Proceedings of the 1987 Winter Simulation Conference
A. Thesen, H. Grant, W. David Kelton (eds.)

USING EXPERT SYSTEMS TO SELECT SOFTWARE FOR TRAFFIC ANALYSIS

Edmond Chin-Ping Chang, PhD., PE.
Assistant Research Engineer
Texas Transportation Institute
Texas A&M University System

ABSTRACT

An expert system was developed by the Texas Transportation Institute (TTI) to assist users in selecting computerized software currently being supported by the Federal Highway Administration (FHWA), U. S. Department of Transportation. This system was designed to investigate the potential expert systems applications in transportation engineering. This study was performed to serve three basic purposes. First, it investigates the feasibility of developing small-scale traffic engineering knowledge-based expert systems using a simple knowledge engineering tool. Secondly, it develops an alternative method of recommending computer programs for user-specified applications. Thirdly, it investigates a possible approach for implementing advisory expert systems to be operated in the IBM PC/XT/AT microcomputer environment.

This paper describes the development of this expert system using a commercially available knowledge engineering tool developed by Level Five Research, Inc. INSIGHT 2+ was used to explore with the expert system programming in an inexpensive microcomputer environment. In operation, this system reviews and analyzes the user input information, evaluates with various reasoning paths and offers a conclusion. With the proper combination of knowledge programming tools and pre-identified 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 improvement, this type of expert system design can assist the user as a stand-alone expert advice system.

1. INTRODUCTION

Several traffic engineering programs are currently being supported by the Federal Highway Administration (FHWA), U. S. Department of Transportation as noted by Bryne, et al. (1982), FHWA (1985), and Transportation Research Board (1981). A prototype expert system was developed at the Texas Transportation Institute to assist in the selection of these microcomputer software packages. It was intended to apply the expert systems concept to assist individual users in selecting computer models for traffic analysis as suggested by Labrum (1980), Lieberman (1980,1982), and FHWA (1977). This system was also used to (1) investigate the potential feasibility of Expert Systems in traffic engineering, and (2) computerize the expert systems through Artificial Intelligence techniques; see Winston (1984), Harmon and King (1985), Waterman (1986), Buchanan and Shortlife (1985).

This paper describes the development of a prototype expert system. This study was performed to serve three basic purposes. First, it explores the feasibility for developing small-scale traffic engineering knowledge-based expert systems using a simple knowledge engineering tool. Secondly, it develops an alternative method of recommending computer programs for user-specified applications. Thirdly, it investigates a possible approach for implementing advisory type expert systems in the IBM PC/XT/AT microcomputer environment. INSIGHT 2+, a commercially available knowledge engineering tool developed by Level Five Research, Incorporated, was selected in 1986 due to the simplicity of its implementation to develop experimental expert systems in the IBM PC/XT/AT microcomputer environment. 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 in particular problem areas from pre-determined 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 as illustrated by Hendrickson (1986) and Chang (1987). The technical information was collected from various traffic engineering computer programs available from or being developed by the Federal Highway Administration (FHWA). For each program, information is processed based on 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 a suitable microcomputer traffic package as a stand-alone expert advice system.

The following sections describe the application of the expert systems design concept in developing this software selection advice system. The study first extracted representative information and converted the selected material acquired into an applicable form to characterize the evaluation criteria for developing a practical expert system. This paper investigates how the decision table was developed and used in the actual design. It also summarizes the working experience acquired and illustrates the basic program structure of the expert system as implemented in the INSIGHT 2+ Production Rule Language (PRL). And last, it discusses the various advantages and disadvantages discovered in the process of designing this simplified prototype expert system within the INSIGHT 2+ knowledge engineering environment.

2. TRAFFIC ENGINEERING SOFTWARE

As urban traffic demands increase, the most efficient coordination is required between existing traffic control devices and proper signal timing settings. Currently, a large number of computer software packages have been developed for providing better traffic analysis. With the increasing availability of microcomputers, numerous programs are accessible for traffic engineering applications. Without having to access a mainframe computer, a traffic engineer at virtually any location can analyze routine traffic engineering problems efficiently, thereby allowing more time for innovative engineering analysis. As more traffic professionals have begun to apply these traffic engineering oriented computer packages, programming development is being emphasized at the federal, state and community research levels to enhance these packages from time to time for more problem-solving capabilities. However, because of the tremendous ongoing enhancements taking place in each program, it is very hard to keep abreast of the developments in each of these traffic analysis packages. Therefore, there is an increasing need for developing expert type advice systems to assist the end user in selecting proper tools to provide specific analysis.

The Traffic Engineering Software packages, as analyzed, are supported and maintained by the Systems and Software Support Team, Federal Highway Administration (FHWA), Department of Transportation. The technical information used is based on the information manual released by the FHWA during the 1986 Annual TRB Meeting in Washington, D. C. The technical information, as in the FHWA Users Software Awareness Report, is illustrated in Figure 1 using the information concerning the MAXBAND program as an example. 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 are illustrated in the following categories by FHWA.

I. Signal Timing Optimization programs -

- A. SOAP -- Signal Operation Analysis,
- B. MAXBAND -- Maximum Band Optimization,
- C. AAP -- Arterial Analysis Package,
- D. TRANSYT-7F -- Traffic Network Study Tool,
- E. SIGOP-III -- Signal Optimization Model.

II. Traffic Flow Simulation Models -

- A. NETSIM -- Network Simulation Model,
- B. TRAFLO -- Macroscopic Network Model,
- C. FRESIM -- Freeway Simulation Model,
- D. ROADSIM -- Two-lane, Two-way Rural Road.

III. Other Traffic Engineering Software -

- A. ITDS -- Integrated Data System,
- B. HCS -- 1985HighwayCapacity Manual,
- C. PPD -- Platoon Progression Diagram,
- D. COUNTS-PC -- Signal Warrants Analysis,
- E. LINKOD -- Origin-Destination Synthesis.

3. EXPERT SYSTEMS DESIGN

An "Expert System" is a collection of computer programs or systems that applies in a specialized domain area. The "Expert System" combines both the 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 from the domain expert in the field; see Winston (1984), Harmon and King (1985), Waterman (1985).

There are five major components involved in Artificial Intelligence applications in the Expert Systems (ES) designs. The design components include the (1) expert system, (2) domain expert, (3) knowledge engineer, (4) expert systems-building tool, and (5) end user. Figure 2 summarizes the basic AI/ES components and their corresponding 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 specific program development. 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 was 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-THEN-AND-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 the computer hardware in a most primitive machine language format. Most conventional programming is made 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.

Harmon and King's (1985) concept of the evolution of artificial intelligence applications is illustrated in Figure 3. As indicated, a more application-oriented research seems to be the trend for the near future. AI/ES programming development can be separated into three areas: expert systems tools, national language queries, and AI languages. Normally, AI programming is made in the languages LISP and PROLOG. LISP, which stands for 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 consisting of both numbers or words. For years, LISP has been preferred by AI engineers in the United States. On the other hand, the PROLOG language, representing 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, Clocksin and Mellish (1984).

3.1. KNOWLEDGE ENGINEERING TOOLS

Knowledge Engineering Shells or Tools are often used to build an expert system. They provide all 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 to explain these programming development steps in ordinary English to enhance the understanding of the decision-support process. Symbolic ES operations using LISP and PROLOG usually take up huge computing memory, and thus may be executed slowly, particularly on microcomputers. The symbolic operations performed in LISP or PROLOG are usually implemented more efficiently using LISP as the operating system.

Knowledge Engineering Tools allow users to prototype a defined problem quickly and develop their own customized applications in less time than it takes an AI programming language. However, PC-based expert systems development tools are not suitable for large scale ES applications. 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 type computer for experimental program development, Waterman (1986), Buchanan and Shortlife (1985). Then the developed expert systems can be transferred to generate ES programs for practical applications that may eventually run on a personal computer in the microcomputer environment.

3.2. INSIGHT 2+ SYSTEM

INSIGHT 2+, developed by Level Five Research, Inc., is a tool that allows prototype AI/ES programming. INSIGHT 2+ is a microcomputer tool. It is for users to apply knowledge, form conclusions from facts, and solve problems in small-scale applications. Unlike ordinary data base systems which merely store, organize and recall information, the INSIGHT 2+ expert system allows the user to review and analyze the information given, evaluate it using various paths of reasoning, and offer 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.

The functional structure of the INSIGHT 2+ system is illustrated in Figure 4. INSIGHT 2+ can implement user expertise for problem-solving. By being able to question the users and maintain the knowledge base interactively, expert systems developed using INSIGHT 2+ may enhance the user's ability to analyze a problem and achieve suitable solutions. By accumulating the answers and probability of success from the trial-and-error process, the expert system can summarize the user modified solutions. During execution, INSIGHT 2+ automatically questions the user for better conclusions. Using the expertise programmed in the knowledge data base as obtained from the user, INSIGHT 2+'s inference is able to reason 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.

4. BASIC DESIGN PROCESS

A prototype expert system was implemented to provide computerized assistance as a potential computerized users' guide for selecting the necessary traffic engineering software. In this automated selection guide, three main goals or categories of each computer software package are evaluated. These packages include the traffic signal timing optimization, the traffic flow simulation model and other traffic engineering analyses. Each computer package is treated as a subgoal as described earlier. 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 the 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 was further explained along with the proper conclusions in this expert system.

This section describes 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 best be described in the seven-step process outlined in Figure 5. This design process includes the following steps: extracting basic information, designing a decision table, setting up evaluation goals, selecting evaluation constraints, developing a production rule language, debugging the production rule language, and completing the program recommendations.

4.1. EXTRACT BASIC INFORMATION

Information used to develop this prototype expert system was extracted from three sources. They include: (1) The Second Issue of the Traffic Software Users Awareness Report, (2) The BASIC program developed and demonstrated by the FHWA at the 1986 Annual Transportation Research Board meeting in Washington, D.C., and (3) The working experience of the Traffic Operations Program of the Texas Transportation Institute. The Traffic Software Users Awareness Report, issued semi-annually to all users who have received any of FHWA's traffic engineering software, contains the latest information on FHWA software. In the FHWA, the Systems and Software Support Team in the Office of Traffic Operations 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 into three categories. These FHWA-supported packages include the Traffic Signal Timing Optimization Programs, Traffic Flow Simulation Models, and other Traffic Engineering Analysis Software. Functionally, these traffic analysis packages include some of the popularly used 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. At first, the relevant information for each of the traffic models was extracted and summarized from the Traffic Software Users Awareness Report. The information was summarized by reading the article very carefully, comparing different traffic engineering software with each other, and differentiating special characteristics according to engineering judgement and the criteria stated in the Traffic Software Users Awareness Report. Based on the results from the comparisons, the current FHWA-supported traffic engineering software packages were divided into three functions:

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. The functions provided by these simulation models include the simulation capability of various types of transportation facilities. Applicable transportation facilities may also include isolated intersections, arterial streets, networks, highways, rural highways, and urban highways.

Other Traffic Engineering Software -- Other traffic engineering software, as 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 the relevant and irrelevant information from the available technical material, the domain expert traffic engineer then modifies the design of the decision-making process according to his past working experience with these models. He then determines which information should be emphasized for analysis and identifies factors from the specialized traffic engineering technology to evaluate the relative importance of criteria used in the practical expert systems design.

4.2. DESIGN DECISION TABLE

Decision table analysis is a decision-making aid. It can assist ES designers as an engineering tool in the design-and-evaluation process, Hendrickson (1986) and Chang (1987). Table 1 displays a simplified version of the decision table used for studying two basic design elements used in this particular expert system. In the decision table, the horizontal components represent the main goals and subgoals defined in the expert system. The first horizontal level components in the table are the main goals. The second horizontal level components are the subgoals. On the other hand, the vertical components of the decision table represent the facts and rules, such as the design constraints and potential areas which could be investigated by each of the traffic analysis packages.

Table 1 demonstrates one example of how the decision table was applied to analyze the MAXBAND program in the design process. As illustrated, 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 summarizing the goal, subgoal, and design constraints, the production rules are then specified for actual program development. The major advantages of using this type of decision table in the practical expert system development include the ability to:

1. Summarize the basic relationships of constraints.
2. Evaluate the independent constraints.
3. Study the detailed interrelationships among major variables in a systematic approach.

   Note - However, 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 in Table 1.

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 and design constraints.

4.3. SET UP EVALUATION GOALS

The basic evaluation goals for this prototype expert system, as described earlier, are to provide specific recommendations to the proper software package depending on user-input requirements. The actual INSIGHT 2+ PRL program is constructed directly toward this objective as follows:

```
1. Program belongs to Signal Timing Optimization
   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 Analysis 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
```

As indicated in the above definition, the INSIGHT 2+ program uses a set of outline type evaluation goals with different degrees of recommended actions being coded as part of the prototype expert system. In this particular system setup, several things are noted. First, the definition of this goal statement is identical to the functional classification as indicated in the Traffic Engineer Section. 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 not to classify the application of these programs into three main goals but rather to separate all programs into fourteen different programs.

4.4. SELECT EVALUATION CONSTRAINTS

Evaluation criteria are very hard to identify because of the basic differences for selecting the proper evaluation constraints. The evaluation criteria used for this analysis were that each transportation-related computer analysis software package be unique, identifiable and classifiable for balanced classification in system design.

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

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

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

4.5. 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 PRL by representing knowledge in terms of IF...AND...THEN...ELSE rules which 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 users to work with specialized knowledge. This study applies only with the known simple-facts and question-answer type query in the evaluation. Knowledge bases created by INSIGHT 2+ can be executed quickly in a microcomputer-based work station environment.

Basically, the key words of the INSIGHT 2+ production rule language (PRL) consist of a set of 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 to determine among 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. Secondly, the explicit information for designing the detailed expert system structure is defined with the commands TITLE, THRESHOLD, CONFIDENCE and GOALS. Thirdly, the production rules are set up according to the nature of the conclusions and recommendations for traffic engineering management. Lastly, the trace report provided in the INSIGHT 2+ knowledge engineering system enables the study of the decision-making process according to the specific production rule defined. An example is illustrated below to describe the information selection process in the MAXBAND program using the basic information obtained from the FHWA software awareness report. It demonstrates the rules coded for the MAXBAND program according to the required constraints extracted from Figure 1 described earlier. As indicated, a set of natural language type program statements first defines the prerequisite conditions for determining the function of the MAXBAND program in this prototype system.

```
RULE For selecting program MAXBAND
IF Program belongs to Signal Timing Optimization
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 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 and relationships about a problem that are embedded in the expert's knowledge. For example, if the average experience of the traffic engineer suggests that the capability of 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 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 to the facts and information entered by users. If the user-input information is incomplete, it 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 made based on the reasoning to reach final conclusions. Moreover, it can provide the user with English-type query question-and-answer prompts rather than just output with 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.

4.6. DEBUG THE PRODUCTION RULE LANGUAGE

Program testing and debugging are essential to successful computer programming and ES applications. It is relatively easy to compile and debug the

Production Rule Language because of the program support environment. 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, "FHWAINFO" in this case, to start program execution. After compiling production rules, INSIGHT 2+ will flag the error messages until the compiled knowledge base can be obtained. Then, run the program by using the function keys which are defined at the bottom of the screen display. 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, the proper recommendations will be given to meet the user's input requirements.

After reviewing the trace report, the production rules can be revised based on 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 prioritize their relative importance and how each goal is analyzed in the system. Since the system separates each goal according to its unique identification numbers and characters, these items are very important in the initial design of the production rule language to avoid any potential problems. Logical errors coded in PRL may create a lot of design problems later on. Therefore, the selection of the proper facts and constraints are important for the successful expert system operation.

4.7. 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 the users how to execute and implement this expert system before actual INSIGHT execution.

   EXECUTION STEPS

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

   II. PRODUCTION RULE ANALYSIS.

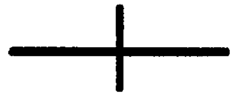
   III. RECOMMEND SOLUTION ALTERNATIVE.

   IV. OPTIONAL TRACE REPORT.

2. EXPAND additional information -

   The function EXPAND can be used to describe the question acquired in the production rules and constraints in the query analysis. This command and other similar keywords 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 an Isolated Intersection.



3. DISPLAY supportive information -

   This particular functional command is used to expand the basic and relevant information for 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 for more information once conclusions have been made from the knowledge inference process.

   DISPLAY MAXBAND (MAXimum BANDwidth)

   MAXBAND - MAXimum BANDwidth

LATEST AVAILABLE VERSION: A time-space diagram was recently added to the original version of the program. The program is also available for mainframe computer application in FORTRAN-77.

5. PROGRAM STRUCTURE

The structure of the expert system, as described in this paper, is illustrated in Figure 6. The program functional structure clarifies the interrelationships among the six components: (1) main goal, (2) subgoal, (3) selection criteria, (4) constraints, (5) rules, and (6) 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. Once the user selects a knowledge base on a particular topic, INSIGHT 2+ will search for all the 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 which INSIGHT 2+ runs. The knowledge engineer creates a knowledge base using the Production Rule Language (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. Applications of the INSIGHT 2+ tools can be performed best on areas that require routine professional judgement. They can assist engineers and managers in designing procedures for implementations. These advantages facilitate the development of expert advice systems to perform the operations of various jobs among many people at different locations.

6. USING INSIGHT 2+

The advantages of using a knowledge engineering tool such as INSIGHT 2+ are its easily understood programming structure and the well-equipped support functions in a user-friendly environment. There are also other advantages of using a knowledge engineering tool such as the INSIGHT 2+ system. By sequencing the definition of goals in the production rule, the statement of goals and subgoals can be better prioritized. Although the order of the constraints in the production rules is not very important in the query input, the interpreter will seek to optimize the execution sequence and the operation of the expert system. Therefore, the order of evaluation constraints within the production rules will not influence execution of the expert system with defined goals and subgoals.

Just as developing a knowledge base in INSIGHT 2+ is simple, using the knowledge engineering tool is also an easy task. The information which needs to be defined is the specific constraints required to determine each individual goal and subgoal using the decision table analysis. The coding of the PRL is efficient within the knowledge engineering programming environment. Then, both forward- and backward- reasoning processes can be performed in this expert system without additional computer programming efforts. The maintenance of this PRL program can be accomplished easily using the built-in editing function or the implementation of 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 in order to analyze, reason, and provide solutions to problems that would normally require human expertise. INSIGHT 2+ uses both backward- and forward-chaining inference mechanisms. 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 pertained to a particular application or to a pattern described similarly by the knowledge rules. In a backward-chaining application, INSIGHT 2+ begins with a specific software package and determines whether or not the pre-conditions justify using that package.

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

7. CONCLUSIONS AND RECOMMENDATIONS

This study evaluates the feasibility of assisting the selection of programs by AI techniques through the Expert Systems Design. 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 either in 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 made through 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-based 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 directions for improving this prototype expert system from the perspectives of the knowledge engineer, domain expert, and end user (12,13,14).

From a knowledge engineer programming standpoint -

- o Modify the goals, subgoals, rules,and constraints,
- o Add a debug error message in the trace report,
- o Include a logic table in the trace report, and
- o Enhance the program through INSIGHT 2+.

From a domain expert applications standpoint -

- o Modify design using manual procedures,
- o Work with other knowledge engineering systems,
- o Obtain experience from teaching end users, and
- o Use other computer programs.

From an end user applications standpoint -

- o Develop the expert system to interface with other software, such as DBASE and PASCAL programs,
- o Expand knowledge bases to help the end users, and
- o Provide determinable threshold settings.

It is further recommended that expansions be enhanced to this prototype expert system to provide a computerized expert system for advising end users to select proper computer programs for effective traffic engineering analysis. This application will be especially useful for aiding users in the future to select suitable computer software packages in the TRAF family as supported by the Federal Highway Administration, U. S. Department of Transportation. 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-orient problem-solving applications.

ACKNOWLEDGMENTS

The author appreciates the research support of the Texas Department of Highways and Public Transportation in cooperation with the Federal Highway Administration, the U. S. Department of Transportation, and Texas Transportation Institute, Texas A&M University System.

DISCLAIMER

IBM Personal Computer (PC) and PC DOS are products of the IBM Corporation.
MS and MS DOS are registered trademarks of the Microsoft Corporation.
TURBO PROLOG is a trademark of Borland International, Incorporated.
INSIGHT 2+ is a trademark of Level Five Research, Incorporated.

REFERENCES

1. Office of Safety and Traffic Operations Research and Development (1985). "Use of Traffic Simulation Models in Project Planning and Evaluating." Traffic Systems Division, Federal Highway Administration, U. S. Department of Transportation, Washington, D.C.

3. Byrne, A.S., De Laski, A.B., Courage, K.G. and Wallace, C.E. (1982). "Handbook of Computer Models for Traffic Operations Analysis." FHWA Report No. TS-82-2134, Washington, D.C.

4. Transportation Research Board (1981). "The Application of Traffic Simulation Models." Transportation Research Board Special Report 194, Washington, D.C.

5. Hsu, Y.S. and Munnjal, P.K. (1974). "Freeway Digital Simulation Models." Transportation Research Record 509, Washington, D.C., pp. 24-41.

6. Labrum, W.D. and Farr, R.M. (1980). "Traffic Network Analysis with NETSIM in a Small Urban Grid." FHWA Report No. TS-80-230, Washington, D.C.

7. Lieberman, E.B. and Worrall, R.D. (1980). "Traffic Network Analysis with NETSIM -- A User's Guide." FHWA Report No. IP-80-3, Washington, D.C.

8. Wicks, D.A. and Andrews, B.J. (1980). "Development and Testing of INTRAS: A Microscopic Freeway Simulation Model," Volumes 1-4. FHWA Report Nos. RD-80-106, RD-80-107, RD-80-108, and RD-80-109, Washington, D.C.

9. Lieberman, E. (1982). "Microscopic Simulation for Urban Traffic Management," Volumes 1-5. FHWA Report Nos. RD-80-113, RD-80-114, RD-80-115, RD-80-116, and RD-80-117, Washington, D.C.

10. Federal Highway Administration (1977). "An Introduction to Urban Travel Demand Forecasting - A Self-Instructional Text." User-Oriented Materials for UTPS. Federal Highway Administration, Urban Mass Transportation Administration, Washington, D.C.

11. Winston, P.H. (1984). Artificial Intelligence, Second Edition. Addison-Wesley, Reading, MA.

12. Harmon, P. and King, D. (1985). Expert Systems: Artificial Intelligence in Business. Addison-Wesley, Reading, MA.

13. Waterman, D.A. (1986). A Guide to Expert Systems. Addison-Wesley, Reading, MA.

14. Buchanan, B.G. and Shortlife, E.H. (1985). Rule-Based Expert Systems: The MYCIN Experiments of the Stanford Heuristic Programming Project. Addison-Wesley, Reading, MA.

16. Borland International (1986). "TURBO PROLOG - The Natural Language of Artificial Intelligence - Owner's Manual," Borland International, Scotts Valley, CA.

17. Level Five Research (1986). "INSIGHT 1 Reference Manual." Level Five Research, Indialantic, FL.

18. Level Five Research (1986). "INSIGHT 2+ Reference Manual." Level Five Research, Indialantic, FL.

19. Hendrickson, C.T., Rehak, D.R. and Fenves, S.J. (1986). "Expert Systems in Transportation Systems Engineering." Transportation Research,

20. Zozaya-Gorostiza, C. and Hendrickson, C.T. (1986). "An Expert System for Traffic Signal Setting Assistance." Civil Engineering Department, Carnegie-Mellon University. Paper prepared for the ASCE Journal of Transportation Engineering. January 1986.

21. Clocksin, W.F. and Mellish, C.S. (1984). Programming in PROLOG, Second Edition. Springer-Verlag, New York.

22. Chang, E.C. (1987). "Select Traffic Analysis Software Using Expert Systems." Paper presented at the 1987 Annual Transportation Research Board Meeting, Washington, D.C.

23. Chang, E.C. (1987). "Expert Systems Application on Left Turn Signal Treatment." Paper presented at the 1987 Annual Transportation Research Board Meeting, Washington, D.C.

EXAMPLES OF THE MATERIALS USED FOR THE MAXBAND MODEL

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 the Material Used for MAXBAND

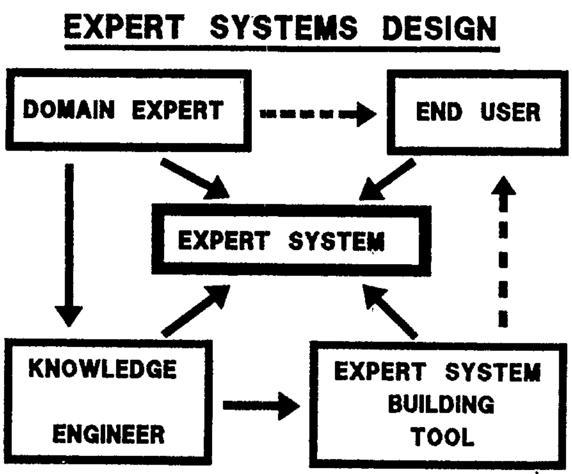


Figure 2. Main Components in Expert Systems Design.

Source: Reference (12,13).

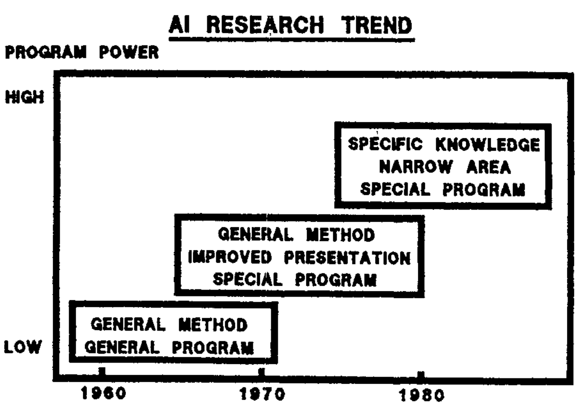


Figure 3. Evolution of Artificial Intelligence

Source: Reference (13).

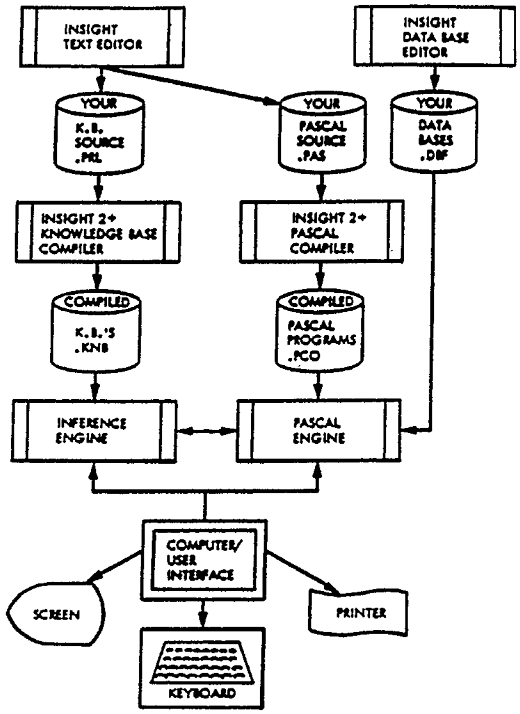


Figure 4. Functional Structure of the INSIGHT 2+

Source: Reference (17,18).

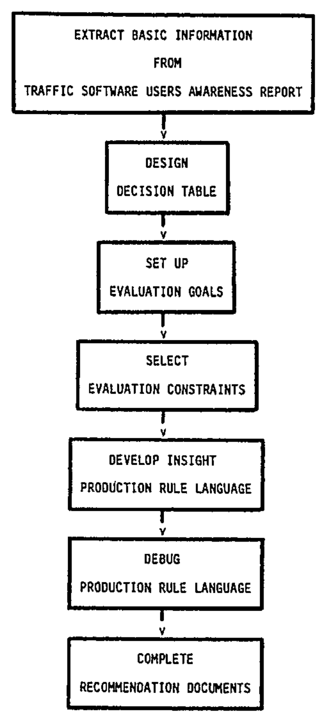


Figure 5. Basic Design Process.

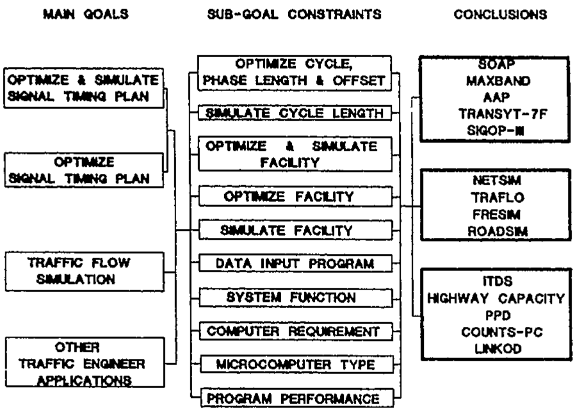


Figure 6. Basic Program Structure.

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	