improve intersection operation. M.1, a knowledge-based expert system development tool, was used to develop the Intersection Advisor to run on IBM or IBM-compatible microcomputers. During an interactive conference the advisor requests information on intersection volumes, critical movements, geometry, and potential lanes available. It then recommends the most efficient and feasible of 448 possible intersection improvements.

Sue McNeil and Anne Margaret Finn describe a prototype system, Bridge PAIRS (Paint Identification and Ranking System), constructed using an expert system building program based on a decision network. The system allows users to establish the facility condition, evaluate the need for bridge painting, identify appropriate painting strategies, and cost the strategies.

Stephen G. Ritchie describes the development of an initial prototype expert system to assist in the design of the structural thickness of asphalt concrete pavement overlays. The system is called OVERDRIVE (OVERlay Design heuRIstic adVisEr) and is a part of an ongoing research effort that is developing an integrated set of expert system tools for the analysis and design of highway pavement rehabilitation strategies.

Application of Expert Systems to Transportation: A Strategy for Safety and Productivity Gains

R. T. LEWIS AND W. F. JOHNSON

The purpose of this paper is to outline what Transport Canada has done and intends to do to achieve applications of expert systems in support of its own and broader sectoral objectives. Expert systems represent one of the first areas of commercial development of new computer hardware and software derived from artificial intelligence research programs. The case for innovation in transportation and the objectives and role of Transport Canada in innovation are briefly reviewed. Within this context, the department's understanding of expert systems technologies and their relationship to the transport sector in general and the department in particular is discussed. Areas of application for expert systems are outlined; criteria for development of expert systems research and development (R&D) programs are discussed; and Transport Canada's expert system R&D strategy and current R&D activities are described. The department's current and medium-term R&D activities in this field are outlined. There are two aspects to this work: first, to stimulate awareness of and seek agreement on applications of expert systems to transportation and, second, to develop limited applications in a number of selected areas and, in so doing, both realize benefits from the applications and lay the base for further work in this field. These first applications require a limited number of rules, are amenable to application with the existing base of skills, and are based on available technology.

In the future, the evolution of transportation systems in Canada will correspond closely to market demands on both the freight and the passenger service sectors. As these demands change, the system will have to cope with needs to expand traditional services or provide new ones and phase out uneconomic service at an acceptable social cost. The economic problems of meeting such challenges are compounded by Canada's geography and climate and by continuing uncertainty about the future direction of the economy. Aggregate forecasts, for example, of a 3.5 percent annual growth in demand for rail freight services, mask the uncertainty of the markets for Canada's raw material exports. Economic deregulation will pose challenges to carriers as they adapt to new competitive forces. Compounding these is the need to replace or rebuild equipment and infrastructure as these approach the end of their economic and technological lives.

In developing an approach to these diverse challenges, two common themes, increased productivity and improved safety, have been defined. The productivity theme reflects the fact that, almost uniformly across all modes, transportation operating and maintenance (O&M) costs are rising. This both reduces the funds available for new capital investment and, through its impact on rates or fares, limits the competitiveness of Canadian exporters and fuels domestic inflation. Solutions to the productivity issue must recognize that capital markets are tight and will continue to be tight for the next 20 years and that the ability of the transport sector to finance large investments will be constrained by competition for funds from other sectors. In some measure, investment in new technologies can be expected to reduce the investment that could otherwise be required in conventional technology. However, O&M costs must also be reduced through investment in innovation. Strategies must be put in place to provide job enrichment opportunities for the transport labor force and to overcome readjustment problems as new technologies are introduced.

The safety theme reflects the fact that a new product or system developed to increase productivity must also meet regulatory requirements for safety or, where possible, enhance safety. In this context, safety is defined as protection of individuals and physical assets and prevention of damage to the environment. Justification for levels of effort and objectives in support of productivity increases will also have a complementary safety thrust.

The development of artificial intelligence, which began some 30 years ago, represents an evolution distinct in nature and power from previous generations of computer development. It is the object of multibillion dollar research programs, for example, Japan's Fifth Generation Computer program, the British Alvey program, the European Strategic Program for Research and Development in Information Technology (ESPRIT), and elements of the U.S. Defense Advanced Research Projects Agency (DARPA) program. The resulting development of expert systems or knowledge-based systems provides an immediate tool to resolve some of the pressing needs for productivity and safety enhancement in the transport sector.

The Research and Development Directorate of Transport Canada has recognized the potential of expert systems and is engaged in a long-term strategy to establish the full scope of their potential and to identify and develop specific applications with short-term benefits. The progress that has been made to date is described here. However, the context within which Transport Canada's R&D strategies are evolved is complex and directly affects the form of the strategy. This context is briefly described in the next section.

Tower C, Place de Ville, Ottawa, Ontario K1A 0N5, Canada.

ROLES IN AND RESPONSIBILITIES FOR INNOVATION

The National Transportation Act of 1967 and the federal government's new "Freedom to Move" policy proposals emphasize that, with some exceptions, market forces will be sufficiently strong to meet the transportation expectations of Canadian users. Government roles include safety and economic regulation, provision of subsidies to pay for uneconomic but socially necessary services, and provision of major segments of infrastructure for ports, airports, roads, and urban transportation.

The federal government also acts as a facilitator to bring together the many members of the transport sector whose concerted action is required to meet the challenges of innovation.

There are a number of federal mechanisms that can be used to stimulate Canadian innovation generally. These include tax incentives, directed R&D support programs administered by federal departments, and the programs of the National Research Council and the Natural Sciences and Engineering Research Council.

Transport Canada views transportation innovation activities in the context of

- Research for policy development: This includes socioeconomic and technical assessments to support policy decision making and as required for policy implementation. Examples include research on alternative fuels and tax incentives for innovation.
- Regulatory innovation: This pertains to the assessment of socioeconomic and technical needs of regulation. Examples include studies to determine the technological specifications required for framing dangerous goods regulations, fuel conservation standards, and regulations to govern operations in arctic marine environments.
- Innovation in transport services operated by the federal government: The products are improved systems, equipment, and operating methods for federally provided services. Examples include the development of equipment for the Canadian Airspace Systems Plan, icebreaker design and propulsion technologies, and marine radars and other navigation aids.
- Support for transport sector innovation: This covers innovation for enhanced transportation safety, efficiency, and productivity. Examples are federal-industry R&D programs for aerospace and rail freight. Other goals of this work reflect energy needs, industrial and regional development needs, and technologies for transport of the handicapped.

This definition of the purposes of innovation provides a yardstick for establishing institutional roles and responsibilities that can be and are related to the program mandates of federal departments and agencies and to their strategic and operational objectives. Definition of these, in turn, permits establishment of specific innovation strategies, operational R&D plans, and R&D programs. This is a complex process but one that becomes relatively easy to implement when agreement has been reached on what needs to be done.

The R&D efforts of transportation carriers vary by mode. In dollar terms, the Canadian railways are the largest performers with their efforts apparently concentrated on short- and medium-term R&D programs. The Canadian marine industry also mounts significant research programs. Highway transportation carriers and the airlines, although they seek to upgrade their performance through innovation, usually do so through acquisition of foreign technology. All modes are now exploring opportunities to meet requirements for improvements to management systems.

The Canadian equipment manufacturing industry is strongest in aerospace equipment, in urban transit technology, and in serving Canadian railways and marine carriers. There is strength also in some areas of the automobile parts manufacturing industry and in manufacturing buses. In these areas of strength, the benefits obtainable through early dialogue between carriers and manufacturers on future needs is beginning to be recognized, although much remains to be done. Both the department's proposed investments in aeronautics equipment and systems and the railways' proposed investment in advanced train control systems offer potentially large markets for Canadian equipment manufacturers.

Academic involvement in meeting the transportation sector's needs for innovation is presently largely restricted to socioeconomic research and systems analysis. This pattern is changing with, for example, the development of new working relationships with Memorial University related to the Marine Group's Arctic and East Coast Marine R&D programs. In addition, both the federal government and the railways have worked closely with the Canadian Institute for Guided Ground Transportation at Queens University. The department is also currently working with the Natural Sciences and Engineering Research Council to sensitize Canadian universities to the transport sector's need for innovation.

The interdependent roles of government, the private sector, and universities in Canadian transportation affairs as well as resource scarcity have dictated a collective and mutually supportive approach to innovation. This is not to say that a government department or a carrier does not fund R&D independently in support of its own objectives. It does mean that a conscious effort to identify collective R&D objectives and priorities is made, and advantage is taken of opportunities for joint or complementary R&D programs. A variety of formal and informal linkages exists. These include formal mechanisms for interdepartmental cooperation on transportation R&D at the federal level; modal R&D advisory boards and committees that provide for discussion of transportation R&D among all levels of government, the private sector, and universities; and international agreements, for example, the Canadian-U.S. Volpe-Jamieson agreement.

EXPERT SYSTEMS APPLICATION STRATEGY

The previous sections have provided a description of the context within which the department plans and programs R&D and of the department's roles in innovation. In cooperation with other members of the sector, the department continuously and systematically assesses the contribution that innovation can or should make to realizing objectives for increased safety and productivity. The public announcements by Japan, the United States, and members of the European Economic Community of new artificial intelligence programs triggered the development in Transport Canada of its current expert systems strategy.

Broadly speaking, the regulation, operation, and maintenance of modern transportation systems break down into a hierarchy of interdependent but nonetheless discrete functions. Equipment and systems controls are built on a modular basis. Modern hardware and software permit each module to be controlled by its own microprocessor with human intervention required on an exceptional rather than a continuous basis. This permits greater equipment and system productivity and safety; it also permits greater human productivity. This latter impact has been perceived as a threat. On the contrary, it creates time for operators to enrich their working experience. It does mean, however, that for the foreseeable future strategies for the introduction of advanced computer-based technologies must recognize

- The attainment of specific objectives for increased safety and productivity and
- The psychological impediments to introduction of the powerful new technologies and, the other side of the coin, the need to define strategies to enrich the jobs of the users of these technologies.

In this context, and as noted in the introduction, expert systems are one of the first commercially applicable derivatives of some 30 years of research in artificial intelligence (AI). Other emerging product areas include natural language software, computer-aided learning, and voice recognition. There are many aspects to AI research; however, it may be characterized as

- The part of computer science concerned with designing intelligent computer systems (i.e., systems that exhibit characteristics that humans consider intelligent);
- A branch of computer science the objective of which is to endow a machine with reasoning and perceptual capabilities; and
- The area of computer science that deals with problems that are incomplete in nature or that have indefinite solutions.

These definitions of AI reflect the academic nature of the work and mark the tremendous interest, matched by commitment of enormous sums of money, that has arisen in the field. The Japanese plan to develop a fifth-generation computing system with an architecture heavily dependent on a variety of technologies embodying concepts usually termed AI. For both military and civilian purposes the United States and countries in Western Europe have mounted similar projects.

Aside from the long-term military and civilian objectives that are being pursued under these programs, two applications of AI are now in the marketplace. The first is marked by the development and introduction of robots and computer-aided design/computer-aided manufacturing (CAD/CAM) that have revolutionized manufacturing processes, notably in the automobile industry. The second derives from advances in knowledge representation and expert systems that are reflected in the variety of medical and other diagnostic systems that has recently entered the marketplace.

An expert system (ES) is a computer program that uses knowledge and inference procedures to solve problems that are so complex as to require significant human expertise for their solution. The knowledge necessary to perform at such a level, plus the inference procedures used, can be thought of as a model of the expertise of the best practitioners in the field.

There are three categories of clients for an expert system:

- Clients who require answers to problems. These are the operators who will require answers in time frames dictated by the nature of the problem. They may range from a dispatcher who is controlling a demand-responsive system, to a scheduler, a maintenance mechanic, a maintenance manager, or a planner. Response time may vary from a requirement for a real-time solution to hours for complex problems.
- Clients who are attempting to improve the quality of the system, to increase its knowledge, or to refine and hone their own expertise. A variant of this would be to use the system as a research assistant.
- Clients who are students taking advantage of the system to upgrade their knowledge and skills.

In summary, an expert system may be used as a decision aid, to transfer knowledge or expertise, to improve the efficiency of an expert's use of time, to improve the quality of an expert, or as a training tool.

The definition of objectives, priorities, and strategy for application of expert systems to transportation reflects in part the concepts of user need for innovative response that are applied on a multimodal and modal basis by Transport Canada's R&D managers. However, at an early stage, two other factors were recognized that have influenced Transport Canada's approach. The first, a scarcity of expertise and resources, requires a centralized focus for work until such time as expert systems capability has been built up in all modes. The second reflects the principal characteristics of expert system technology: it is a humanizing rather than a deskilling technology. Strategies for initial applications of expert systems must be oriented toward enhancing human capability and skills.

In recognition of these factors and of the pragmatic need to build awareness and acceptance of the technology through successfully attaining limited objectives, the department has begun to evolve its expert systems R&D strategy in the context of applications in the following areas:

- Vehicle crews,
- Vehicle maintenance,
- External control of vehicles in transit,
- Permanent way control and operation,
- Terminal facility control and operation,,
- Operational planning and regulation, and
- Strategic planning.

Within each of these areas, and in the two functional areas of administration and regulation that parallel each of these areas, there are classes of jobs, each of which requires its own expertise and each of which should or would benefit from an "expert" assistant.

Preliminary assessments have been made of applications of ES and of some of the further work required to achieve these applications. For example, in the area of on-board systems for vehicle crews, the principal functions of the on-board crew are command and control. These require two main supporting functions, navigation and communications. Secondary functions include engineering (diagnosis and repair of vehicle systems) and cargo monitoring and service. Except in military applications, it appears unlikely that ES will replace humans in these functions in the foreseeable future. Even if it were technically feasible, it would likely be psychologically and sociologically unacceptable. Expert systems will therefore be used to perform trouble-shooting and advisory roles in the on-board environment—roles for which they are particularly well suited.

This suggests that any on-board expert system to be used by the primary operator (captain) of a vehicle in transit will require the successful solution of two problems over and above the technical details of the specific application. These are suitable input-output mechanisms to enable the system and its user to communicate with each other and suitable input-output mechanisms by means of which the system can sense and directly control its environment. The human interface typically considers input and output separately and requires

- A simple mechanism by means of which the operator can consult and direct the expert system without being distracted from the principal task of controlling the vehicle. A computer keyboard is clearly unacceptable, although a limited set of specific function keys may be marginally acceptable, especially in vehicles that require the operator to qualify via a full-time course of instruction. If the input mechanism requires some form of graphics (e.g., icon manipulation, a schematic overlay, or a map), some form of pointing mechanism will almost certainly be required. Current candidates range from digitizing tablets and light pens through touch screens to track-balls and mice. Touch screens and track-balls are probably the only present-day devices that could possibly be integrated into a vehicle operator's control environment.

One of the most promising approaches for textual and command input at present is voice input. Systems such as the Votan VPC 2000 already offer continuous speech recognition of a trained repertoire of command words at approximately $3,500 Canadian. Speech is the most effective vehicle for communication, and humans can readily adapt both vocabulary and speech patterns to match the comprehension level of their audience. Satisfactory interface with an expert system can almost certainly be implemented using a carefully tailored natural language subset.

- A simple mechanism by means of which the expert system can communicate with the operator. This involves a number of issues because some output will be textual, some will be nontextual (e.g., graphic, tabular, numeric), and some may be direct connection to other on-board systems. In a military environment the Head-Up Display (HUD) has proved to be the most effective, albeit costly, solution. The HUD projects a mixture of text and graphics on the inside of the prime viewport, allowing the operator to focus on it or look through it. The technology is currently too expensive to be practical in any but the most sophisticated of vehicles.

An alternative approach is to provide a small display screen, which must be integrated into the control panel in such fashion that the operator is not distracted from the principal control operations when assimilating output from the expert system. Voice synthesis is useful only if the information output is essentially textual in nature; this is an area that requires substantial research, especially the synthesis of textual representations of nontextual data.

The development of mechanisms for direct sensing and control of other vehicle systems, be they electronic, electrical, mechanical, or any combination thereof, has until recently remained the province of engineers and has been largely ignored by the computing community. The problem of real-time sensing and control has been the province of military and process control applications, but the manufacturing sector is now also starting to develop the necessary technology. Typical examples from the automotive industry are electronic fuel-injection systems; engine diagnostic systems; antiskid braking systems; and seat-belt, door, and light status monitoring and warning systems. However, there is considerable research and development yet to be done to investigate how best to interface computer systems to vehicle systems for both monitoring (input to the computer system) and control (output from the computer system).

These two problem areas reveal that fundamental research is required. The human interface problem, usually referred to as the man-machine interface (MMI), requires research that must be directed by psychologists. This will almost certainly come out of the university environment. The machine interface problem requires development of both sensors and real-time computer interconnection mechanisms. This is most likely to be driven by the manufacturing sector of industry, with major focus provided by the military and aerospace sectors.

Possible application areas for on-board expert systems include

- Vehicle control,
- Vehicle system diagnostics and trouble-shooting,
- Navigation,
- Communications,
- Cargo monitoring and service,
- Emergency advice and reaction,
- Specialized on-board applications, and
- Procedural and regulatory advice.

The thrust of all of these systems will be to provide the crew with early warning of problems and advice on how to resolve them. The most difficult application area will undoubtedly be the implementation of on-board systems to cope with emergency situations. Primitive systems exist, for example, fuel-low and brake-failure warning devices in automobiles. More complex systems are in the early stages of development, such as those that would give an airline pilot, for example, a diagnosis of the highest priority problem and outline proposed action. Voice output from the expert system will almost certainly be required, as will research in behavioral psychology both to establish human performance under stress and to devise the most appropriate communication mechanisms. A tricky conceptual problem relates to the correctness of the expert system response in comparison with what the operator "knows" is best. Emergencies by definition invalidate rules. Nevertheless, the expert system can provide an invaluable contribution to problem diagnosis and alert the operator to the required procedure or take action in the event of operator incapacity.

The application of on-board expert systems to the provision of procedural or regulatory advice, or both, will be an early target area. Such systems could, for example, provide advice on procedures at border crossings—tariffs, vehicle configuration, and so on. When married to central office dispatching, scheduling, and routing systems, expert systems will provide an immediate boost to productivity, notably for trucking operators and urban logistic and paratransit fleets. Canadian work in the field will build on work pioneered by the Ontario government, for example, the Computerized Goods Transportation Information System.

Systems for on- and off-board vehicle maintenance have similar characteristics in that they are required to both diagnose the problem and identify remedial action. Both on- and off-board systems require an embedded system capability and a capacity to interface either automatically or on demand of the user.

The off-board systems in particular will require presentation of diagrams, many of which will be extremely detailed. This implies a capacity for high-resolution graphic display and gigabyte (109) storage capacity. The General Electric Diesel Electric Locomotive Troubleshooting Adviser (DELTA) uses an optical disk attached to a dedicated microcomputer system with high-resolution color graphic display. Preliminary areas selected on the basis of their different knowledge base requirements include major structure (hull) maintenance, engine and power train maintenance; control, guidance, and communication system maintenance; auxiliary equipment maintenance, and procedural and regulatory advice.

Systems for external control of vehicles in transit can be divided into two categories: remote control of autonomous or semiautonomous vehicles and control, mainly procedural, via communications links of crewed vehicles. Application areas include vehicle routing and scheduling, traffic control, emergency procedures, weather forecasting, and procedural and regulatory advice. Applications of current computer technology are already widespread in these areas; there is a good understanding of the rules and a good data base that should permit fairly fast introduction of expert systems in this field.

The paratransit vehicle dispatch system being demonstrated now in Vancouver provides an example of the benefits of application in this field. Paul Tuan at Stanford Research Institute (SRI) has pioneered a novel expert system approach to solving this problem by developing a powerful scheduler's workstation that is initially installed in an inexpert form (with background knowledge of the geography and vehicle fleet characteristics) and is capable of learning from the user by observation and analysis. As part of its design, the system has a range of optimization algorithms embedded within its basic mechanism. After some 6 months, the workstation will have gained sufficient expertise that it can handle 80 percent or more of the scheduling situations on its own, at which point it can be networked with a human scheduler. Human schedulers are required to handle unusual situations, which the expert system detects and hands on. The advantage of this approach is that the peculiarities and idiosyncrasies of any system can be accommodated and that, when the system has reached an acceptable level of competence, it can be replicated for the price of duplicating the knowledge base.

Because scheduling is such a skilled job, and because it is much more an art than a hard science, it is an excellent candidate for expert system applications.

Application to permanent way control and operation is foreseen in the following categories: scheduling and capacity planning, communications and control, facility diagnosis and maintenance, weather forecasting, usage accounting, emergency procedures, and procedural and regulatory advice. Systems for operational planning and regulation would include traffic forecasting, financial planning and control, vehicle design planning and configuration, transitway planning and configuration, terminal facility design planning and configuration, communication and navigation system design planning and configuration, emergency procedure planning, and statistical regulatory advice. Applications in these latter areas will be based on the transportation simulation modeling techniques and capability already in existence. The track-train dynamics simulation models developed in Canada provide one example. Another is the Bay Area Rapid Transit model developed at SRI by Bjorn Conrad and Tony D'Esopo. This latter model has proven to be capable of supporting both operational decision making and system planning and development. These are algorithmic solutions from which an ES could evolve.

All of these applications have been designed to improve the performance of experts in the exercise of their daily functions. Because of the psychological risks, as well as the need for further refinement of technology, effective applications in the on-board vehicle category will be most difficult to achieve; conversely these are the areas in which public concern for transportation safety requires immediate action. Immediate action means an immediate start. Immediate results mean the development and implementation of an initial family of "toy" but nonetheless worthwhile applications in the next 2 or 3 years. In the concluding sections of the paper will be discussed how Transport Canada intends both to learn by doing and to systematically build up awareness and competence to reach the potential of the applications discussed here.

APPLICATION TO TRANSPORTATION

The results of preliminary assessments of the relevance of the application of ES to transportation were outlined in the previous section. In this section a methodology for evaluating the suitability of an application is discussed. The principal factors that are recognized in Transport Canada's current strategy are awareness, utility, feasibility, and cost.

Awareness

Media coverage of major research initiatives in the AI field has resulted in both a general understanding of the power and capabilities of advanced computer technologies and a more visceral, potentially luddite, concern about their introduction. Clearly both relate to broader public concerns about transportation safety, job security, and the future of Canada's economy and society. Although not precluding research in any area, this means that Transport Canada's selection of fields for application of ES will focus on enhancement of an individual's skills. Another aspect of this is the need to build an understanding within the transport community of the power of these technologies, of the constraints to their application, of the resources that are required, and of the commitments that must be made to achieve application.

Utility

This requires assessment of the marketability of a given application (that is, a user's readiness to adopt ES) and its impact in traditional cost-benefit terms and in terms of the social or cultural impacts of a particular ES.

Feasibility

Some potential applications are more amenable to implementation than others on purely technical grounds. Areas in which ES applications currently enjoy success include medical diagnosis, computer system configuration, resource exploration, and intelligent computer-aided design. These expert systems tend to operate in relatively static fields, in a relatively "safe" environment, with limited or no external control hierarchies, and with relatively long decision-making time frames. To be applicable to transportation, hardware must be "ruggedized." Other advances are also required. Advances in hybrid knowledge representation, distributed artificial intelligence, fault tolerant computing, on-line real-time and embedded AI systems, and knowledge acquisition will be required to permit expert systems to achieve optimum impacts in transportation.

Cost

The relative cheapness and power of modern computers permit the immediate application of expert systems. Depending on the area of application, ES can be built on a variety of computers. These include the PC/AT that costs less than $5,000 Canadian and ranges to some of the specialized hardware developed by Symbolics, Texas Instruments, Xerox, and others costing from $15,000 to $150,000 (U.S.) up to VAXen mainframes and supercomputers costing millions. The smallest practical system will probably involve some 200 to 500 rules, require at least 6 months to develop, and involve 30 min of expert time and 8 hr of knowledge engineering time per rule. It was believed, until recently, that the lack of availability of skilled knowledge engineers might prove an insurmountable barrier to rapid application of the technology (opening knowledge engineers' salaries are on the order of $70,000 U.S.). There is some evidence, however, that the skills of current computer professionals can be upgraded to a degree that will allow the problem to be tackled. Nonetheless, the demand on the Canadian ES industry already overloads its ability to produce and will constrain what can be achieved in the next 5 to 10 years. Of equal relevance is the ability of the client to assess the technology and fully use it. ES strategy must provide for in-house training and development in parallel with building of initial ES applications.

TRANSPORT CANADA'S EXPERT SYSTEMS STRATEGY

Four factors affect how the department has evolved its ES strategy since it first moved into the field some 2 years ago:

- During the past 10 years there has been an evolution in transportation management and planning processes toward systems analysis and integration. As the technical limits of existing vehicles, ways, and terminals were approached, a search was carried out to find means to improve scheduling, communication, control, and maintenance technologies and procedures. This led to the widespread introduction in transport of computer technologies, to recognition of the rules governing efficient and effective transport operations, and to a better understanding of the expertise involved.
- It was readily apparent that this evolution in transportation management and planning philosophy and practice was amenable to application of expert systems. Indeed, adoption in air and railway modes of advanced computers for scheduling and control; built-in sophisticated microprocessors in automobiles, aircraft, and locomotives; and the technologies and concepts involved in the Canadian National Railroad's hump yard improvement program reflect the state of the art of application of third- and fourth-generation computers. They permit assessment of both the benefits and the costs of applying the next step, expert systems.
- There has been an evolution of ES directly applied to or applicable to transportation. These include DELTA and SPILLS. The latter is a generic name for a set of prototype expert systems designed to assist in the location, assessment, and cleanup of hazardous spills and to train personnel who must deal with such matters.
- A specific problem, one representative of many transportation problems, is the need both to make use of the volume of data and information being generated by research, in this case, by the Vehicle Weights and Dimensions program (a multimodal federal–provincial–private sector R&D activity designed to establish technical specifications for trucking in Canada), and to capture the expertise of highway experts in Canada.

It is fair to state that although these four factors collectively motivated the department's first steps in the field, the internal coalescence was reinforced by growing interest throughout the federal government in AI and, of course, by representatives from the private sector and the academic community.

The following factors shaped the department's ES strategy:

- Applications need to be focused on improving individuals' skills.
- Development of applications of ES is proceeding apace around the world. Regardless of any Canadian efforts in this field, the international nature of virtually all transport operations, equipment, and systems requires Transport Canada to understand what is going on and to develop an ability to amend its own operations and regulations accordingly. "Who has legal liability for failure of a smart car?"
- There is acceptance at the technical level of the benefits of ES application, but, in order to convince senior management, some benefits from R&D in the field have to be demonstrated within the first 12 to 18 months of an ES strategy.
- Development of a capability to fully use ES will take 6 months to 5 years for the delivery of a "finished" AI application. Although an individual expert will be trained as the system is built, there are compelling reasons to start small and learn by doing.
- There is a good base of PC/AT expertise in the department and more broadly within the transport community on which the first limited applications of ES can be built.
- In practice, transportation R&D planning and management are done on a consultative and cooperative basis by all levels of government and the private sector. This reflects both joint or complementary interests in the results of R&D and the

scarcity of resources. In addition to developing in-house capability in ES, it is also necessary to carry out a parallel effort to build awareness and, where possible, engage in joint or complementary ES R&D with this broader community.

STRATEGY

The department strategic expert system goal is, by 1990, to achieve a capability to use ES applications effectively in support of the department's program objectives and to stimulate applications of ES more broadly throughout the transport community.

This strategy is to be implemented by

- Further development and refinement of existing analyses of the applicability of ES to transportation and, where appropriate, conduct of R&D projects to fill in gaps in understanding;
- Development of papers and conduct of workshops to create awareness of ES applicability to transportation and as a device to both generate critical assessments of the department's activity and identify opportunities for joint work in the field;
- Development of limited ES applications (200 to 500 rules) in each modal area of transportation and of complementary training and educational activities to promote effective use of these applications; and
- Development of the capability and expertise to develop or contract for the development of more complex expert systems of up to 5,000 rules after 1990.

It should be noted that although this is a Transport Canada strategy, directed primarily at the Canadian transport community, elements of this program are clearly international in scope. Its conceptualization and at least one major project reflect cooperation with SRI in Palo Alto, California. As Canadian-U.S. cooperation on aeronautics R&D increases, joint ES projects with the United States may be developed. The same will be true for ES applications in all modes.

The department is now well embarked on a 2-year program that represents the first phase of the strategy. Activities and their results to date follow.

- Awareness: A 1-day workshop on applications of ES to transportation was held in Montreal in September 1985. The proceedings of this workshop are available and may be obtained by contacting Transport Canada, Tower C, Place de Ville, Ottawa, Ontario K1A 0N5, Canada. Those who attended included government, private-sector, and university members of the transport community and representatives of the Canadian AI community.
- Heavy vehicle configurations: A contract has been let to develop a feasibility study of the applicability of ES to the management of vehicle weights and dimensions data and information. This contract also provides for development of a prototype ES (200 rules).
- Northern airspace system management: A contract has been let to develop an ES for this purpose. In addition, further work is under way to investigate the applicability of ES to assisting air traffic controllers.
- MV Caribou: A contract has been let to determine the feasibility of an ES fuel-monitoring system on this ferry. In addition, further work is under way to assess the feasibility and utility of developing other ESs to provide expert advisor capability for the navigation of this vessel.
- Demand-responsive transportation: As discussed earlier in this paper, an ES has been developed to assist in the scheduling of demand-responsive transport services. It is on display at Expo '86.
- Transportation of dangerous goods: An algorithmic model is being developed to cut down the response time and improve the quality of responses to requests for information telephoned in to Transport Canada's emergency center. This again can be evolved into an expert system.

The resources committed by the department to this work in FY 1985–1986 amounted to some $175,000 and involved commitment, at varying levels of effort, of some 10 professionals. In 1986–1987, the department expects to spend some $500,000 in this area, mostly for contracts with the private sector.

SUMMARY

The challenges that face transportation over the next decades and the innovation activities that are being carried out by the department and other members of the transport community to address these challenges have been discussed.

R&D programs to address these challenges are established in the context of the degree to which innovation (new equipment, new systems, new procedures) can make a contribution to transportation objectives. Transport systems are both "technologically intensive" and "people dependent"; these characteristics heighten the attraction of a technology, expert systems, that can materially increase the skills and capability of the work force.

The department's strategic objective is thus to stimulate application of, and apply, expert systems as aids to increasing on-the-job safety and productivity. As in any new technology, advances will be required. Components will have to be made more rugged, and technical developments will be required before optimum advantage can be realized. Nonetheless, the promise of this family of technologies, and the success of even the preliminary applications, has resulted in a departmental commitment to "get started" and to "learn by doing."

Applications of expert systems are foreseen on board vehicles and in maintenance, external-to-the-vehicle and way-control systems, terminal operations and planning, and regulation. Eventually, applications are expected in virtually every aspect of a transportation operation.

The department's current and medium-term R&D activities in this field have been outlined. There are two aspects to this work: First, to stimulate awareness of and seek agreement on applications of expert systems to transportation. Second, to develop limited applications in a number of selected areas and, in so doing, both realize benefits from the applications and lay the base for further work in this field. These first applications require a limited number of rules, are amenable to application from the existing base of skills, and are based on available technology.

The department's objectives for work in this area beyond 1990 will in large measure depend on the lessons learned from this relatively low-cost venture. The impact of these technologies on human resources, and the dynamism of expert systems technology, precludes establishment of medium- (past 1990) and longer-term objectives in this field. Experience gained during the next 5 years is absolutely necessary before such objectives can be specified meaningfully.

As in other fields of transportation R&D, the department is developing its plans and implementing R&D projects in cooperation with other members of the Canadian transport community. The department's operational and regulatory roles are inextricably linked to other members of the transport community through technology. Scarcity of resources, particularly in this field, also makes cooperation, as reflected in a number of the expert systems projects, absolutely mandatory.

The results of a workshop on the topic, attended by representatives of governments, the private sector, and universities, confirmed the department's assessment of the need to mount a concerted effort to achieve productivity gains and increased safety through application of expert systems.

Using Expert Systems To Select Traffic Analysis Software

EDMOND CHIN-PING CHANG

An experimental expert system was developed by the Texas Transportation Institute to assist users in selecting computerized software packages currently being supported by the FHWA. This system was designed to investigate potential expert systems applications in transportation engineering. This study was performed to serve three basic purposes: to test the feasibility of developing a small-scale traffic engineer knowledge-based expert system using a simple knowledge engineering tool, to develop an alternative method of recommending computer programs for user-specified applications, and to investigate a possible approach for implementing advisory expert systems to be operated in the IBM PC/XT/AT microcomputer environment. The development of a prototype expert system using a commercially available knowledge engineering tool developed by Level Five Research Incorporated is described. INSIGHT 2+ was used to experiment with expert system programming in the inexpensive microcomputer environment. This system reviews and analyzes user-input information, evaluates it with various reasoning paths, and offers a conclusion. With the proper combination of knowledge programming tools and preidentified decision-making processes, individual users can develop their applications faster than if they had to learn complex artificial intelligence programming languages. It is recommended that the expert advisory system design concept of this prototype model be extended to assist practicing traffic engineers in selecting software packages to optimize traffic control strategies. With proper improvements, this type of expert system design can assist the user as a stand-alone expert advice system.

Several traffic engineering programs are currently being supported by the FHWA *(1–5)*. A prototype expert system was developed at the Texas Transportation Institute to assist in the selection of these microcomputer software packages. The intent was to apply the expert systems concept to assist individual users in selecting computer software for optimum traffic analysis *(6–10)*. This system was also used to *(a)* investigate the potential feasibility of using expert systems in traffic engineering and *(b)* computerize the expert system's advice through artificial intelligence techniques *(11–16)*.

The development of a prototype expert system is described. This study was performed to serve three basic purposes: to test the feasibility of developing a small-scale traffic engineer knowledge-based expert system using a simple knowledge engineering tool, to develop an alternative method of recommending computer programs for user-specified applications, and to investigate a possible approach for implementing advisory expert systems in the IBM PC/XT/AT microcomputer environment. INSIGHT 2+, a commercially available knowledge engineering tool developed by Level Five Research, Inc., was selected, because of the simplicity of its implementation, to develop experimental expert systems in the IBM PC/XT/AT microcomputer environment *(17, 18)*. Normally, LISP- or PROLOG-based expert systems are adequate for customized problem solving, but knowledge engineering tools such as the INSIGHT 2+ system can allow users to quickly represent specialized knowledge by following predetermined guidelines.

This prototype TTI-FHWA expert system reviews and analyzes the information given by the user, evaluates it with various paths of reasoning, then offers a conclusion for a particular application *(19–21)*. The technical information was collected from various traffic engineering computer programs available from or being developed by the FHWA. For each program package, information is processed on the basis of the user's inquiry, expected performance, development status, hardware requirements, and the available information source. This study is intended to provide an advice system for recommending suitable traffic engineering software. It is a stand-alone expert advice system that uses a microcomputer.

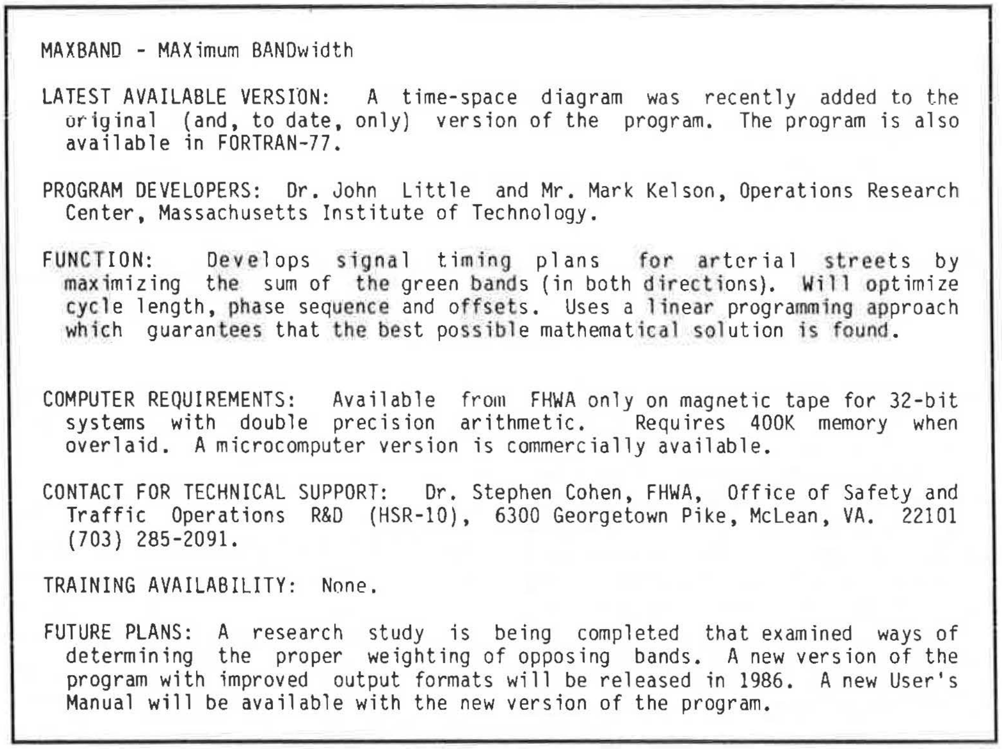
TRAFFIC ENGINEERING SOFTWARE

As urban traffic demands increase, the most efficient coordination is required between existing traffic control devices and proper signal-timing settings. A large number of computer software packages have been developed to provide better traffic analysis. Microcomputers are increasingly available, and numerous programs are available for traffic engineering applications. Without having to access a mainframe computer, a traffic engineer at virtually any location can routinely and efficiently analyze traffic engineering problems and thereby have more time for innovative engineering analysis. As more traffic professionals begin to apply these traffic engineering–oriented computer packages, programming development is being emphasized at the federal, state, and research community levels to enhance the problem-solving capabilities of these packages. However, because of the numerous enhancements being made in each program, it is hard to keep track of developments in each of these traffic analysis packages. Therefore there is an increasing need for an expert advice system to assist the end user in selecting proper tools for specific kinds of analysis.

The traffic engineering software packages analyzed are supported and maintained by the Systems and Software Support Team of the FHWA. The technical information used is based on

---

Texas Transportation Institute, Texas A&M University System, College Station, Tex. 77843.



MAXBAND - MAXimum BANDwidth

LATEST AVAILABLE VERSION: A time-space diagram was recently added to the original (and, to date, only) version of the program. The program is also available in FORTRAN-77.

PROGRAM DEVELOPERS: Dr. John Little and Mr. Mark Kelson, Operations Research Center, Massachusetts Institute of Technology.

FUNCTION: Develops signal timing plans for arterial streets by maximizing the sum of the green bands (in both directions). Will optimize cycle length, phase sequence and offsets. Uses a linear programming approach which guarantees that the best possible mathematical solution is found.

COMPUTER REQUIREMENTS: Available from FHWA only on magnetic tape for 32-bit systems with double precision arithmetic. Requires 400K memory when overlaid. A microcomputer version is commercially available.

CONTACT FOR TECHNICAL SUPPORT: Dr. Stephen Cohen, FHWA, Office of Safety and Traffic Operations R&D (HSR-10), 6300 Georgetown Pike, McLean, VA. 22101 (703) 285-2091.

TRAINING AVAILABILITY: None.

FUTURE PLANS: A research study is being completed that examined ways of determining the proper weighting of opposing bands. A new version of the program with improved output formats will be released in 1986. A new User's Manual will be available with the new version of the program.

FIGURE 1 Example of material used for MAXBAND model *(1)*.

the manual released by the FHWA in 1986. Figure 1 is an example of the material used in the MAXBAND program. Basically, these computer software packages include traffic signal–timing optimization programs, traffic flow simulation models, and other traffic engineering computer software. These computer programs and their particular areas of application fall under the following three categories *(1)*:

1. Signal-timing optimization programs *(2, 3)*
   - SOAP—Signal Operation Analysis Package
   - MAXBAND—Maximum Bandwidth Optimization
   - AAP—Arterial Analysis Package
   - TRANSYT-7F—Traffic Network Study Tool
   - SIGOP-III—Signal Optimization Model
2. Traffic flow simulation models *(4)*
   - NETSIM—Network Simulation Model
   - TRAFLO—Macroscopic Urban Network Model
   - FRESIM—Freeway Simulation Model
   - ROADSIM—Two-lane, Two-way Rural Road Model
3. Other traffic engineering software *(10)*
   - ITDS—Integrated Traffic Data System
   - HIGHWAY CAPACITY—1985 Highway Capacity Manual Software
   - PPD—Platoon Progression Diagram
   - COUNTS-PC—Signal Warrants Analysis
   - LINKOD—Origin-Destination Table Synthesis

EXPERT SYSTEMS DESIGN

An expert system is a collection of computer programs or systems that applies in a specialized domain. The expert system combines both problem-solving and knowledge-support components for specific applications. The expert systems support environment helps the user interact with the program to specify the requirements and the specialized expertise of an expert in the field *(11–13, 19, 20)*.

There are five major components of artificial intelligence (AI) applications in expert system (ES) designs: *(a)* expert system, *(b)* domain expert, *(c)* knowledge engineer, *(d)* expert systems building tool, and *(e)* end user. Figure 2 shows the basic AI/ES components and their relationships to each other *(13)*. The domain or area expert is an articulate, knowledgeable person with a reputation for producing good solutions to problems in a particular field. The knowledge engineer is usually a person with a background in computer science and AI technology who knows how to build expert systems. The knowledge engineer interviews the domain experts, organizes the knowledge, decides how it should be represented in the expert system, and may assist in development of a specific program. The expert systems building tool is the computer-programming environment and language used by the knowledge engineer or computer programmer to build the expert system. The user or the end user is the person for whom the expert system is developed.

As indicated in Figure 2, the user may be a traffic engineer debugging the expert systems building tool or language, a knowledge engineer refining the existing knowledge in the system, a domain expert adding new knowledge to the system, the end user relying on the system for advice, or clerical personnel adding more information to the knowledge engineering data base. A knowledge engineer converts a domain expert's specialized knowledge into sets of IF-AND-THEN-ELSE rules using instructions that a computer understands. However, no matter what software or hardware the expert system has to use, the knowledge-based expert system ultimately has to be implemented on computer hardware in a

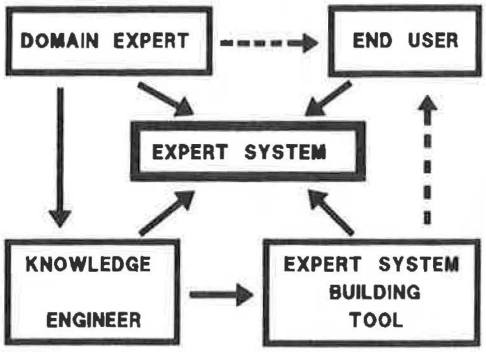


FIGURE 2 Main components of expert systems design.

most primitive machine language format. Most conventional programming is done in high-level languages, such as BASIC, COBOL, FORTRAN, PASCAL, LISP, or C. AI languages are used in ES designs for processing user-input information to derive conclusions and recommendations. Problem-solving AI languages such as LISP and PROLOG are often used.

The evolution of AI applications is shown in Figure 3 *(12)*. As indicated, a more application-oriented research trend has recently become evident. AI/ES programming development can be separated into three areas: expert systems tools, natural language queries, and AI languages. Normally, AI programming is done in LISP and PROLOG. LISP (LISt Processing) is particularly suited for symbolic and numeric processing for decision analysis. LISP is most suitable for manipulating lists of symbols (i.e., strings of numbers or words, or both). For years, LISP has been preferred by AI engineers in the United States. On the other hand, PROLOG (PROgramming in LOGic) is preferred in Europe and Japan. PROLOG contains structures more suitable for writing programs that evaluate logical expressions, whereas LISP contains operators that facilitate the creation of programs that manipulate lists for representing specific expert knowledge.

KNOWLEDGE ENGINEERING TOOLS

Knowledge engineering shells or tools are often used to build expert systems. These tools provide all of the features needed in an expert system, such as help functions, windowing capabilities, graphics support, and other functions, to help the knowledge engineer add the information from the domain expert. The knowledge shell usually includes an explanation subsystem that describes the logical steps needed to reach a conclusion. The natural language interface can further help explain these programming development steps in ordinary English to enhance understanding of the decision-support process. Symbolic ES operations using LISP and PROLOG usually take up a lot of computing memory and thus may be executed slowly, particularly on microcomputers. The symbolic operations performed in LISP or PROLOG are usually implemented in more efficient computers using LISP as the operating system.

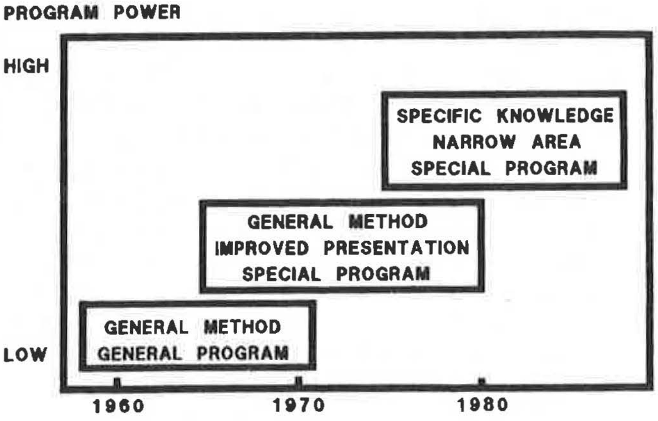


FIGURE 3 Evolution of AI research trend.

Knowledge engineering tools allow users to develop a prototype of a defined problem quickly and develop their own customized applications in less time than it takes an AI programming language. However, PC-based expert system development tools are not suitable for large-scale ES application. The common practice for developing expert systems is to obtain a commercially available LISP machine to use fast-executing knowledge engineering tools, such as ART, KEE, Knowledge Craft, or EMYCIN, on a VAX-type super-mini-computer for experimental program development *(13, 14)*. Then the developed expert system can be transferred to generate ES programs for practical applications that may eventually be run on a personal computer in a microcomputer environment.

INSIGHT 2+, developed by Level Five Research, Inc., is a microcomputer-based tool that allows prototype ES programming *(17, 18)*. It is used to apply knowledge, form conclusions from facts, and solve problems in small-scale applications. Unlike ordinary data base systems that merely store, organize, and recall information, the INSIGHT 2+ expert system reviews and analyzes the information given, evaluates it using various paths of reasoning, and offers recommendations. INSIGHT 2+ provides a programming environment for the design, creation, and use of knowledge systems. INSIGHT 2+ permits the use of natural language to develop knowledge data bases and to interact in an IBM PC/XT/AT microcomputer environment.

INSIGHT 2+ can be used to implement user expertise for developing problem-solving techniques. Because it can make inquiries and maintain a knowledge base, expert systems developed using INSIGHT 2+ may enhance the end user's ability to analyze a problem and achieve suitable solutions. For example, by accumulating the answers and probability of success from the user's trial-and-error process, the expert system can summarize the modified solutions. During program execution, INSIGHT 2+ automatically questions the user for information for better conclusions. Using the expertise obtained from the user and programmed in its knowledge base, INSIGHT 2+'s inference engine is able to reason, even from incomplete or uncertain information. By asking the user to specify a confidence value, INSIGHT 2+ can further evaluate the viability of a path of reasoning or a chain of rules depending on the probability of a certain line of reasoning. At any point during a consultation, the user can request an explanation of the current reasoning status or, optionally, wait until the conclusion of a session and obtain a complete trace report. The functional structure of the INSIGHT 2+ system is shown in Figure 4.

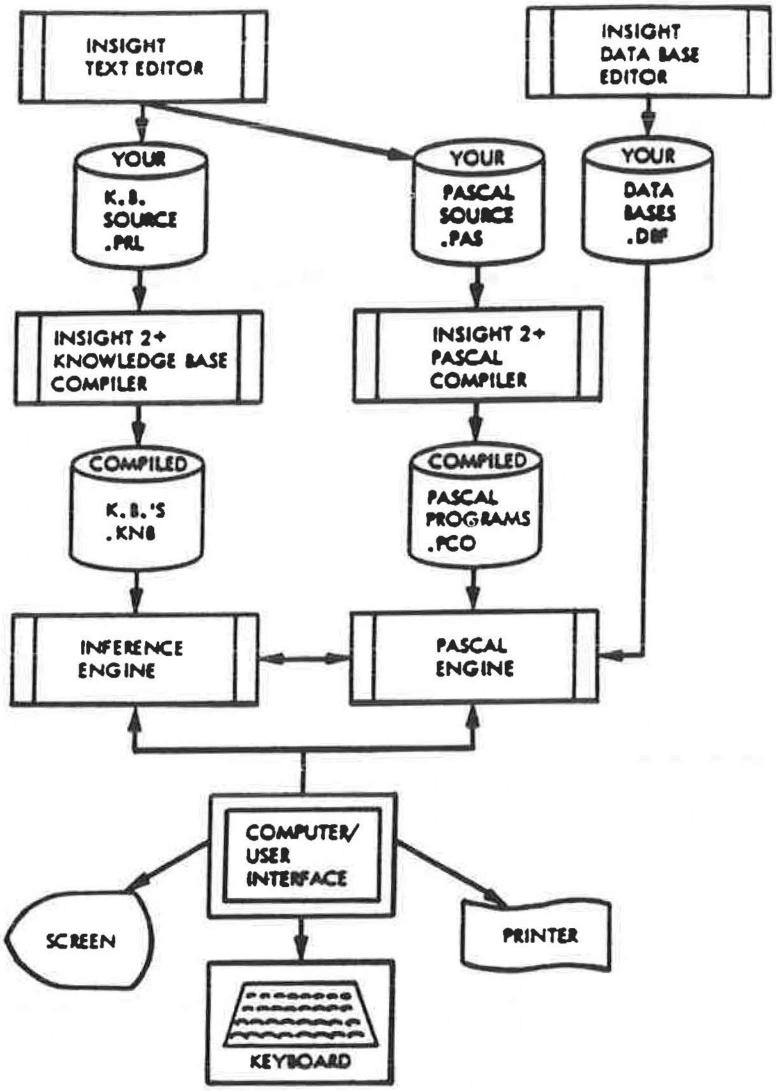


FIGURE 4 Functional structure of the INSIGHT 2+ system *(17, 18)*.

BASIC DESIGN PROCESS

A prototype expert system was implemented to provide computerized assistance in selecting traffic engineering software. In this automated selection guide, three main goals or categories of each computer software package are evaluated. These packages include traffic signal–timing optimization, traffic flow simulation, and other traffic engineering analyses. Each computer package is treated as a subgoal. The knowledge engineering tool summarizes the final recommendations of the traffic engineering software from the user-input requirements in this experimental system. The relevant selective information, such as software functions and computer requirements, was extracted from the information source to identify the conclusion of the production rule. It was then used to program the INSIGHT 2+ production rule language (PRL). The displayed information and the proper conclusions were further explained in this expert system.

In this section is described the basic design process used in preparing this prototype expert system for the selection of computerized traffic analysis software packages. The development process used in this ES design can be illustrated by the seven steps shown in Figure 5: extracting basic information, designing a decision table, setting up evaluation goals, selecting evaluation constraints, developing a PRL, debugging the PRL, and completing the program recommendations.

Extract Basic Information

Information used to develop this prototype expert system was extracted from three sources: *(a)* the second issue of the Traffic Software Users Awareness Report, *(b)* the BASIC program developed and demonstrated by the FHWA at the 1986 Annual Meeting of the Transportation Research Board, and *(c)* the working experience of the Traffic Operations Program of the Texas Transportation Institute. The Traffic Software Users Awareness Report, issued semiannually to all users who have received any of the FHWA's traffic engineering software, contains the latest information on FHWA software. The Systems and Software Support Team in the Office of Traffic Operations of the FHWA is responsible for distributing all of the transportation– and traffic engineering–related computer software.

To simplify the discussion, the traffic engineering software discussed here is classified in three categories. These FHWA-

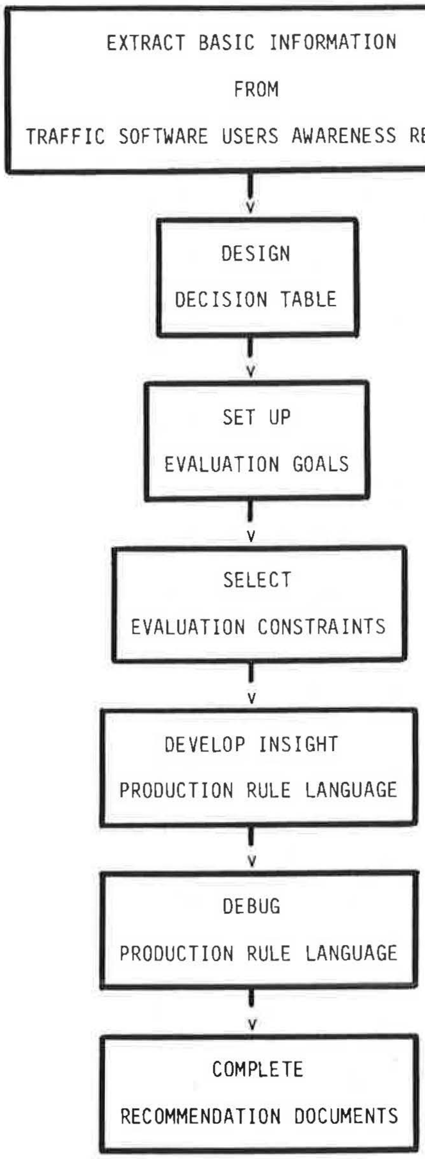


FIGURE 5 Basic design process.

supported packages include the traffic signal–timing optimization programs, traffic flow simulation models, and other traffic engineering analysis software. These traffic analysis packages include some of the popular traffic engineering tools, component models of the TRAF family, step-down versions of the transportation planning tools, and the Highway Capacity Manual software available at present. The relevant information for each of the traffic models was extracted from the Traffic Software Users Awareness Report and summarized. Summaries were based on careful reading of the article, comparing different traffic engineering software, and differentiating special characteristics according to engineering judgment and the criteria stated in the Traffic Software Users Awareness Report. On the basis of the results of the comparisons, the current FHWA-supported traffic engineering software packages were divided into three categories:

- Signal-timing optimization programs: Signal-timing optimization programs optimize the major signal-timing variables, such as cycle length, phase length, sequence, and offset. The transportation facilities analyzed may include isolated intersections, arterial streets, and networks. It should be noted that most of these computer programs, with the exception of MAXBAND, can also perform certain simulation functions for evaluating existing conditions.
- Traffic flow simulation models: Traffic flow simulation models are designed to simulate different traffic control strategies. These models can simulate various types of transportation facilities. Transportation facilities may include isolated intersections, arterial streets, networks, rural highways, and urban highways.
- Other traffic engineering software: Other traffic engineering software, supported by the FHWA, may perform specialized functions, such as computerized data base management, highway capability analysis, time-space analysis, traffic flow profile display, traffic signal warrant analysis, and origin-destination travel analysis.

After separating all of the relevant and irrelevant information from the available technical material, the domain expert traffic engineer modifies the design of the decision-making process according to his past experience with these models. He then determines what information should be emphasized for analysis and identifies factors to be used to evaluate the relative importance of criteria used in the practical design of the prototype expert system.

Design Decision Table

Decision table analysis is a decision-making aid that is used in design and evaluation *(20)*. Table 1 gives a simplified version of the decision table used for studying two basic design elements of this particular expert system. In the decision table, the horizontal components represent the main goals and subgoals defined in the expert system. Horizontal components on the first level in the table are the main goals. Horizontal components on the second level are the subgoals. The vertical components of the decision table represent the facts and rules, such as design constraints and potential application areas, that could be investigated by each of the traffic analysis packages.

Table 1 is an example of how the decision table was applied to analyze the MAXBAND program. The main goal and subgoal of MAXBAND are first identified by applying the backward-chaining concept. Then the major constraints of MAXBAND are separately identified. After the goal, subgoal, and design constraints are summarized, the production rules are specified for actual program development. The major advantages of using this type of decision table in the development of a practical expert system include the ability to

1. Summarize the basic relationships of different constraints.
2. Evaluate the requirements of independent constraints.
3. Study the detailed interrelationships among major variables in a systematic approach. (It should be noted that the original table used in the actual design of this prototype expert system is more complex than the one presented here.)
4. Provide the domain expert's knowledge and skill in completing the background information required in the decision-making process.
5. Set up evaluation goals, subgoals, and design constraints.

TABLE 1 SIMPLIFIED DECISION TABLE ANALYSIS

	SIGNAL TIMING OPTIMIZATION PROGRMAS					TRAFFIC FLOW SIMULATION MODELS				OTHER TRAFFIC ENGINEERING SOFTWARE				
	SOAP	MAXBAND	AAP	TRANSYT-7	SIGOP-III	NETSIM	TRAFLO	FRESIM	ROADSIM	ITDS	CAPACITY	PPD	COUNTS-PC	LINKOD
I. APPLICATION AREAS														
A. SIGNAL TIMING PLAN	X	X	X	X	X	X	X	X	X					
1. OPTIMIZATION	X	X	X	X	X									
a. ISOLATED	X													
b. ARTERIAL		X	X	X	X									
c. NETWORK				X	X									
2. SIMULATION	X	X	X	X	X	X	X	X	X					
a. ISOLATED	X		X		X	X								
b. ARTERIAL		X	X	X	X	X	X							
c. NETWORK			X	X	X	X	X							
d. FREEWAY							X							
e. URBAN FREEWAY								X						
f. RURAL FREEWAY									X					
B. PROVIDED FUNCTION	X	X	X	X	X	X	X	X	X			X		
C. DATA MANAGEMENT SYSTEM	X		X							X				
D. HIGHWAY CAPACITY ANALYSIS											X			
E. SIGNAL WARRANT ANALYSIS													X	
F. ORIGIN-DESTINATION PLANNING														X
II. COMPUTER REQUIREMENTS														
A. MAINFRAME	X	X	X	X	X	X		X	X					X
B. MICROCOMPUTER	X		X	X	X	X				X	X	X	X	

Set Up Evaluation Goals

The basic evaluation goal for this prototype expert system, as described earlier, is to provide specific recommendations about software packages to meet user requirements. The INSIGHT 2+ PRL program is constructed to meet this objective:

1. Program belongs to signal-timing optimization programs
    1.1 Program SOAP is recommended
    1.2 Program MAXBAND is recommended
    1.3 Program AAP is recommended
    1.4 Program TRANSYT-7F is recommended
    1.5 Program SIGOP-III is recommended
2. Program belongs to traffic flow simulation models
    2.1 Program NETSIM is recommended
    2.2 Program TRAFLO is recommended
    2.3 Program FRESIM is recommended
    2.4 Program ROADSIM is recommended
3. Program belongs to other traffic engineering software
    3.1 Program ITDS is recommended
    3.2 Program HIGHWAY CAPACITY is recommended
    3.3 Program PPD is recommended
    3.4 Program COUNTS-PC is recommended
    3.5 Program LINKOD is recommended

The INSIGHT 2+ program uses a set of outline-type evaluation goals with different degrees of recommended action coded as part of the prototype expert system. In this particular system setup, several things are noted. First, the definition of this goal is identical to the functional classification given in the section on traffic engineering software. Second, the purpose of each program package is identified as part of the goal definition. This arrangement indirectly implies the inclusion of the program categories as part of the decision rule. The other possible programming approach is to not classify these programs under three main goals but treat each program as a separate goal.

Select Evaluation Constraints

Evaluation criteria are made hard to identify by differences in traffic engineering evaluation constraints. The evaluation criteria used for this analysis were that each transportation-related computer analysis software package must be unique, identifiable, and classifiable.

1. Unique: The specified constraints should be adequate to describe the characteristics of the computer software to be analyzed.

2. Identifiable: The selection criteria should provide clearly defined characteristics for decision-making support, such as definable application areas and confident answers.

3. Classifiable: The common features selected in the analysis are the potential program applications, such as optimization, simulation, and other transportation-related features. The major application areas of these FHWA computer software packages can be implemented for isolated intersections, arterial streets, generalized signalized networks, rural highways, urban highways, and freeway corridor systems.

Develop the INSIGHT 2+ Production Rule Language

To build expert systems with INSIGHT 2+, the user must first specify a set of goals for decision making. INSIGHT 2+ uses the production rule language (PRL) to represent knowledge in terms of IF–AND–THEN–ELSE rules that contain factual information in the expert knowledge domain. PRL also allows the end user to specify procedural rules and execute dependent conditions to search for any unsatisfied IF conditions. Knowledge bases have a variable threshold, or a minimum confidence acceptability level, that can be adjusted as the knowledge data base is executed. A numeric confidence level may also be assigned to each conclusion to allow the user to work with specialized knowledge. This applies only with the known simple-facts and question-answer type of query in the evaluation. Knowledge bases created by INSIGHT 2+ can be executed quickly at a microcomputer-based work station.

Basically, the key words of the INSIGHT 2+ PRL are command words for programming the main decision-making and other information-supporting functions. They are

AND	DISPLAY	IF	RULE
ARE	ELSE	IS	THEN
OF	END	OFF	THRESHOLD
CONFIDENCE	EXPAND	ON	TITLE

In this study, the necessary constraints are selected for the basic facts, rules, and application areas for each of the computer packages in the analysis. First, the particular application groups are assembled and grouped as basic constraints. Second, the explicit information for designing the detailed expert system structure is defined with the commands TITLE, THRESHOLD, CONFIDENCE, and GOALS. Third, the production rules are set up according to the nature of the conclusions and recommendations for traffic engineering management. Last, the trace report provided in the INSIGHT 2+ knowledge engineering system makes possible the study of the decision-making process according to the specific production rule defined. An example follows to describe the information selection process in the MAXBAND program using the basic information obtained from the FHWA software awareness report. It gives the rules coded for the MAXBAND program according to the required constraints extracted from Figure 1. As indicated, a set of natural language program statements first defines the prerequisite conditions for determining the function of the MAXBAND program in this prototype expert system.

```
RULE for selecting program MAXBAND
IF Program belongs to Signal Timing Optimization Programs
AND Optimize signal timing    Cycle Length
OR Optimize signal timing    Phase Length
OR Optimize signal timing    Offset
AND Optimized Facility is Arterial Street
AND Program can Not simulate Cycle Length
AND Program should Provide plots of Time-Space
      Diagram ?
AND Computer requirement is    Mainframe
AND Type of microcomputer you use is    None
THEN Program MAXBAND is recommended
AND DISPLAY MAXBAND
```

The INSIGHT 2+ system separates the decision rules into two basic categories: knowledge rules and inference rules. Knowledge rules include the facts about and relationships of a problem that are embedded in the expert's knowledge. For example, if the experience of the traffic engineer suggests that optimization of traffic signal timing is required for traffic analysis, then this element becomes the major deciding factor for choosing a particular signal-timing program. The inference mechanisms, on the other hand, can tell the computer how to use knowledge rules to solve a problem. Inference is a reasoning algorithm, not a rule, that provides the reasoning or problem-solving strategy. In a completed AI program, knowledge rules are usually combined with both the knowledge base and the inference rules in the expert system to provide better application.

In operation, the INSIGHT 2+ inference engine or knowledge processor of the expert system compares the decision rules in the knowledge base with the facts and information entered by users. If the user-input information is incomplete, the inference engine will ask the user to provide more descriptions for additional analysis. It can also offer conclusions and explain recommended actions in a natural language interface. Usually, the recommendations are based on the reasoning used to reach final conclusions. Moreover, it can provide the user question-and-answer prompts in English not just output computer codes. The reasoning or inferencing process will link related decisions supplied by the user to appropriate actions from the production rules in the knowledge base. These linked rules form knowledge chains in which the THEN statement may become the IF statement that can eventually lead to the most likely conclusion in the evaluation.

Debug the Production Rule Language

Program testing and debugging are essential to successful computer programming and ES applications. The program support environment makes it relatively easy to compile and debug the PRL. The basic procedure for running this particular expert system is described in the following steps. Load the program in response to the MS DOS prompt command. Load the INSIGHT 2+ interpreter by typing 'I2'. Next, specify the knowledge base, FHWAINF0 in this case, to start program execution. After compiling production rules, INSIGHT 2+ will flag error messages until the compiled knowledge base can be obtained. Then, run the program by using the function keys that are defined at the bottom of the display screen. After loading the compiled program in the interpreter mode, press the function key F3 to start the expert system for analysis. The user will then respond to the questions presented and select the desired answers. At the end, recommendations to meet the user's input requirements will be given.

After the trace report has been reviewed, the production rules can be revised using the trial-and-error method. The different levels of trace reports could be used to evaluate the relative effectiveness of the model. Programming efforts should be continued until the computer program works as designed in the production rules. Some working experience obtained from debugging the INSIGHT 2+ knowledge engineering tools includes revising the search sequence for each study goal. It was noted that the order of the goals and subgoals may be used to rank their relative importance and how each goal is analyzed in the system. Because the system separates each goal according to its unique identification number and characters, these items are important in the initial design of the PRL to avoid any potential problems. Logical errors coded in PRL may create a lot of design problems later on. Therefore selection of the proper facts and constraints is important for successful operation of an expert system.

COMPLETE PROGRAM DOCUMENTATION

To complete the necessary program documentation, knowledge engineering tools usually provide various programming-support functions and commands. Program documentation about the system should be done through the use of three command words in the INSIGHT 2+ system—TITLE, EXPAND, and DISPLAY.

1. TITLE: This function is used to summarize the contents of the expert system under design and provide additional information for program documentation and reference. For example, this function was used in this prototype expert system to explain the proper execution steps for instructing users how to execute and implement this expert system before the actual execution of the INSIGHT program.

EXECUTION STEPS
I. USER INPUT INFORMATION.
- PROGRAM CATALOG
- OPTIMIZATION CAPABILITY
- CYCLE LENGTH SIMULATION
- OPTIMIZE FACILITY
- SIMULATE FACILITY
- INPUT DATA PROGRAM
- TIME-SPACE DIAGRAM
- MAINFRAME OR MICROCOMPUTER
- TYPE OF MICROCOMPUTER

II. PRODUCTION RULE ANALYSIS.
III. RECOMMEND SOLUTION ALTERNATIVE.
IV. OPTIONAL TRACE REPORT.

2. EXPAND additional information: The EXPAND function can be used to describe the questions in the production rules and the constraints in the query analysis. This command and other similar key words can also provide explanations and the characteristics of the problem under analysis. This function was used in this system to describe the question to be asked the user.

EXPAND Optimized Facility is isolated intersection

Diagram of an isolated intersection:

```
              ¶
  ____________¶____________
              ¶
              ¶
```

3. DISPLAY supportive information: This functional command is used to expand the basic and relevant information for

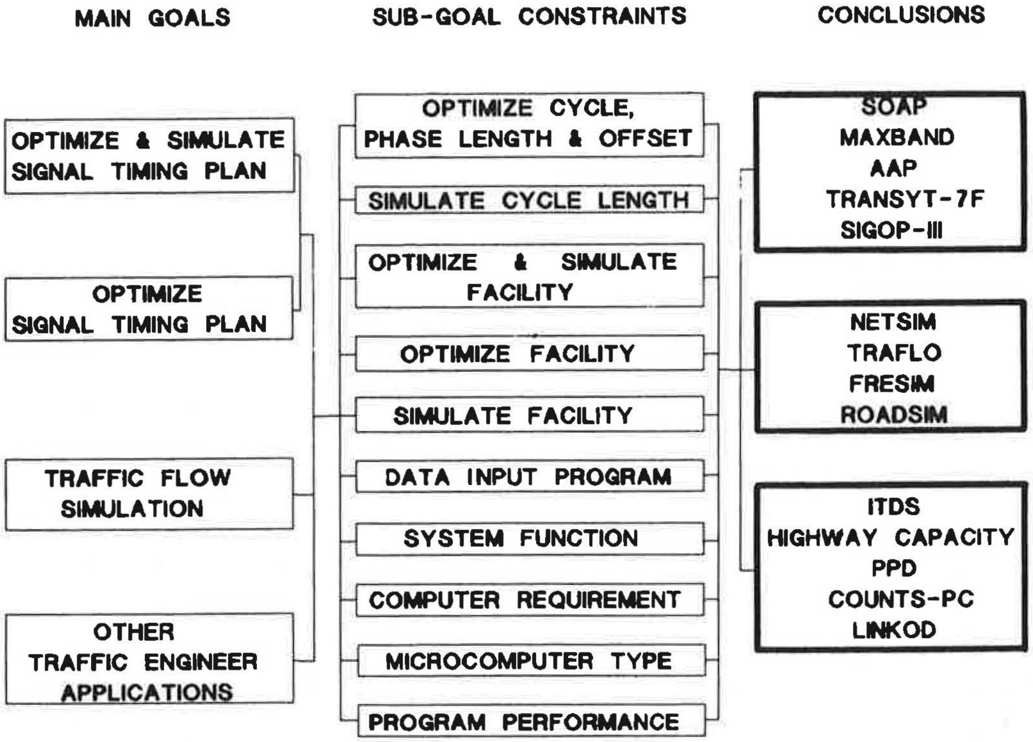


FIGURE 6 **Basic program structure.**

providing supportive suggestions about the recommended design alternatives and the question itself during the query process. The DISPLAY function was used in this expert system to provide additional suggestions to obtain directions for searching more information when conclusions have been reached through the knowledge inference process.

DISPLAY MAXBAND (MAXimum BANDwidth)
MAXBAND — MAXimum BANDwidth
LATEST AVAILABLE VERSION: A time-space diagram was recently added to the original (and, to date, only) version of the program. The program is also available in FORTRAN-77.

PROGRAM STRUCTURE

The structure of the expert system, as described in this paper, is shown in Figure 6. The functional structure of the program clarifies the interrelationships among the six components: *(a)* main goal, *(b)* subgoal, *(c)* selection criteria, *(d)* constraints, *(e)* rules, and *(f)* recommendations. The prototype expert system program was developed to explain the main goals, relate main goals and subgoals using production rules, explain subgoal constraints, demonstrate selection criteria for the production rules, and recommend subgoal conclusions to main goals in the AI/ES analysis.

The INSIGHT 2+ knowledge system consists of an inference system and a knowledge base compiler. The inference mechanism executes the knowledge data base. After the user has selected a knowledge base on a particular topic, INSIGHT 2+ searches for all possible recommendations. INSIGHT 2+ presents the user with questions to answer and goals to select. The knowledge system formulates the goal choices, the questions, and the conclusions from information obtained from the domain expert and the end user. The knowledge base compiler works with the knowledge engineer's input to create the compiled knowledge base that INSIGHT 2+ runs. The knowledge engineer creates a knowledge base using PRL and a standard text editor processor. INSIGHT 2+ takes the knowledge base, translates it, and then streamlines it so the INSIGHT 2+ knowledge system can run faster in execution. The INSIGHT 2+ tools can be best used in areas that require routine professional judgment. They can assist engineers and managers in designing procedures for implementation. They are helpful when many people at different locations need expert advice to do a job.

USING INSIGHT 2+

The advantages of using a knowledge engineering tool such as INSIGHT 2+ are its easily understood programming structure and well-equipped support functions in a user-friendly environment. Other advantages of using a knowledge engineering tool such as the INSIGHT 2+ system follow. Sequencing in the production rule is important only for the definition of goals and subgoals. Although the order of the constraints in the production rules is not important in query input, the interpreter will seek to optimize the execution sequence and the operation of the expert system. The order of evaluation constraints within the production rule for defining goals and subgoals will not influence execution of the expert system.

The knowledge engineering tool is also easy to use. The information that needs to be defined is the specific constraints required to determine each individual goal and subgoal using decision table analysis. The coding of PRL is efficient within the knowledge engineering programming environment. Both forward- and backward-reasoning processes can be performed in this expert system without additional computer-programming efforts. This PRL program can be maintained easily using the built-in editing function or regular word-processing facilities.

With knowledge engineering tools like INSIGHT 2+, a person can develop expert systems, which accumulate knowledge on a subject or knowledge base, to analyze, reason, and provide solutions to problems that would normally require human expertise. INSIGHT 2+ uses a backward- and forward-chaining inference mechanism. In a forward-chaining application, INSIGHT 2+ can be used to acquire user input and try to recommend a software package according to the information it contains on a particular application or a pattern described by the knowledge rules. In a backward-chaining application, INSIGHT 2+ begins with a specific software package and determines whether or not the preconditions justify using that package.

INSIGHT 2+ does have some disadvantages. Because of the interconnected cause-and-effect relationships, errors in program logic are difficult to identify. The INSIGHT 2+ system also limits the type of data that can be analyzed in the PRL system. Moreover, there are limitations on the length of a line to be coded in the PRL program. But perhaps one of the most important improvements between the INSIGHT 1 and INSIGHT 2+ systems is the addition of explicit OR functions for eliminating duplicate definitions of each individual condition. This will provide the benefit of not having to specify every possible decision tree by using duplicate production rules.

CONCLUSIONS AND RECOMMENDATIONS

This study evaluates the feasibility of using expert system designs to aid in the selection of programs. Texas Transportation Institute researchers believe that it is cost-effective to develop computer software using AI techniques to assist the end user in optimizing traffic management strategies. The expert systems design can assist practicing traffic engineers in selecting proper traffic-related software for developing better traffic control strategies in both urban and rural areas. Furthermore, the production rules of the proposed expert systems design, developed with either AI languages or knowledge engineering tools, can provide an alternative means for representing traffic engineering expertise in the decision-making process.

AI languages and tools are generally more flexible for developing expert systems yet more difficult for programming than is a conventional computer language. Because of the complexity involved in AI/ES programming, only well-trained programmers can comfortably use the LISP and PROLOG languages to build expert systems. Knowledge engineering design can be done with a range of knowledge engineering tools for developing specialized applications. Knowledge engineers often have to make decisions about the programming languages to be used. If portability is the primary concern, they will probably choose to translate their codes into conventional programming languages that can later be run on conventional operational systems. On the other hand, if more complex or sophisticated expert systems are to be developed for future applications, the tools may be coded in LISP or PROLOG and designed to run on LISP- or PROLOG-based machines. Usually, AI languages do not have user-friendly programming support for ES development as do knowledge engineering tools, which can be easily used on conventional computer systems.

To enhance this prototype expert system, it is recommended that the information be stored outside the expert system to optimize program execution and compilation of the knowledge data base. To restructure this expert system, additional investigations are also needed in the following three areas *(12–14)*:

1. From a knowledge engineering programming standpoint:
   - Modify the goals, subgoals, rules, and constraints;
   - Add a debug error message in the trace report;
   - Include a logic table or logic tree in the trace report; and
   - Enhance the program through the INSIGHT 2+ system.
2. From a domain expert applications standpoint:
   - Modify design using manual procedures,
   - Work with other knowledge engineering systems,
   - Obtain experience from teaching end users, and
   - Use other computer programs.
3. From an end user applications standpoint:
   - Develop the expert system to interface with other software, such as DBASE, PASCAL, and LOTUS programs;
   - Expand the knowledge bases to help the end users; and
   - Provide determinable information, such as threshold settings.

It is further recommended that expansions be enhanced to provide a computerized expert system for advising end users in the selection of proper computer programs for effective traffic engineering analysis. This application will be especially useful in the future for helping users select suitable computer software packages in the TRAF family as supported by the FHWA. It is also believed that modification of this expert system could be best achieved by improving AI/ES program efficiency and restructuring the formulation of existing programs for object-oriented problem-solving applications.

ACKNOWLEDGMENTS

The author appreciates the research support of the Texas Department of Highways and Public Transportation in cooperation with the U.S. Department of Transportation, Federal Highway Administration, and the Texas Transportation Institute. In addition, the constructive comments and suggestions provided by the TRB reviewers were helpful in the completion of this paper.

REFERENCES

1. *Use of Traffic Simulation Models in Project Planning and Evaluating.* Office of Safety and Traffic Operations Research and Development, Traffic Systems Division, FHWA, U.S. Department of Transportation, Sept. 1985.
2. A. S. Byrne et al. *Handbook of Computer Models for Traffic Operations Analysis—Technical Appendix: Summary of Models and References.* Report TS-82-214. FHWA, U.S. Department of Transportation, Dec. 1982.

3. A. S. Byrne et al. *Handbook of Computer Models for Traffic Operations Analysis.* Report TS-82-2134. FHWA, U.S. Department of Transportation, 1982.
4. *Special Report 194: The Application of Traffic Simulation Models.* TRB, National Research Council, Washington, D.C., 1981.
5. Y. S. Hsu and P. K. Munnjal. Freeway Digital Simulation Models. In *Transportation Research Record 509*, TRB, National Research Council, Washington, D.C., 1974, pp. 24–41.
6. W. D. Labrum and R. M. Farr. *Traffic Network Analysis with NETSIM in a Small Urban Grid.* Report TS-80-230. FHWA, U.S. Department of Transportation, June 1980.
7. E. B. Lieberman and R. D. Worrall. *Traffic Network Analysis with NETSIM—A User's Guide.* Report IP-80-3. FHWA, U.S. Department of Transportation, Jan. 1980.
8. D. A. Wicks and B. J. Andrews. *Development and Testing of INTRAS, A Microscopic Freeway Simulation Model.* 4 Vols. Reports RD-80-106, RD-80-107, RD-80-108, and RD-80-109. FHWA, U.S. Department of Transportation, Oct. 1980.
9. E. Lieberman. *Microscopic Simulation for Urban Traffic Management.* 5 Vols. Reports RD-80-113, RD-80-114, RD-80-115, RD-80-116, and RD-80-117. FHWA, U.S. Department of Transportation, Jan. 1982.
10. *An Introduction to Urban Travel Demand Forecasting—A Self-Instructional Text.* User-Oriented Materials for UTPS. FHWA and UMTA, U.S. Department of Transportation, 1977.
11. P. H. Winston. *Artificial Intelligence*, 2nd ed. Addison-Wesley, Inc., Reading, Mass., July 1984.
12. P. Harmon and D. King. *Expert Systems—Artificial Intelligence in Business.* John Wiley and Sons, Inc., New York, 1985.
13. D. A. Waterman. *A Guide to Expert Systems.* Addison-Wesley, Inc., Reading, Mass., 1986.
14. B. G. Buchanan and E. H. Shortlife. *Rule-Based Expert Systems—The MYCIN Experiments of the Stanford Heuristic Programming Project.* Addison-Wesley, Inc., Reading, Mass., 1985.
15. *PD PROLOG—User's Manual.* PROLOG documentation for the educational and public domain system. Robert Morein and Automata Design Associates, Dresher, Pa., 1986.
16. *TURBO PROLOG—The Natural Language of Artificial Intelligence—Owner's Manual.* Borland International, Inc., Scotts Valley, Calif., May 1986.
17. *INSIGHT 1 Reference Manual.* Level Five Research, Inc., Indialantic, Fla., 1986.
18. *INSIGHT 2+ Reference Manual.* Level Five Research, Inc., Indialantic, Fla., 1986.
19. C. T. Hendrickson, D. R. Rehak, and S. J. Fenves. Expert Systems in Transportation Systems Engineering. *Transportation Research*, Jan. 1986.
20. C. Zozaya-Gorostiza and C. T. Hendrickson. An Expert System for Traffic Signal Setting Assistance. Civil Engineering Department, Carnegie-Mellon University. Submitted to *Journal of Transportation Engineering*, ASCE, Jan. 1986.
21. W. F. Clocksin and C. S. Mellish. *Programming in PROLOG*, 2nd ed. Springer-Verlag, New York, 1984.

The products mentioned here are trademarks of several companies. IBM Personal Computer (PC) and PC DOS are products of the IBM Corporation. MS and MS DOS are registered trademarks of the Microsoft Corporation. PD PROLOG is a trademark of the Robert Morein and Automata Design Associates. TURBO PROLOG is a trademark of Borland International, Incorporated. INSIGHT 2+ is a trademark of Level Five Research, Incorporated.

Designing Optimal Transportation Networks: An Expert Systems Approach

ROBERT SHIENG-I TUNG AND JERRY B. SCHNEIDER

A knowledge-based expert system (KBES) approach can be used to solve the single-mode (automobile), fixed-demand, discrete, multicriteria, equilibrium transportation network design problem. Previous work on this problem revealed that mathematical programming methods perform well on small networks with only one objective. A solution technique is needed that can be used on large networks that have multiple, conflicting criteria with different weights of relative importance. The KBES approach discussed in this paper represents a new way to solve network design problems. The development of an expert system involves three major tasks: knowledge acquisition, knowledge representation, and testing. For knowledge acquisition, a computer-aided network design and evaluation model (UFOS) was developed to explore the design space. This study investigated the problem of designing an optimal transportation network by adding and deleting capacity increments to or from any link in the network. Three weighted criteria were adopted for use in evaluating each design alternative: cost, average volume-to-capacity ratio, and average travel time. The best nondominated design is determined by a multicriteria evaluation technique called concordance analysis. The research started with a design exercise conducted by a group of students who were asked to find a series of link capacity changes that would produce a series of successively better designs. The results were carefully examined and used to generate the facts and rules that make up the knowledge base of the network design expert system (EXPERT-UFOS). It has two phases of analysis. The macrolevel analysis recommends a total budget using trade-off functions for each pair of criteria. The microlevel analysis provides advice about how to add or delete capacity on each link to avoid paradoxes. Test results show that EXPERT-UFOS found, with fewer design cycles, designs that were better than any of the 76 student designs included in the test. EXPERT-UFOS may have enough simplicity to deal with large networks. The results of this study, in which a laboratory-based knowledge acquisition method was employed successfully to generate a functional knowledge base, suggest that the KBES approach is an appropriate method for dealing with the computational complexities of network design problems.

Contemporary transportation network designers face two major problems. The first is computational complexity that has restricted the classical solution method (mathematical programming) to small problems. The second is that traditional single-objective formulations are not well suited to dealing with practical multicriteria problems. A new design process that provides a capability for dealing with a multicriteria evaluation and decision-making process and is computationally feasible for large problems is needed.

The limitations of traditional mathematical programming models for dealing with network design problems have been examined thoroughly by many researchers *(1–3)*. Basically, any transportation network design problem encounters a combinatorial explosion because of the discrete nature of the attributes of the transportation network. When searching for a unique optimum solution, intensive calculation is required. Experience has shown that these models can handle only small- to medium-sized networks with currently available computers within a reasonable time. This computing requirement has limited the applicability of such techniques to small problems only.

Approximation methods (i.e., machine-based heuristics or heuristic search and man-machine interaction or interactive problem solving) have shown more promise for dealing with large problems. Heuristic search techniques use empirically derived rules (e.g., add, delete, interchange) to systematically search for near-optimal solutions. In contrast, interactive methods use man's intuitive capabilities and knowledge to guide a search of the combinatorial solution space. Both heuristic search and interactive methods can reduce the size of the search space to some extent. Heuristic search techniques, which usually use a single global heuristic, can often consistently find optimal solutions, but the computational requirements are still prohibitive for dealing with large problems *(2)*. Interactive methods use more heuristics and domain knowledge and usually can find acceptable solutions within reasonable computing times, but these techniques lack consistency because they depend heavily on human knowledge, experience, and perceptual skills.

The need for a multicriteria evaluation component has added to the complexity of the transportation network design task. Only a few researchers have used optimization methods to tackle the multicriteria network design problem *(4)*. The results indicate that such methods are generally not applicable to large problems because of computational difficulties. The results suggest again that approximate methods are likely to be most appropriate for large real-world problems.

Approximate methods that include a multicriteria evaluation component are still under development. The purpose of an approximate method is to derive robust search heuristics that can find high-quality solutions within a reduced solution space. Both heuristic and interactive methods have been developed to achieve this objective. The difference between them is the way in which the search strategy and knowledge (heuristics and facts) are generated and used. Heuristic methods often emulate optimization algorithms. They integrate a search strategy with available knowledge. Heuristics can consistently find solutions

---

R. S.-I. Tung, Department of Public Works, 900 King County Administration Building, Seattle, Wash. 98104. J. B. Schneider, Department of Civil Engineering, University of Washington, Seattle, Wash. 98195.

that are locally optimal. However, they have no pattern recognition capability and cannot recognize local constraints. Interactive methods have separate simulation and control functions. The control function tends to rely more on human guidance and such inputs can often produce an efficient and effective search. Such search behavior is usually called "intelligent search" and relies on man's powerful pattern recognition capabilities and domain expertise. However, this pattern recognition capability and domain expertise cannot be stored and coded in an explicit form, which makes system performance user dependent. Thus this method lacks consistency, and its reliability cannot be guaranteed.

There are no conflicts between heuristic search and interactive methods. An ideal system would use them both. Heuristics can be created by using interactive methods and then coded as machine-based algorithms. However, not every idea discovered using interactive techniques can be algorithmically defined as required by the traditional heuristic search method. More important, traditional heuristic search methods cannot flexibly handle various rules, facts, and associated domain knowledge while maintaining a user-friendly dialog with the designer. This is why the knowledge-based expert systems approach can be used to tackle this problem.

The knowledge-based expert systems (KBESs) approach has evolved from research in artificial intelligence. In contrast with traditional algorithmic methods, the expert systems approach has separated the control strategy from the knowledge base. It can flexibly handle various heuristics (or rules) and facts. Also, it is interactive and user friendly. The expert systems approach has been found to be useful in many fields *(5)*. However, most applications so far are diagnosis oriented. Only a few applications are in the transportation area *(6)*. No previous research has attempted to deal with the multicriteria equilibrium network design problem using this new approach. It is hypothesized that this new method will be useful for dealing with design problems on large networks.

KNOWLEDGE ACQUISITION

To develop an expert system, normally either a human expert or some written expertise must exist and be available for use. Unfortunately, no human experts exist who can handle the complexity of transportation network design problems. Written expertise does not exist either. However, not all expertise must come from long experience. Design expertise can also be generated by using a simulation model. In dealing with combinatorial design problems, simulation may be the only effective way to generate such expertise. To test this fundamental hypothesis, a computer-aided design and evaluation model (UFOS) was developed. Using the UFOS model, a design exercise was conducted with a group of students. The results of this design exercise provided much valuable design knowledge that was then used to develop the knowledge base for a network design expert system.

Network Simulation Model

UFOS is designed to allow a user to formulate and test a wide variety of ideas about the design of a transportation network. It has the capability of performing both fixed-demand analysis and elastic-demand analysis. In this study, only fixed-demand analyses have been used. UFOS contains four individual modules (Figure 1).

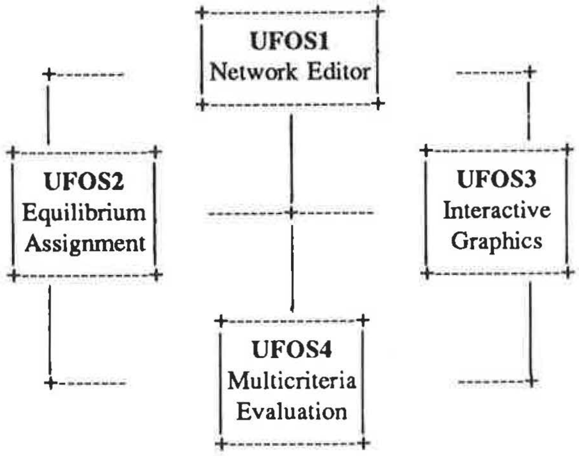


FIGURE 1 Modules of UFOS.

These four modules are linked to provide a user-friendly interactive design and evaluation environment. UFOS has built-in link attribute settings. Five roadway link types are available for conducting network design activities:

Type	*Lanes*	*Capacity (vehicles/hr)*	*Speed (mph)*	*Cost Factor ($/lane-mi)*
1	1	250	35	1
2	2	800	35	1
3	3	1,300	35	1
4	2	2,000	60	10
5	3	3,000	60	10

Types 1–3 represent arterial standards. Types 4 and 5 represent freeway standards. These different link types have different construction costs, and these costs represent one of the performance criteria used in the evaluation process.

Design Exercise

The network for the design exercise is defined by nine nodes that represent nine large zones in the eastern part of the Central Puget Sound region (Figure 2). These nodes are connected by 24 roadway links. This network forms a linear urban shape that usually generates high congestion in its central area. An evening peak origin-destination pattern with a total trip volume of 19,500 vehicles per hour represents the travel demand requirement.

This network was used as the basis for a design problem that was assigned to eight graduate students enrolled in a course on transportation and land use models. The purpose of this exercise was to search for a network design that would produce an efficient loading pattern with minimal congestion, minimal average trip times, and the lowest possible cost. Using the given travel demand pattern, each student was asked to search for an efficient roadway network design by increasing or decreasing the capacity and speed on various links. The travel behavior of each trip maker was assumed to follow the user-optimum principle. As a result, trip makers change their routes

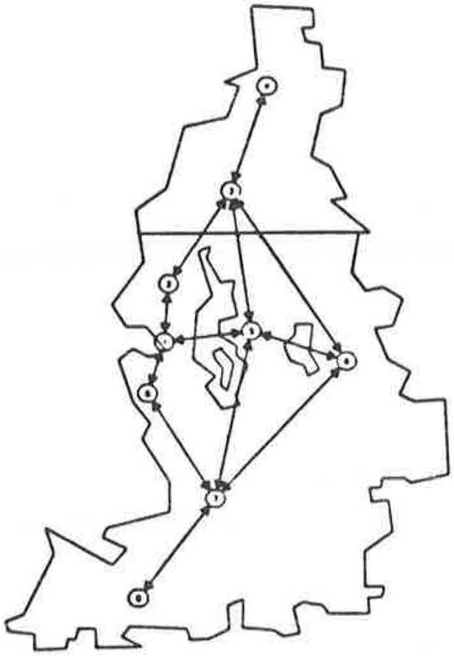


FIGURE 2 Puget Sound test network.

in response to various capacity allocations that produce different congestion patterns. The performance of each design is evaluated using three weighted criteria: total cost, average vehicle-to-capacity (V/C) ratio, and average trip time (ATT).

Theoretically, given a travel demand (origin-destination) pattern and a fixed number of link types, the possible lower-bound and upper-bound values of each design criterion can be calculated by simply setting all links at minimum capacity (Type 1) or at maximum capacity (Type 5). The bounding values for this problem are given in Table 1.

TABLE 1 BOUNDING VALUES

Criterion	Design 999 (maximum capacity)	Design 888 (minimum capacity)	Objective
Cost ($)	8,915.5 (UB)	356.62 (LB)	Less is better
V/C	0.56 (LB)	3.99 (UB)	Less is better
ATT (min)	29.76 (LB)	4795.04 (UB)	Less is better

NOTE: LB = lower bound and UB = upper bound.

The best design for this problem is one of the 5^{24} = 59,604,644,775,380,625 possible alternatives in this design space. Each will have criterion values that lie somewhere between the bounds. Note that the design with a minimum capacity roadway network has the best (lowest) cost but the worst V/C and ATT. On the other hand, the design with all links set at maximum capacity has the best performance in V/C and ATT but is worst (highest) in cost. In reality, there are many conflicts like this among performance criteria. A preferred design is a design that can satisfy most of the objectives well, that is, a best-compromise design.

The exercise started with a network in which all links had been set at minimum capacity. Each student was asked to generate at least six designs to explore the combinatorial design space. The students were asked to record their link-specific design decisions and design performance expectations. It was hoped that they would detect some cause-and-effect relationships between link-type changes and performance criteria that might become rules that could be used to build the knowledge base for an expert system.

The evaluation was a two-stage process. In the first stage, the students evaluated their design against their own design set and the two given bounding designs (Designs 888 and 999). In the second stage, the best designs from each student were aggregated and comparatively evaluated. A single weighting scheme was used throughout: Cost, 0.5; V/C, 0.25; and ATT, 0.25

Design Strategy Analysis

Given such a partially structured problem, the designer wants to devise a strategy that will produce a sequence of successively better designs. Such strategies can be developed from the knowledge and experience of the designer (expert). Usually, the construction of a design strategy involves using both "deep knowledge" or "hard information" (e.g., an explicit model with assumptions, relationships, and constraints) and "surface knowledge" and "soft information" (e.g., intuitive constructs). Moreover, the designer's ability to implement a certain strategy depends on the ability to interpret the hard information in the results and effectively integrate it with the soft information. Usually, for engineering-oriented design problems, a clear understanding of the deep knowledge aspects of the problem is necessary. Thus a domain expert can usually perform better than a novice designer. However, given the right computer-based design aids, many novice designers can reach high performance levels rather quickly. This design exercise allows the student to explore the performance of certain design concepts quickly and easily. By observing their progress, it is possible to learn how effective various design strategies are in dealing with a partially structured problem.

Two basic types of design strategies were used: incremental exploration and logic based. Each strategy has its strengths and weaknesses. Two criteria can be used to evaluate these strategies in terms of their effectiveness and efficiency in producing high-performance designs. First, how consistent was the strategy in finding successively better designs? Second, how efficient was the search strategy used to find an optimum design? The following two examples are used to illustrate these points.

- Incremental exploration design strategies: Most students used an incremental exploration strategy. They simply added some capacity to the links with the highest V/C ratios during each design session. They kept on driving congestion levels down while keeping costs as low as possible. This type of strategy was conservative but did produce better designs easily. The experiences of Student A are typical of this type of design behavior. Table 2 gives his experience for the six designs. The total capacity trends clearly show the incremental changes he made. Only a few of the most congested links were upgraded to the next level of capacity during each design cycle. Table 3 gives the overall design performance of the six designs and Table 4 gives the ranking results produced by a series of multicriteria evaluations. As the ranking results show, this student found a series of designs that were, except for the last, successively better. In addition, his best design (005) was the

TABLE 2 DESIGN EXPERIENCE OF STUDENT A

Link No.	888		001		002		003		004		005		006	
	Cap	V/C	Cap	V/C	Cap	V/C	Cap	V/C	Cap	V/C	Cap	V/C	Cap	V/C
1	250	6.2	800	2.4	800	3.8	800	3.4	1300	1.9	1300	1.6	3000	0.7
2	250	2.4	250	6.7	800	2.4	800	2.4	800	2.2	1300	1.7	2000	1.0
3	250	6.8	800	4.1	800	4.0	800	3.5	1300	2.4	2000	1.8	3000	0.9
4	250	1.6	250	6.6	1300	1.2	800	1.6	800	1.6	800	1.8	800	1.3
5	250	6.7	800	4.5	800	4.1	800	3.8	1300	2.5	2000	2.1	3000	1.5
6	250	2.9	250	4.1	250	4.3	800	0.4	800	0.9	800	1.5	800	1.8
7	250	5.0	250	5.2	800	2.9	800	2.4	800	2.0	800	1.4	800	1.4
8	250	7.7	800	1.8	800	1.5	800	1.3	800	1.2	800	1.5	800	1.2
9	250	3.7	250	5.5	800	2.5	800	2.5	800	2.5	1300	1.5	1300	1.5
10	250	3.7	800	6.9	800	2.5	800	2.5	1300	1.5	1300	1.5	1300	1.5
11	250	5.6	250	2.6	800	3.1	800	3.0	1300	1.7	1300	1.7	3000	0.9
12	250	2.2	250	5.1	250	1.8	800	0.4	800	0.7	800	1.1	800	1.1
13	250	4.6	250	4.1	800	3.2	800	2.8	1300	2.0	1300	1.7	2000	1.0
14	250	0.0	250	2.0	250	0.0	800	0.0	800	0.0	250	0.0	250	0.0
15	250	4.0	250	6.2	800	3.3	800	2.4	800	2.2	1300	1.6	2000	1.1
16	250	1.1	250	1.8	250	1.8	800	0.4	800	0.2	800	0.6	800	0.3
17	250	3.7	250	5.5	250	5.1	800	2.2	800	1.9	800	1.4	800	0.9
18	250	2.0	250	3.8	250	3.1	800	1.0	800	0.8	800	0.4	800	0.3
19	250	1.4	250	2.7	250	4.5	800	1.1	800	1.2	800	0.4	800	0.4
20	250	3.1	250	4.6	250	3.3	800	1.9	800	1.6	800	1.3	800	1.0
21	250	5.1	250	6.3	800	2.7	800	2.8	1300	1.8	1300	1.7	3000	0.9
22	250	4.7	250	4.7	250	5.6	800	2.5	800	2.2	1300	1.6	3000	1.0
23	250	3.7	250	6.0	800	2.5	800	2.5	800	2.5	1300	1.5	1300	1.5
24	250	6.9	800	4.3	800	4.4	800	4.4	1300	2.7	2000	1.8	3000	1.2
Tot. Cap.	6000		9300		14750		19200		23200		27250		39150	
Avg. V/C		3.9		4.5		3.1		2.1		1.7		1.4		1.0

second best in the class. However, an incremental strategy may cause paradoxical results. As the data in Table 3 indicate, Design 001 has a higher capacity than the lower-bound design (888), but it also has higher V/C and ATT. Paradoxes can cause problems in design and must be avoided.

- Logic-based design strategy: Some students chose to try to develop more sophisticated design strategies. They wished to use mathematical principles to tackle the problem. To do this, they needed to develop a deeper understanding of various aspects of the design problem such as the design-to-performance and node-to-link relationships. For example, they knew that the low-cost designs would be preferred because the weight of "cost" is greatest (0.5). They determined that a link should have the highest possible capacity if it was congested in Design 999, in which all links had the maximum capacity setting. Similarly, they knew a link should be given the lowest capacity setting if it had no loading even in Design 888, in which all links had the maximum capacity setting. They devised efficient strategies for adding and deleting capacity. They knew that a network with an average V/C ratio of around 1.0 could produce the best combination of values for the criteria. Thus they simply added capacity in rough proportion to the V/C ratios. However, even with this knowledge, they could not always predict the results correctly. Student B used this type of design strategy. Table 5 gives his design results for the six

TABLE 3 DESIGN CRITERIA MEASURES OF DESIGNS OF STUDENT A

Design	Cost ($)	V/C	ATT (min)
888	356.62	3.94	4795.04
999	8,915.50	0.56	29.76
002	565.31	4.48	4881.95
002	804.51	3.06	1404.37
003	1,141.18	2.13	659.69
004	1,360.78	1.68	196.10
005	1,584.31	1.37	85.87
006	3,351.41	1.01	47.88

NOTE: 888 and 999 are bounding designs.

TABLE 4 MULTICRITERIA EVALUATION RANKS OF DESIGNS OF STUDENT A

	Design						
Run	888	999	002	003	004	005	006
1	2*	2*	1				
2	3*	3*	2	1			
3	4*	4*	3	2	1		
4	5*	5*	4	3	2	1	
5	6*	6*	5	3	2	1	4

NOTE: 1 is best. * = not above average.

TABLE 5 DESIGN EXPERIENCE OF STUDENT B

Link No	888		001		002		003		004		005		006	
	Cap	V/C	Cap	V/C	Cap	V/C	Cap	V/C	Cap	V/C	Cap	V/C	Cap	V/C
1	250	6.2	2000	1.5	2000	1.3	2000	1.2	2000	1.2	2000	1.6	2000	1.4
2	250	2.4	1300	1.4	1300	1.2	1300	1.1	1300	1.1	1300	1.7	2000	1.0
3	250	6.8	1300	1.4	2000	1.2	2000	1.4	2000	1.4	2000	1.8	2000	1.2
4	250	1.6	800	1.3	800	2.0	800	1.7	800	1.7	1300	1.8	1300	0.9
5	250	6.7	3000	1.5	3000	1.5	3000	1.4	3000	1.4	3000	2.1	3000	1.5
6	250	2.9	800	1.6	1300	1.2	1300	1.4	1300	1.3	1300	1.5	1300	1.2
7	250	5.0	1300	1.5	1300	1.3	800	1.8	800	1.7	1300	1.4	1300	1.3
8	250	7.7	800	1.0	250	2.1	250	1.7	250	1.6	800	1.5	800	1.3
9	250	3.7	1300	1.5	1300	1.5	1300	1.5	2000	1.0	1300	1.5	2000	1.0
10	250	3.7	1300	1.5	1300	1.5	1300	1.5	1300	1.5	1300	1.5	2000	1.0
11	250	5.6	2000	1.3	2000	1.4	2000	1.4	2000	1.5	2000	1.7	2000	1.5
12	250	2.2	250	0.2	250	1.4	250	1.4	250	1.5	250	1.1	250	1.1
13	250	4.6	2000	1.0	2000	1.1	2000	1.1	2000	1.1	1300	1.7	1300	1.0
14	250	0.0	250	0.0	250	0.0	250	0.8	250	0.9	250	0.0	250	0.0
15	250	4.0	1300	1.4	2000	1.1	2000	1.4	3000	1.0	2000	1.6	2000	1.3
16	250	1.1	250	1.4	250	1.5	250	1.4	250	1.4	250	0.6	800	0.6
17	250	3.7	800	1.1	250	1.8	800	1.1	800	1.2	800	1.4	800	1.1
18	250	2.0	250	0.3	250	0.7	250	1.4	250	1.5	250	0.4	250	0.2
19	250	1.4	250	0.9	250	1.7	250	1.3	250	1.3	250	0.4	250	0.0
20	250	3.1	800	1.5	800	1.0	250	1.5	250	1.6	1300	1.3	800	1.2
21	250	5.1	2000	1.3	2000	1.4	3000	1.0	2000	1.5	1300	1.7	2000	1.3
22	250	4.7	2000	1.4	2000	1.4	2000	1.3	2000	1.3	2000	1.6	2000	1.3
23	250	3.7	1300	1.5	2000	1.0	1300	1.5	800	2.5	1300	1.5	1300	1.5
24	250	6.9	3000	1.2	3000	1.2	3000	1.2	3000	1.2	2000	1.8	2000	1.8
Tot. Cap.	6000		30350		31850		31650		31850		30850		33700	
Avg. V/C		3.9		1.2		1.3		1.4		1.4		1.3		1.1

designs. He made dramatic changes on the first design and only marginal changes on the rest of the designs. Table 6 gives the performing measures of these six designs, and Table 7 gives their multicriteria evaluation rankings.

In general, the logic-based strategies produced good first designs but did not always produce high-performance designs quickly. As the data in Table 6 indicate, Student B finally reached his best design in the sixth design session and it was the best in the class. He had a good initial design (001), which was not very different from the best design. However, he did have problems making consistent progress (Table 7). The reason for this is that when a design is close to the optimum it is more difficult to predict the flow pattern that will be produced by link capacity changes. Actually, a network designer can never precisely predict the flow pattern that will be computed by the equilibrium assignment algorithm, but it is often possible to do so in general terms.

Ideally, a good design strategy should contain elements from both of these approaches. An incremental exploration strategy may be better for large problems because it is too difficult to do a sophisticated analysis before the first design action is taken. However, as much logic as possible should be used to reduce the size of the search space and to avoid paradoxes that usually occur when an incremental exploration strategy is used.

TABLE 6 DESIGN CRITERIA MEASURES OF DESIGNS OF STUDENT B

Design	Cost ($)	V/C	ATT (min)
888	356.62	3.94	4795.04
999	8,915.50	0.56	29.76
001	2,112.38	1.20	56.57
002	2,225.08	1.31	53.15
003	2,345.43	1.36	55.43
004	2,324.63	1.38	66.66
005	1,873.64	1.25	61.73
006	2,090.21	1.07	49.79

NOTE: 888 and 999 are bounding designs.

TABLE 7 MULTICRITERIA EVALUATION RANKS OF DESIGNS OF STUDENT B

	Design							
Run	888	999	001	002	003	004	005	006
1	2*	2*	1					
2	2*	2*	1	2				
3	3*	3*	1	2	3*			
4	4*	4*	1	2	3*	5*		
5	5*	5*	2	3	4	5*	1	
6	6*	6*	3	4	5*	7*	2	1

NOTE: 1 = best. * = not above average.

KNOWLEDGE REPRESENTATION

The design strategies identified in the design exercise are valuable for building an expert system. First, the incremental exploration strategy indicates that an incremental simulation approach is an effective way to deal with large design problems. A series of small improvements may be the easiest way to approach a good design. Second, the logic-based design strategy indicates that use of some deep knowledge can produce rapid progress toward the best design. This deep knowledge can be represented as facts and rules and used to construct the knowledge base for an expert system.

Expert System Shell

An expert system shell is a convenient tool for developing application-oriented expert systems. The PC-based expert system shell M.1 *(7)* was used in this study to provide a user-friendly interface and the capability to link with external functions. The extensive number crunching of EXPERT-UFOS was handled by using C-based external functions. By using these external functions, it is possible to maintain the transparency of the knowledge base while having the computational efficiency of a C-program. The relationship between the knowledge base and external functions is shown in Figure 3.

Facts and Rules in EXPERT-UFOS

The knowledge base of EXPERT-UFOS consists of various facts and rules. Facts are link specific and are represented by object-attribute-value (O-A-V) triplets. Objects are the specific links of a network. Attributes describe aspects of the network-related deep knowledge and form the basis for making effective design designs. The term "value" specifies the particular nature of an attribute for a given object. The attributes are further divided into static and dynamic categories. Attributes that are fixed during the entire design process are static attributes (e.g., type, criticality, and length). All other attributes are dynamic attributes (e.g., V/C, add-priority, or delete-priority). The dynamic attributes have to be recalculated during each consultation with EXPERT-UFOS. The full structure of the O-A-V framework is given in Table 8.

Rules are used to implement the heuristics for finding a series of successively better designs. There are four basic types of rules in EXPERT-UFOS:

1. Control rules: These rules control the main flow of a consultation. For instance, the following rule is used to determine if a best nondominated solution has been found:

```
if    no_more_improvement(1) is true
then  best_nondominated_solution is true.

if    evaluation_rank(CYCLE_N) = X
and   X > 1
then  no_more_improvement(CYCLE_N) is true.

if    CYCLE_M + 1 = CYCLE_N
then  nextcycle to CYCLE_M = CYCLE_N.

if    nextcycle to CYCLE_M = CYCLE_N
and   no_more_improvement(CYCLE_N) is true
then  no_more_improvement(CYCLE_M) is true.
```

The following rule controls the process of equilibrium assignment:

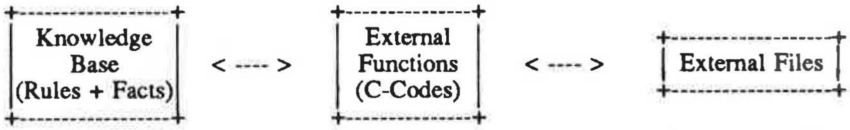


FIGURE 3 Relation between knowledge base and external functions.

TABLE 8 LINK O-A-V TRIPLET REPRESENTATION

Object		Attribute	Possible Values
Link #	Static	Type	(production, attraction, production-attraction, single-production, single-attraction, buffer)
		Criticality	volume of free-flow loadings
		Length	miles
		Capacity	(250, 800, 1300, 2000 or 3000 vehicles per hour)
	Dynamic	V/C	real number as derived
		Add-priority	integer rank number
		Delete-priority	integer rank number

```
if     initial_equilibrium_assignment is true
and    do_shortest_path(ITER_N) is true
and    do_loading_assign(ITER_N) is true
and    do_fibonacci_search(ITER_N) is true
and    do_convergence_check(ITER_N) is true
then   equilibrium_ assignment(ITER_N) is true.
```

2. External rules: These rules activate the external functions for doing extensive calculations. For example, the following rules check the convergence of the assignment process:

```
if     no_of_links = NARC
and    ITER_N = ITER
and    external(eq_converg_check, [NARC,
       ITER]) = FLOW_STD
then   eq_convergence_check(ITER_N) = FLOW_STD.

if     eq_convergence_check(ITER_N) = FLOW_STD
and    ITER_N > 1
and    1.0 – FLOW_STD = X
and    convergence_criterion = Y
and    X > Y
then   do_convergence_check(ITER_N) is true.
```

3. Macrorules: These rules are used to determine the design goal and budget limit. For example, the following rules are used to determine the design goal:

```
if     weight(cost) = X
and    X > 0.5
then   design_goal = decrease_capacity.

if     weight(cost) = X
and    X <= 0.5
then   design_goal = increase_capacity.
```

The budget limit is determined by the following two macrorules and an external rule:

```
if     design_goal = decrease_capacity
then   search_type = downward.

if     design_goal = increase_capacity
then   search type = upward.

if     search_type = SEARCH
and    cost(CYCLE_N) = COST
and    vc(CYCLE_N) = VC
and    att(CYCLE_N) = ATT
and    cost_weight = CW
and    vc_weight = VCW
and    att_weight = ATTW
and    external(budget, [SEARCH,COST,VC,ATT,
       CW,VCW,ATTW]) = BUDGET
then   budget_check(CYCLE_N) = BUDGET.
```

4. Microrules: These rules are used to determine which links should have more or less capacity and how much. Such decision making is based on several facts, such as priority, higher-capacity, cost-factor, link-length, and criticality. For example, the following rules determine whether capacity of a link should be increased to a higher level:

```
if     design_goal = increase_capacity
and    capacity_check(LINK_N) is true
and    budget_check(LINK_N) is true
and    critical_check(LINK_N) is true
then   add_action(LINK_N) is true.
```

To verify that capacity_check(LINK_N) is true, the following two rules are used:

```
if     not(no_more_capacity(LINK_N))
then   capacity_check(LINK_N) is true.

if     design_goal = increase_capacity
and    capacity(LINK_N) = CA
and    CA>= 3000
then   no_more_capacity(LINK_N) is true.
```

The following rule is used to verify that the budget_check(LINK_N) is true to ensure that the capacity increase will not cause the budget limit to be exceeded:

```
if     link_cost(LINK_N) = X
and    budget(LINK_N) = Y
and    X<= Y
then   budget_check(LINK_N) is true.
```

The link cost is calculated by the following rule:

```
if     design_goal = increase_capacity
and    capacity_check(LINK_N)
and    capacity(LINK_N) = CA
and    higher_capacity(CA) = HC
and    cost_factor(CA,HC) = CF
and    length(LINK_N) = LN
and    CF * LN = Z
then   link_cost(LINK_N) = Z.
```

The higher-capacity and cost-factor data are provided by reference to the following facts:

```
higher_capacity(250) = 800.
higher_capacity(800) = 1300.
higher_capacity(1300) = 2000.
higher_capacity(2000) = 3000.

cost_factor(250, 800) = 2.2.
cost_factor(800, 1300) = 2.0.
cost_factor(1300, 2000) = 14.8.
cost_factor(2000, 3000) = 25.0
```

TESTING AND CONCLUSIONS

EXPERT-UFOS was tested by giving it the same design problem as was given to the students. The result was that EXPERT-UFOS needed only four cycles to conclude the best nondominated design (003) given in Table 9. Design 003 is substantially better than the three best student designs. The better rank (Table 10) indicates that Design 003 is the best known nondominated design for the given weighting scheme. This

TABLE 9 CRITERIA MEASURES OF EXPERT-UFOS DESIGN AND BEST STUDENT DESIGNS

Design	Cost ($)	V/C	ATT (min)
888	365.62	3.94	4795.04
999	8,915.50	0.56	29.76
003	1,525.76	1.33	126.07
S01	2,090.21	1.07	49.79
S02	1,584.31	1.37	85.87
S03	1,544.42	1.43	149.09

NOTE: S01 to S03 are the best student designs.

TABLE 10 MULTICRITERIA EVALUATION RANKS OF THE DESIGN CONTEST

	Concordance		Discordance			
Design	Dominance Value	Rank	Dominance Value	Rank	Average Rank	Final Rank
888	0	2	0.97	4	3.0	3*
999	0	2	0.97	4	3.0	3*
003	1.5	1	–0.51	1	1.0	1
S01	0	2	–0.47	3	2.5	2
S02	–0.5	3	–0.49	2	2.5	2*
S03	–1.0	4	–0.47	3	3.5	4*

NOTE: * = not above average.

result indicates that EXPERT-UFOS did find a solution that is better than all of the 79 designs generated by the students. EXPERT-UFOS performed well in its first test. It needed only four cycles to find its best design. Because an equilibrium assignment problem must be solved in each cycle, the fewer cycles needed, the greater the efficiency of the method. There can be no absolute measure of efficiency because different machines have different computational speeds. Because the equilibrium assignment algorithm is a standard procedure for finding an optimal flow pattern, the less execution time needed to solve the assignment, the more efficient the method. EXPERT-UFOS quickly reduced the search space to a minimum. Part of the success of EXPERT-UFOS is the result of its successful prevention of the design paradox. As the results show, increasing cost does reduce values of V/C and ATT. As long as EXPERT-UFOS can avoid paradoxes, the system should be able to find a high-performance design quickly using a cyclic approach.

Because good results were obtained in a few cycles, EXPERT-UFOS is cost-effective. However, this does not mean that EXPERT-UFOS will always be superior to interactive methods. The design exercise discussed was conducted by novice designers under loosely defined conditions. Trade-off functions were unknown, and all of the students were doing this design task for the first time. Given the trade-off functions, an experienced designer might produce a design the performance of which was the same as or even better than that of Design 003. However, this might be true only for a small network. It is unlikely that a human could deal effectively with a large network using only an intuitively guided approach. On the other hand, because EXPERT-UFOS can efficiently reduce the search space and effectively avoid paradoxes, it offers a reasonable approach for dealing with large problems. Tests of this type are currently under way.

REFERENCES

1. L. LeBlanc et al. An Efficient Approach to Solving the Road Network Equilibrium Traffic Assignment Problem. *Transportation Research*, Vol. 9, 1975, pp. 309–318.
2. R. Wong. Transportation Network Research: Algorithmic and Computational Questions. *Transportation Research*, Vol. 19A, 1985. pp. 436–438.
3. T. Friesz. Transportation Network Equilibrium, Design and Aggregation: Key Developments and Research Opportunities. *Transportation Research*, Vol. 19A, 1985, pp. 413–427.
4. T. Friesz and P. Harker. Multicriteria Spatial Price Equilibrium Network Design: Theory and Computational Results. *Transportation Research*, Vol. 17B, 1983, pp. 411–426.
5. F. Hayes-Roth et al., eds. *Building Expert Systems*. Addison-Wesley, Inc., Reading, Mass., 1983.
6. C. Hendrickson. Comment on Research in Transportation System Characteristics and Operations. *Transportation Research*, Vol. 19A, 1985, pp. 367–369.
7. *M.1 Reference Manual: Updated Edition for Software Version 1.3*. Teknowledge, Inc., Palo Alto, Calif., 1985.

Application of Expert Systems to Left-Turn Signal Treatment

EDMOND CHIN-PING CHANG

Left-turn treatments are essential to signal capacity and operational safety at signalized intersections. Left-turn warrants and guidelines are sets of evaluation procedures designed to maximize level of service, minimize approach delay, and reduce left-turn–related accidents. Currently, three left-turn phasings are used to allow vehicles to make left turns on a green arrow or a circular green indication: permissive, exclusive, and exclusive/permissive. An experimental expert system design for recommending alternative left-turn phase selection on microcomputer systems was investigated. The goal of this study was to computerize left-turn phase selection by using artificial intelligence languages and knowledge engineering. This study focuses on investigating expert systems programming using PROLOG and the INSIGHT 1 system in an IBM PC/XT/AT microcomputer environment. Three experimental systems were developing using the PD PROLOG system, the TURBO PROLOG system, and the INSIGHT 1 knowledge engineering system. The background of the study, the artificial intelligence concept, the basic systems design, and the practical experience gained are discussed. Potential advantages and disadvantages of developing expert systems using different artificial intelligence languages and the knowledge engineering for traffic engineering applications are evaluated. The results of this limited study indicate that it is feasible to combine artificial intelligence and traffic engineering technologies for alternative traffic signal analysis.

This study was developed by the Texas Transportation Institute to investigate the feasibility of applying artificial intelligence (AI) technology and expert systems design concepts to a confined traffic engineering problem using an IBM PC/XT/AT microcomputer. Prototype expert systems were experimented with to analyze user input; evaluate it using various paths of reasoning; offer a conclusion; and, finally, suggest suitable left-turn phase treatment. The guidelines applied in this study were developed from a paper by Jonathan E. Upchurch *(1)*.

Three prototype expert systems were developed with AI programming tools for expert systems using PROLOG and the INSIGHT 1 system in IBM PC/XT/AT-compatible microcomputer systems *(2–4)*. Two slightly different expert systems were designed using AI languages; another system was built with a knowledge engineering tool. These systems include the ones developed in the AI programming languages PD PROLOG and TURBO PROLOG as well as the INSIGHT 1 production rule language *(5–11)*. All three expert systems were completed and observed to perform successfully; advantages and disadvantages were noted for each of the expert system programming techniques.

PD PROLOG is a public-domain experimental PROLOG system that follows very closely the structure and syntax of an AI computer programming language as described by W. F. Clocksin and C. S. Mellish *(5)*. This A.D.A. PROLOG interpreter was developed for educational and public-domain usage *(8)*. TURBO PROLOG is a commercially available AI programming language compiler developed and released in May 1986 by Borland Incorporated *(9)*. It follows more closely the function and syntax of the LISP AI programming language than did the original PROLOG languages, such as the PD PROLOG system. The major advantage of the TURBO PROLOG system is its capability of compiling and generating object codes as quickly as the TURBO PASCAL compiler. It also has built-in editing and tracing functions, a knowledge inquiry environment, knowledge data base management systems, and programming development environments.

INSIGHT 1, as mentioned previously, is a commercially available knowledge engineering tool developed by Level Five Research *(10, 11)*. It was used in this study to investigate the feasibility of designing expert systems using knowledge engineering tools in an IBM PC/XT/AT-based microcomputer environment. In general, AI programming can be implemented through the LISP- or PROLOG-based language system with a minimum of difficulty. Knowledge engineering tools like the INSIGHT 1 system can allow noncomputer-oriented users and knowledge engineers to prototype a specialized problem area quickly. Knowledge engineering tools can assist users to develop their own customized expert system applications and define the logical reasoning structure in less time than it would take any other computer-programming language or system.

BACKGROUND

Left-turn treatments are essential to signal capacity and operational safety at an intersection. Left-turn guidelines are sets of procedures designed to maximize level of service, minimize approach delay, and reduce left-turn–related accidents. Three left-turn phasings are commonly used to allow vehicles to make left turns on a green arrow or circular green indication: permissive, exclusive, and exclusive/permissive left-turn treatments *(1)*. Selecting proper left-turn phasings involves a series of engineering decisions instead of an algorithmic process. The experience and knowledge of a traffic engineer can greatly improve final solutions. The design process begins with describing the intersection geometry, traffic movements, and available signal control equipment. Next, traffic volume data are investigated. When enough information has been collected,

---

Texas Transportation Institute, Texas A&M University System, College Station, Tex. 77843.

the traffic engineer can propose alternatives. Then the traffic engineer can modify or insert new production rules based on his design experience.

This left-turn phase expert system follows the guidelines recommended by Upchurch *(1)*. Many preidentified factors and rules are required to determine the logical choices among different design alternatives. The evaluation guidelines, as shown in Figure 1, recommend different phasing selections by considering left-turn volumes, opposing through volumes, number of opposing lanes, cycle length, approach speed, sight distance restrictions, and historical records of severe left-turn–related accidents. This selection guideline represents the typical analysis process of *(a)* an algorithmic method, *(b)* knowledge inference capabilities, and *(c)* the knowledge base of a traffic engineer. The first evaluation determines the critical volume cross-product calculation from the input. The forward-chaining inference mechanism models the dependencies among different decision-making activities in the human reasoning process. The reasoning or inference process optimizes design objectives by starting from known information. The third process models the domain knowledge in IF-AND-THEN-ELSE rules to resemble the human decision-making process. For example, the existence of sight distance restrictions and severe left-turn accidents can justify the provision of protected left-turn signal treatments.

These decision rules and reasoning processes are particularly useful for solving problems in instances that may not be covered by established guidelines. Problem-solving expert systems based on established guidelines can provide users with reasoning knowledge similar to that of a human expert constantly available for assistance in the specialized area *(3, 4)*. The expert system can generate solutions that resemble the traditional design and that may be used by other traffic engineers for determining proper traffic control. Because only a few heuristic decisions that might lead to the best solutions are selectively analyzed each time, the system is quite efficient. Most traffic engineering problems have characteristics similar to left-turn phasing selection as described in this paper. Traffic engineering expert systems are useful for assisting users to solve recurring design problems, sharing common working experience for mutual learning, and providing better design alternatives in the future. By correctly constructing the knowledge-based expert system, traffic engineers can further refine their mental decision-making process to reflect experience obtained from the previous design process.

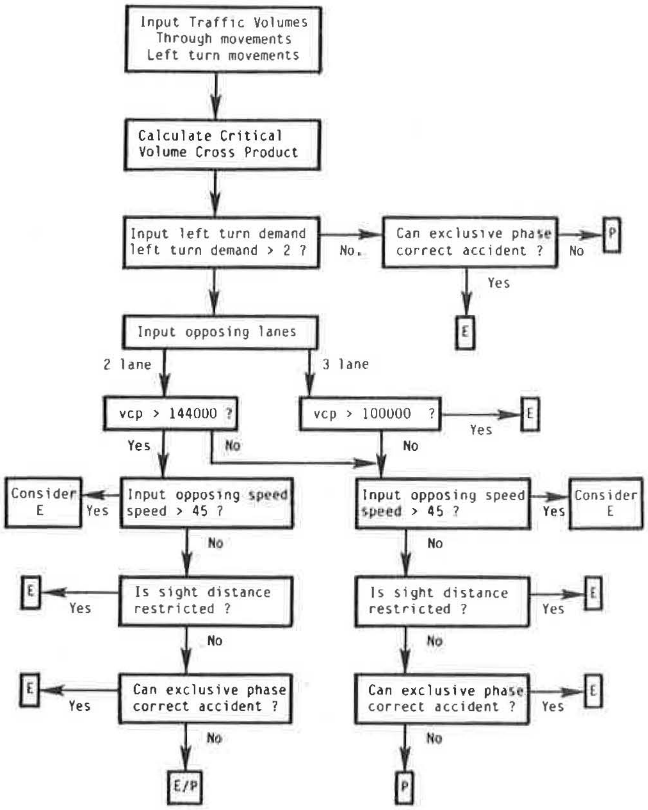


FIGURE 1 Recommended guidelines for selecting type of left-turn phasing *(1)*.

ARTIFICIAL INTELLIGENCE TECHNOLOGY

Artificial intelligence (AI) technology, including knowledge-based systems and expert systems, has promising applicability to engineering problems *(2–4)*. The relationships among AI, knowledge-based expert systems (KBESs), and expert system (ES) design are shown in Figure 2. Since World War II, scientists have developed computerized techniques to simulate human behavior and decision making. Behavioral scientists, mechanical engineers, and computer scientists are all active in AI research to produce programs that can solve problems that humans solve well. It is anticipated that the AI study will eventually lead to intelligent computerized applications in specialized areas. The research includes decision-making systems, robotic devices, and various approaches to computerized speech synthesizing. Today, the United States, Japan, Britain, and other countries of the European Economic Community are all implementing knowledge-based systems and expert systems. However, expert systems research in this country is confined to only a few university research laboratories, mainly those at Stanford, Carnegie-Mellon, and the Massachusetts Institute of Technology.

Knowledge-Based Expert Systems

The knowledge-based expert system (KBES) is a collection of AI techniques and analysis processes that enables a computer to assist people in analyzing specialized problems. KBESs were introduced to extend computer applications. A KBES

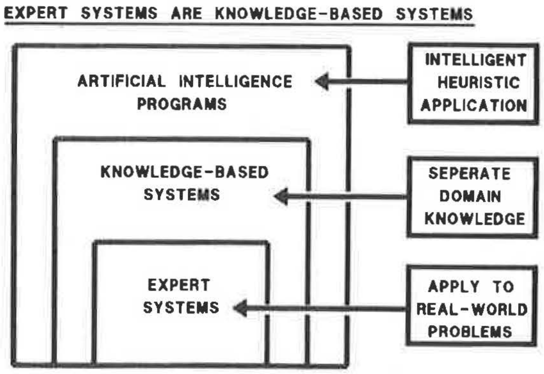


FIGURE 2 Artificial intelligence, knowledge system, and expert systems *(3)*.

provides human expertise through both the knowledge engineering language and the program-supporting environment *(3)*. The AI/KBES application requires development of a generalized knowledge base that permits traffic engineers to interact with the following three components: the traffic characteristic data, the theoretical or simulation results, and the specific hypothesis for measuring the effects of traffic control system measures. The structured guidelines for traffic engineering problems are suitable for KBES applications because explicit algorithms do not exist and the traditional programs can provide only restricted problem-solving capability. A rule-based expert system (RBES) is another knowledge-based mechanism available for design applications. It should be noted that a KBES may also be an RBES.

Expert Systems Design

Expert systems (ESs), as part of the AI/KBES technology, are computer programs that incorporate the knowledge and thinking processes of experts to provide operational people the insights gained from years of experience. Expert systems differ from conventional data-processing programs. The latter rely on defining logical algorithms for a program. The major differences among ESs are expert performance, symbolic reasoning, depth of knowledge representation, and self-knowledge for logical operation. Traditional programs are developed by explicitly stating all of the applicable rules and execution sequences. Usually, algorithmic programming states only the action parts of the rules. A KBES, on the other hand, uses the same action rules as algorithmic programs but specifies independently all of the heuristic parts of the selection sequence. The rules can be programmed in symbolic relationships and treated as the knowledge base.

A practical ES includes three elements: the knowledge data base, the support environment, and the end user. These are usually specified by a knowledge engineer or AI programmer who specializes in ES and a domain expert who understands the specific problem or domain area of the designated program. By conducting extensive interviews with the domain expert, the knowledge engineer can summarize the expert's knowledge into commonly known facts and rule-of-thumb tricks that the expert has acquired from years of experience. Three modules are generally programmed in ES: the explanation module, the knowledge acquisition module, and the user interface module. The explanation module provides the ES with the ability to recommend problem-solving strategies based on the reasoning process. The knowledge acquisition module coded in the knowledge base is usually constructed in rigid format for logical processing. The knowledge interface module often uses a set of problem-oriented questions presented through a friendly interface. The interface module helps the user to monitor system performance, supply information, request explanations, and redirect possible recommendations.

The ES design combines the decision-making process and rule-of-thumb guidelines for specific problem solving. This design process combines the algorithmic method, knowledge inference capabilities, and the knowledge base of the traffic engineer. Sequential control is used to evaluate the critical volume cross-product from input traffic. The forward-chaining concept evaluates the dependencies among different activities in the human reasoning process. This reasoning process optimizes objectives by starting from known information. In the decision-making process, the domain knowledge is written with IF-THEN-ELSE rules to resemble the human decision-making process. For example, the existence of sight distance restrictions and severe left-turn accidents may justify the use of a protected left-turn signal treatment, which might also be recommended by an experienced traffic engineer. Expert systems have been applied in many disciplines. However, not all areas are suitable for expert system formulation *(6)*.

Other Representation Frameworks

Representing knowledge in an AI program means choosing a set of conventions and structures for describing the objects, relations, and processes *(4)*. First, a conceptual framework is chosen to represent the problem, either symbolically or numerically. Then conventions within given computer languages are chosen for implementing the design. The former is difficult and important; the latter is less difficult and of less importance because good programmers can find ways of working with almost any concept within any kind of programming language. Representing knowledge in procedures is one alternative that domain experts in every scientific field have tried hard to avoid. The definition of production rules offers opportunity for making a knowledge base easier to understand and modify.

Artificial Intelligence Languages

A knowledge engineer converts an expert's knowledge into rules that a computer understands. Most programming is done in high-level languages, such as BASIC, COBOL, FORTRAN, PASCAL, and C. AI languages are useful in designing an ES. They include *(a)* high-level AI conventional languages, *(b)* knowledge engineering development tools, and *(c)* portability among different operating systems. Figure 3 shows AI language development *(2)*. AI researchers have been developing LISP machines that can run the knowledge systems more efficiently than does conventional hardware using a standard operating system. If portability is the primary concern, the researchers will choose to translate their codes into conventional languages that can be run on conventional operational systems. On the other hand, if more sophisticated ESs are needed, the tools may be coded for LISP- or PROLOG-based machines.

Currently, several AI languages are available for building expert systems. Specifically, an ES may be implemented as part of the KBES using a general-purpose programming language, general-purpose representation language, or domain-independent expert system framework. These high-level AI languages contain some special features, such as developing reasoning strategies. AI languages contain powerful abstract mechanisms that make the programming of human reasoning logic flexible and easy. Currently, KBESs built using LISP and PROLOG are popular among researchers. ES development tools or knowledge engineering tools can compile these English-like rules into an efficient machine code for developing production expert systems.

The AI programming language normally used is LISP (LISt Processing). LISP is preferred by AI engineers in the United

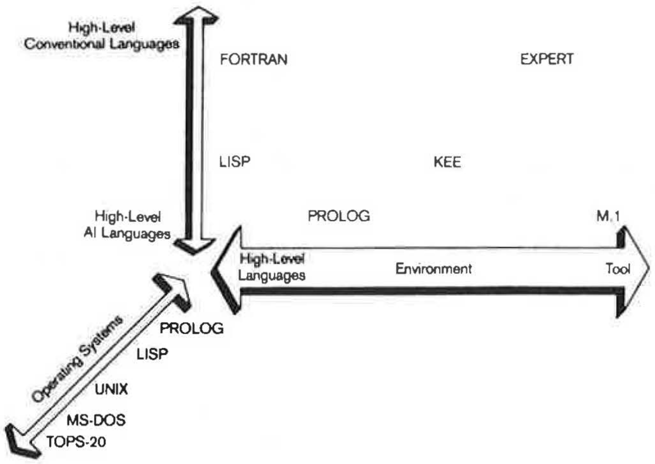


FIGURE 3 Development of artificial intelligence languages *(3)*.

States whereas a similar language, PROLOG (PROgramming in LOGic), is preferred in Europe and Japan. PROLOG contains structures suitable for programs that manipulate logical expression, whereas LISP contains operators that facilitate the creation of programs that manipulate lists. These structures are useful for developing symbolic computing programs for numerical programming. LISP has been used for AI programming and ES design for nearly two decades *(3, 4)*. It is a symbolic manipulation language with structures based on the number of constructs or statements. LISP was created by John McCarthy in 1958. Of all the major programming languages still in use, only FORTRAN is older. LISP is highly recursive, and both data and programs can be represented as lists. The lists can be nested like a Chinese "puzzle box." LISP is a "function application" language that uses a set of simple functions, such as (Plus 2 2). PROLOG, in contrast, was initially developed as a symbolic programming language in 1972 by A. Colmerauer and P. Roussel at the University of Marseilles *(5, 8, 9)*. This programming language implements a simplified predicate calculus as a true logical language. PROLOG is a "declarative query" AI language that uses simple relations among fact, rule, and query.

Neither LISP nor PROLOG is an algorithm or a procedure to be executed in fixed sequence, but both program languages can represent the human inference process. In general, AI languages are more flexible and difficult to use in prototype expert system development. Knowledge engineering tools can help an engineer design an ES.

DESIGN PROCESS

Evaluation and design procedures are needed to develop a practical expert system application. The basic design process used to implement this left-turn phase selection ES is shown in Figure 4. The design process implemented includes six different steps: extracting basic information, defining the determining factors, defining the goals and objectives, determining the analysis constraints, developing the program, and finishing program documentation.

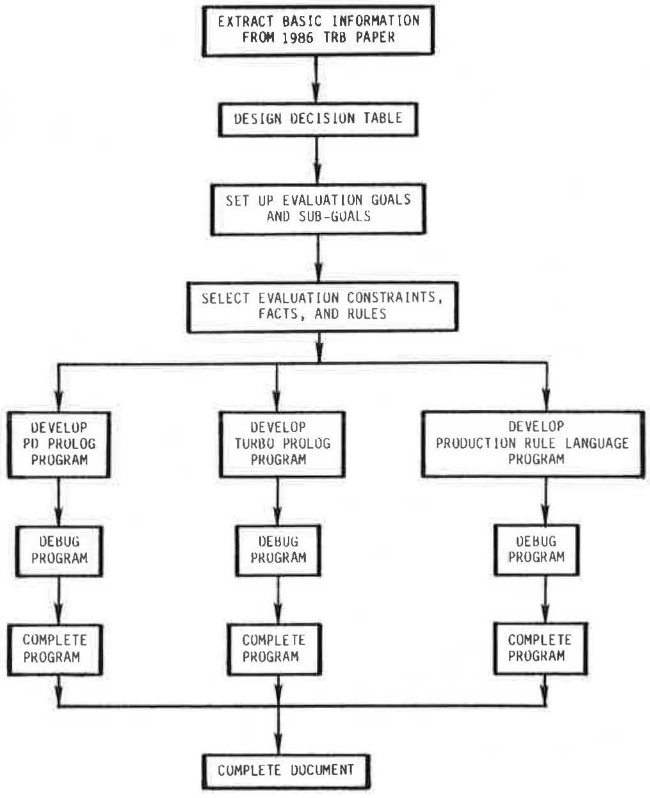


FIGURE 4 Basic design process.

Basic Design Process

Extracting Basic Information

The first step is to describe the characteristics of the problem. The description was first extracted from the design procedure recommended by Upchurch *(1)*. This information contains the analytical format for optimizing the selection of left-turn signal treatments. Basically, it can be classified into main goals, subgoals, facts, rules, and constraints. This basic functional relationship is used in the later analysis. The functional relationship of this particular ES application can be illustrated as follows:

- Study objective: Provide computerized left-turn phase selection process.
- Main goal: Recommend left-turn treatments
  1. Permissive left-turn phase,
  2. Exclusive left-turn phase, and
  3. Exclusive/permissive left-turn phase.
- Subgoal: Different variations of the main goals.
- Fact: Part of the preselected necessary condition.
- Rule: Set of prerequisite conditions to describe each of the subgoals as defined earlier.
- Constraint: Each of the selected prerequisite conditions.

Defining Determining Factors

On the basis of the guidelines recommended for optimizing the left-turn signal treatment process, a decision table was extracted to study the relationships among all of the goals, subgoals, facts, rules, and constraints. As indicated in Figure 1, there are six major constraints or user inputs needed in the decision-making process to determine proper left-turn signal treatments. The constraints, as defined, are the common traffic input information. These user inputs include amount of left-turn demand, number of opposing through lanes, volume cross-product of the conflicting left-turn and through movement pairs, opposing travel speed, sight distance restriction, and possibility of severe left-turn accidents.

Defining Goals and Objectives

From the evaluation procedure recommended in Upchurch's paper, the decision-making process was simplified into three main goals to illustrate this important study result of left-turn treatments. Because the major purpose of this ES is to computerize left-turn phase selection, the study objective is to recommend the phase sequence to be used on the basis of user input. As recommended in Upchurch's guidelines, the goals of this ES design are to recommend exclusive phase, permissive phase, or exclusive/permissive phase treatment as described earlier. To clarify the basic relationship among the outcomes of different data input, these main goals were further divided into 16 different subgoals to accommodate various possible cases involved in the logical design. These conditions or subgoals were separately described as Conditions A through P, depending on their probabilities of occurrence.

Defining Analysis Constraints

The basic analysis constraints are the major factors that can be used to define and describe the goal and subgoal needed in the analysis. Design constraints, facts, and rules were then evaluated according to decision table analysis. The constraints used in this ES include the following user-input variables:

1. Amount of left-turn demand greater than or less than two per cycle,
2. Number of opposing lanes equal to two or three,
3. Volume of cross-product value,
4. Opposing speed greater than or less than 45 mph,
5. Sight distance with or without restriction, and
6. Existence of severe left-turn accidents.

As indicated, some of the questions require numerical data input and other questions need logical data input for expanding the constraints as well as the answer for the object-answer or the simple-facts type of query. Each subgoal or condition is described by the preselected conditions from the user input. Their probability of occurrence depends on the fulfillment of each preselected condition in the actual execution. It should also be noted that the execution sequence of the ES is not predefined but results from evaluating the user's input information.

Developing the Program

After the program was designed, the computer codes were developed using PD PROLOG, TURBO PROLOG, and the INSIGHT 1 production rule languages. It should be noted that these programming tools were selected to implement this ES in the microcomputer environment. The left-turn signal phase selection expert systems were programmed for each goal and subgoal by using the constraints, rules, and facts defined previously. For each expert system, a prototype program was developed, coded, and debugged, and the basic program code was completed. As indicated earlier, AI/ES programming is different from conventional programming because greater emphasis is placed on description of the solution itself than on the solution process. Because most AI/ES programming tools are equipped with a programming-support environment, program development can be completed efficiently.

Finishing Program Documentation

All of the necessary program documentation was implemented inside the ES program to provide information for each of the expert systems. It was also noted that the structured syntax and program code of the AI languages and knowledge engineering tools was useful for internal program documentation.

Decision Table Analysis

A decision table can assist in the evaluation of the major study factors and their corresponding relationships as the different goals, subgoals, and constraints apply in the design process. Table 1 is a simplified decision table to illustrate how an action is represented in the evaluation. In this case, if one of the conditions is not satisfied, no action will be recommended. As

TABLE 1 EXAMPLE OF A SIMPLE DECISION TABLE

CONDITIONS	CHOICE (TRUE OR FALSE)	
LEFT TURN DEMAND > 2 PER CYCLE	TRUE	ELSE
ARE THERE TWO OPPOSING LANES ?	TRUE	ELSE
IS VOLUME CROSS PRODUCT > 100,000	TRUE	ELSE
ACTIONS		
1. SUGGEST USING EXCLUSIVE PHASE	X	
2. CHECK OTHER INPUT VARIABLE		X

TABLE 2 DECISION TABLE DESIGN

LEFT TURN SIGNAL TREATMENTS	PERMISSIVE PHASE			EXCLUSIVE PHASE											E/P PHASE
CONDITIONS	A	O	P	C	D	E	G	H	I	J	K	L	M	B	F
LEFT TURN DEMAND															
o DEMAND > 2		X	X	X	X	X	X	X	X	X	X	X	X		X
o DEMAND <= 2	X													X	
OPPOSING THROUGH LANES															
o OPPOSING LANES = 2		X		X	X	X	X	X	X						X
o OPPOSING LANES = 3			X							X	X	X	X		
VOLUME CROSS PRODUCT															
o > 144,000				X	X	X									X
o <= 144,000		X					X	X	X						
o > 100,000													X		
o <= 100,000			X							X	X	X			
OPPOSING SPEED															
o > 45				X			X			X					
o <= 45		X	X		X	X		X	X		X	X			X
SIGHT DISTANCE															
o W/ RESTRICTION					X			X			X				
o NO RESTRICTION		X	X			X			X			X			X
SEVERE LEFT ACCIDENT															
o COULD BE CORRECTED BY EXCLUSIVE PHASE						X			X			X		X	
o COULD NOT BE CORRECTED BY EXCLUSIVE PHASE	X	X	X												X

indicated, satisfying only one condition in Table 1 may indicate that other input data are needed in the decision process.

Table 2 is the detailed decision table used in this prototype expert system. The vertical column of the decision table lists all of the major decision factors and their constraints. In this particular example, there are six major determining factors. These factors include left-turn demand > 2 or < 1, opposing lanes = 2 or = 3, volume cross-product > 144,000 or ≤ 144,000 or > 100,000 or ≤ 100,000, opposing speed > 45 or ≤ 45 mph, sight distance with restriction or no restriction, and the possibility of correcting severe left-turn accidents by exclusive phase.

In the decision table, the first horizontal row lists all of the main goals (i.e., permissive phase, exclusive phase, and exclusive/permissive phase). The second horizontal row lists all of the possible subgoals ranging from Conditions A through P. In each column, X represents the requirements for fulfilling a certain decision condition. For example, in Condition A under the selection of the permissive phase, there are two Xs, one representing left-turn demand ≤ 2 and the other the severe left-turn accidents that cannot be corrected by using exclusive left-turn phasing. The existence of these two conditions causes the permitted left-turn phase treatment to be recommended.

This basic decision table structure can also be transformed into the pseudocode shown in Figure 5. Two different mechanisms, using either flowcharts or decision tables, are used to illustrate both decision analysis by domain experts and expert systems programming by knowledge engineers. A program flowchart can help the domain expert trace a path in the program and the knowledge engineer or programmer develop the AI program. On the other hand, the detailed decision table can also be used to identify the requirements for each condition and the desired goals from the available information. By using results from the decision table, an efficient pseudocode for later

```
1.  Define decision rules.
2.  Input 4 arterial NEMA traffic movements: 2, 5, 6, 1.
3.  Return the maximum among the volume cross products of 2 * 1 and 5 * 6.
4.  Input left turn demands and sight distance restriction.
5.  (If (left turn demand <= 2) then
         ("Can exclusive phase correct accidents?" and
         Echo printouts and
         (If can then output "Exclusive phase - suggested".) or
         (If cannot then output "Permissive phase - suggested".))) or

    (Else Input opposing lane and
          (If (the maximum product <= 144000 ; opposing lane = 2) then
              (If (opposing lane = 3, maximum product > 100000) then
                  (Output "Exclusive phase - suggested".) or
              (Else Input opposing speed and
                    (If (opposing speed > 45) then
                        (Output "Exclusive phase - suggested".) or
                    (Else "Is sight distance restricted?" and
                          (If sight distance is restricted then
                              (Echo printouts and
                               Output "Exclusive phase - suggested".) or
                          (Else "Can exclusive phase correct accidents?" and
                                Echo printouts and
                               (If can then
                                   (Output "Exclusive phase - suggested".) or
                               (Else
                                   (Output "Permissive phase -suggested".)
                               ))))))))).

          (Else input opposing speed and
                (If (opposing speed > 45) then
                    (Output "Exclusive phase - suggested".) or
                (Else "Is sight distance restricted?" and
                      (If sight distance restricted then
                           Echo printouts and
                          (Output "Exclusive phase - suggested".) or
                      (Else "Can exclusive phase correct accidents?" and
                           Echo printouts and
                          (If can then
                             (Output "Exclusive phase - suggested".) or
                          (Else
                            (Output "Exclusive/Permissive phase - suggested".)
                          ))))))).
```

FIGURE 5 Program pseudocode.

programming can be developed. In summary, the basic advantages of using the decision table include definition of all constraints individually for each goal and subgoal, presentation of all information clearly and systematically, and provision of more efficient program structure.

Program Structure

As indicated in Figure 4, this logical structure was implemented in three expert systems. They were programmed with the PD PROLOG and TURBO PROLOG computer languages and the INSIGHT 1 knowledge engineering tool *(8–11)*.

PD PROLOG Program

A PD PROLOG program is defined with the IF-THEN-OR-AND-ELSE rule *(8)*. There are two major advantages to this programming language. First, it uses the OR function that can greatly reduce redundant rules in programming. Second, it is quite similar to the pseudocode, as illustrated, for internal program documentation. These advantages make PD PROLOG programs easier to understand. However, this programming approach also has two major disadvantages: *(a)* the program is hard to trace for program execution and *(b)* it cannot trace backward to provide backward-chaining analysis for evaluating a specific subgoal in this ES design.

TURBO PROLOG Program

A TURBO PROLOG program consists of four basic programming blocks that include definitions of the domains, predicates, goals, and clauses *(9)*. The domain and predicate blocks identify all of the variables, types, and functions. The goal block declares the desired destinations or recommendations of searching. The clause block properly defines all of the facts, rules, functions, and procedures. The major advantages of TURBO PROLOG are that it is easy to understand, easy to debug, suitable for modular programming, able to generate execution files, linkable with other language programs, and equipped with editing and tracing functions. The major drawback to TURBO PROLOG is its incapability of using the OR function in the production rule. This drawback means duplicate production rules are needed to define specific conditions for each similar alternative in the expert systems design.

INSIGHT 1 Production Rule Language

INSIGHT 1 programs contain two basic parts *(10, 11)*. The first declares the goals and subgoals; the second defines all of the rules and facts. In this study, the first part of the INSIGHT 1 program describes the goals and subgoals of the left-turn signal treatments, such as different left-turn treatments and Conditions A through P. The second part of the INSIGHT 1 program summarizes the interrelationships of the prerequisite rules for fulfilling the conditions in the evaluation process.

There are two ways to program an INSIGHT 1 rule with knowledge engineering tools. The first is to separate the programming into two separate but coordinated modules. One module deals with the definition of the goals and subgoals. The other module declares the individual rules, facts, and functions for each definable case in the decision-making process. In this approach, the user can search for each individual goal or subgoal as a separate entity. The second approach mixes the goals with the facts and rules. That is, the main goals or subgoals are defined again as separate facts and rules to be included in the decision evaluation process. In either case, the user is required to supply only the necessary information related to the query; the expert system will search for certain main goals and decide which alternative is most suitable according to the user's choice for generating the optimum solution. The user may prefer the second approach that mixes the goals and rules. However, from the programmer's point of view, the first approach of separating the goals and rules is much easier to use to develop and debug the INSIGHT 1 program codes.

CONCLUSIONS AND RECOMMENDATIONS

This study investigates the feasibility of applying AI technology to the development of a prototype expert system in transportation engineering for microcomputer application. The basic procedure for generalized expert systems design generated from established guidelines has been summarized. Three slightly different expert systems were developed using the PD PROLOG and TURBO PROLOG languages and the INSIGHT 1 knowledge engineering tool. Table 3 gives a comparison of the advantages and disadvantages of the three prototype expert systems. In this table is summarized some of the design experience gained from programming this simplified traffic engineering analysis for application in the IBM PC/XT/AT microcomputer environment. This investigation is focused on knowledge acquisition, knowledge representation, system programming, and future applications. However, it is believed that this prototype expert system still requires some improvements before any practical applications can be made.

Conclusions

The following conclusions were reached in the course of designing the prototype system to optimize left-turn signal analysis:

1. Expert systems are appropriate for preidentified problem solving,
2. AI languages and knowledge-based engineering tools have advantages and disadvantages, and
3. The ES programming approach may be tailored for practical applications.

It was concluded that knowledge engineering tools, such as INSIGHT 1, do indeed have some advantages over conventional AI languages. The major advantages are their easy-to-read-and-write programs, the user-friendly menus supported by the programming environment, and the clearly defined goals and subgoals. Built-in functions are available for explaining questions in the knowledge engineering programming environment. For program debugging, a trace report is provided to study program execution and the knowledge inference process. The program-supported windows and functions are also useful for easy program development and operation. Most of all, both

TABLE 3 COMPARISONS OF TURBO PROLOG, PD PROLOG, AND INSIGHT 1

	TURBO PROLOG	PD PROLOG	INSIGHT 1
DECLARATION OF VARIABLES, OBJECTS.	YES (DOMAINS)	NO	NO
DECLARATION OF FUNCTIONS OR RELATIONS.	YES (PREDICATES)	NO	NO
DECLARE THE SEARCHING ROUTE	YES (GOAL)	NO	YES (GOALS & SUBGOALS)
SET UP RULES, FACTS, & FUNCTIONS.	DEFINE INSIDE CLAUSES BLOCK	DEFINE INSIDE THE PROGRAM	USE RULE FUNCTION TO DO THE JOB
READABILITY	VERY EASY	EASY	VERY EASY
PROGRAMMING DIFFICULTY	EASY	EASY	VERY EASY
CAN GENERATE .OBJ FILE .EXE FILE	YES	NO	NO
INTERPRETER MODE	YES	YES	YES
TRACE FUNCTION	YES	NO	YES. (GENERATE TRACE REPORT)
STRUCTURED PROGRAM	YES	YES	YES
USE OF PARENTHESE	NO	YES	NO
USE OF OR FUNCTION	NO	YES	NO
CAN LINK WITH OTHER COMPUTER PROGRAM	YES	NO	NO
ADD EXPLANATION FOR EACH GOALS & SUBGOALS	BY PRINT COMMAND NEED TO DIFINE FORMAT	BY PRINT COMMAND NEED TO DIFINE FORMAT	BY EXPAND FUNCTION VERY HANDY TO USE
DEBUGGING DIFFICULTY	VERY EASY	NOT EASY	EASY
SELF PROVIDE MENU, WINDOW, & FUNCTION KEYS	NO NEED DEFINED BY PROGRAMMER	NO NEED DEFINED BY PROGRAMMER	YES
DO MATH OPERATION	YES	YES	NO
WORK FORWARD & BACKWARD	NO NEED TO MODIFY THE PROGRAM	NO NEED TO MODIFY THE PROGRAM	YES
BUILT-IN EDITIOR	YES	NO	YES

the forward- and backward-chaining capabilities are available in the INSIGHT 1 system to *(a)* check whether goals or subgoals fit the input (forward chaining) and *(b)* check the input for the fulfillment of specific goals or subgoals in the analysis (backward chaining).

Recommendations

It is recommended that future expansions and improvements be made to develop large-scale expert systems for practical traffic management and engineering applications. Four points were ignored in this prototype expert systems design. These were the abilities to

1. Provide input data checking in the query process,
2. Return to previous steps to make changes during the search process,
3. Abandon the current searching process without losing input data, and
4. Change the data base at any time.

At present, AI researchers are trying to develop large-scale expert systems for production usage. They have devoted a lot of effort to making the expert systems design more flexible and understandable for general applications. The stylized condition-action knowledge representation provides many advantages because of its simplicity and restricted syntax in the natural language interface mechanism. Similarly, explanations and reasoning are also simplified because the convenient backward-chaining structure dynamically links the knowledge rules for logical reasoning from both directions.

Because expert systems represent a relatively new technology, there are two challenges that face most AI/ES applications today. One is how to define and represent knowledge for intelligent application by computers. The other is to develop better ways to use expert knowledge for intelligent problem solving. Although new knowledge representations in AI/ES designs have not been fully developed, AI experts are still experimenting with various knowledge representations in different discipline areas. Specialists in many fields are encountering difficulty in encoding field expert knowledge. Many refinements to the design process are still needed to satisfy demands for expert problem solving. These efforts include the modification of production rules for future reasoning in a detailed and ill-structured domain environment.

An expert system is best suited to applications in which the subject is highly detailed but tightly defined, such as practical traffic engineering applications. In the AI/ES design, the requirements of each goal or subgoal may be displayed to allow the user to understand the decision-making process. Because both forward- and backward-chaining capabilities are provided, users can refine their expertise by evaluating in both directions to search for a specified goal or subgoal. The AI/ES approach has a dual function. On the one hand, it provides an approach to computerizing decision analysis based on expert knowledge. On the other hand, AI/ES development presents a new opportunity for domain experts in many fields to review their expertise systematically through AI techniques. Therefore the development of specialized problem-solving tools can also contribute to refinement of the reasoning logic of particular applications.

A recommended AI/ES approach is shown in Figure 6. As indicated, the user's decision-making process can be used to construct the basic decision table. Then sets of knowledge engineering tools can be used to refine the logical reasoning from the knowledge representation and acquisition process during the construction of the production expert system. If more user-oriented applications are needed in the future, the expert system can later be translated into programs using AI languages for fast execution. In this way, the domain expert and user may cooperate more quickly to develop productive expert systems for their specialized applications.

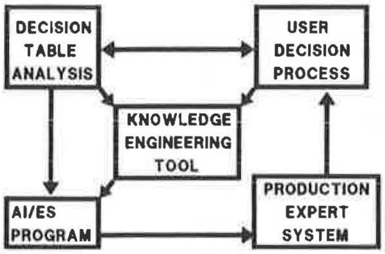


FIGURE 6 Recommended AI/ES programming approach.

ACKNOWLEDGMENTS

The author appreciates the research support of the Texas Department of Highways and Public Transportation in cooperation with the Federal Highway Administration, U.S. Department of Transportation, and the Texas Transportation Institute. In addition, the constructive comments and suggestions provided by the TRB reviewer were helpful during the completion of this paper.

REFERENCES

1. J. E. Upchurch. Guidelines for Selecting Type of Left Turn Phasing. Presented at 65th Annual Meeting of the Transportation Research Board, Washington, D.C., Jan. 1986.
2. P. H. Winston. *Artificial Intelligence*, 2nd ed. Addison-Wesley, Inc., Reading, Mass., 1984.
3. P. Harmon and D. King. *Expert Systems: Artificial Intelligence in Business*. John Wiley and Sons, Inc., New York, 1985.
4. B. G. Buchanan and E. H. Shortlife. *Rule-Based Expert Systems: The MYCIN Experiments of the Stanford Heuristic Programming Project*. Addison-Wesley, Inc., Reading, Mass., 1985.
5. W. F. Clocksin and C. S. Mellish. *Programming in PROLOG*, 2nd ed. Springer-Verlag, New York, 1984.
6. C. T. Hendrickson, D. R. Rehak, and S. J. Fenves. Expert Systems in Transportation Systems Engineering. Submitted to *Transportation Research*, 1986.
7. C. Zozaya-Gorostiza and C. T. Hendrickson. An Expert System for Traffic Signal Setting Assistance. Submitted to *Journal of Transportation Engineering*, ASCE, 1986.
8. *PD PROLOG—User's Manual*. PROLOG documentation for the educational and public-domain system. Robert Morein and Automata Design Associates, Dresher, Pa., 1986.
9. *TURBO PROLOG—The Natural Language of Artificial Intelligence—Owner's Manual*. Borland International, Inc., Scotts Valley, Calif., May 1986.
10. *INSIGHT 1 Reference Manual*. Level Five Research, Inc., Indialantic, Fla., 1986.
11. *INSIGHT 2+ Reference Manual*. Level Five Research, Inc., Indialantic, Fla., 1986.

The products mentioned in this paper are trademarks of several companies. IBM Personal Computer (PC) and PC DOS are products of the IBM Corporation. MS and MS DOS are registered trademarks of the Microsoft Corporation. PD PROLOG is a trademark of the Robert Morein and Automata Design Associates. TURBO PROLOG is a trademark of Borland International, Incorporated. INSIGHT 1 is a trademark of Level Five Research, Incorporated.

Knowledge-Based Expert System Technology Can Benefit Pavement Maintenance

J. J. HAJEK, G. J. CHONG, R. C. G. HAAS, AND W. A. PHANG

Timely and judicious selection of pavement maintenance treatments can significantly extend pavement life. To facilitate this task, an expert system for recommending routing and sealing (ROSE) of asphalt concrete pavements in cold areas was developed. The system incorporates data transmitted by 41 variables, such as pavement serviceability, age, and types of pavement surface distress, and encodes expertise derived from recent research and development studies and from experience. It contains about 360 rules. The system recommendations are given as a desirability of routing and sealing on a scale from 0 to 10. The interactive version of ROSE was developed and calibrated using an expert system development shell. This resulted in significant savings in programming, testing, and calibration. An automatic version of ROSE was implemented in FORTRAN and successfully applied to about 900 pavement sections, representing about 7200 km of highway. This application makes it possible to quantify funding requirements for different routing and sealing policies.

There are many maintenance and rehabilitation treatments that a pavement engineer can use to preserve or improve the way in which asphalt concrete pavements serve the traveling public. Described in this paper is a knowledge-based computer program that can function like an expert when selecting and recommending routing and sealing (R&S) of cracks in cold areas. This computer program, or knowledge-based expert system, was named ROSE. It is a part of a larger knowledge-based expert system for the selection and recommendation of all common pavement preservation treatments *(1)*.

Using R&S as an example, the principal objective of this paper is to show how knowledge-based expert system technology can be used to improve the selection and planning of pavement maintenance and rehabilitation actions.

ROSE was designed specifically for the Ontario Ministry of Transportation and Communications (MTC). It is based on MTC pavement monitoring and evaluation procedures, interacts with the existing pavement management information data bank, and contains the MTC knowledge base (i.e., decision logic for when to rout and seal). Although the direct application of ROSE in other jurisdictions may be difficult or even inadvisable, it is hoped that the methodology and programming approach described herein will have general applicability in other jurisdictions and to other problems.

An overview of expert systems, including their position in the field of artificial intelligence and description of their architecture and existing applications, can be found elsewhere *(2, 3)*.

ROSE was developed to satisfy the following specific objectives:

- Capture and encode expertise. Readily available knowledge associated with the selection of the routing and sealing treatment, as well as with the selection of other pavement preservation treatments, is not detailed enough to be suitable for direct use. Much of this knowledge is heuristic, unpublished, and dispersed among many users. Gathering and encoding this knowledge within an expert system structure should be especially valuable for organizations that want to capture and effectively use the expertise of senior pavement design and maintenance engineers today and for many years after their retirement. Encoding and computerizing knowledge also forces engineers to carefully organize knowledge by formulating detailed R&S guidelines.
- Provide means for consistent application of R&S guidelines.
- Provide a decision support system for preparation of preservation plans for individual pavement sections.
- Support network-level pavement management decisions. The MTC, and many other agencies, has developed a pavement management data bank that contains section-specific, detailed technical data for hundreds of pavement sections that make up the highway network. This wealth of data should be used to improve management decisions involving the total network.

ROUTING AND SEALING

The objective of R&S is to prevent surface water, particularly water containing deicing salts, from entering and damaging the pavement structure. Routing, usually done with a carbide-tipped circular cutter, opens up a crack to a width of from 20 to 40 mm and a depth of about 10 mm. This opening, cleaned and dried by hot compressed air, is required to accommodate enough sealant (hot-poured rubberized or polymerized asphalt cement) to provide an effective seal even after the pavement contracts at low temperatures *(4)*. Because of continuing improvements in sealants and in routing and sealing technology, it is difficult to estimate the benefits of R&S on the basis of past experience. However, it appears that R&S, if timed and executed properly, can prolong pavement life by about 30 percent

---

J. J. Hajek and G. J. Chong, Research and Development Branch, Ontario Ministry of Transportation and Communications, 1201 Wilson Avenue, Downsview, Ontario M3M 1J8, Canada. R. C. G. Haas and W. A. Phang, Department of Civil Engineering, University of Waterloo, Waterloo, Ontario N2L 3G1, Canada.

(5 years). This estimate is based on continual observation of seven pavement test sections routed and sealed in 1981, on Highway 17 east of Ottawa, and on long-term observation of many other sealed and unsealed sections.

The MTC has been intermittently routing and sealing asphalt concrete pavements for many years. During the last 2 years, for example, R&S work averaged about $1.5 million in cost. However, the MTC does not have any firm policy for R&S, and opinions differ among MTC personnel regarding its implementation and usefulness.

The economic significance of the R&S treatment should not be judged by its past funding or even required funding. The true economic significance emerges if the benefits of the treatment in prolonging pavement life and its cost are considered. Although a typical cost of R&S for a two-lane highway is about $1,000/km, a typical resurfacing cost is about $40,000/km.

To fully realize the significant benefits of this treatment, *(a)* the pavement sections must be selected judiciously for cost-effectiveness and *(b)* R&S applications must be timely and well executed. ROSE was designed to help pavement engineers with the first part of the task—selecting sections that would most benefit from R&S.

DEVELOPMENT OF ROUTING AND SEALING GUIDELINES

The first step in the development of ROSE was formulation of detailed R&S guidelines. The objective was to capture the best available experience and expertise, not just a general consensus among different practitioners, to be encoded in the system. The developed guidelines are thought to be the best available, but they are not yet official MTC R&S guidelines. Such guidelines may be issued after the results of long-term monitoring of an extensive 1986 experimental R&S program are known. The following brief description is included to outline the main features of the problem solved by the system. The conceptual objective is to demonstrate that, given any guidelines of this nature, expert system technology can play a key role in their implementation.

The guidelines were developed in two stages that correspond to two levels of detail: a macro level and a micro level.

Macrolevel Guidelines

The macrolevel guidelines describe an overall philosophy of R&S and were formulated by studying available literature *(4)* and the performance of existing R&S experimental pavement sections, by interviewing and working closely with one MTC research engineer, and by consulting two other MTC experts. During the interviews, the experts were individually asked whether or not they would recommend R&S for a variety of different pavement sections, with what degree of confidence, and for what reasons. Although some interviews were done in the field, the majority of the interviews was done indoors using pavement deterioration data on existing sections. The macrolevel guidelines made possible construction of a prototype of ROSE.

In general, it is recommended that R&S be used as a preventive pavement maintenance treatment. That is, R&S should be done before the initially formed single pavement cracks deteriorate (ravel; branch out into multiple cracks; or, in the case of transverse cracks, become stepped). On the other hand, it is not always practical to R&S hairline cracks. Also, if there are only a few cracks suitable for R&S, the operation may not be economically viable. Conversely, if cracking is quite extensive, it is usually better to resurface the entire pavement than to rout and seal it.

R&S decisions depend on the following factors in addition to the amount and width of cracks.

- Crack type. It is usually important to rout and seal transverse cracks that follow a course approximately at right angles to the pavement centerline. Transverse cracks directly affect riding quality of the pavement and there is some evidence that R&S may prevent or retard their stepping. As a preventive maintenance treatment, pavement edge cracks may not be routed and sealed and alligator cracks should never be.
- Pavement serviceability. Pavements with low (deteriorated) pavement serviceability should not be routed and sealed. Pavement serviceability was measured using the Pavement Condition Index (PCI) on a scale from 0 to 10 (5).
- Pavement structure. It is particularly important to R&S asphalt concrete overlays placed over portland cement concrete (PCC) pavements. Pavement condition, such as stepping, before overlay placement also affects R&S decisions.
- Presence of pavement distress. Pavement distress, such as raveling, flushing, and rutting, that reaches certain critical levels affects routing and sealing decisions. For example, a pavement section with severe raveling on most of its length should not be routed and sealed.
- Existence of pavement maintenance treatments. The presence of some maintenance treatments, such as spray patching or manual patching, usually makes R&S inadvisable.

Microlevel Guidelines

Microlevel guidelines were developed during the calibration and testing phase with only limited input from experts. The guidelines deal in detail with the influence of all variables and factors affecting R&S decisions. For example, a macrolevel guideline may state that the presence of manual patching reduces chances for cost-effective R&S. The corresponding microlevel guideline quantifies this statement by taking into account all (five) possible density levels used to describe the frequency of manual patching (few, intermittent, frequent, extensive, and throughout).

INTEGRATION AND COMPATIBILITY WITH PMS DATA BASE

Knowledge-based expert systems must be integrated with existing pavement management systems. The pavement evaluation procedure, together with the pavement information data bank, represents a significant investment. This investment is not just in software and data bases but, more important, in personnel knowledge, acceptance of the system, and training. For ROSE to be a useful decision-making tool, it must be integrated and made fully compatible with pavement management processes, including terminology, pavement evaluation

TABLE 1 GUIDE FOR DESCRIBING SEVERITY OF PAVEMENT DISTRESS *(5)*

DISTRESS TYPE / DISTRESS SEVERITY	1 Ravelling and Coarse Aggregate Loss	2 Flushing	3 Rippling and Shoving	4 Wheel Track Rutting	5 Distortion	Single & Multiple Cracks		Alligator Cracking 7. Longitudinal Wheel Track 9. Centreline 13. Transverse	10-11 Pavement Edge Cracking
						6. Longitudinal Wheel Track 8. Centreline 14. Meander and Midlane 15. Random	12. Transverse (half, full and multiple)		
1 Very Slight	Barely Noticeable	Very faint colouring	Barely noticeable	Barely noticeable (< 6 mm)	Noticeable swaying motion	Crack width < 2 mm Hairline	Crack width < 2 mm Full and partial cracks	Alligator pattern forming Depression < 12 mm	Single longitudinal or single wave-formation
2 Slight	Noticeable	Colouring visible	Noticeable	6 to 12 mm	Good control of car still present	2 to 12 mm width Single cracks	2 to 12 mm width Single full-width cracks	Alligator pattern established with corners fracturing Depression > 12 mm	Multiple parallel longitudinal or wave-formation less than 0.5 m from pavement edge
3 Moderate	Pock-marks well-spaced, open texture	Distinctive appearance with free asphalt	Rough ride Washboard appearance	12 to 19 mm Multiple cracks may be starting	Fair control of car	12 to 19 mm width Multiple cracks starting	12 to 19 mm width Single full cracks with slight cupping or lipping or multiple cracks starting	Alligator pattern established with spalling of blocks Depression > 19 mm	Progressive multiple cracks extend over 0.5 m but less than 1 m from edge. Crack begins to braid.
4 Severe	Pock marks closely-spaced, disintegration, small pot holes	Free asphalt on surface, has wet look	Very rough ride Pronounced washboard appearance	19 to 25 mm May include multiple longitudinal cracks	Poor control of car	19 to 25 mm width Multiple cracks, spalling begins to develop	19 to 25 mm width Single full cracks with moderate cupping or lipping, or multiple cracks	Blocks begin to lift, patching required. Depression > 25 mm	Progressive multiple cracks extend over 1.0 m but less than 1.5 m from edge. Begins to alligator.
5 Very Severe	Disintegrated with large pot holes	Wet look with tire noise like wet pavement surface	May cause loss of control of vehicle	Rutting>25mm May include multiple longitudinal cracks	Continuous distortion, may be dangerous at speeds > 60 km/h	Width > 25 mm Multiple cracks with spalling developed. May begin to alligator.	Width > 25 mm Severe cupping or lipping, multiple cracks with spalling. May begin to alligator.	Complete disintegration of affected area, pot holes from missing block. Depression > 50 mm	Progressive multiple cracks extend over 1.5 m from edge. Outermost area near edge is alligatored.

methodology, operating practices, and existing computer hardware and software.

The cornerstone of ROSE is the method MTC uses for evaluating and rating pavement surface distress *(5)*. Fifteen types of typical pavement surface distress, given in Table 1, are evaluated. Each type of distress is evaluated separately on a severity scale and on a density scale ranging from 0 to 5. The severity and density of distress are assigned using the guides in Tables 1 and 2, respectively, considering the entire length of the section. The average section length is about 10 km.

Pavement distress data are stored in a pavement management data bank on a mainframe computer. The bank is also designed to store all other pavement-related data that influence R&S decisions, such as pavement age; PCI; and type, extent, and cost of existing pavement maintenance treatments as well as pavement structural characteristics.

SYSTEM ARCHITECTURE

Traditionally, pavement preservation decisions have been made either at a project level or at a network level. Project-level decisions are based on detailed technical information about a specific pavement section. Network-level decisions are based on summary condition information about the entire highway network. Knowledge-based expert systems have the potential to use detailed site-specific data for network-level decisions by operating in two modes *(1)*:

- An interactive mode that queries the user for required input data and is intended to process one pavement section at a time and
- An automatic mode that is designed to interact only with other computer files and programs and is able to process many sections at the same time.

The overall architecture of the two operating modes for ROSE is shown in Figure 1.

Interactive Mode

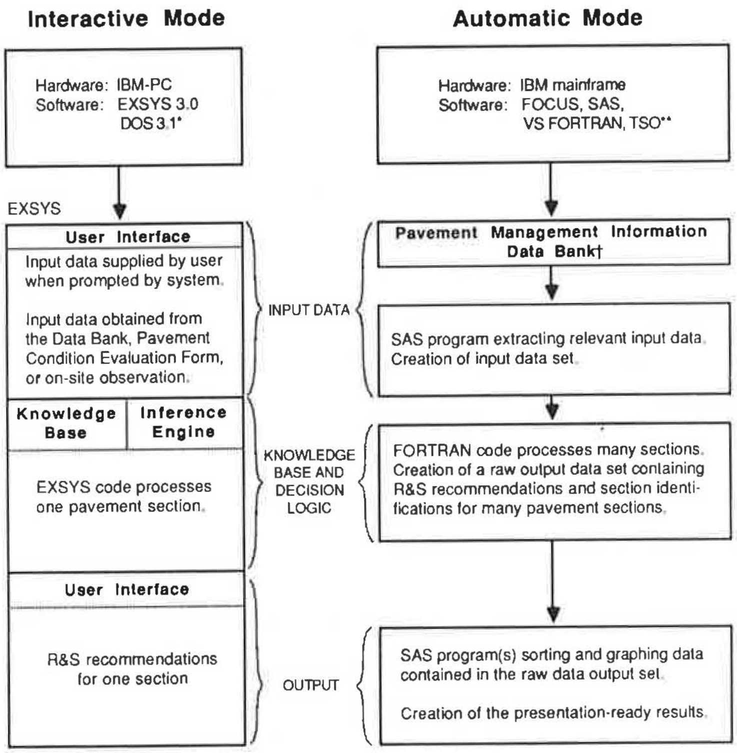
The interactive mode was developed first using an EXSYS expert system development package *(6)* that runs on IBM-compatible microcomputers. This type of hardware is readily available to the intended users.

Selection of EXSYS was based on a detailed evaluation of several expert system development shells and programming

TABLE 2 GUIDE FOR DESCRIBING DENSITY OF PAVEMENT DISTRESS *(5)*

Class or Code	Description	For all Distresses Except Transverse Cracking[a]	For Transverse Cracking Only
1	Few	< 10%	Cracks (full and/or half cracks) are more than about: 40 m apart
2	Intermittent	10 - 20%	No set pattern. Cracks (full and/or half) are about: 30 to 40 m apart
3	Frequent	20 - 50%	A set pattern. Cracks (full and/or half) are about: 20 to 30 m apart
4	Extensive	50 - 80%	Rather regular pattern. Cracks (full and/or half) are about: 10 to 20 m apart
5	Throughout	80 - 100%	Regular pattern. Cracks (full and/or half) are less than about: 10 m apart

[a] *Based on percent of surface area within the section affected by distress.*



* Disc Operating System
** Time Sharing Option
† User data are stored in FOCUS files.
FOCUS is software for information retrieval and data management.

FIGURE 1 Overall architecture of ROSE.

languages *(1)*. EXSYS was selected mainly because of its simple, rule-oriented language and powerful editing capabilities. It has a user-friendly interface that can be used to emulate the interaction a user might have with an expert to solve a problem. It may be also noted that EXSYS has been used previously for a similar problem *(7)*.

EXSYS provides a suitable programming environment for the development, calibration, testing, and running of expert systems for solving structured selection problems. The objective of such problems is a knowledgeable selection from a finite set of possible solutions. In this case, the problem was formulated as the selection from a set of numbers, 0, 1, 2, 3, . . . 10 that were used to indicate the desirability of routing and sealing. For example, definite rejection of R&S is indicated by 0, 5 may be interpreted as "maybe," and 10 means that R&S is highly desirable and is recommended with total confidence as a cost-effective treatment.

The programming was done with "if-then" rules that were used to represent knowledge about R&S. For example, if PCI is 60 or less, then do not R&S. The rules were interpreted by the EXSYS inference engine using backward chaining *(1)*. Prototype development and rule formulation and coding were greatly assisted by the EXSYS editing program and inference mechanism.

Automatic Mode

The interactive version of ROSE (programmed in EXSYS) is incompatible with the existing mainframe-based pavement management data bank. To achieve direct access to the data bank, the EXSYS rules were translated into FORTRAN using, again, the "if-then" format used by EXSYS. The recoding made possible high-speed processing of sections, direct access to the data bank, and subsequent statistical analysis of R&S recommendations obtained for hundreds of sections using SAS programs *(8)*. The purpose and sequence of programming steps are shown in Figure 1.

The bulk of the program development work was data verification and transfer, file access, system integration, and planning. The translation from EXSYS to FORTRAN alone was relatively easy, mainly because the rules in the EXSYS code had already been formulated and arranged to obtain a correct solution *(1)*.

DECISION LOGIC

The major challenge in the development of knowledge-based rules for ROSE was to take into account the influence of 15 types of surface distress in a systematic, quantifiable manner because each of the 15 types can occur at five levels of severity (Table 1) and five levels of density (Table 2) for a total of 375 (15 × 25) different conditions. Each condition may have a slightly different influence on R&S decisions.

In addition to the 15 distress variables, the desirability of R&S is also influenced by another 11 variables (Table 3) for which data are stored in the data bank. The total number of variables or factors considered by ROSE is 41. Of these, 39 are numerical variables—measured on at least ordinal scales. The task was to use the values of these 41 variables and convert

TABLE 3 QUANTIFICATION OF KNOWLEDGE-BASED PARAMETERS

VARIABLE OR FACTOR		EXTENT OR RANGE				
PCI	Range, PCI	0 - 59	60 - 65	66 - 69	70 - 74	75 and up
	CCM	0	0.2	0.5	0.9	1.0
AGE	Range, yr.	1 - 8	9 -12	13 - 15	16 and up	
	CCM	1.0	0.9	0.8	0.7	
MAINTENANCE TREAT.	Extent, %	< 10	10 - 20	20 - 50	50 - 80	80 -100
Manual Patching	CCM	0.9	0.5	0.1	0	0
Machine Patching		1.0	0.7	0.3	0	0
Spray Patching		0.9	0.5	0.1	0	0
Rout and Seal Cracks		1.1	1.0	0.9	0.8	0.7
Chip Seal		0.9	0.5	0.1	0	0
SECTION LENGTH	Range, km	0 - 10	10 - 15	>15		
	CCM	1.0	1.05	1.1		
Pavement Structural Characteristics						
TOTAL THICKNESS OF ASPHALT CONCRETE	Range, mm	< 50	50 - 70	70 - 90	90 - 100	>100
	CCM	0	0.5	0.8	0.9	1.0
PCC BASE OR PC TREATED BASE	Range, RSP	0	1	2 - 4	5 - 6	7 - 9
	PM	5	7	8	9	10
OVERLAY OF ASPH. CONC. PAV. WITH STEPPED TRANSVERSE CRACKS	Range, RSP	1	2 - 3	4 - 5	6 - 7	8 - 9
	PM	2	4	6	8	10

Legend: CCM - Cracking Condition Modifier. Multiplication Coefficient for R&S Desirability.

RSD - Routing and Sealing Desirability.

PM - Adjusts R&S Desirability according to pavement structural data.

them (using heuristic rules based on the previously outlined R&S guidelines) into one variable: desirability of R&S.

The conversion was done by developing (and calibrating) micro guidelines (based on the macro guidelines) and expressing them as rules. Moreover, to analyze fiscal consequences of R&S decisions, it was also necessary to estimate the amount of R&S for any given section. The inevitable result is a data-intensive solution procedure containing about 360 rules. The following description of the solution procedure and decision logic is abbreviated and includes only the main features.

A general decision model is shown in Figure 2. The model follows the reasoning an expert is likely to use to solve the problem. ROSE considers first the condition of (half, full, and multiple) transverse cracking in terms of severity and density using the variable BASE (as defined in the figure). Values of this variable, for all possible conditions of transverse cracking, are given in Table 4. (All values in Table 4 are based on engineering judgment.) If the condition of transverse cracking is judged to be the deciding factor (BASE ≥ 5), the left side of the decision tree of Figure 2 is used, and a preliminary conclusion regarding the desirability of R&S (MODIFIED BASE in Figure 2) is made by including two additional considerations:

- Influence of all of the remaining (14) types of distress. To provide a graduated relationship between the state of the 14 types of distress and R&S desirability, cracking distress modifiers (CDMs) given in Table 4 were established. If more than one of the remaining 14 types of distress were present, a final value of CDM was obtained by multiplying CDM-values for individual types of distress (CDMs are multiplicands).

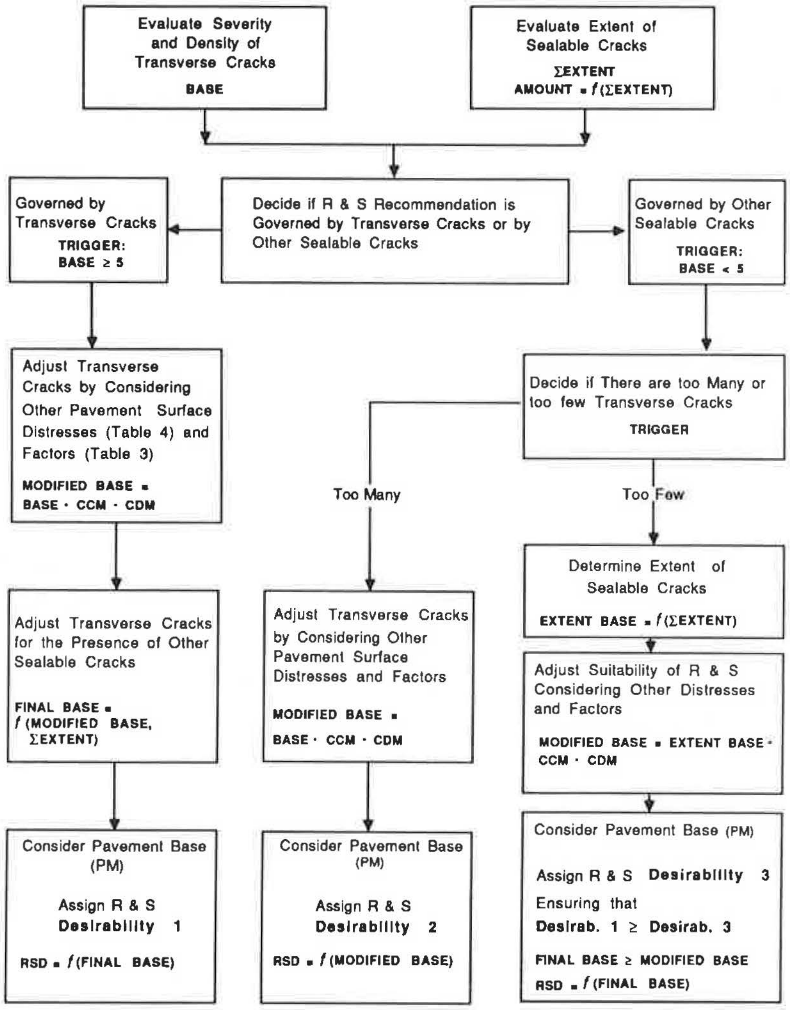


FIGURE 2 General decision model for ROSE.

- Influence of PCI, age, existing maintenance treatments, section length, and thickness of asphalt concrete. The influence of these variables was captured using cracking condition modifiers (CCMs) given in Table 3. For example, if pavement serviceability, measured in terms of the PCI, was below 60, R&S was not recommended (CCM = 0). If the PCI was in the range of 60 to 65, CCM = 0.2. CCMs for pavement age were used to capture a heuristic rule that old pavements with good performance in the past without R&S are not prime candidates for R&S in the future. An analogous approach was used to incorporate the influence of the remaining variables. CCMs were estimated using engineering judgment; operationally, CCMs are also multiplicands.

Next, the desirability of R&S was adjusted (to yield FINAL BASE in Figure 2) by considering the total amount of cracks suitable for R&S (ΣEXTENT) obtained by adding the values of the variable EXTENT (Table 4) estimated for individual types of distress. For example, if an exceedingly large amount of cracks suitable for R&S was detected, the desirability of R&S was reduced. The variable ΣEXTENT was also used to estimate the amount of R&S.

Finally, the influence of pavement structure on R&S recommendations was modeled using PM factors (Table 3). For example, if a pavement section with an asphalt concrete layer was placed over an existing asphalt concrete pavement with distinctly stepped transverse cracks (rather than over an unstepped pavement or over a granular base), its R&S desirability, which was up to this point in the range of, say 8 to 9, was increased to 10.

Returning to the top of Figure 2, if the condition of (half, full, and multiple) transverse cracking was not considered a deciding factor for R&S (BASE < 5), it was assumed that this condition existed because there were either too many or too few transverse cracks. If there were too few transverse cracks (right side of Figure 2), the total amount of cracks suitable for R&S (ΣEXTENT) was considered to assign a preliminary R&S desirability (EXTENT BASE). The preliminary R&S desirability was again adjusted by considering

- The presence of the remaining 14 types of distress (using CDMs of Table 4);
- The influence of PCI, age, and other variables (CCMs, Table 3);
- The influence of pavement base (PMs, Table 3); and
- R&S desirability based only on the condition of transverse cracks.

TABLE 4 QUANTIFICATION OF KNOWLEDGE-BASED PARAMETERS FOR PAVEMENT SURFACE DISTRESS

PAVEMENT DISTRESS MANIFESTATION				DISTRESS CONDITION, SEVERITY AND DENSITY																								
				1. VERY SLIGHT					2. SLIGHT					3. MODERATE					4. SEVERE					5. VERY SEVERE				
	NAME		PARAMETER	FEW	INT.	FREQ.	EXT.	THR.	FEW	INT.	FREQ.	EXT.	THR.	FEW	INT.	FREQ.	EXT.	THR.	FEW	INT.	FREQ.	EXT.	THR.	FEW	INT.	FREQ.	EXT.	THR.
CRACKING	**Transverse**	Half, Full and Multiple	BASE/ TRIGGER	0	1	1	2	3	1	3	7	10	8	7	8	6	5	3	4	2	1	1	0	1	0	0	0	0
			EXTENT	0	0.5	1	2	3	1	2	4	6	8	2	4	5	6	8	3	0	0	0	0	0	0	0	0	0
		Alligator	CDM	0.9	0.4	0.1	0	0	0.8	0.3	0	0	0	0.7	0	0	0	0	0.6	0	0	0	0	0.6	0	0	0	0
	Longitudinal Wheel Track	Single and Multiple	CDM	1	1	1	1	1	1	1	1	1	1	1	0.9	0.8	0.7	0.6	0.9	0.6	0.4	0.1	0	0.7	0.3	0.1	0	0
			EXTENT	0	0.5	1	2	3	1	2	4	6	8	2	4	5	5	7	2	0	0	0	0	0	0	0	0	0
		Alligator	CDM	0.9	0.4	0.1	0	0	0.8	0.3	0	0	0	0.7	0	0	0	0	0.6	0	0	0	0	0.6	0	0	0	0
	Centerline	Single and Multiple	CDM	1	1	1	1	1	1	1	1	1	1	1	1	0.9	0.8	0.7	0.9	0.7	0.6	0.3	0	0.7	0.4	0.1	0	0
			EXTENT	0	0.5	1	1	2	0.5	1	2	4	5	1	2	5	7	9	1	0	0	0	0	0	0	0	0	0
		Alligator	CDM	1	0.7	0.4	0.2	0.1	0.9	0.6	0.3	0.2	0.1	0.9	0.4	0.1	0	0	0.8	0.3	0.1	0	0	0.6	0.2	0	0	0
	Longitudinal Meander and Midlane		CDM	1	1	1	1	1	1	1	1	1	1	1	0.9	0.8	0.7	0.6	0.9	0.6	0.4	0.1	0	0.7	0.3	0.1	0	0
			EXTENT	0	0.5	1	1	2	0.5	1	2	4	5	1	2	4.5	5	7	2	0	0	0	0	0	0	0	0	0
	Random		CDM	1	1	1	1	1	1	1	1	1	1	1	0.9	0.8	0.7	0.6	0.9	0.6	0.4	0.1	0	0.7	0.3	0.1	0	0
			EXTENT	0	0.5	1	1	2	0.5	1	2	4	5	1	2	3.5	3	4	1	0	0	0	0	0	0	0	0	0
	Pavement Edge	Single and Multiple	CDM	1	1	1	1	1	1	1	1	1	1	1	1	0.9	0.8	0.7	0.9	0.7	0.6	0.3	0	0.7	0.4	0.1	0	0
			EXTENT	0	0	0.5	1	1.5	0.5	1	2	3	4	1	2	3.5	3	4	1	0	0	0	0	0	0	0	0	0
		Alligator	CDM	1	0.7	0.4	0.2	0.1	0.9	0.6	0.3	0.2	0.1	0.9	0.4	0.1	0	0	0.8	0.3	0.1	0	0	0.6	0.2	0	0	0
Ravelling and C. Agg. Loss			CDM	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0.6	0.1	0.1	0	0.4	0.2	0.1	0	0
Flushing, Ripping and Shoving, Wheel Track Rutting, and Distortion			CDM	1	1	1	1	1	1	1	1	0.9	0.9	0.9	0.8	0.4	0.1	0	0.9	0.7	0	0	0	0.6	0.2	0	0	0

Legend:
BASE - Indicates suitability of transverse half, full & multiple cracks for R&S.
EXTENT - Approximate relative extent of transverse half, full & multiple cracks.
TRIGGER - Determines if R&S desirability is governed by transverse cracks or by other sealable cracks.
CDM - Cracking Distress Modifier. Multiplication coefficient for R&S desirability.
EXTENT - Approximate relative extent of sealable cracks.

Last, in the case in which too many transverse cracks were deemed to exist to justify R&S, the situation was duly noted as a basic section characteristic and an R&S desirability, however small, was also established.

APPLICATION

The input to ROSE is the present pavement condition. The outputs are R&S recommendations that are considered valid for up to 1 year. This should be acceptable in practice because R&S treatments are not usually planned more than 1 year in advance and any changes in pavement performance during this period are often too small to be measured. Also, the experts interviewed during the development of R&S guidelines worked on the assumption that although their R&S recommendations cannot be implemented immediately, they should be before the end of the next construction season.

In addition to assigning R&S desirability, ROSE also estimates for each section the total amount of cracks recommended for R&S in terms of meter per kilometer of two-lane highway. Further, ROSE classifies each section in one of the following three categories:

1. Sections with too few sealable cracks to warrant R&S next year but that may require R&S in the future,
2. Sections that may require R&S within 1 year, and
3. Sections that already have too many cracks to benefit from R&S.

ROSE was designed to fully use all available surface distress data, and other data stored in the data bank, without any unnecessary assumptions or simplifications. It would be possible to significantly reduce the number of rules (360) by asking the user to input more global data. For example, by asking questions such as "what is the approximate amount of sealable cracks in meters per kilometer?" instead of inputting detailed data and expecting ROSE to calculate the amount.

Both the interactive and the automatic versions of ROSE use identical knowledge base, input data, and decision logic. The exceptions are input data and relations concerning pavement structural characteristics (last part of Table 3). The data bank does not yet contain detailed pavement structural data for all pavement management sections. For this reason, the automatic version assumes that asphalt concrete thickness is about 100 mm or more and that it was placed over a granular base or asphalt concrete base without distinctly stepped transverse cracks. These assumptions are usually met, and in many MTC districts the degree of compliance is about 95 percent.

ROSE was calibrated and tested on about 100 pavement sections, located in different parts of Ontario, until a satisfactory level of system reliability and accuracy was achieved. The calibration was done by using ROSE in the interactive mode and taking advantage of the editing features and the inference engine supplied by EXSYS.

Field verification of the results indicates that the main limitation on the reliability and accuracy of ROSE is the correctness of input distress data obtained from the data bank. This should be overcome with time when it is realized how the use of distress data has been expanded by ROSE.

RESULTS

Interactive Mode

In interactive mode, ROSE can be used as a decision-making or a decision-support system. It performs at the level of a pavement maintenance professional who is roughly in agreement with the R&S methodology embedded in the system and applies this methodology consistently. This assumes that the input data used are the routinely available data taken from the data bank or directly from a field evaluation form *(5)*. However, ROSE does not outperform an expert because the expert, if he or she so chooses, can benefit from evaluating the pavement in situ and obtaining specific, up-to-date pavement deterioration data for the sole purpose of recommending R&S.

Analysis of one pavement section on an IBM XT microcomputer, including supplying data for up to 40 variables, takes about 4 min. ROSE operates as any other well-designed interactive program. In addition, it contains several enhancements. For example,

- The user, when prompted by ROSE for input data, can ask "Why?" ROSE answers why the data are needed. This is done by displaying, on-screen, the first applicable rule for which the data are needed.
- The change and rerun option and the editing program enable the user to easily review and change any input data, or part of the EXSYS code, and rerun the program.

An example of an R&S problem, solved by ROSE in the interactive mode, is shown in Figure 3.

Automatic Mode

ROSE's performance in the automatic mode is excellent. Assuming that an expert cannot visit hundreds of sections and uses the same information as that available to ROSE, ROSE's accuracy is similar to that of a patient and consistent expert and the results are available more or less instantaneously.

The desirability of R&S treatments was evaluated by ROSE for two MTC regions, Southwestern Region and Northern Region, using the most recent pavement deterioration data. In all, 488 sections were evaluated in the Southwestern Region and 396 sections in the Northern Region. The highway networks of the two regions are roughly equal in size and, together, comprise about 7200 centerline kilometers (about 40 percent of the total provincial highway network). An example output listing is given in Table 5. The listing identifies 10 pavement sections in the Southwestern Region that would most benefit from R&S. The sections on the list should be considered prime candidates for R&S in 1987. The distribution of the desirabilities with which the sections were recommended for R&S in the two regions is shown in Figure 4.

ROSE can also be used to evaluate the funding consequences of different R&S strategies. For example, assuming that the cost of R&S is $1 per meter, the R&S cost for all sealable cracks in the Southwestern Region was estimated to be $2.6 million (Figure 5), and the cost for the sections recommended for R&S next year with a desirability of 7 or more was estimated to be $1.2 million.

Given:

A two-lane, 9-km-long, 10-year-old pavement section. It has an 80-mm-thick original asphalt concrete layer placed over a granular base. Its PCI is equal to 70, and the section has only three surface distresses (unusual but simple):

a) Transverse cracking (half, full, and multiple), which is rated as slight and occurring extensively.
b) Centerline cracking (single and multiple) rated as slight and frequent.
c) Wheel track rutting considered to be slight and extensive.

In addition, there are also few manual patches.

Task:

Estimate R&S desirability for this section and the approximate cost of R&S.

Solution by ROSE:

1. Considering transverse cracking, BASE value is 10 (Table 4) and R&S desirability is governed by transverse cracking (Figure 2). EXTENT/TRIGGER is 6.
2. Considering centerline cracking, CDM is 1, and EXTENT is 2 (Table 4).
3. Considering wheel track rutting, CDM is 0.9. (There is no EXTENT because rutting is not a sealable distress.)
4. PCI has the corresponding CCM equal to 0.9 (Table 3), CCM for age is 0.9, CCM for a few manual patches is 0.9, CCM for length is equal to 1, and CCM for total thickness of asphalt concrete is 0.8.
5. MODIFIED BASE = $10 \times 1 \times 0.9 \times 0.9 \times 0.9 \times 0.9 \times 1 \times 0.8 = 5.2$ (based on equation in Figure 2).
6. ΣEXTENT = $6 + 2 = 8$.
7. MODIFIED BASE is adjusted by a multiplication coefficient of 0.9 (the amount of cracks for R&S is considered to be somewhat on the low side) resulting in 4.7 (5.2×0.9).
8. The amount of cracks for R&S (AMOUNT) is estimated to be 663 m/km. The estimate is done using the heuristic equation

   AMOUNT = $104 \times \Sigma$EXTENT $- 165$, where AMOUNT > 0.

Report by ROSE:

1. Desirability of R&S: 5 (rounded from 4.7).
2. Amount of sealable cracks: 663 m/km.

Conclusions:

1. The section may still benefit from R&S. However, do not R&S before considering first sections with R&S desirability higher than 5.
2. Assuming R&S cost of $1 per meter, the total cost is estimated to be $6,000.

FIGURE 3 Example of R&S solution by ROSE.

TABLE 5 LISTING OF ALL PAVEMENT MANAGEMENT SECTIONS IN SOUTHWESTERN REGION WITH R&S DESIRABILITY OF 9

OBS	RSD	LHRS	Offset	Length	DIST	PCI	Age	BASE	RSCG	Amount	Total
479	9	12170	10.0	16.0	1	90	2	10	20	663	10,608.0
480	9	23930	0.6	17.0	1	85	4	8	20	1,183	20,111.0
481	9	29210	4.0	2.9	1	88	4	10	20	559	1,621.1
482	9	47920	0.0	24.0	1	78	5	8	20	923	22,152.0
483	9	29168	0.0	14.0	1	80	6	10	20	455	6,370.0
484	9	11840	0.0	5.5	2	90	6	8	20	1,027	5,648.5
485	9	16190	1.3	16.0	3	91	1	10	20	559	8,944.0
486	9	24070	1.6	18.0	3	86	5	8	20	923	16,614.0
487	9	38400	0.6	23.0	3	75	7	8	20	1,079	24,817.0
488	9	24510	0.0	25.0	3	75	8	8	20	1,547	38,675.0

NOTE: RSD = routing and sealing desirability; OBS = section number (sections are sorted according to RSD; the total number of sections analyzed was 488); LHRS and Offset = section identification parameters used by location referencing system; Length = section length in km; DIST = MTC district number; PCI = pavement condition index; Age = pavement age in years; BASE = defined in Table 4; RSCG = R&S classification category (20 indicates that the section should be routed and sealed within 1 year); Amount = estimated amount of cracks to be routed and sealed in m per km for a two-lane highway; and Total = total estimated amount of cracks to be routed and sealed in m per section.

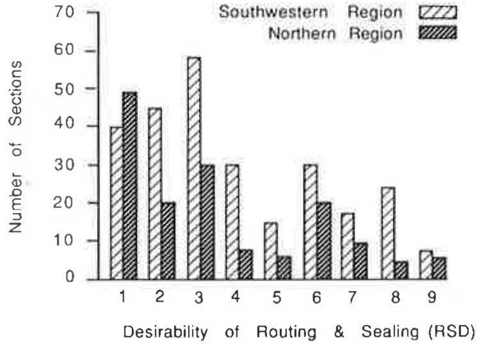


FIGURE 4 Routing and sealing recommendations for all sections in Southwestern and Northern regions.

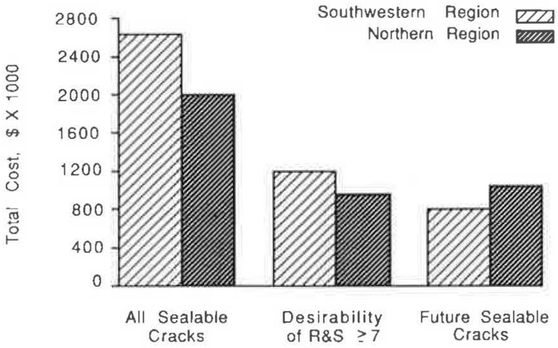


FIGURE 5 Consequences of different routing and sealing policies.

CONCLUSIONS AND RECOMMENDATIONS

Expert system technology can improve the design, planning, and programming of pavement preservation treatments. This can be achieved by efficient and consistent application of the encoded knowledge and experience of many pavement engineers. At a project level, knowledge-based expert systems can recommend routine preservation treatments enabling experts to concentrate on more difficult tasks. At a network level, these systems can quantify the consequences of pavement preservation policy decisions for planning and programming.

The development, testing, and calibration of a prototype version of ROSE were made much easier and more efficient by using the inference engine and editing features of the EXSYS expert system development shell (and, of course, the interactive mode of ROSE runs under EXSYS and uses its user interface). It is thus possible to realize significant productivity advantages in developing prototype expert systems, or other computer programs, using artificial intelligence techniques (for example, mechanical interpretation of the knowledge base by an inference engine), even though the finished expert systems or computer programs may not employ any artificial intelligence techniques *(9)*.

ROSE, a knowledge-based expert system for recommending routing and sealing of asphalt concrete pavements in cold areas, can quickly and reliably analyze and rank pavement sections in terms of their suitability for routing and sealing. The routing and sealing recommendations given by ROSE and their correctness are governed by the preliminary routing and sealing guidelines. Any future changes in the guidelines should be incorporated in ROSE.

Because of huge investments in existing pavement management systems, knowledge-based expert system technology must be integrated and made fully compatible with the existing pavement management processes.

EXSYS, in common with most existing rule-based expert system software, has many advantages, but it does not yet represent an "ideal" programming environment. For example, it requires use of domain rules to create contextual assertions that control the application of other rules.

Because of their potential for increasing effectiveness through improvement of pavement management information, the development of knowledge-based expert systems should continue.

ACKNOWLEDGMENT

The authors wish to acknowledge the significant assistance of K. C. Choi, engineering student, University of Waterloo. Choi did most of the programming work and also participated in testing the prototype system and data analysis.

REFERENCES

1. J. J. Hajek. *Life-Cycle Pavement Behaviour Modeling Using A Knowledge-Based Expert System Technology.* Ph.D. dissertation. University of Waterloo, Ontario, Canada, 1987.
2. F. Hayes-Roth. Knowledge-Based Expert System. *IEEE Computer*, Vol. 17, No. 10, 1984, pp. 263–273.
3. K. A. Godfrey. Expert Systems Enter the Marketplace. *Civil Engineering*, ASCE, May 1986, pp. 70–73.
4. G. J. Chong and W. A. Phang. *Sealing Cracks in Flexible Pavements in Cold Areas: An Audio-Visual Script.* Report PAV-83-01. Ministry of Transportation and Communications, Downsview, Ontario, Canada, 1983.
5. J. J. Hajek, W. A. Phang, W. A. Wrong, A. Prakash, and G. M. Stott. *Pavement Condition Index (PCI) for Flexible Pavements.* Report PAV-86-02. Ontario Ministry of Transportation and Communications, Downsview, Ontario, Canada, July 1986.
6. *EXSYS—Expert System Development Package.* EXSYS, Inc., Albuquerque, N.Mex., 1985.
7. S. G. Ritchie, C-I. Yeh, J. P. Mahoney, and N. C. Jackson. Development of an Expert System for Pavement Rehabilitation Decision Making. In *Transportation Research Record 1070*, TRB, National Research Council, Washington, D.C., 1986, pp. 96–103.
8. *SAS User's Guide: Statistics, Version 5 Edition.* SAS Institute, Inc., Cary, N.C., 1985.
9. J. Doyle. Expert Systems and the Myth of Symbolic Reasoning. *IEEE Transactions on Software Engineering*, Vol. SE-11, No. 11, Nov. 1985, pp. 1386–1390.

Intersection Advisor: An Expert System for Intersection Design

DONALD A. BRYSON, JR., AND JOHN R. STONE

The Intersection Advisor is a prototype expert system that recommends geometric modifications to improve intersection operation. It complements existing microcomputer programs that consider the other two aspects of intersection design, volumes and signalization. Intersection Advisor is intended for eventual incorporation in a comprehensive, interactive intersection design package. M.1, a knowledge-based expert system development tool, was used to develop the Intersection Advisor to run on IBM or IBM-compatible microcomputers. During an interactive consultation the advisor requests information on the intersection volumes, critical movements, geometry, and constraints on approach improvement. It then recommends the most efficient improvements for each approach by generating one of nearly 600 possible reports. Recommendations are arrived at by determining an "ideal" lane configuration for the given traffic flows. The ideal design is checked against the improvement constraints, and a next-best design is selected, if necessary. The best feasible design is then compared with the existing design, and the user is informed of any modifications required. Intersection designs produced with the advisor compare well with those produced using the guidelines of the 1985 *Highway Capacity Manual.*

The operational characteristics of a signalized intersection are determined by the interactions of three basic components: traffic flows, geometry, and signalization. Intersection design involves the manipulation of geometry and signalization with respect to traffic flows in order to maximize operating efficiency (Figure 1). Optimization is difficult to achieve, however, because of the high degree of interdependence among the design parameters and the variety of constraints commonly encountered. Even experienced engineers cannot always determine the optimum design for a complex intersection, although their initial solution to a given problem is usually remarkably close. New engineers proceed by trial-and-error, gradually developing a "feel" for appropriate solutions. Obviously, there is systematic, reproducible reasoning involved—a combination of acquired factual knowledge and problem-solving techniques—that can be defined as expertise.

Computer techniques have had a significant impact on the design and analysis of signalized intersections. Capacity analysis programs (MCTRANS, CAPSSI, SIGNAL, CMA) have been available for some time, and the publication of the new *Highway Capacity Manual* (HCM) *(1)* has generated a number of new programs (SICA, NCAP, HCS) based on its methodology. Signal optimization programs, such as SOAP and INTERCALC, have also proven useful *(2)*.

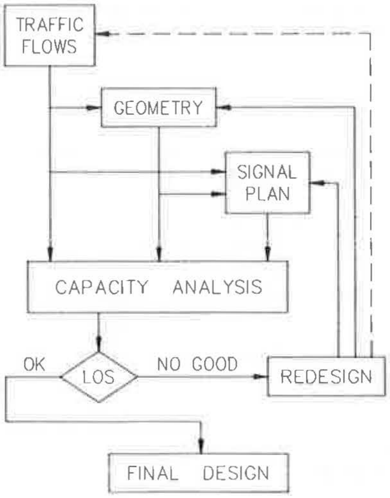


FIGURE 1 Intersection design process.

Existing programs are algorithmic in nature and deal with the numerical components of intersection design, such as volume-to-capacity calculations and signal optimization. These programs enhance the design process by increasing the speed and accuracy of calculation-intensive tasks. These procedural programs cannot, however, assure that the options being analyzed are necessarily reasonable, nor can they assure that the best alternative will be selected. They offer little guidance in deciding the best way to improve the operation of a deficient intersection. An engineer's experience and judgment are the most useful aids in solving this aspect of the design problem—in determining, for example, whether to add a turn lane or revise a signal plan, or both. When the engineer has chosen a design alternative, he must then determine the operating characteristics of the intersection either manually or by computer. The process is repeated until an acceptable design is achieved.

To provide an integrated, interactive environment for selecting and analyzing intersection designs, it is necessary to develop software tools that incorporate professional experience and judgment. These tools must be able to perform the kind of heuristic reasoning required to efficiently generate and evaluate options for improving a design. Such a system has the potential to greatly improve the existing design process by combining tasks that are currently performed manually or with various isolated programs (Figure 2). An essential component of this system is software that can handle the various aspects of geometric design.

Department of Civil Engineering, North Carolina State University, Box 7908, Raleigh, N.C. 27695.

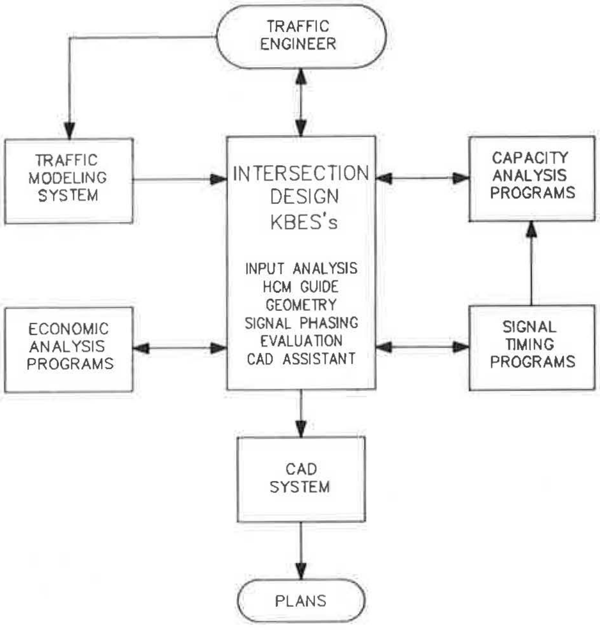


FIGURE 2 An integrated intersection design system.

The purpose of this paper is to demonstrate the applicability of knowledge-based expert system (KBES) technology to signalized intersection design, and to approach geometry in particular. The Intersection Advisor is a functional KBES prototype that recommends geometric modifications to improve the operation of signalized intersections. Like other knowledge-based systems, the advisor reproduces knowledge typically possessed by an expert in a particular field and reaches conclusions that lead to the solution of a problem by systematically applying appropriate reasoning techniques (rules) to its knowledge of the problem area (facts). Experience with the Intersection Advisor suggests that such a system has practical applications in making decisions concerning the geometric design of signalized intersections.

The geometric design problem and the objectives for the development of the Intersection Advisor are discussed. A sample problem is presented to familiarize the reader with the operation of the advisor and to lay the groundwork for an explanation of the solution strategy and its implementation in the knowledge base. The paper concludes with a discussion of results and their influence on the direction of future development of the Intersection Advisor.

INTERSECTION ADVISOR

Geometric Design Problem

Of the three design components described earlier (traffic flows, geometry, and signalization), only one, the geometric element, includes design variables essential for planning, design, and analysis. Of the parameters that the engineer can readily control, those with the greatest direct impact on intersection performance are the number of lanes per approach and the movements permitted from each lane. This was the specific area chosen to test the feasibility of applying KBES technology to intersection design.

The problem of determining the most efficient lane configuration for a given set or sets of turning movements is representative of the intersection design process as a whole. It incorporates both the generation of solutions (How can operations be improved?) and their evaluation (Which solution is best?). The lane geometry problem is narrow enough to provide a reasonably simple prototype yet complex enough to be challenging and realistic. The representational logic required to relate geometric features and traffic flows transfers directly to signalization and other problems. It offers a suitable foundation for the eventual development of a comprehensive intersection design system.

Guidelines for Development

Two major objectives guided the development of the Intersection Advisor. The first objective was to define a set of rules and facts for identifying the most efficient lane configuration for a given set of turning movements. This body of knowledge corresponds to the "expertise" of an experienced traffic engineer and reproduces the reasoning process he employs in developing an appropriate design. These rules are transferable to any KBES for intersection design.

The second objective was to define a logical system for representing the physical and operational relationships among approach legs, individual lanes, and turning movements. This representation scheme provides a framework for implementing the reasoning processes identified previously. The second objective is essential to an efficient interactive environment and applies directly to other aspects of intersection design. Successfully achieving these two objectives helps assure that the Intersection Advisor can be easily expanded from a prototype to a functional system, regardless of changes in the problem domain or in the implementation environment.

Several secondary objectives were also established: *(a)* Recommendations should be presented as incremental changes to an existing design rather than as a complete intersection. The use of an incremental approach as opposed to an absolute approach models typical intersection improvement projects and facilitates the evaluation of various alternatives, as in a cost-benefit analysis. *(b)* The user must be able to constrain the set of potential design solutions to reflect considerations such as limitations in right-of-way availability. *(c)* To provide maximum flexibility and ease of use, the user should be able to analyze individual components of an intersection without analyzing the intersection as a whole. *(d)* Recommendations should include an explanation of the anticipated impacts of the proposed modifications on intersection performance. *(e)* Input requirements must be simple, rational, and consistent with standard practice.

Simplifying Assumptions

It was decided to make the Intersection Advisor consistent with, and complementary to, the planning analysis methodology described in Chapter 9 of the *Highway Capacity Manual*. The advisor would address all of the parameters included in the planning method, with the goal of identifying the lane configuration that would most economically result in a sum of critical movements of fewer than 1,200 vehicles per hour. The set of intersections to be analyzed was limited to standard four-way intersections with no more than four lanes per approach (no severe skewing or offsets and no one-way streets). The solution set was limited to approaches with fewer than eight lanes (a maximum of three through lanes with dual turn lanes in each direction). The expertise incorporated in the Intersection Advisor was obtained from Chapter 9 of the HCM and from the authors' own experience in teaching and performing intersection analysis and design.

Using the Intersection Advisor

A consultation with the Intersection Advisor combines knowledge from two different sources—the user and the program itself. The user supplies his knowledge of a particular intersection in response to questions based on generalized facts and rules pertaining to efficient relationships between turning movements and lane geometry (Figure 3). For each intersection approach being analyzed, the advisor also asks the user to provide intersection volumes, critical movements, and the existing geometry. Turning movements and critical movements can be obtained from the simple manual planning analysis calculations described in Chapter 9 of the HCM or from other microcomputer programs [*3* and work by S. Gayle and J. Papaleo on Signalized Intersection Capacity Analysis (SICA) Using the 1985 *Highway Capacity Manual*, 1986].

User-supplied geometric information is determined from Figure 3 and input to the advisor in response to prompts. The combination of eight basic approach types and two critical movement conditions enables the user to choose from 16 different approach cases for a given set of turning movement volumes.

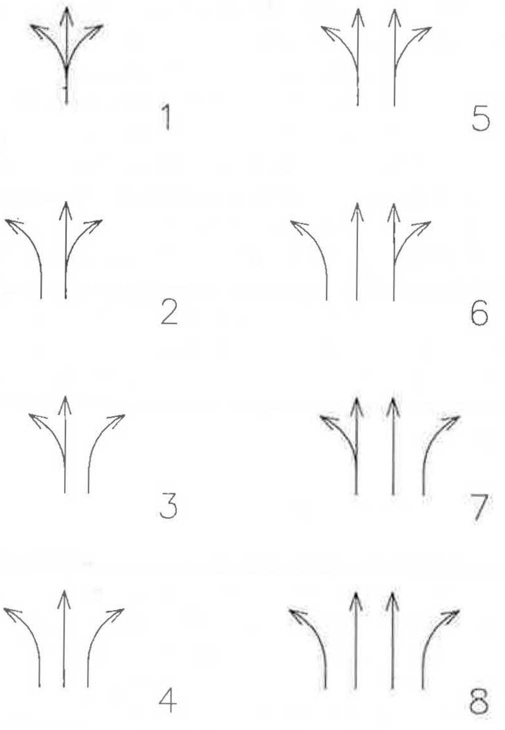


FIGURE 3 Basic lane configurations.

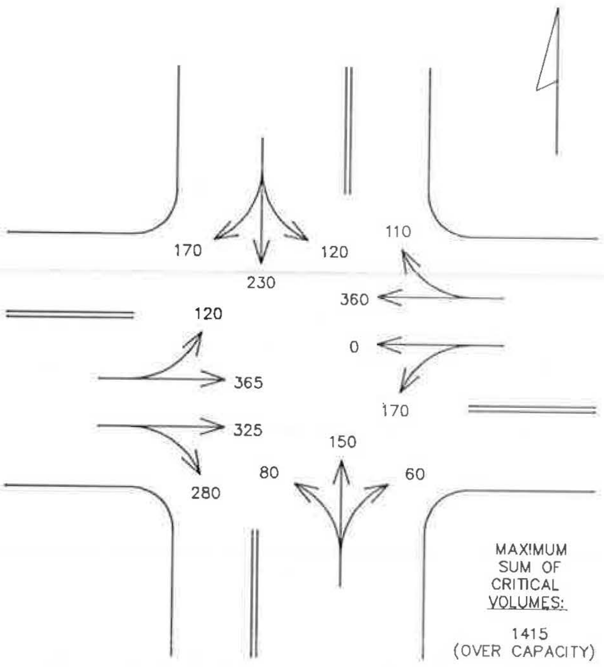


FIGURE 4 Initial conditions.

Finally, the advisor asks the user to specify the maximum number of lanes available for improving the approach. This allows the user to account for physical or right-of-way restrictions.

Sample Problem

To demonstrate how the Intersection Advisor is used, a consultation was run based on Sample Calculation 5 in Chapter 9 of the HCM. Figure 4 shows the geometry and lane volumes of the HCM problem. The existing intersection operates with an overcapacity critical movement summation of 1,415 vehicles per hour, and the HCM advises that "separate left-turn lanes might be considered for each approach, subject to physical constraints." This recommendation leads to a new critical summation of 1,135 vehicles per hour and improved intersection operation.

The Intersection Advisor, on the other hand, recommends the addition of separate left-turn lanes to only two approaches (Figure 5), a design improvement that, like the HCM approach, leads to a critical movement summation of 1,135 vehicles per hour. The advisor's recommendation, therefore, provides an equivalent improvement at half the cost. An argument for the HCM recommendation might be made, however, on the basis of symmetry. This type of policy issue will be addressed in future versions of the Intersection Advisor and is discussed in more detail later in this paper.

An excerpt from the consultation session follows.

Do you wish to consider improvements to the southbound approach?
> > yes.
Enter the number of southbound vehicles turning left during the design hour.
> > 120.

Enter the number of southbound vehicles turning right during the design hour.

>> 170.

Enter the number of southbound vehicles going straight during the design hour.

>> 230.

Are left-turn or through movements the critical movements on the southbound approach? Enter "left" or "thru."

>> thru.

Enter the code number (1–8) of the basic lane configuration corresponding to the existing southbound approach.

>> 1.

Enter the maximum number of lanes desired for the southbound approach.

>> 4.

Enter the maximum number of through lanes desired for the southbound approach.

>> 2.

Recommendations for southbound approach: Providing an exclusive left-turn lane (Lane Configuration 2) will significantly improve the critical through movement on this approach.

Four consultations of this type result in Figure 5.

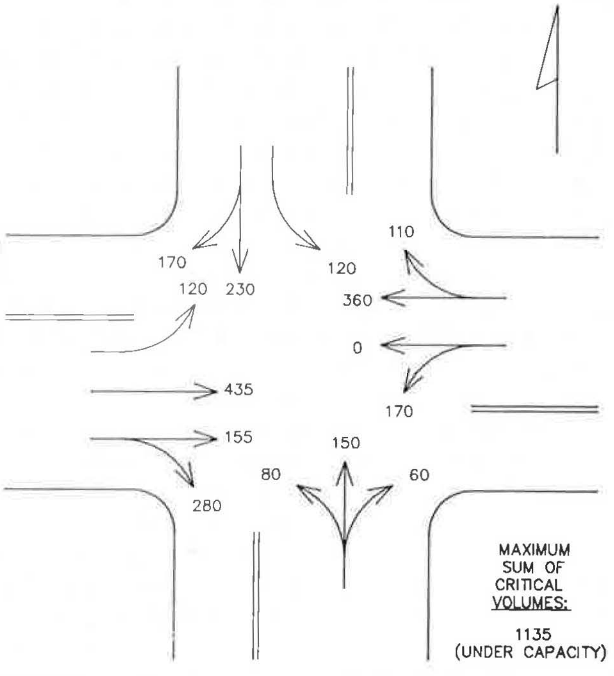


FIGURE 5 Solution recommended by Intersection Advisor.

KNOWLEDGE BASE

Solution Strategy

Appendix I to Chapter 9 of the HCM describes some general rules-of-thumb traffic engineers use to help determine a suitable lane configuration for a given set of traffic flows. These heuristic "suggestions" address such factors as the capacity of a given type of lane and the volumes and types of movements occurring in adjacent lanes. One such guideline states, "As a general suggestion, an exclusive right-turn should be considered when the right-turn volume exceeds 300 vph and the adjacent main-lane volume also exceeds 300 vphpl" (p. 9–64). The expert uses heuristic reasoning of this type along with pertinent standards and policy guidelines and the results of the planning analysis methodology to determine what, if any, modifications need to be made to the existing design. This is the body of knowledge required to meet the first objective in developing the Intersection Advisor, the capturing of expertise.

Before this knowledge could be incorporated into an expert system, however, it was necessary to meet the second objective, the development of an efficient representational schema. This objective was achieved by defining the configuration of lanes in a given approach as the basic logical unit in the knowledge base. Identifying the entire approach as a single entity eliminates the need to explicitly define the properties of each of the six lane types typically used in four-way intersections. This is a significant simplification because the properties of a given type of lane are often influenced by an adjacent lane. These interactions (such as the shifting of traffic between two lanes or the blocking of one movement by another) are implicit in the definition of each approach configuration. Furthermore, it is no longer necessary to prevent the occurrence of illogical lane configurations (such as a right-turn lane to the left of a left-turn lane). Assuming a maximum cross section of seven lanes, there are more than 300,000 ways of arranging the six different lane types. The incorporation of rules to eliminate all of the illogical and impractical solutions would involve considerable overhead, even in the simple problem domain of the Intersection Advisor.

Because the number of feasible configurations makes up a small fraction of the total number of possible configurations, the solution set and overhead are greatly reduced by predefining the acceptable approach configurations. The Intersection Advisor requires only eight basic lane configurations (Figure 3). Each one has specific properties that determine how it will handle various ranges and combinations of turn volumes. For a given set of traffic flows, one of these configurations will provide the most efficient service, based on the heuristic strategy implemented.

KBES Development Tool

Conceptually, traffic flows can be thought of as operands, and the basic lane configurations as operators, each of which has an associated cost. The result of an operation is a level of service. Thus the geometric design problem becomes one of selecting for a given operator the operand that provides a result above some minimum value at the lowest cost. Although this analogy is somewhat oversimplified, it serves to demonstrate how the geometric design process can be represented as a structured selection problem. Structured selection problems are generally appropriate for solution by knowledge-based systems. In particular, M.1 is well suited for solving this type of problem.

The Intersection Advisor was developed using M.1, a knowledge engineering software tool intended for the design and implementation of stand-alone expert systems on IBM personal computers or compatibles *(4)*. Although M.1 has limitations as a practical application tool, it does allow the rapid development of prototype systems and is useful for validating concepts and logic before intensive software development is undertaken.

An M.1 system consists of a knowledge base, a cache, and an inference engine. The knowledge base contains facts and rules pertaining to a specific application. The cache is the storage area for all intermediate and final conclusions, as well as all user input. The inference engine is the mechanism by which M.1 systematically searches for needed values in order to reach a particular goal. This search mechanism obtains values from the cache, from the knowledge base, or from the user. Figure 6 shows how the system architecture relates to the intersection design process in the Intersection Advisor.

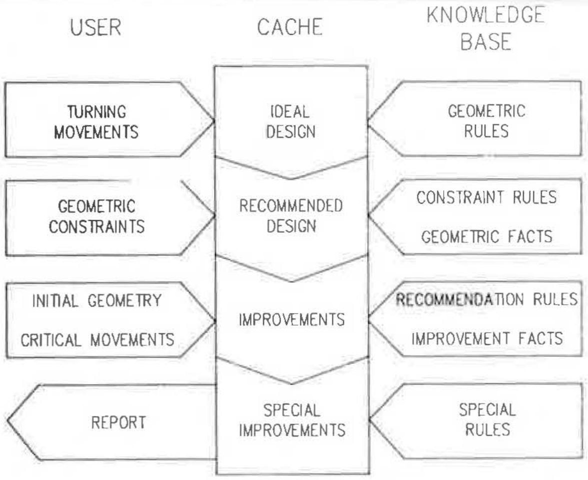


FIGURE 6 Organization of a consultation.

Implementation

The Intersection Advisor uses a set of mutually exclusive geometric rules to reach its first subgoal: selection of an initial "ideal" approach configuration that most efficiently accommodates the given turning movements. These rules involve a series of comparisons between the turning movement volumes and the capacity characteristics of the eight basic lane configurations (Table 1). Because there is always one ideal basic configuration for a given set of turning movements, as soon as a geometric rule has fired, the advisor proceeds to the next subgoal.

TABLE 1 DECISION TABLE FOR BASIC GEOMETRIC RULES

CONDITIONS	OPTIONS										
T < 450	T	T	T	T	T	T	F	F	F	F	ELSE
L < 100	T	T	T	T	F	F	T	T	F	F	
R < 300	T	T	F	F	-	-	T	F	T	F	
T+L+R > 450	T	T	T	T	-	(T)	(T)	(T)	(T)	(T)	
R+T < 500	-	-	-	-	T	F	-	-	-	-	
L+T < 400	-	-	T	F	-	-	-	-	-	-	
3L < R	T	F	-	-	-	-	-	-	-	-	
ACTIONS											
NUMBER OF IDEAL LANE CONFIGURATION	3	2	3	4	2	4	5	7	6	8	1

" - " indicates immaterial value.
"( )" indicates redundant value.

When the ideal basic configuration has been established, a set of constraint rules ensures that this design does not violate any restrictions, caused by physical constraints or limits in right-of-way availability, the user may have placed on approach expansion. The goal of these rules is to identify the best of the allowable basic lane configurations (the "recommended" configuration). If the constraint rules reject the ideal configuration, the advisor generates the next-best alternative by comparing the ideal design with the available approach cross section defined by the user. The properties inherent in the definition of each of the basic lane configurations supply enough information for the advisor to select the next-best configuration directly. It is not an iterative process. The recommended design is the one that most closely matches demand volumes without exceeding lane limitations.

The next subgoal is to determine the differences between the recommended and the existing design. These differences are defined in terms of improvements to the existing design. The Intersection Advisor uses a set of relational improvement facts and recommendation rules to describe the geometric and operational differences between logical combinations of existing and recommended designs. By combining knowledge of the inherent characteristics of the recommended configuration with the geometric and operational data provided by the user, the recommendation rules identify a specific set of modifications to improve the intersection approach. Together, these modification descriptions define every feasible transition from one basic approach configuration to another. The elimination of infeasible solutions from the solution set, combined with the approach-based representational scheme, means that only 56 modification descriptions are needed. This is a significant reduction, considering the total number of combinations of lanes that is mathematically possible. The Intersection Advisor provides a brief description of the nature and magnitude of the benefits expected if its recommendations are implemented, and it recognizes when the suggested improvements will require widening the opposite leg of the intersection.

Finally, there is a set of special rules that generates recommendations related to multiple turn lanes and very high through volumes. The three independent special rules complement the eight basic approach configurations and provide a total of 64 ultimate designs. In all, nearly 600 unique reports for feasible recommendations are possible for each approach analyzed because of the various combinations of ideal designs, recommended designs, modification descriptions, and special recommendations.

DISCUSSION OF RESULTS

Evaluation Strategy

Validating a knowledge-based expert system is not as straightforward as validating an algorithmic program. The nature of the problem is often such that there is no single solution that can be proven "best." Two experts can come up with different solutions to the same problem, both of which are acceptable and completely defensible, but neither of which is necessarily optimum. These differences can usually be attributed to variations in the policies or practices being followed. Most expert systems are derived from the knowledge of a single expert (or at most a small number of experts). They therefore reflect the policies and practices preferred by that expert. In evaluating the performance of expert systems, it is important to consider two questions. First, does the system accurately and consistently reproduce an accepted approach to the problem? And second, can the system be modified to reflect other valid policies and practices?

The design policy implemented in the Intersection Advisor is a conservative one. It encourages balanced volume-to-capacity ratios for all lanes and is intended to provide a basis for a signal plan that will offer a high level of service. The advisor concentrates on improving the operation of the entire intersection by increasing the capacity of each approach. This policy gives some consideration to critical movements, but otherwise it is not very sensitive to interactions between approaches. This is not a major limitation, however, because approach capacity is primarily a function of lane geometry, whereas conflicts and other interactions between approaches are highly dependent on signalization.

The results of dozens of test cases are consistent with this policy and with the guidelines and examples contained in Chapter 9 of the HCM. The test cases include simple design problems like Sample Calculation 5 in the HCM, intersections with severe improvement constraints, and those with volumes requiring up to seven lanes in each direction. A typical consultation lasts about 5 min, and most users find the advisor easy to run and understand.

In no case has it been possible to significantly improve the operation of an intersection by making improvements in addition to, or instead of, those recommended by the advisor. It should be noted that the advisor does not recommend the removal of existing lanes that are unnecessary because this would not typically be practical. The advisor will, however, suggest that existing lane uses be redefined if warranted. The advisor does have a tendency to overdesign in certain cases because it seeks to achieve a target volume-to-capacity ratio for each lane and does not consider signalization. The advisor also ignores intersection symmetry in making its recommendations. All of these traits are consistent with the planning method and with the advisor's design policy.

Not only can the Intersection Advisor accurately and consistently implement a specific design policy, its policy can be changed relatively easily. Constants in the geometric rules can be increased to reflect a lower acceptable level of service. Other modifications can be made to reflect a policy of more liberal warrants for exclusive left-turn lanes. Approach symmetry can be assured by adding several simple rules and modifying a few more. The ability of a simple prototype like this to reflect various policies and practices is significant. It emphasizes the potential of expert systems as practical transportation engineering tools. It also suggests a completely new application for the Intersection Advisor and related systems: as tools for evaluating various policies or changes to existing policies.

CONCLUSIONS

As a prototype system, the main purpose of the Intersection Advisor is to test the feasibility of a concept and to identify areas for future development. The advisor has demonstrated the feasibility of using a knowledge-based system to solve geometric problems in intersection design. Although the advisor in its present form may not be considered a useful design aid in terms of time savings or increased accuracy, it does make valuable contributions to the development of an integrated, interactive system for the design of signalized intersections. Continued progress in this area is the focus of further development of the Intersection Advisor.

Several improvements to the advisor are necessary if it is to become a more useful tool. The use of code numbers to represent lane configurations is inconvenient, so an icon-based graphics interface is planned. The problem domain must be expanded and generalized to contain the wide range of problems confronted in actual practice, including one-way streets and multileg and T-intersections. Parking conditions, pedestrian activity, and lane width need to be considered. The user should be able to specify a minimum desired level of service. The advisor should be able to select a design based on more than one set of turning movements (a.m. and p.m. peak hour volumes, for instance), and it should be able to determine critical movements from the information provided. Sensitivity to the interactions among approaches and to the performance of the intersection as a whole must be increased.

The enhancement that appears to be the most difficult to implement is generalization of the problem domain, particularly the inclusion of a graphic interface. The other improvements can be achieved through the addition of rules and facts to the knowledge base and by introducing a cyclic format to the solution strategy. The cyclic format would not be iterative but would allow the advisor to consider the intersection as a whole, including the influence of signalization, before making individual recommendations. A major decision about the future of the Intersection Advisor involves selecting the most appropriate hardware and software environment for its development. M.1 does not have the power and flexibility needed for the implementation of the larger design system. Nevertheless, it did provide a suitable environment for gaining a better understanding of the nature of the geometric design problem and for developing and testing a knowledge base and solution strategy that will be the basis for future efforts.

REFERENCES

1. *Special Report 209: Highway Capacity Manual.* TRB, National Research Council, Washington, D.C., 1985.
2. *Microcomputers in Transportation: Software and Source Book.* Report UMTA-URT-86-1. U.S. Department of Transportation, Feb. 1986.
3. E. Shenk et al. *Prototype Highway Capacity Manual Programs for Intersection Analysis.* TTRC/FHWA, U.S. Department of Transportation, 1986.
4. *M.1 Reference Manual.* Teknowledge, Palo Alto, Calif., March 1985.

Expert System To Cost Feasible Bridge-Painting Strategies

SUE MCNEIL AND ANNE MARGARET FINN

Knowledge-based expert systems have been developed for many civil engineering applications including bridge deck condition assessment, selection of optimal strategies for bridge deck rehabilitation, and traffic signal setting. Such expert systems incorporate both heuristic knowledge and algorithmic approaches to problem solving. Identification of bridge-painting strategies is perfectly suited to such an approach. Bridge-painting decisions are based on measurement of condition; qualitative assessment of deterioration; and heuristics describing the incompatibilities among different types of steel, paint, and surface preparation. Further, uncertainty plays a crucial role because surface treatment, paint application, and bridge condition are nonuniform. Optimization or current approaches to decision making are unable to effectively include all of these variables. A prototype system, Bridge PIARS (Paint Identification and Ranking System), constructed using an expert system building program is based on a decision network. The system allows the user to establish the facility condition, evaluate the need for bridge painting, identify appropriate painting strategies, and cost the strategies. The system and its operation are described, and several areas for research to extend and enhance the system are identified.

More than 45 percent of the bridges in the nation's bridge inventory *(1)* are structurally obsolete or functionally deficient. The magnitude and extent of the bridge problem have spurred additional funding for bridge maintenance and rehabilitation, the development of innovative repair and rehabilitation methods, and the application of computer techniques to bridge management systems *(2)*. Such bridge management systems, based on the National Bridge Inventory (NBI), are used or proposed for use in Kansas, Maryland, Minnesota, New York, North Carolina, and Pennsylvania and are under development in an NCHRP research project. These approaches are structured within traditional algorithmic computer-programming frameworks. In comparison, Seymour *(2)* has conceptualized a bridge management system based on rule-based expert system application modules. One suggested module is a bridge-painting management system that includes establishing the facility condition and the need for repair and selecting an appropriate compatible and economic paint system.

A bridge-painting management system is designed to protect the investment in the bridge. Steel bridges are coated to prevent corrosion, which leads to loss of section and ultimately structural deficiency. Coatings include paint, galvanizing, and the oxidized steel formed on weathering steel or a combination of these. These coatings deteriorate as a result of exposure and the application of deicing salts, which necessitates their replacement from time to time. The impact of this practice on the lifetime costs of a bridge must be evaluated to select an appropriate time interval between paintings, the thickness and number of coats of paint, and a system for application.

The solution to this problem has been formulated and computerized in the Bridge Corrosion Cost (BCC) model developed by Frondistou-Yannis *(3)*. The model is based on simulations of the deterioration and coating costs over the life of the bridge. Because the model is programmed, with respect to the choice of painting systems, using a traditional algorithmic program, many judgments are left to the experienced coating and maintenance engineers. A knowledge-based expert system (KBES) that identifies bridge-painting strategies and their costs is described in this paper.

BRIDGE-PAINTING SYSTEMS

Coatings are used to prevent steel bridges from corroding. Paints are the most commonly used protection, but galvanizing, the application of a zinc coating to steel usually through hot dipping, and the use of weathering steel (A588) are other forms of protection. Paints are not used exclusively on low-carbon steel; they may also be used as additional coatings on galvanized and weathering steel.

Paints are applied using a brush, roller, or spray in several coats each a few mils thick (1 mil = 0.025 mm). The first coat is the primer, followed by intermediate coats and then top coats. Paint types include

- Oil-based paints with or without alkyd resins;
- Zinc-rich primers with organic top coats; and
- Vinyls, epoxies, and polyurethanes known as high-performance paints.

Good paint performance requires a good bond between the metal surface and the paint. The best bonds are achieved when the metal surface is properly prepared. Typical surface preparation specifications are defined by the Steel Structures Painting Council (SSPC) specifications *(4)*. Different paints have minimal surface preparation requirements. For example, zinc-rich primers require blast cleaning. Therefore paints should not be considered in isolation but as a paint system consisting of

- Types of prime, intermediate, and top coats and
- Surface preparation.

This expert system uses two sets of painting systems. The first set of systems is based on those defined by Frondistou-

---

Department of Civil Engineering, Massachusetts Institute of Technology, Cambridge, Mass. 02139. Current affiliation of S. McNeil: Department of Civil Engineering, Carnegie-Mellon University, Pittsburgh, Pa. 15213.

Yannis *(3)* and information in Hare *(5)*. This set includes 12 paint systems of which 4 are defined for low-carbon steel, 3 for galvanized steel, and 5 for weathering steel. The systems are based on common practice, paint compatibilities, system performance, and minimum surface preparation requirements. The 12 systems are summarized in Table 1. The second set is based on the SSPC systems *(4)*. Three other conclusions may be reached for any bridge:

13. Painting is not required,
14. More information is required or information is out of date,
15. Painting should be deferred because of other work pending or possible reconstruction of the bridge.

Several factors influence the choice of a painting system. These include the environment, the bridge condition, the coating condition, the existing paint, and some of the limitations of various types of painting systems. To describe environments effectively, yet avoid detailed measurements, the following four environments, in descending order of severity, are used by ASTM and the SSPC:

- Industrial or urban,
- Marine,
- Rural, and
- Desert.

In this study the desert environment was ignored. Definitions of environments can be confusing because of windborne ocean spray and pollutants, acid rain, and deicing salts. The most severe environment for a bridge is assumed to ensure minimum performance. For example, ocean spray or deicing salt classifies a rural bridge as being in a marine environment.

The performance of a coating system is measured using a grade from 0 to 10 as defined in ASTM D 610 and summarized in Table 2. Each grade is associated with a percentage of rust.

This performance measure is somewhat unsatisfactory because areas near the edges of members may have a rating of 0 when other areas have a rating of 10. Also given in Table 2 for each rating is the percentage of the bridge that needs to be painted. These values are derived in Frondistou-Yannis *(3)*. When 30 percent of the bridge shows rusting or the grade declines to 2 it is assumed that the complete bridge will be repainted.

TABLE 2 CORROSION PERFORMANCE RATING ACCORDING TO ASTM D 610

Rating	Description	Area to be Painted (%)
10	No rusting or less than 0.01% of surface rusted	0
9	Minute rusting, less than 0.03% of surface rusted	0
8	Few isolated rust spots, less than 0.1% of surface rusted	0
7	Less than 0.3% of surface rusted	0
6	Extensive rust spots but less than 1% of surface rusted	8
5	Rusting to the extent of 3% of surface rusted	18
4	Rusting to the extent of 10% of surface rusted	40
3	Approximately 1/6 of surface rusted	60
2	Approximately 1/3 of surface rusted	100
1	Approximately 1/2 of surface rusted	100
0	Approximately 100% of surface rusted	100

The existing paint and thickness and the year last painted influence the choice of a system:

- The system to be applied should be the same as the existing paint to ensure compatibility between the paint applied and the existing system, unless the complete bridge is to be blasted (Systems 3, 4, 11, and 12).
- The type of existing paint together with its age may also be used as an indicator to check the condition as represented by the grade described in Table 2. Frondistou-Yannis *(3)* gives performance curves for each painting system and environment as well as empirical adjustment factors for the type of bridge and thickness of existing paint.

Other variables and considerations that affect the selection of a paint system are summarized in Table 3 for each of the systems.

The cost of applying any bridge painting system varies with

- The system,
- The condition of the surface of the structure,

TABLE 1 PAINT SYSTEMS FOR HIGHWAY STRUCTURAL STEEL [modified from Frondistou-Yannis *(3)*]

Protection Method	Paint	Surface Preparation
Low-Carbon Steel		
1. Paint System I	Oleoresinous paints (oils and alkyds)	Hand cleaned
2. Paint System II	Same as above	Commercial blast
3. Paint System III	High-performance paints	Near-white metal or commercial blast
4. Paint System IV	Zinc-rich primers and organic top coats	Near-white metal
Galvanized Steel		
5. Zinc coating	None	None
6. Alkyds	Zinc dust or zinc oxide alkyd paints	Wash with soap then rinse
7. High performance	High-performance paints	Hand cleaning
Weathering Steel		
8. Steel surface	None	None
9. Paint System I	Oleoresinous paints (oils and alkyds)	Hand cleaned
10. Paint System II	Same as above	Commercial blast
11. Paint System III	High-performance paints	Near-white metal or commercial blast
12. Paint System IV	Zinc-rich primers and organic top coats	Same as above

TABLE 3 FACTORS THAT INFLUENCE PAINT SYSTEM SELECTION [modified from Frondistou-Yannis *(3)* and Hare *(5)*]

System	Advantages/Uses	Disadvantages/Limitations
1	Suitable for use on existing oil/alkyd paints	
2	Suitable for use on existing oil/alkyd paints	
3	Good system when existing lead paint removed	Difficult to apply on complex designs; requires skilled contractors
4	Good system when existing lead paint removed	Difficult to apply on complex designs; must be sprayed
5		Unsuitable in industrial or marine environments
6		Must be sprayed; use on unpainted or similar existing system
7		Difficult to apply on complex designs; use on unpainted or similar existing system
8		Unsuitable in industrial or marine environment
9	Suitable for use on existing oil/alkyd paints	Use on unpainted or similar existing systems
10	Suitable for use on existing oil/alkyd paints	Use on unpainted or similar existing systems
11	Good system when existing lead paint removed	Difficult to apply on complex designs; requires skilled operators
12	Good system when existing lead paint removed	Difficult to apply on complex designs; must be sprayed

- The number and thickness of coats, and
- Local conditions.

The BCC model *(3)* provides unit costs for estimating the total cost of applying a painting system. Similar costs are also provided by the SSPC *(4)*.

KNOWLEDGE-BASED EXPERT SYSTEMS

Knowledge-based expert systems (KBESs) are a major area of research in artificial intelligence. They are interactive computer systems based on the facts, rules of thumb, and approaches used by human experts to solve a problem. KBESs provide a practical alternative to conventional programs when the problem is ill-structured and solution algorithms do not exist or do not provide a complete solution *(6–8)*. The organization of KBESs, comparisons with conventional programs, applications, and the development of KBESs are described in detail elsewhere *(6, 8)*.

The success of MYCIN *(6)*, a large KBES developed in the mid-1970s for medical diagnosis, has led to development of small-scale KBESs for many different applications. In civil engineering, examples include traffic signal setting *(9)*, preliminary design of high-rise buildings *(10)*, and selection of bridge deck rehabilitation strategies *(2)*. These applications in civil engineering are often characterized by the integration of algorithmic programming, which typically identifies conventional programs, with heuristics and symbolic manipulation, which commonly identify KBESs. The result is a more complete and correct solution that no longer functions as a "black box" that is unable to explain or easily change the solution procedure used.

A KBES has four basic components:

- Knowledge base—contains all knowledge and rules used in solving the problem.
- Context—contains information that is specific to the problem currently being solved.
- Inference mechanism—links the knowledge base and context. The object of the inference mechanism is to reach a goal or conclusion and solve the problem.
- User interface—allows the user to interact with the system just as others confer and interact with an expert.

Although these elements are common to all KBESs, they may also include the ability to explain their reasoning and acquire knowledge.

The success of a system is dependent on whether an expert's method of problem solving, knowledge, and experience can be conveyed to the knowledge engineer. A number of language tools ranging from high-level languages to problem-specific tools and environments are available to implement the knowledge and problem-solving process *(6)*.

The knowledge acquisition component of the bridge-painting problem is described in the following section.

BRIDGE-PAINTING PROBLEM

Like other areas of bridge maintenance, bridge painting is often deferred because of financial constraints. For example, in Massachusetts for many years the budget for the Massachusetts Department of Public Works for bridge painting was on the order of $500,000. The department aims to paint bridges every 7 years, but with this budget limitation it is significantly behind schedule. To begin to correct this shortfall, the bridge-painting budget for fiscal year 1986 in Massachusetts is $5 million. The allocation of either a large or a small budget to particular bridges requires trade-offs. Furthermore, lack of trained inspectors and poor information on the condition of existing bridge paint make planning difficult.

In developing a bridge-painting program an agency often makes decisions hierarchically. At the uppermost level, a bridge-painting budget is set. At the next level, decisions about which bridge has to be painted are required. At the lower levels, type, thickness, and number of coats of paint; method of application; and amount of cleaning are decision variables. Because of problems with paint compatibility, local environment, and environmental conditions, there is usually a small subset of all possible painting systems that is feasible for a particular bridge. Identification of this feasible set of strategies is based on heuristics and qualitative data on the bridge, existing paint, environment, and local preferences.

The BCC model *(3)* seeks a strategy that minimizes the discounted costs of bridge painting over the life of the bridge. The optimal strategy for any one bridge is found by simulation. The simulation simply calculates the discounted bridge painting over the life of the bridge, for a range of reasonable thicknesses and for each possible value of the frequency of repainting until a minimum cost is observed. The process is repeated for each system that is appropriate for that type of steel. The minimum cost of all of the systems is identified as the optimal strategy. This approach fails to explicitly recognize that some systems may not be feasible for reasons of paint compatibility or local conditions. The expert system described in this paper interactively queries the decision maker to ensure that all available information is included in the identification of feasible strategies. The system heavily depends on the relationships and empirical data of the BCC model *(5)* to identify optimal strategies that are feasible.

The bridge-painting problem is ideally suited to the utilization of a KBES *(7, 9)* because

- The problem is well defined and has a relatively narrow problem domain,
- Experts exist and can describe their methods,
- The task does not require common sense but heuristic solution, and
- Expertise takes time to acquire and experts are in relatively short supply.

The knowledge base captures information from experts in the field and structures it in the form of an expert system to derive appropriate actions. The knowledge base is accessed for the identification of feasible painting strategies and determination of costs.

To isolate the engineer from the computer science–related details of building an expert system, tools to include problem-solving knowledge, allow the user and system to communicate, and complete problem solving are used. These tools, which are described in the following section, allow the knowledge engineer to focus on knowledge acquisition and inclusion.

BRIDGE-PAINTING IDENTIFICATION AND RANKING SYSTEM

The knowledge-based expert system Bridge PIARS (Bridge Painting Identification and Ranking System) is written in GEPSE (General Engineering Problem Solving Environment), a set of knowledge-based expert system building tools written by Chebayeb and Connor *(11)*. GEPSE is written in the C programming language and provides users with the flexibility to incorporate knowledge as rules and algorithmic procedures, but it also includes a predefined rule-based inference engine (forward chaining), interpreter, and mechanisms for the inclusion and alteration of objects, goals, and rules.

Bridge PIARS has two parts. The first part identifies feasible paint systems and the second part costs all feasible strategies for the bridge. Bridge PIARS defines objects such as a bridge or paint, which have attributes such as percentage corrosion. Relationships between objects and attributes are defined as rules. For example, if the attribute percentage corrosion of object bridge has a value of 10 percent, the attribute grade is set to 4. The inference engine fires rules until a goal is reached. In Bridge PIARS goals are reached at the end of the first phase. Functions are defined throughout the program to perform calculations, input, and output. For example, expected deterioration is calculated and compared with observed deterioration. The second part calculates the cost of each feasible painting system. The cost calculations of the second part can be repeated to perform a number of simulations and produce an optimal strategy similar to the BCC model *(3)*.

The first part of Bridge PIARS identifies feasible bridge-painting systems by asking the user a series of questions about the bridge and the existing paint. Alternatively, the program could query a data base if the appropriate data were available. The National Bridge Inventory as it now exists includes no information on the condition of bridge paint. The basic questions used to identify the most feasible system are

- Q1: What is the bridge made of?
- Q2: What type of steel is the bridge made of?
- Q3: What is the bridge painted with?
- Q4: What environment is the bridge in?
- Q5: Do environment regulations permit spraying?
- Q6: Is the bridge a truss or does it have complicated geometry?
- Q7: What is the bridge grade according to ASTM D 610?
- Q8: Are skilled operators available?

All questions are not necessarily asked for every bridge because a strategy may be decided on without asking all questions. In addition to asking the questions listed here, the system also queries the user about the painting history of the bridge. It then calculates the expected condition and compares this with the actual condition. If the conditions differ, the user is given the option of choosing the forecast or the observed condition or aborting the session.

The general relationships between questions and goals and objects are depicted in the decision network shown in Figures 1–4. Figure 1 shows the part of the decision tree common to all three types of steel considered by the system and the determination of the existing coating system. The intermediate steps required to reach a goal are shown in Figures 2–4 for galvanized, weathering, and low-carbon steels, respectively. The nodes on the network represent the questions listed previously. At each node identified as Q7 the system asks the user "what is the grade of the bridge?" At this node the user is also asked additional questions about the painting history of the bridge. This ensures that the actual deterioration is consistent with the deterioration predicted since the bridge was last painted. Each branch on the decision tree terminates when a goal is reached. The goals are depicted in Figures 1–4 as squares with numbers corresponding to the paint systems defined in Table 1 and the conclusion that painting should not be done.

The second part of Bridge PIARS estimates the painting costs. It uses the feasible paint systems derived in the first phase of the program. The user is asked to specify thickness. The calculation is repeated for each system and different thicknesses can be specified as shown in Figure 5. The program has a simple data base of unit costs and correction factors that the user can modify if necessary. It is this area of the knowledge base that could benefit most from further refinement.

Both parts of the program are interactive and ask the user for input for the particular problem to be solved. The user is given a menu of possible answers as shown in Figure 6. This question is one of several used to establish the painting history of the bridge to permit comparison of actual deterioration with predicted deterioration. The question and menu appear on the screen when a rule is fired. The user's choice is then assigned to an attribute through the rule. For example, the rule in Figure 7 asks the user "what type of steel is the bridge made of?" and assigns the answer to the attribute type of the object steel. The rule base includes approximately 80 such rules.

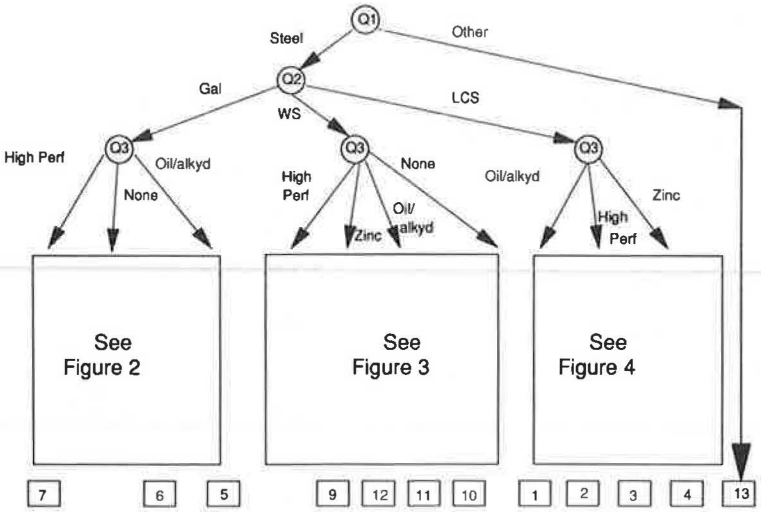


FIGURE 1 Decision tree for identifying feasible bridge-painting strategies.

To provide some preliminary testing of the program, Bridge PIARS was used to evaluate and cost bridge-painting strategies for two bridges that were recently contracted out for painting by the Massachusetts Department of Public Works. Both bridges were painted with the same painting system. Bridge PIARS identified three feasible painting strategies for each bridge including the one used. Table 4 gives a summary of the test results for the three bridges. For the actual painting system Bridge PIARS was within 12 percent of the lowest bid whereas office estimates tended to be around 30 percent under the lowest bid. The preliminary testing highlighted the role of the

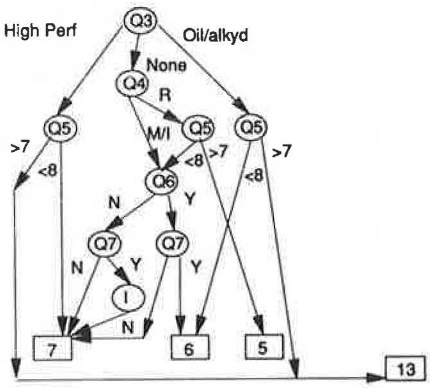


FIGURE 2 Decision tree for identifying strategies for galvanized coatings.

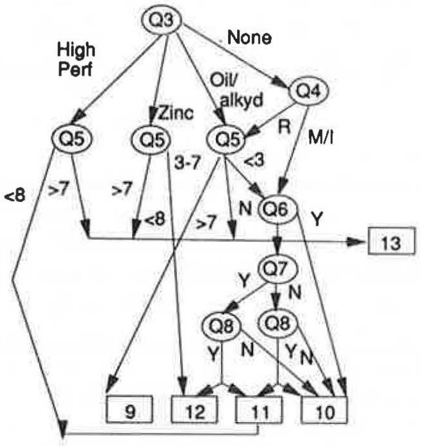


FIGURE 3 Decision tree for identifying strategies for weathering steel.

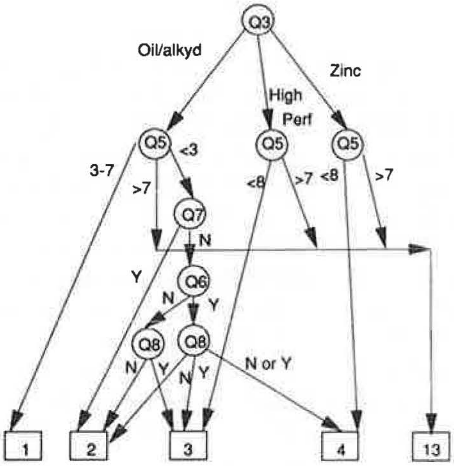


FIGURE 4 Decision tree for identifying strategies for low-carbon steel.

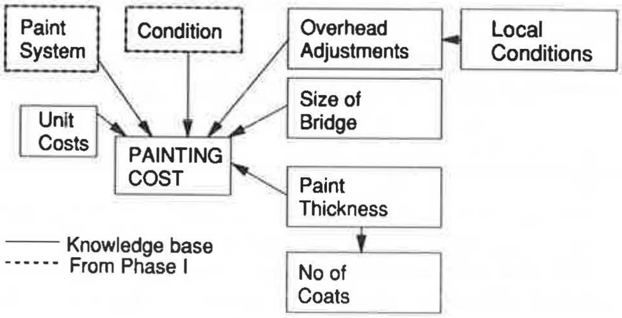
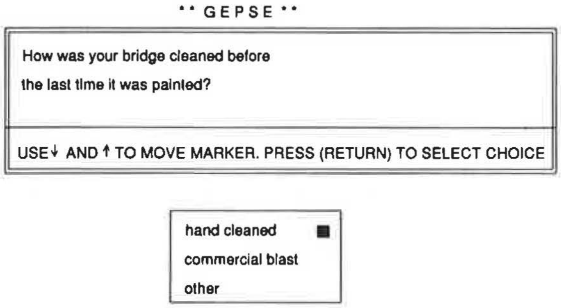


FIGURE 5 Determinants of painting costs.



** GEPSE **

How was your bridge cleaned before the last time it was painted?

USE ↓ AND ↑ TO MOVE MARKER. PRESS (RETURN) TO SELECT CHOICE

hand cleaned ■

commercial blast

other

FIGURE 6 Typical user query.

```
(make-rule "typing"

   (is? type-of steel nil)

   (set type-of steel

     (menu 1
     "What is the type of steel?"
      3
     galvanized
     low-carbon
     weathering)))
```

FIGURE 7 Rules for finding type of steel.

TABLE 4 RESULTS OF PRELIMINARY TESTING OF BRIDGE PIARS

Bridge	Systems Proposed by Bridge PIARS	Cost ($)	System Used	Office Estimate ($)	Lowest Bid ($)
1	2	50,000	4	63,000	94,000
	3	65,000			
	4	89,000			
2	2	593,000	4	800,000	1,174,000
	3	779,000			
	4	1,056,000			

current version of Bridge PIARS as an aid and the importance of life-cycle costs in paint selection.

CONCLUSIONS

Bridge PIARS, as a demonstration prototype, shows the use of KBESs for solving engineering problems. The system shows sufficient potential that several enhancements are warranted. Improvements to the problem-solving approach include

- Determining the optimal painting system using optimization methods that incorporate budget constraints rather than using simulations,
- Accounting for the uncertainty and variability of the information provided by the user and the performance of coating systems, and
- Including the time-varying properties of paint reliability.

These features demonstrate the importance of an integrated approach to problem solving that permits users to include algorithmic approaches within a KBES. Other enhancements will be to the user interface and include refinements to qualitative questions about paint condition and the geometry of the bridge and the addition of an explanation facility and interfaces to a data base for historical information that is typically unavailable in the field.

To be completely robust, the system also requires some refinement of the deterioration relationships that are used to check condition, unit cost estimates, the use of multiple and alternative paint systems, and the ability to update cost information and alternative approaches for assessing paint condition.

Bridge PIARS has demonstrated the ability of KBESs to

- Provide friendlier user interfaces than have been common in algorithmic program solutions to similar problems,
- Integrate qualitative and quantitative information processing, and
- Present a more complete solution to a problem than is convenient in an algorithmic program environment.

The resultant system can assist in making decisions by consistently accounting for all the variables. However, the system is only as powerful as the knowledge base. The flexibility needed for the user to update the knowledge base and query the system is not included in the present problem-solving environment but may be an appropriate enhancement. However, the KBES approach to this problem allows the knowledge engineer to easily update the knowledge base to reflect new technologies in coatings, applications methods, and paint removal as well as changes in costs over time.

ACKNOWLEDGMENTS

The authors wish to acknowledge the assistance of Nina Ross in obtaining data, the comments and information provided by Jim Gould and Richard McGinn of the Massachusetts Department of Public Works, and the helpful and thorough comments of an unknown reviewer.

REFERENCES

1. *6th Annual Report to Congress on the Status of the National Bridge Program.* FHWA, U.S. Department of Transportation, 1985.
2. W. J. Seymour. *Preliminary Development of a Bridge Management System with Rule Based Expert System Decision Support Enhancements.* Master's thesis. Massachusetts Institute of Technology, Cambridge, 1985.
3. S. Frondistou-Yannis. *Coating and Corrosion Costs of Highway Structural Steel.* Report FHWA-RD-79-121. FHWA, U.S. Department of Transportation, 1980.
4. J. D. Keane. *Steel Structures Painting Manual,* Vol. 2: *Systems and Specifications.* Steel Structures Painting Council, Pittsburgh, Pa., 1982.
5. C. Hare. Factors in Selection of Anti-Corrosive Coating Systems for Bridge Maintenance. *Proc., 2nd World Congress on Coating Systems for Bridges,* La Guardia Airport, New York, 1982.
6. P. Harmon and D. King. *Artificial Intelligence in Business: Expert Systems.* John Wiley and Sons, Inc., New York, 1985.

7. C. T. Hendrickson, D. Rehak, and S. Fenves. *Expert Systems in Transportation Systems Engineering*. Working Paper. Carnegie-Mellon University, Pittsburgh, Pa., 1985.
8. D. A. Waterman. *A Guide to Expert Systems*. Addison-Wesley, Reading, Mass., 1985.
9. C. Zozaya-Gorostiza and C. Hendrickson. An Expert System for Traffic Signal Setting Assistance. *Journal of Transportation Engineering*, ASCE, forthcoming.
10. M. L. Maher. *HI-RISE: A Knowledge Based Expert System for Preliminary Structural Design of High Rise Buildings*. Ph.D. dissertation. Carnegie-Mellon University, Pittsburgh, Pa., 1984.
11. F. Chebayeb and J. Connor. *GEPSE—A Computer Environment for Engineering Problem Solving*. Research Report R86-11. Intelligent Engineering Systems Laboratory, Department of Civil Engineering, Massachusetts Institute of Technology, Dec. 1986.

A Knowledge-Based Approach to Pavement Overlay Design

STEPHEN G. RITCHIE

Described is the development of an initial prototype expert system to assist local engineers in designing the structural thickness of asphalt concrete pavement overlays. The system is called OVERDRIVE (OVERlay Design heuRIstic adVisEr) and is part of ongoing research at the University of California, Irvine, that is developing an integrated set of expert system tools for the analysis and design of highway pavement rehabilitation strategies. The paper provides an overview of expert systems concepts and overlay thickness design methods. A discussion of the OVERDRIVE system follows, including the main components of the first prototype, Version 1.1. It is concluded that a knowledge-based approach to pavement overlay design is feasible and, even in prototype form, OVERDRIVE is a potentially useful tool for local highway engineers. Ongoing research will refine and expand the knowledge base and user interface of OVERDRIVE to enhance its performance as an expert design tool.

In recent years, asphalt concrete pavement overlays have become the principal treatment used in the United States for rehabilitating deteriorated pavements subjected to moderate or heavy traffic. Highway agencies in the United States spend billions of dollars annually on such overlays, which comprise relatively thick layers of bituminous-bound aggregate placed over the existing pavement. An overlay can level out a distorted or rough road surface that is providing poor ride quality and high operating costs to users. It can also increase the structural capacity and service life of an existing pavement. The design life of an asphalt concrete overlay is typically 10 years, and sometimes it is as long as 20 years. Properly designed, an overlay can be a cost-effective means of correcting pavement deficiencies for a substantial period of time.

Three-quarters of the highway mileage in the United States is contained within local highway systems *(1)*, which are the responsibility of tens of thousands of cities, counties, and other local jurisdictions. Successful and cost-effective pavement rehabilitation strategies are generally developed by pavement engineering specialists who use their judgment and experience as well as empirically based design procedures. Typically, these experts are only to be found within federal and state agencies, universities, and private firms. In addition, although conventional computer tools are useful in the overlay design process, their role is limited because the tasks involved tend to be complex and ill-defined so engineering judgment must be relied on. The difficulties facing local highway agencies nationally are therefore not only financial but include the availability of, and access to, specialized human resources and expertise.

The nature of this problem suggests that a new technological approach, involving knowledge-based or expert systems, could be especially useful *(2)*. Such systems are basically interactive computer programs that emulate the knowledge of a human expert to provide advice and guidance to local users.

In this paper is described the development of an initial prototype expert system to assist local engineers in designing the structural thickness of asphalt concrete pavement overlays. The system is called OVERDRIVE (OVERlay Design heuRIstic adVisEr) and is part of ongoing research at the University of California, Irvine, that is developing an integrated set of expert system tools for the analysis and design of highway pavement rehabilitation strategies. The paper provides an overview of expert systems concepts and overlay thickness design methods. A discussion of the OVERDRIVE system follows, including the main components of the first prototype, Version 1.1. It is concluded that a knowledge-based approach to pavement overlay design is feasible and, even in prototype form, OVERDRIVE is a potentially useful tool for local highway engineers. Ongoing research will refine and expand the knowledge base and user interface of OVERDRIVE to enhance its performance as an expert design tool.

EXPERT SYSTEMS

Knowledge-based expert systems are computer programs that have recently emerged from decades of research on artificial intelligence (AI). In general, AI is the study of how to make computers perform tasks that, currently, people perform better *(3)*. Such tasks include natural language processing, speech recognition, vision, and expert problem solving.

Expert systems are designed to emulate the performance of an expert, or group of experts, in a particular problem domain (such as pavement overlay design) through the use of symbolic reasoning. Expert systems therefore address "ill-structured" problems for which a numerical algorithmic solution is not available or is impractical; such problems are solved using expert knowledge, skill, judgment, and heuristics.

A review of potential applications of expert systems in transportation is reported by Yeh et al. *(4)*, and a state-of-the-art review of expert systems in transportation engineering is presented by Ritchie *(5)*. A recent symposium also elaborated on expert systems in civil engineering *(6)*. There are also several more general and comprehensive guides to expert systems *(7, 8)*.

An expert system is fundamentally different from a conventional computer program. One of the principal differences is the

---

Department of Civil Engineering and Institute of Transportation Studies, University of California, Irvine, Calif., 92717.

separation of domain knowledge and the inference or control mechanism. This distinction identifies two of the main components of an expert system, the knowledge base and the inference engine. The knowledge base is the power of an expert system in the sense that it contains all of the empirical and factual information for the problem domain. The inference engine decides how to apply the knowledge in the knowledge base in order to infer new knowledge. It is the control mechanism for the system that attempts to progressively solve each subgoal and thus the entire problem.

There are various ways to represent the knowledge in the knowledge base. The most common is by means of production rules, expressed as IF-THEN statements (e.g., IF surface course is asphalt concrete AND condition is excellent THEN conversion factor = 1.0). When the IF portion or premise of a rule is satisfied by the facts, the action specified by the THEN portion is performed. The rule is then said to "fire." There are two ways in which rules are accessed in a rule-based system: forward chaining and backward chaining. Forward chaining is an inference method that proceeds from information on the left side of the rules to derive information on the right. In other words, rules are matched against facts to establish new facts. Backward chaining involves starting with a conclusion or hypothesis on the right side of one or more rules and trying to establish the facts that would verify that hypothesis. Only rules that are relevant to establishing the hypothesis are executed. Backward chaining therefore proceeds from information on the right side to establish information on the left.

To build an expert system, a symbol manipulation language, such as LISP or PROLOG, can be used. These have been designed specially for AI applications. A variety of dialects exist, including increasingly powerful versions for microcomputers. In addition, a large number of shells or knowledge engineering tool kits are now available for microcomputers. These offer a faster route to expert system development but often involve some sacrifice in flexibility. In either case, the system developer (knowledge engineer) must acquire the expertise and knowledge of the expert or experts and encode it into the knowledge base. An iterative process of testing and refinement then ensues to ensure that the system reaches the desired level of performance.

Finally, it is important to note that although virtually all expert systems are knowledge based, the converse is not necessarily true. In other words, knowledge-based systems are a subset of AI programs, and expert systems are a subset of knowledge systems. A truly "expert" system implies the use of "expert" knowledge to attain high levels of performance in the problem domain. The iterative development process referred to earlier typically involves the successive refinement of an initial knowledge system to produce an expert system that performs at a level comparable to that of recognized human experts. Of course, not all knowledge-based systems need to perform at such a high level to be useful. The nature of real-world problems, and experience, indicates that a spectrum of knowledge-based tools is appropriate, including assistant, colleague, and expert knowledge-based systems.

OVERLAY DESIGN METHODS

Pavements with bituminous surfaces are often called flexible, in contrast with rigid pavements of portland cement concrete. The initial version of OVERDRIVE described in this paper focuses on the design of flexible asphalt concrete overlays on existing flexible pavement. The term asphalt concrete denotes a dense-graded road surface made of hot mineral aggregates plant mixed with hot asphalt. This is the highest type of dense-graded bituminous pavement and is suitable for even the most heavily traveled roads *(1)*.

The two most commonly used design methods for asphalt concrete overlays are component analysis and deflection analysis. These methods reflect empirically based design procedures developed during the last several decades for new pavements. In practice, effective application of the methods requires considerable engineering judgment.

The component analysis overlay design method involves a comparison of the existing pavement structure and a new pavement design for site-specific service conditions. The evaluation of the existing structure requires identification of each of the pavement layers (components) such as the surface course, base, and subbase (if any), as shown in Figure 1. The type, thickness, and condition of each layer must then be determined. Evaluation of the condition of each layer involves selection of a conversion factor that reflects the layer's structural adequacy. Even if the results of sampling and testing in-place materials are available, substantial judgment is required to effectively select the value of each factor. The factors apply reductions of up to 100 percent to the structural adequacy of each layer and can therefore have a major impact on determining the need for an overlay and its design thickness. Further, if either or both site-specific traffic data or subgrade soil strength are not available, judgment must be used to select appropriate design values.

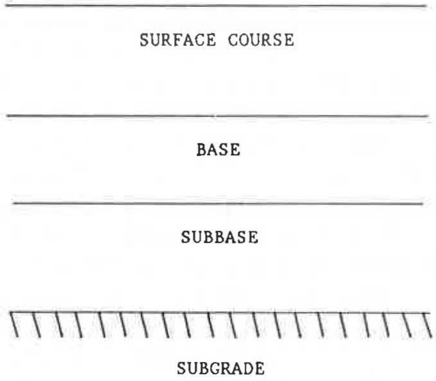


FIGURE 1 Cross section of typical pavement structure.

Deflection analysis design methods involve nondestructive testing of the pavement to yield measured surface deflection caused by a standard load. Although this method does not generally consider individual pavement layers, it directly reflects the effective strength and response of the in situ pavement structure to traffic levels and types. If the measured deflection is greater than an acceptable value, an overlay is required.

Component analysis is a traditional design method that has been used in various forms for many years. More recently, deflection-based procedures have begun to gain wide acceptance, particularly among state departments of transportation. Standard design guidelines, such as those of the Asphalt Institute *(9)*, suggest that it may sometimes be desirable to use both methods before making a final decision based on engineering judgment. However, a majority of local highway agencies in the United States do not own the equipment necessary to perform deflection tests. Many of the smaller agencies also do not have laboratory equipment or personnel and rely either on standardized pavement designs that have worked well in the past or on other judgmental or rule-of-thumb methods.

In building OVERDRIVE 1.1, the first priority was to build a knowledge-based system for overlay design using a component analysis design method. This method provides an improved and more rational overlay design procedure for many users. When implemented using a knowledge-based approach, a powerful design tool results.

DESCRIPTION OF OVERDRIVE

OVERDRIVE 1.1 is an initial prototype of a knowledge-based system to provide interactive expert advice and guidance on the detailed design of asphalt concrete pavement overlays to local highway engineers. OVERDRIVE is a part of a more extensive system named PARADIGM (PAvement Rehabilitation Analysis and DesIGn Mentor), a proposed integrated set of expert systems, now under development, for local highway agencies *(2)*.

The first expert system developed as part of the PARADIGM project was SCEPTRE (Surface Condition Expert for PavemenT REhabilitation). This system is described elsewhere *(10, 11)*. SCEPTRE evaluates project-level pavement surface distress and other user inputs to recommend feasible rehabilitation strategies for subsequent detailed analysis and design by OVERDRIVE. The two systems have been designed so that many of the inputs to SCEPTRE can also be used by OVERDRIVE. SCEPTRE has been developed using the knowledge engineering shell EXSYS *(12)* on a Compaq portable microcomputer (and runs on any MS-DOS–compatible PC). The system is rule based and uses a backward-chaining inference method. The knowledge base in Version 1.4 contains about 140 complex rules, derived from the combined expertise of two pavement specialists. SCEPTRE 1.4 currently addresses state-maintained flexible pavements in Washington State and has been made available for field testing in district offices of the Washington State Department of Transportation (WSDOT). Ongoing research will refine and adapt the knowledge base for local agencies.

Like SCEPTRE, OVERDRIVE is a microcomputer-based production rule system. OVERDRIVE 1.1 accesses its rules using a forward-chaining inference method and has been implemented using EXSYS. The system also interfaces with an external program to pass and receive values of design parameters.

The knowledge base of OVERDRIVE 1.1 is the result of knowledge engineering efforts with a pavement specialist combined with a synthesis of state-of-the-art and other reports, papers, and manuals relating to the Asphalt Institute overlay design method for asphalt concrete overlays on flexible pavement, *(1, 9, 13–15)*. The knowledge base contains more than 100 rules. It is expected that successive versions of OVERDRIVE will incorporate additional knowledge acquired from pavement engineering specialists. The natural evolution of the system's performance will be toward that of a human pavement expert, which is the ultimate objective.

OVERDRIVE is implemented as an interactive microcomputer program to make it accessible to a broad range of potential users and to permit relatively fast determination of overlay structural thickness requirements. This also allows quick assessment of the impact or impacts of varying assumptions and input values for design parameters.

Several major tasks are addressed by OVERDRIVE in formulating a recommendation for the structural thickness of a new overlay. These include determining the effective thickness of the existing pavement structure, determining a new full-depth asphalt concrete construction thickness, and assessing the consequent need for an overlay. Each of these tasks is discussed in more detail in subsequent sections.

EFFECTIVE THICKNESS

OVERDRIVE 1.1 is applicable to existing pavement structures containing up to three layers, excluding the subgrade, as shown in Figure 1. To design the structural thickness of an overlay using a component analysis method such as that of the Asphalt Institute *(9)*, the effective thickness of the existing pavement structure must be determined. The effective thickness of the structure is the sum of the effective thicknesses of each layer. Effective thicknesses are found by multiplying the actual layer thicknesses by appropriate conversion factors. Each effective thickness represents an equivalent depth of new asphalt concrete. To perform this analysis requires assessment of the following items for each individual structure:

- Number of layers (e.g., surface course only; surface course and base; surface course, base, and subbase);
- Thickness of each layer;
- Layer material type; and
- Layer condition.

Pavement segments in OVERDRIVE 1.1 are user defined and should be homogeneous with respect to geometry and features. A conversion factor based on the layer material type and condition is selected. This determination may be assisted by past records of design, construction, or maintenance; by field inspection; and if possible by at least limited sampling and laboratory testing of in-place materials (OVERDRIVE 1.1 does not provide guidance for such sampling and testing). Ultimately, however, the selection of each conversion factor involves engineering judgment. For example, asphalt concrete that is in very good condition with little cracking or rutting may be assigned a conversion factor of 0.9 to 1.0. Asphalt concrete exhibiting greater distress should be assigned a conversion factor in the range 0.5 to 0.8. Granular bases and subbases may be assigned values of 0.2, and so on. The effective thickness for the structure is then determined as follows:

$$\text{Effective thickness} = (T1)\,(S1) + (T2)\,(S2) + (T3)\,(S3) \quad (1)$$

where $T1$, $T2$, and $T3$ are the actual layer thickness for surface course, base, and subbase, respectively, and $S1$, $S2$, and $S3$ are conversion factors for each layer.

OVERDRIVE 1.1 includes three possible surface course layer types, eight base course layer types, and five subbase course layer types. The number of condition levels available for each layer varies between 1 and 28 depending on the layer and material. For example, the condition level, and hence conversion factor, for each surface course layer type is dependent on the extent and severity of alligator cracking in combination with the severity of rutting. Twenty-one possible condition levels result for a given surface course layer type.

To illustrate the rule-based knowledge representation relating to effective thickness determination in OVERDRIVE, consider the partial inference net in Figure 2. An inference net portrays all of the possible inference chains that can be generated by a set of rules. An inference chain indicates how the system uses the rules to infer a result and is formed by matching the IF portions of rules to the facts. The rules corresponding to the inference net in Figure 2 are shown in Figure 3. These rules are contained in OVERDRIVE's knowledge base and were created using the EXSYS editor. As a result of the first rule firing, the user is requested to enter the thickness of the existing surface course layer, in inches, because the numeric variable [SURFACE THICKNESS] has not yet been assigned a value. For the second rule to fire, the user is queried about the material type in the surface layer (this query is in the form of a multiple choice question), and the user indicates that there is an asphalt concrete surface layer. Finally, as a result of the user indicating that there is no alligator cracking or rutting present (in response to further queries by the system), the third rule would fire and a conversion factor of 1.0 would be selected. In response to a system query, the user can enter WHY, and the system will respond with the rule or rules it is attempting to verify, thereby revealing its reasoning.

Figure 4 shows two of the rules that are used to analyze the next possible layer in the existing structure, the base course. As a result of the first rule firing, OVERDRIVE attempts to apply the second rule and queries the user about the existence of a base layer. If such a layer exists, its thickness is determined and the system proceeds to establish the effective thickness of that layer, and then of the subbase layer (if one exists). If a base course layer does not exist because, for example, the existing structure is full-depth asphalt concrete, the appropriate conversion factors and thicknesses are deduced as shown in the ELSE part of the rule in Figure 4.

NEW FULL-DEPTH THICKNESS

Determination of the new full-depth construction thickness involves developing a new design for a full-depth asphalt concrete pavement over the existing subgrade. The procedures incorporated in OVERDRIVE 1.1 are based on the elastic layered theory approach of the Asphalt Institute. This approach assumes that the subgrade is infinite in the vertical direction and that all layers are infinite in the horizontal direction. It includes consideration of limiting strains, material properties, environmental considerations (temperature and frost effects), and traffic. The basic activities of OVERDRIVE in this overall task include:

- Subgrade assessment,
- Traffic analysis, and
- Design of new full-depth thickness.

The sequence of these activities is shown in Figure 5.

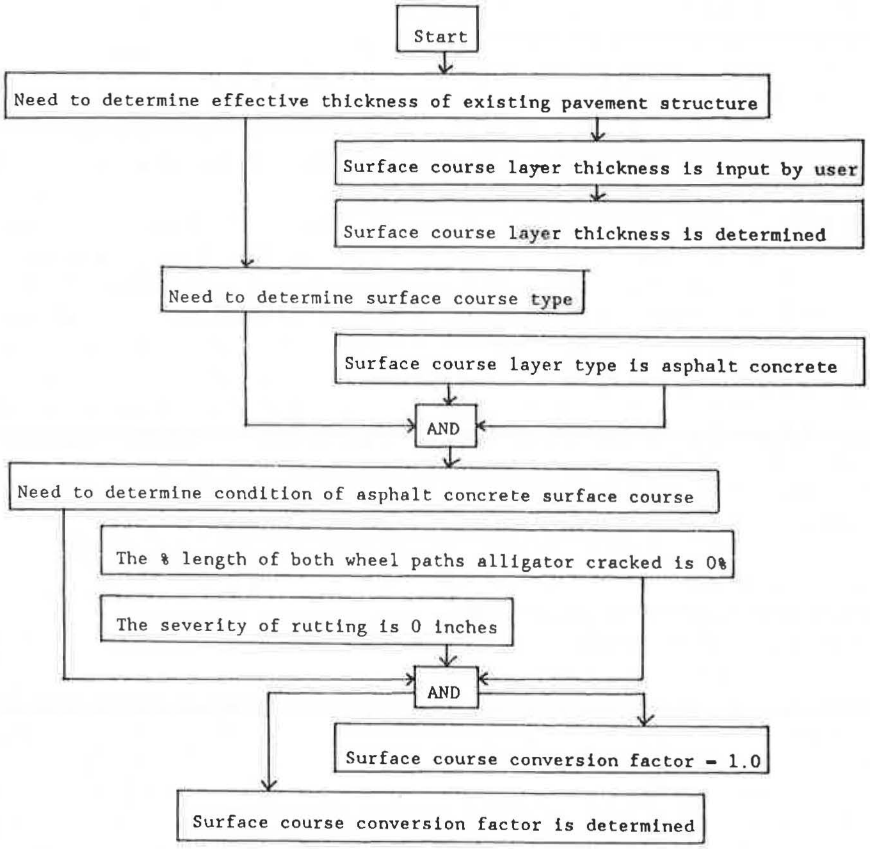


FIGURE 2 Partial inference net for surface course analysis.

```
RULE NUMBER:  1

IF:
          NEED TO DETERMINE EFFECTIVE THICKNESS OF EXISTING PAVEMENT

THEN:
          [T1] IS GIVEN THE VALUE [SURFACE THICKNESS]
 and      SURFACE COURSE THICKNESS IS DETERMINED
 and      NEED TO DETERMINE SURFACE COURSE TYPE

- - - - - - - - - - - - - - - - - - - - -

RULE NUMBER:  2

IF:
          NEED TO DETERMINE SURFACE COURSE TYPE
 and      SURFACE COURSE LAYER TYPE IS ASPHALT CONCRETE

THEN:
          NEED TO DETERMINE CONDITION OF ASPHALT CONCRETE SURFACE COURSE

- - - - - - - - - - - - - - - - - - - - -

RULE NUMBER:  3

IF:
          NEED TO DETERMINE CONDITION OF ASPHALT CONCRETE SURFACE COURSE
 and      THE % LENGTH OF BOTH WHEEL PATHS ALLIGATOR CRACKED IS 0%
 and      THE SEVERITY OF RUTTING IS 0 INCHES

THEN:
          [S1] IS GIVEN THE VALUE 1.0
 and:     SURFACE COURSE CONVERSION FACTOR IS DETERMINED
```

FIGURE 3 Rules corresponding to inference net.

```
IF:
          SURFACE COURSE CONVERSION FACTOR IS DETERMINED

THEN:
          NEED TO DETERMINE IF A BASE COURSE LAYER EXISTS

- - - - - - - - - - - - - - - - - - - - -

IF:
          NEED TO DETERMINE IF A BASE COURSE LAYER EXISTS

 and      IN THE EXISTING PAVEMENT STRUCTURE, A BASE COURSE LAYER EXISTS

THEN:
          [T2] IS GIVEN THE VALUE [BASE THICKNESS]
 and      BASE THICKNESS IS DETERMINED
 and      NEED TO DETERMINE BASE COURSE TYPE

ELSE:
          [BASE THICKNESS] IS GIVEN THE VALUE OF 0.0
 and      [S2] IS GIVEN THE VALUE 0.0
 and      [SUBBASE THICKNESS] IS GIVEN THE VALUE 0.0
 and      [S3] IS GIVEN THE VALUE 0.0
 and      BASE COURSE CONVERSION FACTOR IS DETERMINED
 and      BASE THICKNESS IS DETERMINED
 and      SUBBASE COURSE CONVERSION FACTOR IS DETERMINED
 and      SUBBASE THICKNESS IS DETERMINED
```

FIGURE 4 Example rules for base course analysis.

The two basic design parameters that OVERDRIVE attempts to determine in this phase are the subgrade modulus and the number of equivalent 18,000-lb single-axle loads [18-kip equivalent axle loads (EALs)] due to truck traffic during the user-specified design period.

Sampling and laboratory testing of subgrade materials are encouraged even if original design records are available. The results of this testing can provide an indication of the subgrade resilient modulus. However, when this is not available, as may be the case for many smaller local agencies, OVERDRIVE allows for a more subjective characterization of subgrade strength. Rules in the knowledge base allow the user to classify the subgrade into three categories for design purposes on the basis of subgrade characteristics. A summary of this is given in Table 1.

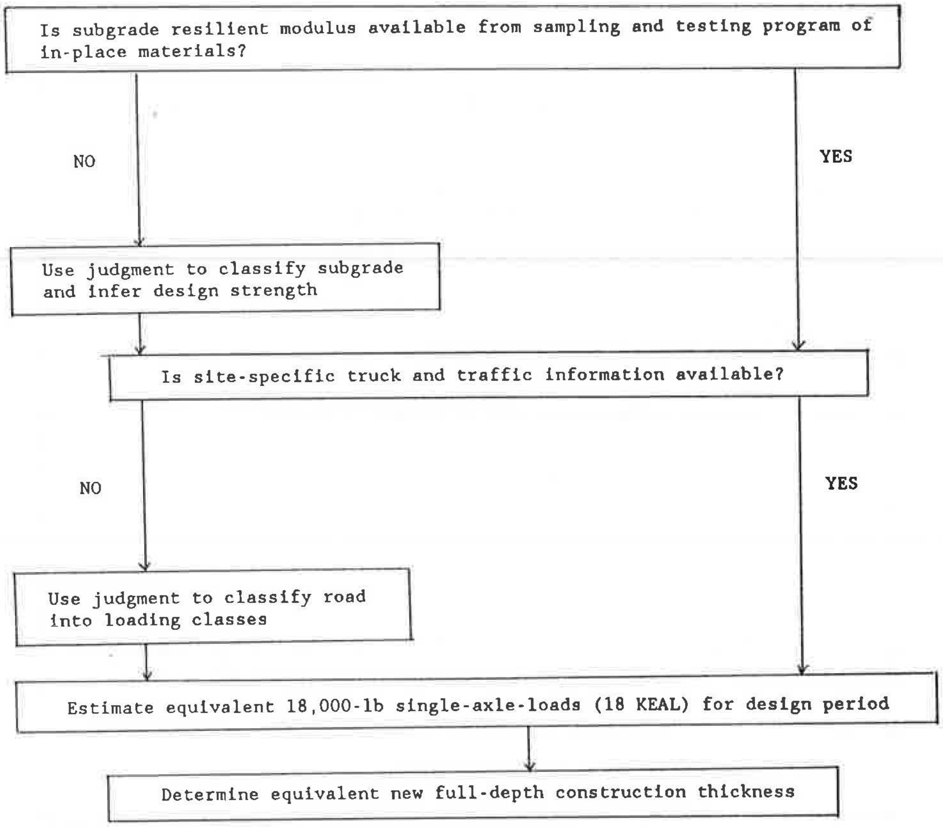


FIGURE 5 Sequence of activities for determining new construction thickness.

TABLE 1 SUBGRADE CLASSES FOR DESIGN PURPOSES *(9, 14)*

Soil Class	Characteristics	Design Modulus (psi)
Poor	Soft and plastic when wet, generally composed of silts and clays (CBR ≅ 3)	4,500
Medium	Includes loams, silty sands, and sand-gravels that contain moderate amounts of clay and silt; can be expected to lose only a moderate amount of strength when wet (CBR ≅ 8)	12,000
Good	Expected to retain substantial amount of strength when wet; includes clean sands and sand-gravels (CBR ≅ 17)	25,000

NOTE: CBR = California bearing ratio.

For the estimation of 18-kip EALs for the design period, several methods are provided in OVERDRIVE depending on the availability of site-specific truck and traffic information. If site-specific data are not available, OVERDRIVE queries the user to determine the appropriate traffic class (Table 2) with its associated estimate of 18-kip EALs. If site-specific data are available, several more detailed procedures are available to determine 18-kip EALs for the design period. The exact procedures and rules that are applied depend on the level of disaggregation of the data (e.g., ranging from average annual daily traffic and percentage trucks at one extreme to whether truck volumes can be estimated for single and multiple units by axle class). OVERDRIVE also provides for traffic growth factors over the design period, if the user so desires.

TABLE 2 ASPHALT INSTITUTE TRAFFIC CLASSIFICATIONS *(9, 14)*

Type of Street or Highway	Traffic Class	Estimated 18-kip EALs
1. Parking lots 2. Lightly trafficked residential streets and farm roads	I	5,000
1. Residential streets 2. Rural farm and residential roads	II	10,000
1. Urban and rural minor collectors	III	100,000
1. Urban minor arterial and light industrial streets 2. Rural major collector and minor arterial highways	IV	1,000,000
1. Urban freeways and other principal arterial highways 2. Rural Interstate and other principal arterial highways	V	3,000,000
1. Urban Interstate highways 2. Some industrial roads	VI	10,000,000

Having established both subgrade and traffic design parameters, OVERDRIVE then determines the new full-depth asphalt concrete construction thickness using the design chart shown in Figure 6. This chart is for a mean annual air temperature of 60°F with frost action possible. Design charts for other conditions can also be readily incorporated. In OVERDRIVE 1.1 the determination of new full-depth thickness is carried out by a conventional external program that is called by rules within the knowledge base. The design parameter values are passed out to

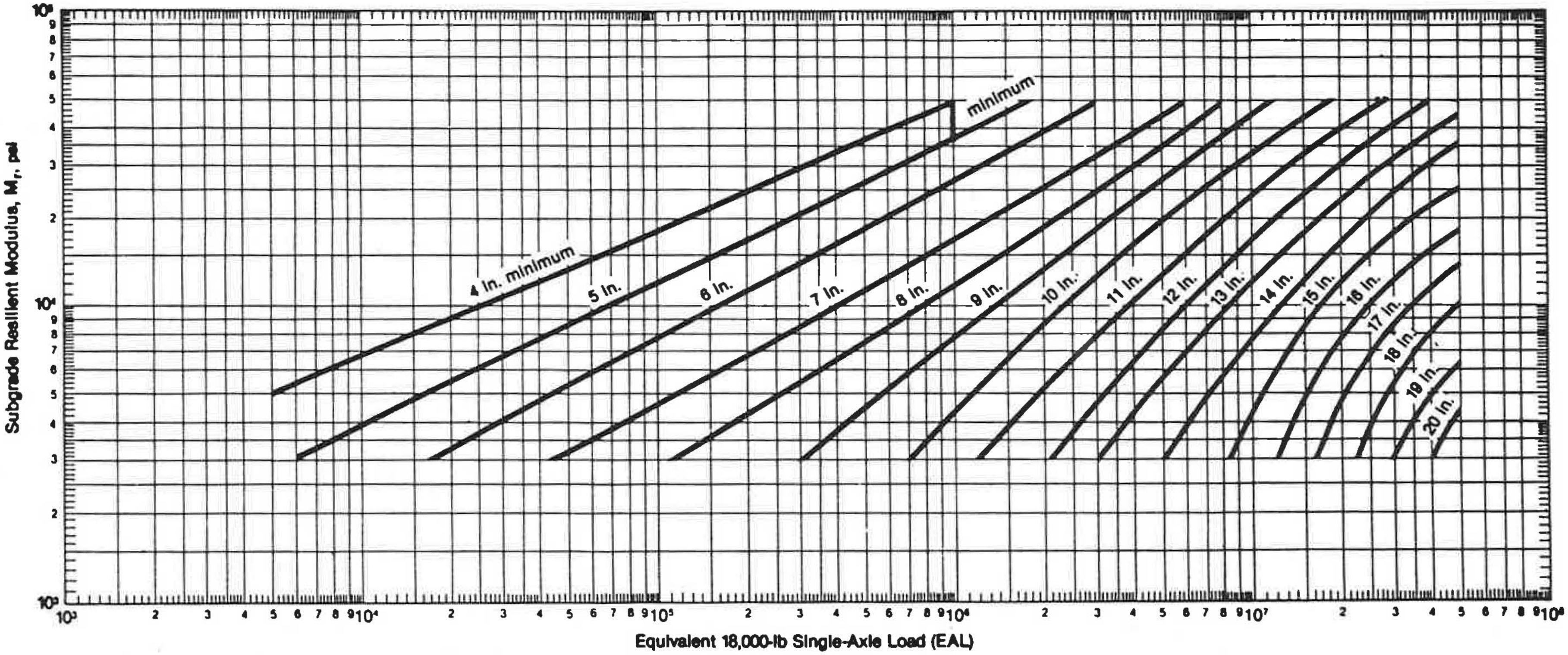


FIGURE 6 Asphalt Institute design chart for full-depth asphalt concrete pavement *(9)*.

this program, which essentially uses the look-up table representations of Figure 6, and the new design thickness is returned to OVERDRIVE to be used in determining the need for an overlay.

NEED FOR OVERLAY

Given the effective thickness and new full-depth construction thickness, it is a simple calculation to determine the structural design thickness of any required asphalt concrete overlay for the given service conditions. OVERDRIVE determines the overlay design thickness as

$$\text{Design thickness} = (\text{New full-depth thickness}) - (\text{Effective thickness}) \qquad (2)$$

In practice, the design thickness must obviously be nonnegative. If the full-depth thickness is less than the effective thickness, this simply indicates that the existing pavement is structurally adequate for the specified service conditions. In this case OVERDRIVE informs the user that an overlay is not required to enhance structural capacity of the section.

It is important to realize that, even if the existing pavement structure is structurally adequate, there may be deficiencies related to the pavement's functional performance in terms of ride quality and safety, for which an overlay would be an appropriate or necessary solution. OVERDRIVE can provide qualitative advice to the user in such situations. For example, if a section of pavement has unacceptable skid resistance or hydroplaning potential, but is otherwise structurally adequate, it may be necessary to overlay the pavement or apply some type of corrective surface treatment. If rutting of the pavement is also present, an overlay may be the only appropriate solution.

The overlay design thickness recommended by OVERDRIVE 1.1 is rounded to the nearest 0.5 in. for the sake of practicality (0.5 in. is also the minimum recommended overlay thickness). However, a quick calculation demonstrates the potential impact on an agency's budget of putting down overlays that are even 0.5 in. too thick. At a cost of about $10,000 per inch per lane-mile for asphalt (material only), the additional cost associated with an overlay that is 0.5 in. too thick, for a two-lane highway with paved shoulders, is about $12,500/mi. In just 100 mi, this misallocation is well over $1 million, which is a substantial amount compared with the maintenance and rehabilitation budgets of many local highway agencies. This underscores the substantial benefits that can be derived from development of improved design tools, especially knowledge-based tools such as OVERDRIVE, in this domain.

Finally, a powerful feature that is exploited in OVERDRIVE is the ability of the user, at the end of a design session, to view and then change any of the inputs for that session and have OVERDRIVE automatically redesign the structural thickness of an overlay. If, because of the user's changes, OVERDRIVE requires further information, this will be requested from the user. However, it is not necessary for the user to reenter all of the inputs. This feature is invaluable in investigating the impact of design inputs and assumptions (e.g., in the characterization of existing pavement layers or the subgrade).

CONCLUSIONS

The development of an initial knowledge-based system for assisting local engineers in designing the structural thickness of asphalt concrete pavement overlays has been discussed. The prototype version of this system, OVERDRIVE 1.1, has been implemented as an interactive microcomputer-based tool. The user can query the system for its reasoning, and the system allows the user to selectively modify input values or assumptions and to quickly assess the impacts of such changes on the structural thickness of overlay required.

In general, the potential for knowledge-based systems to become useful engineering tools in this domain is thought to be high.

Future research and development of OVERDRIVE will involve expanding and refining the knowledge base, incorporating additional knowledge acquired from pavement engineering specialists, providing a life-cycle cost analysis of each design thickness, and addressing the issue of uncertainty inherent in system inputs and conclusions.

On the basis of research to date with OVERDRIVE, it is concluded that a knowledge-based approach to pavement overlay design is feasible and, even in its present prototype form, OVERDRIVE is a potentially useful tool for local highway engineers.

ACKNOWLEDGMENTS

This research was supported by the National Science Foundation. This support, and the assistance of J. P. Mahoney of the University of Washington, is gratefully acknowledged.

REFERENCES

1. C. H. Oglesby and R. G. Hicks. *Highway Engineering*, 4th ed. John Wiley & Sons, New York, 1982.
2. S. G. Ritchie. Expert Systems in Pavement Management, *Transportation Research*, Part A, 1987, in press.
3. E. Rich. *Artificial Intelligence*. McGraw-Hill Book Company, New York, 1983.
4. C. Yeh, S. G. Ritchie, and J. B. Schneider. Potential Applications of Knowledge-Based Expert Systems in Transportation Planning and Engineering. In *Transportation Research Record 1076*, TRB, National Research Council, Washington, D.C., 1986, pp. 59–65.
5. S. G. Ritchie. State-of-the-Art of Expert Systems in Transportation Engineering. In *State-of-the-Art of Expert Systems in Civil Engineering* (M. L. Maher, ed.), U.S. Army Construction Engineering Laboratory, 1986.
6. C. W. Kostem and M. L. Maher, eds. *Expert Systems in Civil Engineering*. Symposium Proceedings, ASCE, 1986.
7. D. A. Waterman. *A Guide to Expert Systems*. Addison-Wesley, Inc., Reading, Mass., 1986.
8. F. Hayes-Roth, D. A. Waterman, and D. B. Lenat. *Building Expert Systems*. Addison-Wesley, Inc., Reading, Mass., 1983.
9. *Asphalt Overlays and Pavement Rehabilitation*. Manual Series 17 (MS-17). The Asphalt Institute, College Park, Md., June 1983.
10. S. G. Ritchie, C. Yeh, J. P. Mahoney, and N. C. Jackson. Development of an Expert System for Pavement Rehabilitation Decision Making. In *Transportation Research Record 1070*, TRB, National Research Council, Washington, D.C., 1986, pp. 96–103.
11. S. G. Ritchie, C. Yeh, J. P. Mahoney, and N. C. Jackson. A Surface Condition Expert System for Pavement Rehabilitation Planning. *Journal of Transportation Engineering*, ASCE, Vol. 113, No. 2, 1987.
12. *EXSYS Expert System Development Package User's Manual*. EXSYS Inc., Albuquerque, N. Mex., 1987.
13. F. N. Finn and C. L. Monismith. *NCHRP Synthesis of Highway Practice 116: Asphalt Overlay Design Procedures*. TRB, National Research Council, Washington, D.C., 1984.
14. J. P. Mahoney. State-of-the-Art on Asphalt Concrete Overlay Design Procedures. Presented at Tri-Regional Pavement Rehabilitation Workshop, Salt Lake City, Utah, 1984.
15. *Interim Guide for Design of Pavement Structures*. AASHTO, Washington, D.C., 1972.