

DTIC

DEPARTMENT OF THE ARMY
ENGINEERING (CASE) PROJECT

2

CONTRACT REPORT ITL-89-1

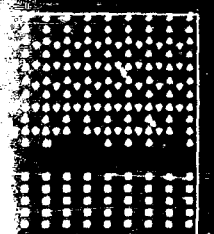
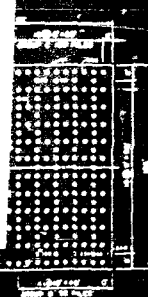
STATE OF THE ART ON EXPERT SYSTEMS APPLICATIONS IN DESIGN, CONSTRUCTION AND MAINTENANCE OF STRUCTURES

by

M. Arockiasamy, Sunghoon Lee

Department of Ocean Engineering
Florida Atlantic University
Boca Raton, Florida 33431

AD-A214 428



DTIC
ELECTR
NOV 14 1989
S B D



September 1989
Final Report

Approved For Public Release; Distribution Unlimited

Prepared for DEPARTMENT OF THE ARMY
US Army Corps of Engineers
Washington, DC 20314-1000

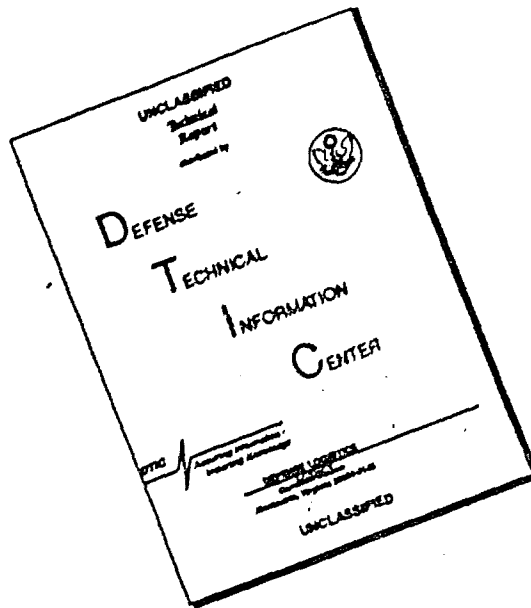
Under Contract No. DACA39-88-P-0114

Monitored by
US Army En

3809 Hallway East

89 11 13 052

DISCLAIMER NOTICE



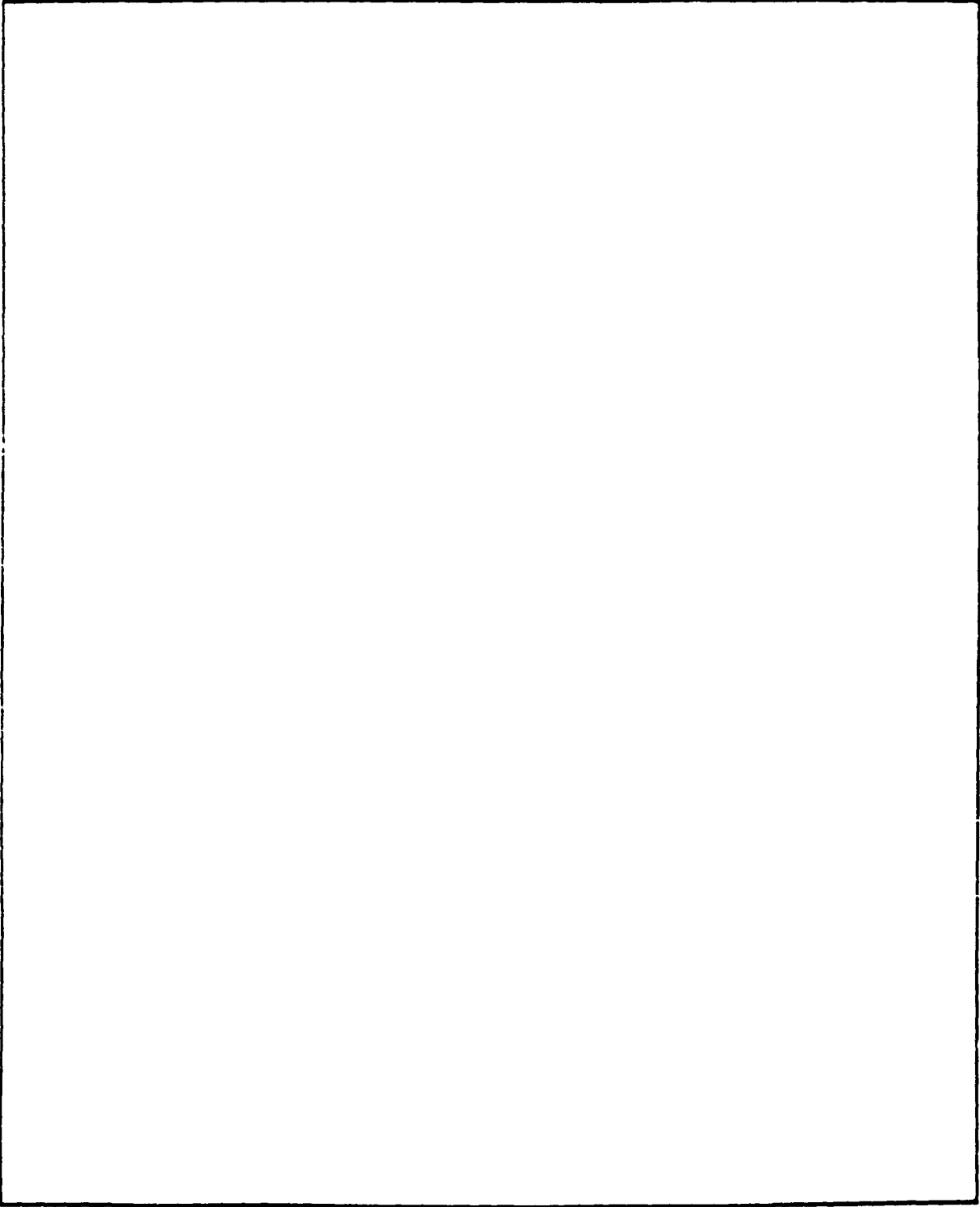
THIS DOCUMENT IS BEST QUALITY AVAILABLE. THE COPY FURNISHED TO DTIC CONTAINED A SIGNIFICANT NUMBER OF PAGES WHICH DO NOT REPRODUCE LEGIBLY.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE				Form Approved OMB No 0704-0188	
1a REPORT SECURITY CLASSIFICATION UNCLASSIFIED			1b RESTRICTIVE MARKINGS		
2a SECURITY CLASSIFICATION AUTHORITY		3 DISTRIBUTION AVAILABILITY OF REPORT Approved for public release; distribution unlimited.			
2b DECLASSIFICATION/DOWNGRADING SCHEDULE					
4 PERFORMING ORGANIZATION REPORT NUMBER			5 MONITORING ORGANIZATION REPORT NUMBER Contract Report IITL-89-1		
6a NAME OF PERFORMING ORGANIZATION Florida Atlantic University		6b OFFICE SYMBOL (if applicable)	7a NAME OF MONITORING ORGANIZATION USAEWES, Information Technology Laboratory		
6c ADDRESS (City, State, and ZIP Code) Boca Raton, FL 33431		7b ADDRESS (City, State, and ZIP Code) 3909 Halls Ferry Road Vicksburg, MS 39180-6139			
8a NAME OF FUNDING SPONSORING ORGANIZATION US Army Corps of Engineers		8b OFFICE SYMBOL (if applicable)	9 PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER Contract No. DACA 89-98-W-0114		
8c ADDRESS (City, State, and ZIP Code) Washington, DC 20314-1000		10 SOURCE OF FUNDING NUMBERS			
		PROGRAM ELEMENT NO	PROJECT NO	TASK NO	WORK UNIT ACCESSION NO
11 TITLE (Include Security Classification) State of the Art on Expert Systems Applications in Design, Construction, and Maintenance of Structures					
12 PERSONAL AUTHOR(S) Brooksbary, M.; Lee, Sunghoon					
13a TYPE OF REPORT Final report		13b TIME COVERED FROM _____ to _____		14 DATE OF REPORT (Year, Month, Day) September 1989	15 PAGE COUNT 153
16 SUPPLEMENTARY NOTES Available from National Technical Information Service, 5285 Port Royal Road, Springfield, VA 22161.					
17 COSAT CODES			18 SUBJECT TERMS (Continue on reverse if necessary and identify by block number)		
FIELD	GROUP	SUB-GROUP	Construction engineering and management Expert system Structural design Design standards processing		
19 ABSTRACT (Continue on reverse if necessary and identify by block number) <p>A comprehensive review of knowledge-based expert system applications to structural design, design standards, and construction planning for typical structures is presented in this report. The example systems reviewed in structural design include buildings, retaining walls, bridges and trawlers. Other review in design standards contributed to improvements in the representation and organization, analysis, and use of standards. The review includes typical representative expert system applications in construction management and management, soil exploration, site layout, real-time construction operations, construction planning and scheduling, etc.</p>					
20 DISTRIBUTION STATEMENT OF ABSTRACT <input checked="" type="checkbox"/> UNCLASSIFIED <input type="checkbox"/> LIMITED <input type="checkbox"/> RESTRICTED			21 ABSTRACT SECURITY CLASSIFICATION Unclassified		
22 LIMITATION OF ABSTRACT UNCLASSIFIED			23 LIMITATION OF ABSTRACT UNCLASSIFIED		

SECURITY CLASSIFICATION OF THIS PAGE



SECURITY CLASSIFICATION OF THIS PAGE

PREFACE

This report provides a comprehensive review of knowledge-based expert system applications in the areas of structural design, design standards, and construction planning. This study will aid in the development of a comprehensive expert system for typical hydraulic structures. Funding for this report was provided by the US Army Engineer Waterways Experiment Station (WES), Information Technology Laboratory (ITL), Vicksburg, Mississippi. Matching funds were provided by the Department of Ocean Engineering, Florida Atlantic University (FAU), Boca Raton, Florida, Dr. S. E. Dunn, Chairman and Acting Dean of Engineering.

Principal Investigator of this study and author of this report was Dr. M. Arockiasamy, FAU, with assistance from Sunghoon Lee, Graduate Assistant. Ms. Barbara Steinberg, FAU, typed the report and coordinated the text and table layout. This work was managed and coordinated at WES by Dr. N. Radhakrishnan, Chief, ITL.

Acting Commander and Director of WES was LTC Jack R. Stephens, EN. Technical Director was Dr. Robert W. Whalin.

Accession For	
NTIS GRA&I	<input checked="" type="checkbox"/>
DTIC TAB	<input type="checkbox"/>
Unannounced	<input type="checkbox"/>
Justification	
By _____	
Distribution/	
Availability Codes	
Dist	Avail and/or Special
A-1	

TABLE OF CONTENTS

	<u>Page</u>
PREFACE	i
LIST OF FIGURES	vii
LIST OF TABLES	x
CHAPTER	
1 INTRODUCTION	1-1
1.1 WHAT IS AN EXPERT SYSTEM?	1-1
1.2 EXPERT SYSTEM ARCHITECTURE	1-2
1.3 ARCHITECTURAL VARIATIONS	1-5
1.3.1 Production System Model	1-6
1.3.2 Blackboard Model	1-6
1.4 PROGRAMMING LANGUAGES AND TOOLS FOR BUILDING EXPERT SYSTEMS	1-8
1.4.1 General Purpose Programming Languages	1-8
1.4.2 Research Tools	1-9
1.5 KNOWLEDGE ELICITATION PROCESS	1-10
2 EXPERT SYSTEMS IN STRUCTURAL DESIGN	2-1
2.1 INTRODUCTION	2-1
2.2 STRUCTURAL DESIGN PROCESS	2-1
2.2.1 Preliminary Design	2-2
2.2.2 Analysis	2-2
2.2.3 Detailed Design	2-2
2.3 EXPERT SYSTEM METHODOLOGIES FOR DESIGN	2-3
2.4 ES APPLICATIONS IN STRUCTURAL DESIGN.	2-6
2.4.1 Preliminary Design: HI-RISE	2-6
2.4.2 Design System for Low-Rise Industrial Buildings: LOW-RISE	2-10

		<u>Page</u>
2.4.3	Preliminary Design of Frameworks by Expert System	2-11
2.4.3.1	Data preparation	2-11
2.4.3.2	Preliminary design	2-12
2.4.4	Bridge Design System: BDES	2-14
2.4.5	ES for the Optimum Design of Bridge Trusses: BTEXPERT	2-16
2.4.5.1	Knowledge base	2-16
2.4.5.2	Inference mechanism	2-18
2.4.5.3	User interface	2-19
2.4.5.4	Explanation facility	2-19
2.4.5.5	Debugging facility	2-21
2.4.5.6	Knowledge acquisition	2-21
2.4.5.7	Knowledge base development	2-23
2.4.5.8	Mathematical optimization	2-25
2.4.6	Retaining Wall Design: RETWALL	2-25
2.4.6.1	Introduction	2-25
2.4.6.2	The selection module	2-26
2.4.6.3	The blockwork module	2-33
2.5	CONCLUDING REMARKS AND SUGGESTIONS FOR FURTHER WORK	2-40
3	DESIGN STANDARDS PROCESSING	3-1
3.1	INTRODUCTION	3-1
3.2	GENERIC DESIGN STANDARDS PROCESSING	3-1
3.2.1	Query Monitor	3-1
3.2.2	Roofload checker	3-3
3.2.3	Generic Standards Processing in a Knowledge- Based Expert System Environment: SPIKE.	3-4

		<u>Page</u>
3.3	AASHTO BRIDGE RATING SYSTEM	3-6
3.4	AUSTRALIAN MODEL UNIFORM BUILDING CODE: AMUBC	3-7
3.5	KNOWLEDGE-BASED STANDARDS PROCESSOR: SPEX	3-8
3.6	A PC BASED ES FOR AUTOMATED REINFORCED CONCRETE DESIGN CHECKING	3-11
3.6.1	Introduction	3-11
3.6.2	The Spread Sheet	3-11
3.6.3	Spread Sheet Conversion to M.1 Data	3-13
3.6.4	Expert System	3-13
3.6.4.1	The ACI knowledge base	3-13
3.6.4.2	Interface configuration	3-13
3.6.4.3	External function	3-14
3.6.4.3.1	External code	3-14
3.6.4.3.2	Rule partner	3-15
3.6.5	Report Generator	3-15
3.7	CONCLUDING REMARKS AND SUGGESTIONS FOR FURTHER WORK	3-15
4	CONSTRUCTION ENGINEERING AND MANAGEMENT	4-1
4.1	INTRODUCTION	4-1
4.2	EXPERT SYSTEMS IN CONSTRUCTION ENGINEERING	4-3
4.2.1	Soil Exploration Consultant: SOILCON	4-3
4.2.2	Layout of Temporary Construction Facilities: SIGHTPLAN	4-3
4.2.2.1	Introduction	4-3
4.2.2.2	Example of SIGHTPLAN'S design actions	4-6

		<u>Page</u>
4.2.3	Brickwork Expert: BERT	4-8
4.2.4	Site Layout Expert System: CONSITE . . .	4-10
4.2.4.1	Introduction	4-10
4.2.4.2	Facility and site representation	4-10
4.2.4.3	Domain knowledge and design status representations	4-13
4.2.4.4	Alternative representation	4-13
4.2.4.5	Constraints representation	4-13
4.2.4.6	Knowledge base organization	4-14
4.2.4.7	Problem solving strategy	4-14
4.2.5	ES for Contractor Prequalification . . .	4-16
4.2.5.1	Introduction	4-16
4.2.5.2	Knowledge acquisition strategy	4-16
4.2.5.3	Knowledge base design	4-18
4.2.6	Expert Systems in Real-Time Construction Operations	4-23
4.2.6.1	Introduction	4-23
4.2.6.2	Methodology	4-24
4.2.6.2.1	Single-scraper expert system	4-24
4.2.6.2.2	Fleet management expert system	4-25
4.3	EXPERT SYSTEMS IN CONSTRUCTION MANAGEMENT	4-27
4.3.1	Introduction	4-27
4.3.2	ES Architecture for Construction Planning: CONSTRUCTION PLANEX	4-27
4.3.3	Analysis of Contingencies in Project Plans: PLATFORM-II	4-34
4.3.4	Know-How Transfer Method	4-38

	<u>Page</u>	
4.3.5	Microcomputer-Based ES for Safety Evaluation: HOWSAFE	4-43
4.3.6	Construction Scheduling Knowledge Representation: CONSAES	4-46
4.3.6.1	Introduction	4-46
4.3.6.2	Methodology	4-47
4.3.6.3	Knowledge organization	4-49
4.3.6.4	Knowledge representation	4-50
4.3.6.5	Knowledge implementation	4-52
4.3.6.6	Mappings	4-52
4.3.6.7	Relevance of CONSAES	4-54
4.4	EXPERT SYSTEMS IN MAINTENANCE	4-54
4.4.1	Introduction	4-54
4.4.2	Centrifugal Pump Failure Diagnosis: PUMP PRO TM	4-55
4.4.3	Vibration Analysis Interpretation	4-58
4.4.4	Field Diagnosis of Welding Defects	4-59
4.4.5	ES for Concrete Pavement Evaluation and Rehabilitation	4-60
4.5	CONCLUDING REMARKS AND FUTURE TRENDS	4-65
REFERENCES	R-1

LIST OF FIGURES

- Fig. 1.1 Component of expert system
- Fig. 1.2 Production system model
- Fig. 1.3 Blackboard model (Maher, 1987)
- Fig. 1.4 ES building tool identification framework (Maher, 1987)
- Fig. 2.1 Inference network for a derivation problem (Maher, 1986)
- Fig. 2.2 Unconnected graph for a formation problem (Maher, 1986)
- Fig. 2.3 Graphical representation of input (Maher, 1984)
- Fig. 2.4 Variation of the beam moment of inertias (Ovunc, 1988)
- Fig. 2.5 Variation of column moment of inertias (Ovunc, 1988)
- Fig. 2.6 Architecture of BTEXPERT (Adeli, 1988)
- Fig. 2.7 A sample Pratt truss plotted by BTEXPERT (Adeli, 1988)
- Fig. 2.8 Example of explanation generated by BTEXPERT in response to how it arrived at value of parameter allowable-stress-range-in-fatigue (Adeli, 1988)
- Fig. 2.9 Example of explanation generated by BTEXPERT in response to WHY it is asking value of string parameter bridge-location (Adeli, 1988)
- Fig. 2.10 Example of WHAT explanation command for providing additional information about parameter steel-type (Adeli, 1988)
- Fig. 2.11 Structure of FCB's (Adeli, 1988)
- Fig. 2.12 Outline of an expert system for the selection of earth retaining structures (Hutchinson, 1988)
- Fig. 2.13 Schematic layout of all the knowledge blocks in the higher level module (Hutchinson, 1985)
- Fig. 2.14 Flow chart for knowledge on the requirement for earth retaining structures (Hutchinson, 1985)
- Fig. 2.15 The different wall footing (base) types used (Hutchinson, 1985)

- Fig. 3.1 Decision table for AISC specification provision 1.5.1.4 (Rasdorf and Wang, 1988)
- Fig. 3.2 Architecture of SPIKE standards processing system (Rasdorf and Wang, 1988)
- Fig. 3.3 The functional modules of SPEX (Garrett, 1986)
- Fig. 3.4 System architecture (Saouma, Jones, and Doshi, 1981)
- Fig. 4.1 SIGHTPLAN software environment (Tommelein, Levitt, Hayes - Roth, 1987)
- Fig. 4.2 Object description (Tommelein, Levitt, Hayes - Roth, 1987)
- Fig. 4.3 Objects in the site layout domain (Tommelein, Levitt, Hayes - Roth, 1987)
- Fig. 4.4 Example of reasoning about site layout (Tommelein, Levitt, Hayes - Roth, 1987)
- Fig. 4.5 Both subcontractors are located on site (Tommelein, Levitt, Hayes - Roth, 1987)
- Fig. 4.6 Convex polygon, representation of a job office (Hamiani and Popescu, 1988)
- Fig. 4.7 Representation of the job site of an office building (Mamiani and Popescu, 1988)
- Fig. 4.8 Organization of the knowledge base (Mamiani and Popescu, 1988)
- Fig. 4.9 Output of CONSITE after solving the office building problem (Mamiani and Popescu, 1988)
- Fig. 4.10 Structure of a rule-based expert system for contractor prequalification (Russell and Skibniewski, 1988)
- Fig. 4.11 Succession of points defining a portion of the load-growth curve (Paulson and Sotoodeh-kooch, 1987)
- Fig. 4.12 Overview of CONSTRUCTION PLANEX (Fenves, Flemming, Hendrickson, Maher, and Schmitt, 1988)
- Fig. 4.13 Illustration of a CONSTRUCTION PLANEX knowledge source (Fenves, Flemming, Hendrickson, Maher, and Schitt, 1988)
- Fig. 4.14 A portion of the platform - II project knowledge base (Kunz, Bonura, and Stezlmer, 1986)

- Fig. 4.15 Rule identifying the project and its location as issues (Kunz, Bonura, and Stezlner, 1986)
- Fig. 4.16 Rule used to identify the issues and alternatives in the problem analysis (Kunz, Bonura, and Stezlner, 1986)
- Fig. 4.17 Storage of know-how "standard work package matrix" method (Niwa and Okuma, 1982)
- Fig. 4.18 Total framework of risk management system (Niwa and Oluma, 1982)
- Fig. 4.19 Examples of use of know-how transfer method ES (Niwa and Okuma, 1982)
- Fig. 4.20 A portion of the HOWSAFE inference net (Levitt, 1986)
- Fig. 4.21 Knowledge structure (O'connor De La Garza, and Ibbs, 1986)
- Fig. 4.22 Knowledge metamorphosis (O'connor, De La Garza, and Ibbs, 1986)
- Fig. 4.23 Knowledge base taxonomy (O'connor, De La Garza, and Ibbs, 1986)
- Fig. 4.24 Typical question and associated text display in pump failure diagnosis (Finn and Reinschmidt, 1986)
- Fig. 4.25 Example rule using the MAIDS English-like format (Finn and Reinschmidt, 1986)
- Fig. 4.26 Decision tree for selection of rehabilitation approach for JPCP (Hall, Connor, Darter, and Carpenter, 1988)

LIST OF TABLES

Table No.

- 1.1 Characteristics of traditional programs and expert systems
- 2.1 Expert systems in structural design
- 2.2 Example of the design charts used for blockwork walls
- 4.1 Major factors utilized in the contractor prequalification process
- 4.2 Sample rules for the subfactor "Balance Sheet"

STATE OF THE ART ON EXPERT SYSTEMS APPLICATIONS IN DESIGN,
CONSTRUCTION, AND MAINTENANCE OF STRUCTURES

CHAPTER 1

INTRODUCTION

1.1 WHAT IS AN EXPERT SYSTEM ?

Expert systems (ES) are emerging as a means for automating the solution of problems that have not yet been formalized as algorithms. Applications of expert systems range from medical diagnosis to architectural design. Although there are many tools available for the development of expert systems that use classification or diagnostic problem solving strategies, the available tools which provide an environment for the development of a hierarchical planning or design strategy are very few. ES is a useful tool for solving ill-defined problems such as those in structural design, where intuition and experience are necessary ingredients. This section defines expert systems so as to establish a common vocabulary and a brief review of available tools.

Expert systems are generally defined as interactive computer programs incorporating judgment, experience, rules of thumb, intuition, and other expertise to provide knowledgeable advice about a variety of tasks (Gaschnig, Reboh, and Reiter, 1981; Fenves, 1986; Maher, 1987; Adeli, 1987). The above definition does not clearly distinguish expert systems from traditional computer programs. The traditional programs can be interactive,

and contain judgment and rules of thumb, yet they are not expert systems. The characterizing features of conventional programs and expert systems are listed below in Table 1.1:

Table 1.1 Characteristics of traditional programs and expert systems (Maher, 1987)

Traditional programs	Expert systems
i) Representation and use of data	Representation and use of knowledge
ii) Knowledge and control integrated	Knowledge and control separated
iii) Algorithmic (repetitive) process	Heuristic (inferential) process
iv) Effective manipulation of large data bases	Effective manipulation of large knowledge bases
v) Programmer must ensure uniqueness and completeness	Knowledge engineer inevitably relaxes uniqueness and completeness restraint
vi) Midrun explanation impossible	Midrun explanation desirable and achievable
vii) Oriented toward numerical processing	Oriented toward symbolic processing

1.2 EXPERT SYSTEM ARCHITECTURE

Knowledge-based expert systems (KBES) have been identified based on research in artificial intelligence as practical problem-solving tools. The basic architecture of an expert system has three basic components: the knowledge base, the context, and the inference mechanism. User interface and an explanation facility are two additional components which make the expert system more usable. Besides a knowledge acquisition

facility is desirable to enhance extensibility of the expert system. The components of an expert system are shown in Fig. 1.1.

The knowledge base in the expert system contains the facts and heuristics associated with the domain in which the expert system is applied. The facts are typically represented as declarative knowledge whereas heuristics take the form of rules. Modification of knowledge base is important in most engineering domains since knowledge is continually changing and expanding. Many expert system environments provide higher level representation schemes than procedural code, such as rules or frames to make the knowledge base as transparent as possible.

The context is the component of the expert system which initially contains the information that defines the parameters of the problem. As the ES reasons about the given problem, the context expands to include the information generated by the expert system to solve it. At completion of the problem solving process, the context contains all the intermediate results of the problem solving process as well as the solution. The context is a declarative form of the current state of the problem the expert system is solving.

The inference mechanism contains the control information and uses the knowledge base to modify and expand the text. It controls the reasoning strategy of the expert system through assertions, hypotheses, and conclusions. The reasoning process is controlled by the inference mechanism at different levels. When it operates at very low levels providing flexibility in

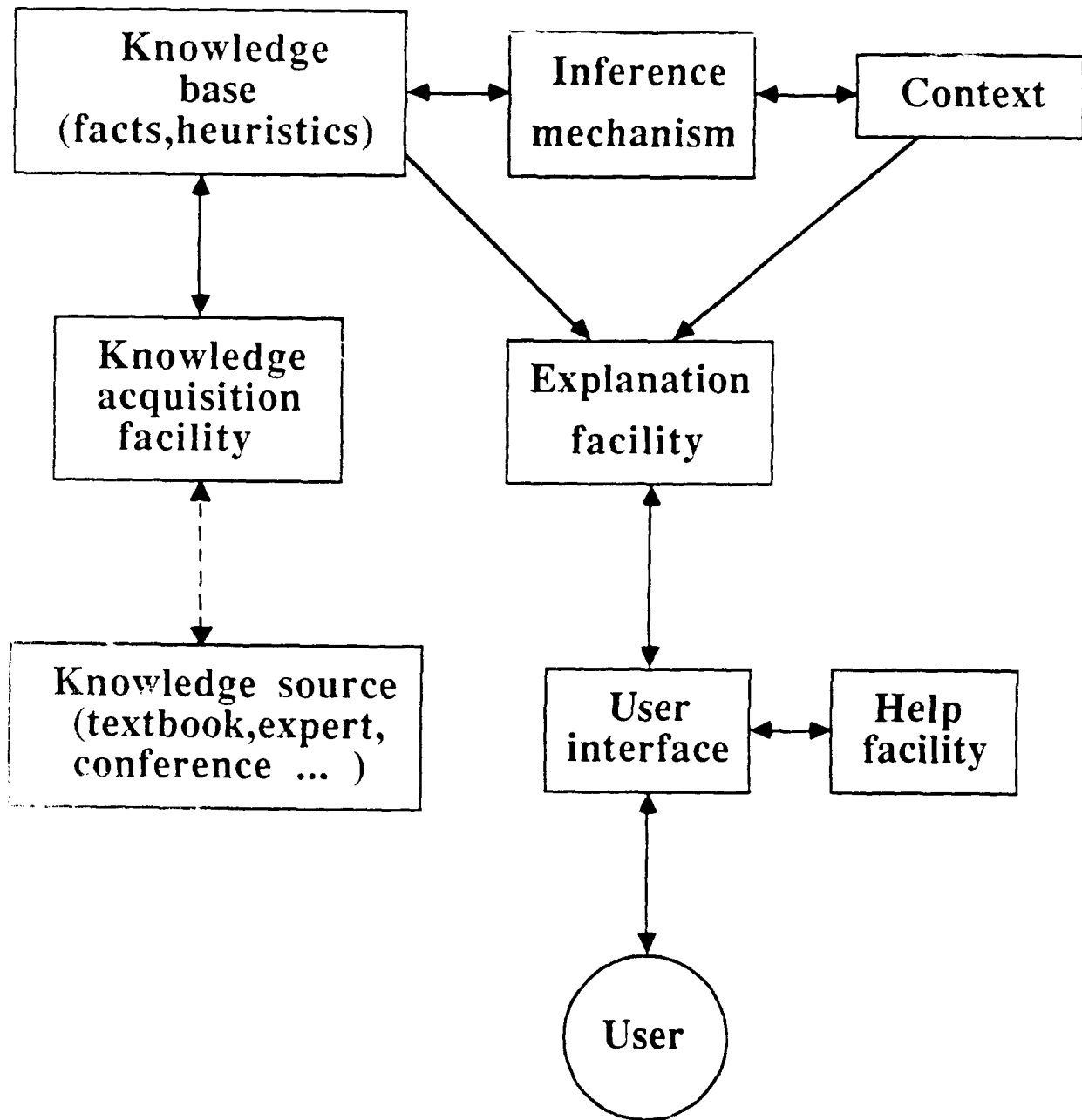


Fig. 1.1 Component of expert system

solution strategy, the knowledge base shall contain additional control information specific to the application domain. With more specific inference mechanism, the control information will be less in the knowledge base.

The explanation facility in an expert system provides answers to questions about the reasoning process used to develop a solution. A good explanation facility can explain both why a certain fact is requested and how a certain conclusion was reached. The knowledge acquisition facility in an expert system is the component that facilitates the structuring and development of the knowledge base. This facility acts as an editor, and the expert should be able to add to or modify the knowledge base as and when the expert system reveals gaps in the knowledge base. The knowledge acquisition facility understands the inference mechanism being used and can actively aid the expert in defining the knowledge base.

The user interface in the expert system allows the traditional capabilities of conventional user interfaces. It allows the user to interact with and query the expert system. In addition to being highly interactive, perhaps with 'HELP' facilities, an expert system user interface needs a transparency of dialogue, whereby some form of an explanation facility indicates the inference, or reasoning process used.

1.3 ARCHITECTURAL VARIATIONS

The production system model and the blackboard model are two of the most common variations in the basic architecture. The production system represents a powerful model for human

information processing and problem-solving ability. The blackboard model introduces the concept of multiple knowledge sources for handling complex problems.

1.3.1 Production System Model

The production system model considers the knowledge base as a set of rules termed as the production memory. A production system consists of three main elements:

- i) A set of IF-THEN rules or knowledge base
- ii) A global database or working memory

and

- iii) An inference mechanism

The rules are developed by the expert and need not be specified in the order in which they are to be considered. The inference mechanism in a production system provides the underlying strategy for identifying the productions that are eligible to be executed and the selection of one of these productions. The inference mechanisms, viz. forward-chaining and backward chaining fire rules according to the built-in reasoning process. Fig. 1.2 shows an illustrative production system model. The earliest implementations of the production system model (VanMelle, 1979) are EMYCIN and OPS5 (Forgy, 1981).

1.3.2 Blackboard Model

The blackboard model illustrated in Fig. 1.3 is based upon the separation of the knowledge base into knowledge sources and the use of a blackboard as a context. The blackboard, a central global database, plays as a communication vehicle among knowledge sources and keeps track of incremental changes made in the

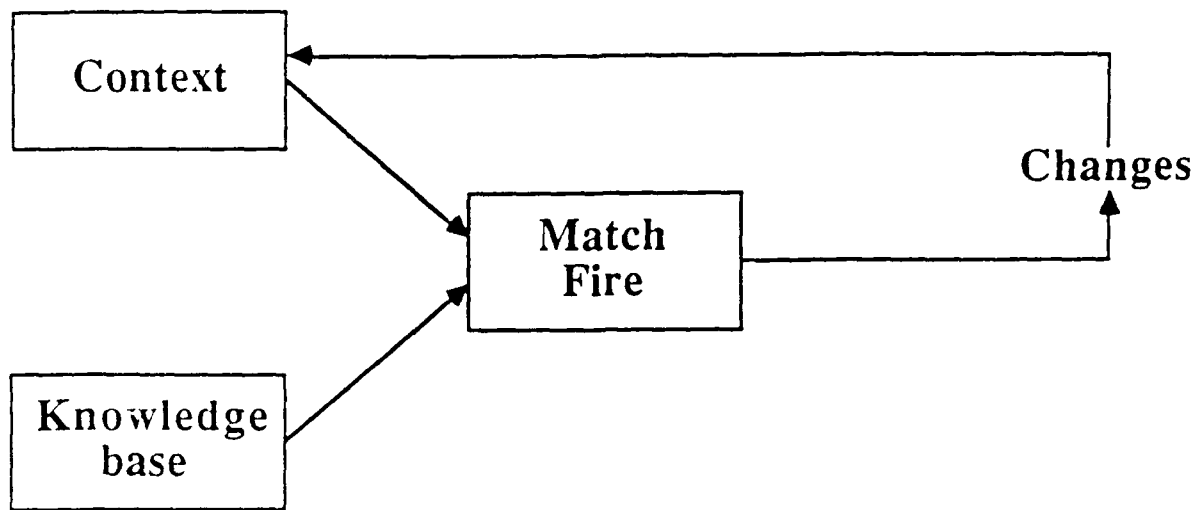


Fig. 1.2 Production system model

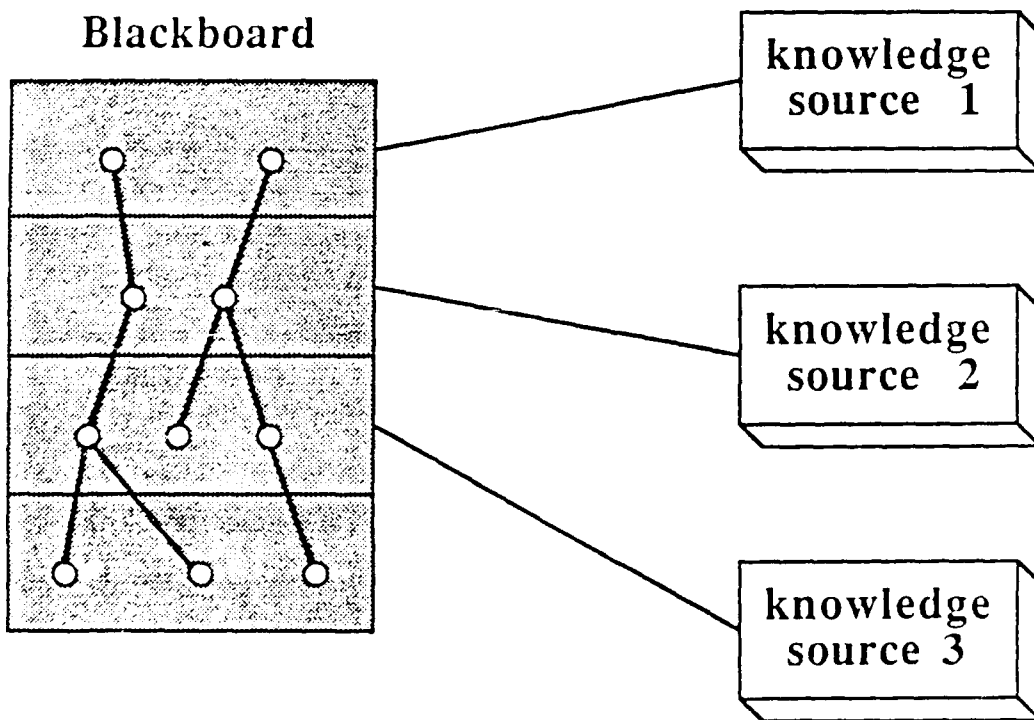


Fig. 1.3 Blackboard model (Maher, 1987)

current state of the problem until a solution is found. The blackboard will utilize a combination of forward and backward reasoning chains. The blackboard concept was first implemented in HEARSAY-II (Reddy, Erman and Neely, 1973). The blackboard model has been applied to problems involving distributed processing, multiple levels of knowledge, and multiple sources of knowledge. The problems being solved by the use of a blackboard model tend to be complex and hence require partitioning into subproblems.

1.4 PROGRAMMING LANGUAGES AND TOOLS FOR BUILDING EXPERT SYSTEMS

1.4.1 General Purpose Programming Languages

Expert systems can be written in any programming language, such as LISP, PROLOG, C, FORTRAN or PASCAL. LISP which is still the choice of many developers in the United States was one of the first languages directed toward symbolic representation and list processing. The concept of structured programming incorporated into PASCAL reduces the complexity through modular programming and effective communication; it allows the programmer to define variable types such as character, string, boolean (with values of either true or false), integer, real number, and array. PASCAL has the variable-type pointer which makes it possible to define logical trees. It can also be used for dynamic storage allocation. Turbo PASCAL has excellent string manipulation and powerful graphic capabilities. C is a very efficient language and is specially suitable for graphic-based programs. While LISP is memory intensive and requires large processing power, C has

limited symbolic manipulation and memory management capabilities.

PROLOG (PROgramming LOGic) which is based on formal logic is popular in Europe and Japan. It has its own inference mechanism. Experience with PROLOG based ES shells shows that PROLOG is a versatile language for database-type applications (Allwood, Steward and Trimble, 1985). However, certain limitations regarding numeric data types, large memory requirement, and slow execution with many implementations of the language are reported for ES development.

1.4.2 Research Tools

Selection of an expert system (ES) shell for engineering applications should be based on type of application, type of machine and operating systems, maximum number of rules allowed (in production systems), response time (in solving problems or answering questions), type of control strategy and inference mechanism, user interface (graphics, natural language processing, etc.), availability of complex mathematical routines, ability to interface with other programs written in the language of the shell, programming aids (editors, debuggers, and a help facility), user support, etc. For engineering problems numerical algorithmic routines must usually be combined with heuristics.

Although a number of expert systems have been developed, only a few of the more relevant ES tools are described below:

The first widely-used ES shell was created by stripping the medical knowledge base from MYCIN and called EMYCIN (for Essential MYCIN or Empty MYCIN) which is used to construct

diagnosis systems. EMYCIN is LISP based and uses production rules which have the form of associative (object-attribute-value) triples for knowledge representation and backward-chaining as the inference mechanism.

It has been used to develop SACON (Structural Analysis Consultant), an expert system for the application of a general purpose finite element structural analysis program (VanMelle, 1979; Bennett and Engelbre, 1979). PROSPECTOR also led to the development of another ES shell called KAS (Knowledge Acquisition System) which uses rule-based representation with a partitioned semantic net for organizing the process of rule matching. KAS which was implemented in INTERLISP uses both backward-chaining and forward-chaining and certainty factors and has explanation knowledge acquisition, and tracing facilities (Reboh, 1981). EXPERT, which is a major ES shell implemented in FORTRAN has explanation, knowledge acquisition, consistency checking, and trace facilities. When the ES developer adds a new rule EXPERT tests the consistency of the rule with the solutions of the representative cases stored in the database. A framework of ES tools shown in Fig. 1.4 can be used as comparative criteria to make the best choice of possible tools for a specific application.

1.5 KNOWLEDGE ELICITATION PROCESS (Firlej, 1985)

The real problems involved in building expert systems are those related to knowledge representation. The emphasis in the building of expert systems seems always on investigating technical issues and implementation of the knowledge already

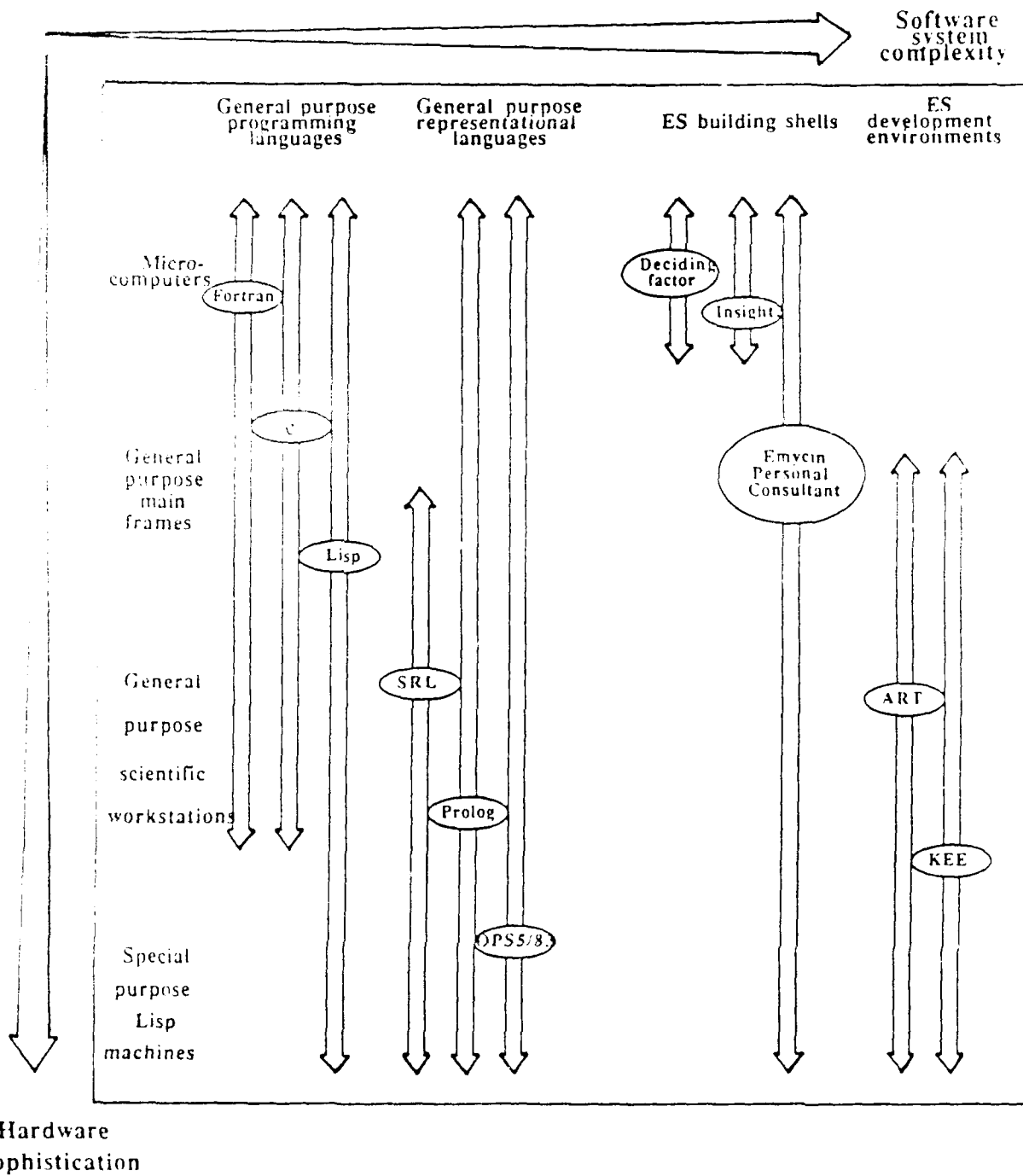


Fig. 1.4 ES building tool identification framework

(Maher, 1987)

elicited. The overall nature of the task is to extract knowledge from an expert in such a way as to reduce the risks and costs involved in the construction of a knowledge-based system. The information on the knowledge base of the expert systems can be obtained from two sources - literature and domain specific knowledge from experts. Literature sources include technical journals, textbooks, manuals, public and commercial documents and reports. A second source of domain specific knowledge is from experts to aid in the development of the system by providing their experience, intuition, judgment, rule of thumb, etc. Before contacting domain experts, the knowledge engineer, the system developer needs to review relevant literature to structure questions for the experts in such a way that the specific information sought is given naturally without tension.

It is essential to avoid dislocations within the interview, for example, to know when to keep quiet and when to prompt, when to direct and when to let the information flow. Since eliciting information from the expert on a large project might take several years, it is essential that the expert's interest and motivation is upheld throughout that period. An expert who finds the whole process tiring and unpleasant will show his feelings in the quality of his response. The obstacles and problems must be identified well in advance so that the elicitation process can proceed without interruptions. Practical issues, like tape recording and transcribing of interviews must be organized efficiently beforehand, so that the analysis of information is not delayed unnecessarily.

CHAPTER 2

EXPERT SYSTEMS IN STRUCTURAL DESIGN

2.1 INTRODUCTION

An overview of expert systems in civil engineering is presented in recent references (Fenves, Maher and Sriram, 1984; Maher, 1987; and Adeli, 1988). Potential applications of artificial intelligence (AI) in structural engineering design and detailing were first proposed by Fenves and Norabhoompipat (1978). An expanded model of the design process was proposed by Rooney and Smith (1983) by introducing a feedback mechanism consisting of i) acquisition of experience, ii) application of experience, and iii) database management. This model was then applied to a single span simply supported steel wide-flange beam. Most expert systems developed so far are basically experimental systems which show the present status and potential applications or present conceptual frameworks.

2.2 STRUCTURAL DESIGN PROCESS

The need to transmit loads in space to a support or foundation is first defined subject to constraints on cost, geometry and other criteria. The design process finally yields the detailed specifications of a structural configuration which would transmit the given loads with the desired levels of safety and serviceability. The three sequential stages in the design process are: preliminary design, analysis and detailed design.

2.2.1 Preliminary Design

The conceptual design relates to synthesis of potential configurations satisfying a few principal constraints. Synthesis of feasible structural configurations based on subsystems applicable to the particular design at hand, formulation and evaluation of specific constraints applicable to the chosen configurations and choice of one or more of these configurations are the important aspects of the preliminary design stage.

2.2.2 Analysis

This is the process of modeling the selected structural configuration and determining its response to external effects. Transformation of real structural configuration to a mathematical model, selection and use of analysis procedure and interpretation of analytical results in terms of the actual physical structure form the important components of this stage.

2.2.3 Detailed Design

This stage refers to the selection and proportioning of structural components which would satisfy all applicable constraints. This is again subdivided into a series of essentially hierarchical subproblems such as detailing the main structural components (beams, columns, etc.) followed by detailing of their subcomponents (connections, reinforcement, etc.) Within each subproblem, a further subdivision is made for selection based on certain controlling constraints (load-carrying capacity or buckling) followed by the evaluation of secondary constraints (e.g. local buckling or crippling).

A reanalysis would be required if the properties of components assumed at the analysis stage show significant deviations from those determined at the detailed design stage. Major and minor cycles of redesign may be necessary until a satisfactory optimal design is obtained. The conceptualize-analyze-detail is characteristic of any design example.

2.3 EXPERT SYSTEM METHODOLOGIES FOR DESIGN

The derivation approach and the formation approach are the two basic approaches used in expert systems. The derivation approach involves deriving the most appropriate solution for the given problem from a list of predefined solutions stored in the knowledge base of expert systems whereas the formation approach yields a solution from the eligible solution components stored in the knowledge base. An ES may use one or both of the approaches described above depending on the complexity of the problem being solved.

The search for a solution of the problem solving using a formation approach begins at an initial state of known facts and conditions which are combined to form a goal state. In a derivation approach, the known facts and conditions are used to derive the most appropriate goal state.

Forward-chaining, backward-chaining and mixed initiative are appropriate strategies for the implementation of a derivation approach. The goal states represent the potential solutions and the initial state represents the input data. The development of an inference network representing the connections between initial

states and goal states is illustrated in Fig 2.1. The advantage of using one of these strategies is that they are currently implemented in a variety of expert system tools so that the development process involves defining, testing and revising an inference network.

Problem reduction, plan-generate-test and agenda control are problem-solving strategies appropriate for implementing a formation approach. The concepts of hierarchical planning and least commitment, backtracking and constraint handling techniques could supplement these strategies. Fig. 2.2 shows an illustration of the unconnected graph of components. The solution is not completely defined by a goal state, but requires that the solution path should also be known. The disadvantage of using one of these strategies is the lack of a standard implementation or ES tool that employs a strategy appropriate for the formation approach. These strategies are typically implemented using a lower level language such as LISP or an ES shell such as KEE.

Representation and use of constraints are essential in any design application. Three operations on constraints are proposed by Stefik (1980):

- i) Constraint formulation is the operation of adding new constraints representing restrictions on variable bindings.
- ii) Constraint propagation is the operation of combining old constraints to form new constraints. This operation deals with interactions between subproblems through the reformulation of constraints from different subproblems.

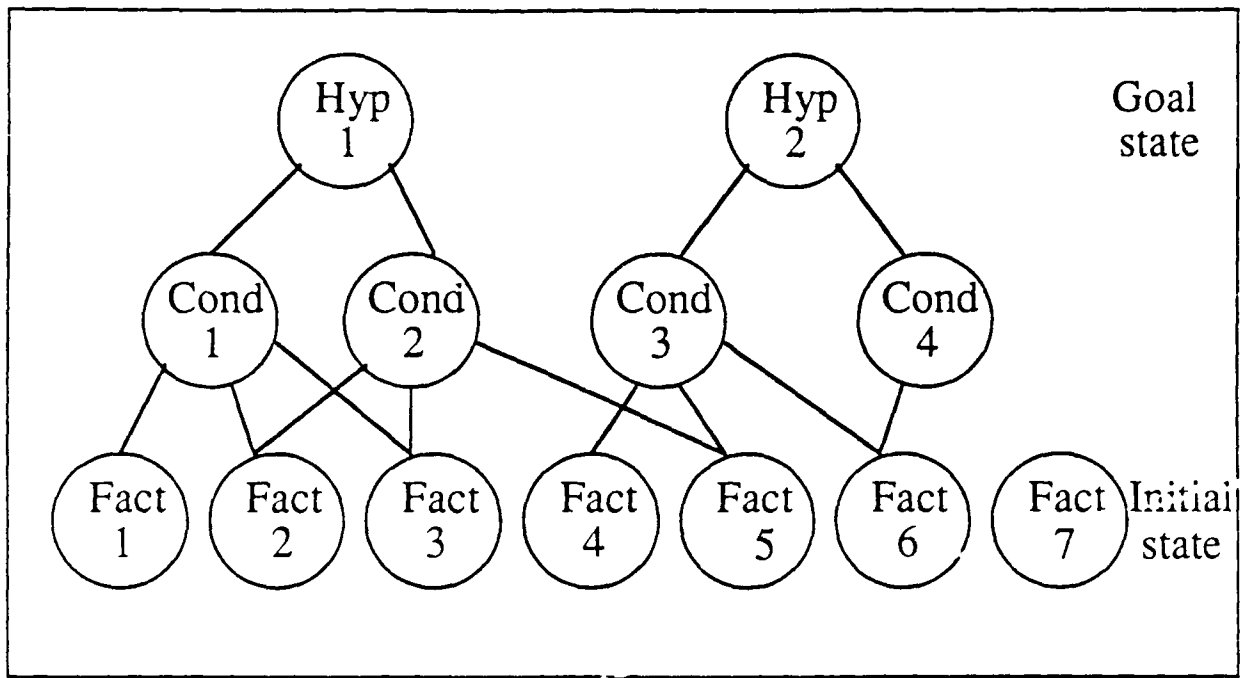


Fig. 2.1 Inference network for a derivation problem
(Maher, 1986)

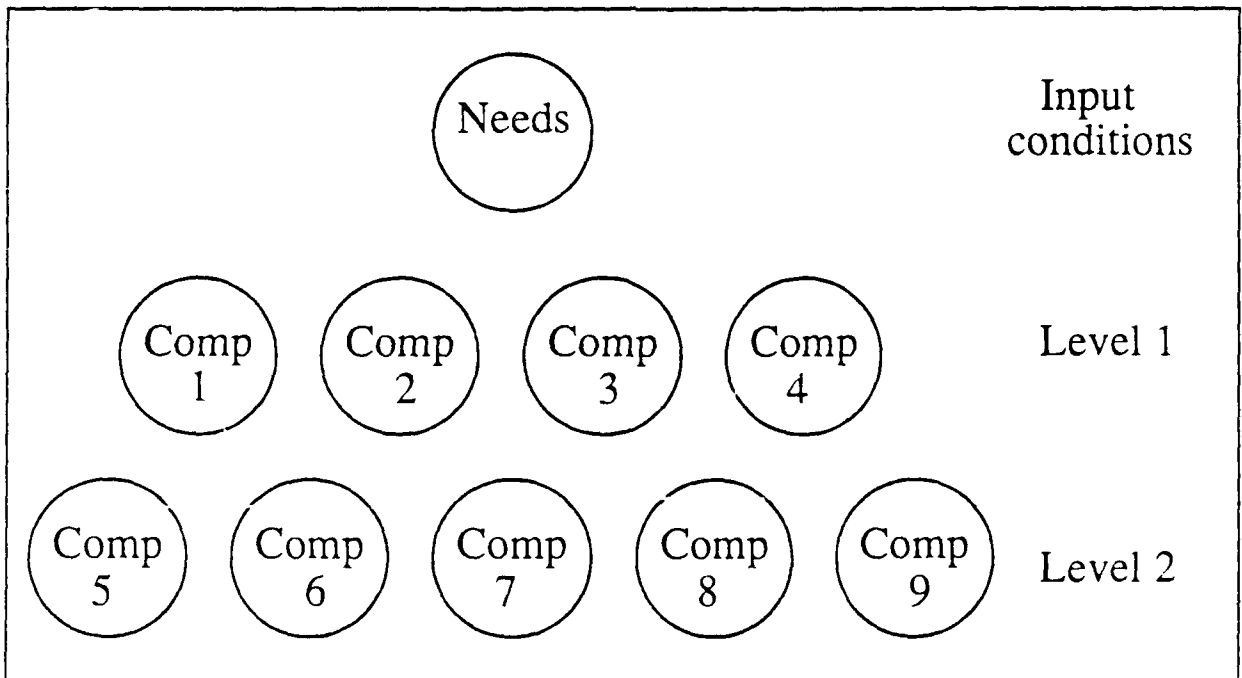


Fig. 2.2 Unconnected graph for a formation problem
(Maher, 1986)

iii) Constraint satisfaction is the operation of finding values for variables so that the constraints on these variables are satisfied.

Table 2.1 presents selected ES applications to structural design. Brief descriptions of only certain specific applications of ES are described in the following sections: Each application is presented with a general description of the problem, the methodology employed, the current state of the system and references.

2.4 ES APPLICATIONS IN STRUCTURAL DESIGN

2.4.1 Preliminary Design: HI-RISE

The preliminary structural design is based on the designer's experience as well as the understanding of the behavior of structural systems. Outlining a structural system for a given building requires a combination of structural system knowledge, experience and creativity. HI-RISE is an ES that forms and evaluates several alternative structural systems for a given three dimensional grid. The expertise in HI-RISE is derived primarily from a recent publication on preliminary structural design (Lin and Stotesburg, 1981) using approximate analysis techniques and applicable design heuristics.

Classes of generic structural subsystems are used as a basis for the generation of feasible systems. Some examples of structural subsystems are: rigidly connected frames, cores, trussed tubes, and braced frames. The generic structural subsystems are expanded and combined to fit the conditions of the

Table 2.1 Expert systems in structural design

System	Current State	Machine/Developer
RETWALL	developmental prototype	SUN 2
BDES	developmental prototype	IBM PC
WISER	developmental prototype	Symbolics 3640
HI-RISE	developmental prototype	VAX 11/750
LOW-RISE	operational prototype	VAX 11/750
ALL-RISE	operational prototype	VAX 11/750
SFOLDER	operational prototype	VAX 11/750
HI-COST	operational prototype	VAX 11/750
DESTINY	developmental	VAX 11/750
SSPG	developmental	Ohio State University
BTEXPERT	prototype	Ohio State University
RTEXPERT	developmental	Ohio State University
PRELIMINARY DESIGN OF FRAMEWORKS BY EXPERT SYSTEMS	developmental	University of South Western Louisiana

particular building. HI-RISE was developed using the PSRL language running on a DEC VAX system. PSRL provides a combination of frame-based and rule-based reasoning. Frames are used in HI-RISE to represent the knowledge of structural systems, subsystems, and components in a hierarchical manner. Rules are used to represent strategy and heuristic knowledge. LISP functions are used to represent approximate analysis procedures.

HI-RISE decomposes the structural design process into five subtasks: synthesis, analysis, parameter selection, evaluation and system selection. The synthesis subtask functions as a first search through the hierarchy of structural subsystems, using heuristic constraints to eliminate infeasible alternatives. The analysis subtask provides for an approximate analysis of a feasible alternative in order to determine the load distribution. The parameter selection subtask proportions key components. The evaluation subtask ranks all feasible alternatives using a heuristic evaluation function. System selection can be done by the user or by defaulting to the system with the best evaluation.

The input to HI-RISE is a three-dimensional grid as illustrated in Fig 2.3. The spatial constraints such as the location of vertical service shafts or internal spaces are specified in terms of their location on the input grid. The intended occupancy of the building, and the wind and live load are the additional input information required by HI-RISE. Once the input has been specified, the interaction between the user and HI-RISE is graphical. The user participates in the selection of a structural alternative from the set of feasible alternatives

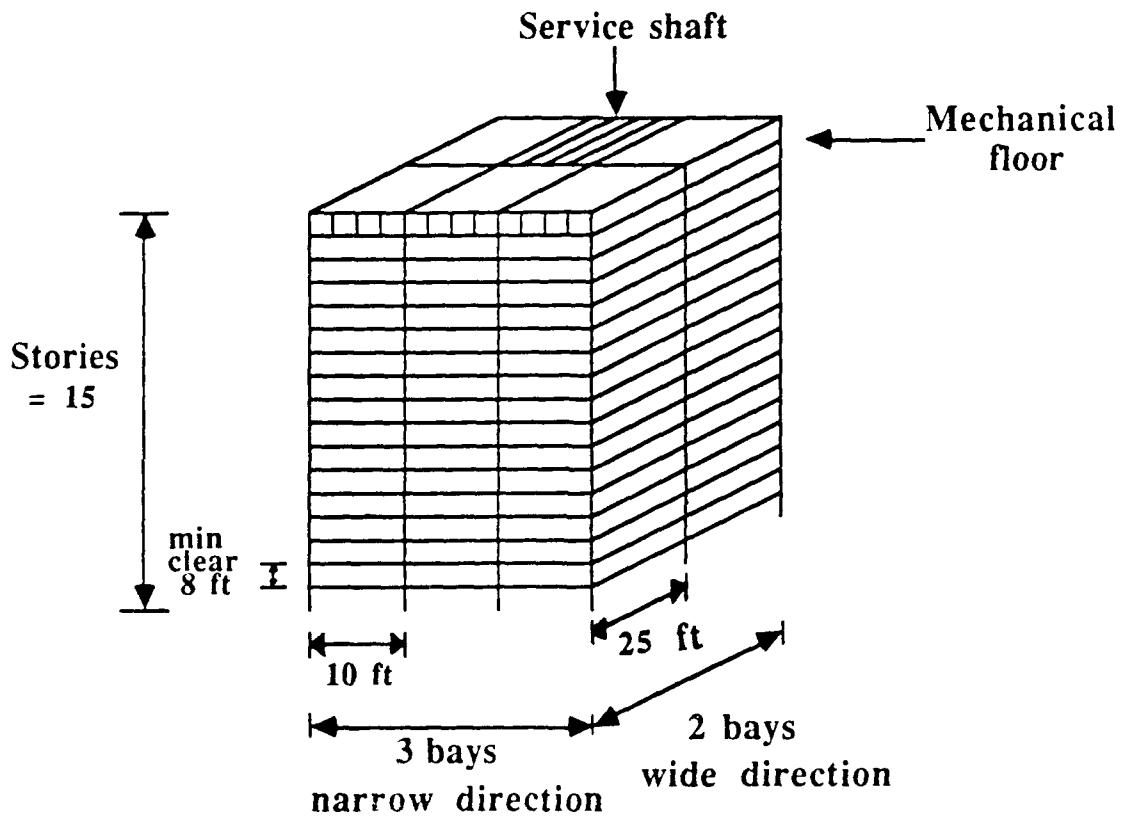


Fig. 2.3 Graphical representation of input (Maher, 1984)

generated by HI-RISE. The user may also request information about the building components of any feasible alternative.

HI-RISE is a developmental prototype ES which serves as a starting point for exploring the use of ES techniques for preliminary structural design process. Currently, HI-RISE is being extended and implemented in Knowledge Craft on a Micro Vax II (Maher, 1984, 1986).

2.4.2 Design System for Low-Rise Industrial Buildings: LOW-RISE

LOW-RISE aids in structural planning, preliminary design and evaluation of industrial type buildings. Planning consists of determining the components of the gravity and lateral load systems of various framing layouts that satisfy user input spatial constraints. Each alternative is ranked heuristically for comparison with other alternatives.

It was implemented in a combination of OPS5, LISP and C. Heuristic knowledge, generation of framing schemes and layouts for components of the gravity and lateral load systems were written in OPS5. More algorithmic parts such as analysis were coded in LISP. C was used to communicate with the database management system.

LOW-RISE relaxes the rigid spatial constraints of HI-RISE. The building is described in terms of large areas called departments, with each department identified by a column placement constraint. It first selects feasible structural configurations satisfying the column placement constraint separately for each department; it then attempts some global 'smoothing' strategies to align the grid across departments.

Finally, preliminary analysis, component sizing, costing, evaluation and ranking are performed on each alternative. This is an operational prototype expert system which has been developed with expertise supplied by experts from the Carnegie-Mellon University Architecture Department, American Bridge Company, and other industries (Camacho, 1985).

2.4.3 Preliminary Design of Frameworks by Expert System (Ovunc, 1988)

A knowledge-based ES is used in the preprocessor of a general purpose software. The first part includes information related to the geometry, quality of the materials and loads acting on the framework as data, whereas the second part contains the approximate sizes of all the members of the framework which are evaluated from the data provided in the first part. The second part which constitutes the knowledge-based expert system determines the member sizes using either the code requirements or certain approximate expressions. Moreover a cost analysis is also included in the second part depending on the type of structures and the quality of materials used.

The software for the preliminary design is developed mainly in FORTRAN language in order to provide the ability to handle complex mathematics and to facilitate interfacing the various final design or other softwares. The modules related to the graphics are written in BASIC language.

2.4.3.1 Data preparation

The first external data required by the preprocessor are

related to i) the selection of the computer type, ii) the processor to be interfaced, iii) the type of structural system to be analyzed, and iv) the type of the analysis to be performed with or without the preliminary design. The remaining external data of the specific structural system under consideration include i) the locations of the columns, ii) the types and qualities of the materials, iii) the dead loads such as floor covering, floor finishing, etc. iv) the gravitational live loads and v) soil conditions, types of foundations, etc.

2.4.3.2 Preliminary design

The preliminary design begins by checking the locations or spacings of the columns by considering the inference mechanisms or database depending on the structural plans, number of floors, floor heights, externally applied loads, the type and quality of materials used, etc. The thickness of the slabs are first evaluated for an optimum spacing of columns. The final design of the slab is performed by using the theory of plates or finite element method or the code requirements or from the database. After the final design of all the slabs, the transfer of the gravitational loads from the slabs to the beams are evaluated. Besides the dead and live loads transferred from the slabs, the wall loads, self weight of beams, horizontal loads due to the wind and earthquake are computed and absolute sizes of the members estimated using moment coefficients for continuous beams under gravitational loads, portal method for the frames under horizontal loads, inference mechanisms or database.

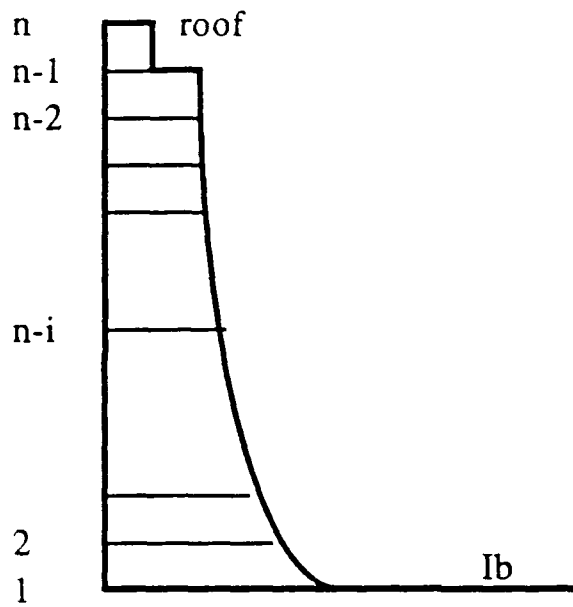


Fig. 2.4 Variation of the beam moment of inertias
(Ovunc, 1988)

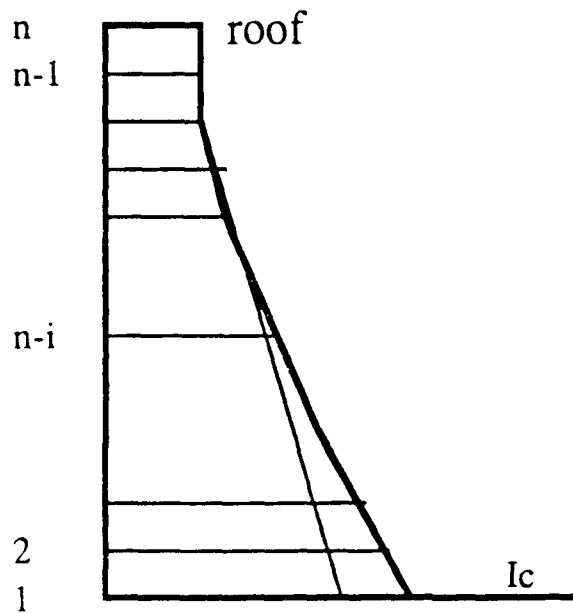


Fig. 2.5 Variation of column moment of inertias.
(Ovunc, 1988)

Fig. 2.4 represents the variation of the moment of inertias of the beams on the abscissa with respect to the level $n-i$ of the floors on the ordinate of the graph where n is the floor number of the roof. The minimum beam moment of inertia appears on the roof floor since the magnitudes of the gravitational loads on the roof are smaller than those of the lower floors. The sudden increase in the beam moment of inertias from the roof to the floor right below it is due to the increase in the rigidity of the floor right below the roof due to the columns above the floor level and the increase in the gravitational loads from roof to the lower floors. The axial forces in the columns increase from floor to floor in proportion to the tributary load area of the floor for that column. Fig. 2.5 shows the variation of column moment of inertias at different floor levels. The column moment of inertias may remain constant for the very few top floors because of the minimum size requirements. The column sizes in the lower floors increase due to the increase in the gravitational loads and the effect of the wind and earthquake. The variation of the column moment of inertias are different for the interior and exterior columns since the axial forces in interior columns are larger than those of the exterior columns.

The knowledge-based expert system is incorporated in the preprocessor which is in a modular form which can be interfaced with the general purpose structural softwares.

2.4.4 Bridge Design System: BDES

The design of highway bridges is an ill-structured problem

in which a large number of solutions are possible. Design decisions include selection of span type (continuous or simple), girder type (rolled beam, prestressed concrete, plate girder), clearance, material types, etc. The expert system, BDES (Bridge Design System) was developed to aid engineers in the decision, modeling and analysis process of highway bridges in North Carolina. It incorporates expert knowledge to aid the decision process as well as knowledge of serviceability and safety criteria of AASHTO and the state of North Carolina. The input to the system consists of graphical definition of bridge geometry, bridge function and the environment in which the bridge is to be constructed. Feasible alternatives to the problem are generated by the ES using approximations and assumptions. The designs are checked using the load factor approach and decisions on the best design to be adopted is based on least weight. The system is capable of designing bridge superstructures of short to medium, simple or continuous spans.

BDES was developed in PASCAL and uses a forward-chaining production rule approach since it facilitates the decision making process of design. Graphics are used for both the input process and output. The rule base is comprised of IF-THEN rules containing information of experts as well as AASHTO bridge specifications and local ordinances of the state. The factual knowledge includes AASHTO requirements, material properties and typical superstructure designs whereas the heuristic knowledge includes rules for superstructure selection, girder spacing determination and selection between simple or continuous span design. BDES is capable of selecting and proportioning short to

medium span bridge superstructures (Welch, 1986).

2.4.5 ES for the Optimum Design of Bridge Trusses: BTEXPERT

BTEXPERT (for Bridge Truss EXPERT) has been developed for optimum design of four types of bridge trusses, i.e. Pratt, Parker, parallel-chord K truss and curved-chord K truss for a span range of 100-500 ft. The system has been developed using the Expert System Development Environment (ESDE) and the Expert System Consultation Environment (ESCE). The two programs, ESCE and ESDE collectively referred to as the Expert System Environment (ESE) are a pair of complementary programs developed recently by the IBM Corporation. The first program is used to develop expert systems and in particular, knowledge bases whereas the second program provides the facilities for interactive execution of the ES. A graphics interface has been developed using the Graphical Data Display Manager (GDDM) (IBM, 1984). It was developed by interfacing an interactive truss optimization program developed in FORTRAN 77 to an ES environment developed in PASCAL/VS. Design constraints and the moving loads acting on the bridge are based on the American Association of State Highway and Transportation Officials (AASHTO) specifications (AASHTO, 1983). The structure and functions of various components of BTEXPERT are presented in Fig. 2.6.

2.4.5.1 Knowledge base

The knowledge base of BTEXPERT consists of the domain-specific knowledge and the control knowledge. The domain specific knowledge consists of rules and algorithmic procedures.

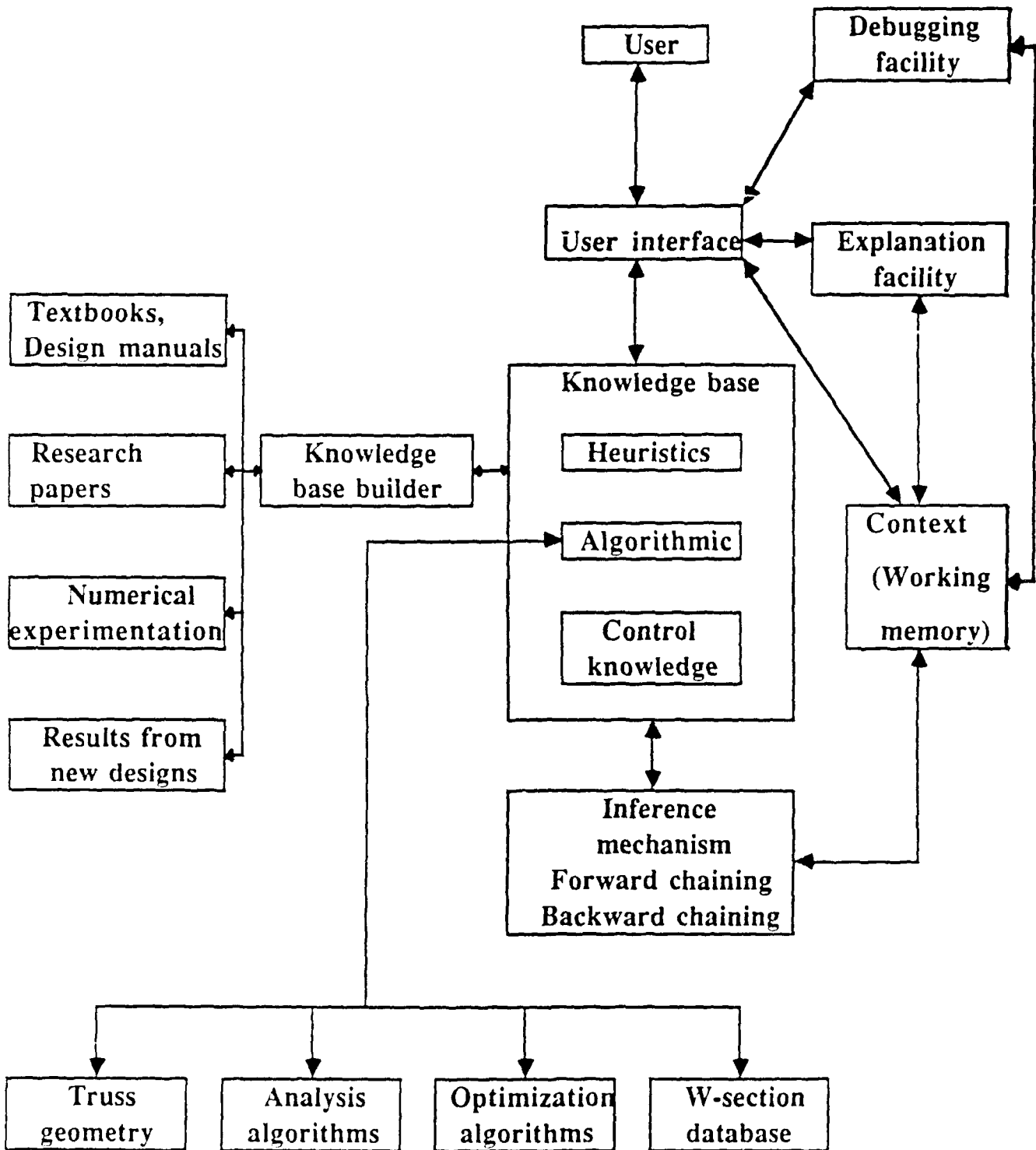


Fig. 2.6 Architecture of BTEXPERT (Adeli, 1988)

The control knowledge consists of control commands for solving a problem. The rules consist of an IF part and a THEN part or premise-action parts. Each rule represents an independent piece of knowledge. Knowledge representation consists of facts or parameters, rules and focus control blocks (FCBs). FCBs are the main building blocks in the ESE.

Rules are classified into the following three categories:

i) Inference rules: The default type of any rule is the inference rule. These rules are processed either by forward or backward chaining.

ii) Single fire monitors: Single fire monitors function independently without any reference to inference rules. The single fire monitor is processed once a parameter in the IF part of a rule gets a value.

iii) Multiple fire monitors: They are processed exactly like a single fire monitor except that they may be executed many times.

2.4.5.2 Inference mechanism

The ESE has both backward-chaining and forward-chaining mechanisms for problem solving. In backward-chaining, the facts for which values have to be determined are regarded as goals or subgoals. The goals and sub-goals of an FCB are selected by the knowledge base builder. The rules are processed one at a time until all the goals and sub-goals are found.

In forward-chaining inference mechanism, the applicable inference rules are collected in a rule list. Known facts in the

FCB are collected in a fact list. The expert system processes the rule list in a top-down manner. Based on the values of the facts in the fact list, the THEN part is executed for rules having their IF parts satisfied. The fact list is subsequently updated. The processing of rule list stops after one complete cycle through the applicable rule list, if single cycle strategy is used; in the case of multiple cycle strategy the rules are processed in the applicable rule list again and again until the applicable rule list is empty or no remaining rules can be fired.

2.4.5.3 User interface

User interface is provided in the form of visual edit screens and menus in which the user has to type in the values of the required parameters at appropriate fields. The user can have graphical displays of the truss configuration with joint or member numbering (Fig.2.7), influence line diagrams (ILD's) for various member axial forces and joint displacements and the design AASHTO live loads.

2.4.5.4 Explanation facility

The explanation facility helps the user to examine the reasoning process. The explanation consists of both the RULE text and RULE comments coded by the knowledge base builder. The explanation facility commands are:

- i) EXHIBIT: It displays the current value(s) of a specific parameter.
- ii) HOW: It displays an explanation of how the system determined a value for a parameter. Fig. 2.8 shows an

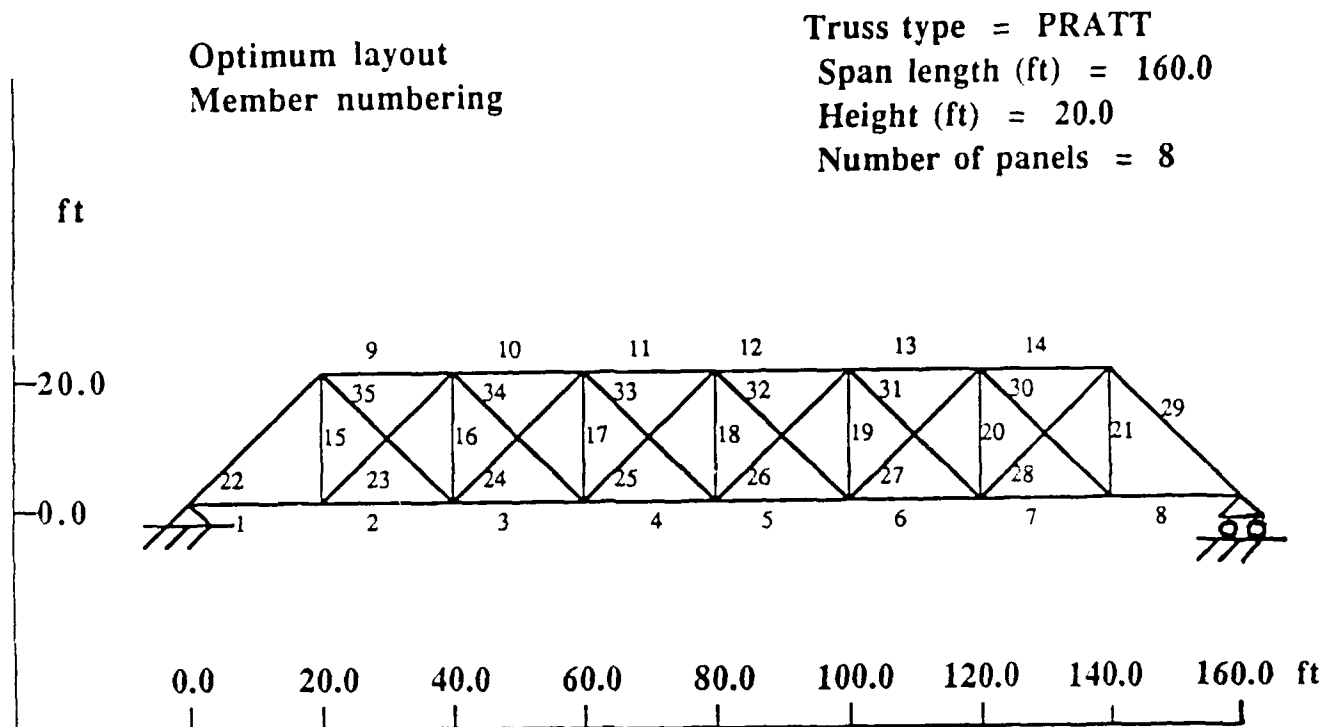


Fig. 2.7 A sample Pratt truss plotted by BTEXPERT
(Adeli, 1988)

Focus : FCB11 (1)

I assigned ^{How} value to allowable stress range in fatigue of FCB11 by

1. Rule RULE0039 which states that

If AASHTO LIVE LOAD = 'HS-15'
or AASHTO LIVE LOAD = 'HS-20'
Then Number of stress cycles = 500000
and Allowable stress range in fatigue = 24.

This rule is based on the AASHTO specification.

As a result of this rule
Allowable stress range in fatigue assigned = 24 (1)

To continue Consultation, Press ENTER

==>

PF1	Help
PF2	Review
PF3	End
PF4	What
PF5	Question
PF6	Unknown
PF7	Up
PF8	Down
PF9	Tab
PF10	How
PF11	Why
PF12	Command

Fig. 2.8 Example of explanation generated by BTEXPERT
in response to How it arrived at value of parameter
allowable_stress_range_in_fatigue (Adeli, 1988)

example of the explanation generated by BTEXPERT in response to the HOW command during a sample consultation.

- iii) WHY: It displays an explanation of why the system is asking a given question (Fig. 2.9).
- iv) WHAT: It displays more information about a given parameter (Fig. 2.10).

2.4.5.5 Debugging facility

The ESDE knowledge acquisition editors check each parameter, rule and FCB for syntax errors whenever they are typed into the system. However, the knowledge base builder should make sure that the knowledge base is consistent and complete, since inconsistencies either between individual rules or in various parts of a rule are not checked by ESE. The knowledge base builder can use the TRACE facility to debug errors detected in the results.

2.4.5.6 Knowledge acquisition

Domain knowledge is partly obtained from textbooks, design manuals, design specifications (e.g. AASHTO 1983), research papers and journal articles. Besides these sources, the gaps in the knowledge base are filled with optimum values of primary design parameters obtained from a detailed numerical machine experimentation in the problem domain. The machine experimentation can be conducted using a software IOTRUS developed in FORTRAN 77 for layout optimization of trusses: The optimum values for the height, number of panels, and initial

Focus : FCB2 (1)

Enter the 'Location of bridge'

(Choose one of the following:)

.... State-Road
.... Trunk-highway
.... Interstate

==> WHY LOCATION OF BRIDGE

To continue Consultation, press ENTER

PF1	Help
PF2	Review
PF3	End
PF4	What
PF5	Question
PF6	Unknown
PF7	Up
PF8	Down
PF9	Tab
PF10	How
PF11	Why
PF12	Command

Focus : FCB2 (1)

I am asking about WHY
to find AASHTO LIVE LOAD which I am trying to
determine.

These rules are used for this line of reasoning.

RULE RULE0012
If BRIDGE LOCATION is 'Trunk-highway'
Then AASHTO LIVE LOAD = 'HS-15'.

(Choose one of the following:)

.... State-Road
.... Interstate
.... Trunk-highway

PF1	Help
PF2	Review
PF3	End
PF4	What
PF5	Question
PF6	Unknown
PF7	Up
PF8	Down
PF9	Tab
PF10	How
PF11	Why
PF12	Command

Fig. 2.9 Example of explanation generated by BTEXPERT
in response to WHY it is asking value of string
parameter bridge_location (Adeli, 1988)

Focus : FCB3 (1)

....What....
If the bridge location is in an area of high corrosion,
the recommended choice of steel will be M244. It
should be noted that the relative costs of M183, M223,
M222, and M244 are 1.0, 1.15, 1.33, and 1.73,
respectively and the resistance to corrosion is poor,
not bad, good, and best, respectively (Ref.Heins 1979).

(Choose one of the following:)

x M183
___ M223
___ M222
___ M244

==>

PF1	Help
PF2	Review
PF3	End
PF4	What
PF5	Question
PF6	Unknown
PF7	Up
PF8	Down
PF9	Tab
PF10	How
PF11	Why
PF12	Command

Fig. 2.10 Example of WHAT explanation command for
providing additional information about parameter
steel_type (Adeli, 1988)

cross-sectional areas of truss members for various span lengths, AASHTO live loads and grades of steel are subsequently used in the knowledge base of BTEXPERT.

2.4.5.7 Knowledge base development

The rules and procedures used in BTEXPERT are classified into a number of FCBs (Fig. 2.11). Each FCB contains rules and procedures for a specific task. FCBs are used to classify all the rules and procedures required in an expert system according to their intended uses and sequences of application. For example, the rules for selecting the right type of truss for the span length specified by the user are:

If Span _ length \geq 100 and Span _ length \leq 200

Then Recommended _ Truss_ type is 'Pratt'

If Span_ length $>$ 300 and Span_ length \leq 380

Then Recommended_ Truss_ type is 'Parallel-chord K truss'

Sample rules used in FCB2 for selecting the right type of design live loads for the bridge under consideration are:

If Bridge_ location is 'State-Road' and Traffic is 'Light'

Then AASHTO_ live_ load is 'H-15'

If Bridge_ location is 'Interstate-Highway'

Then AASHTO_ live_ load is 'HS-20'

More rules on FCBs are given in Reference of Adeli and Balasubramanyam (1988).

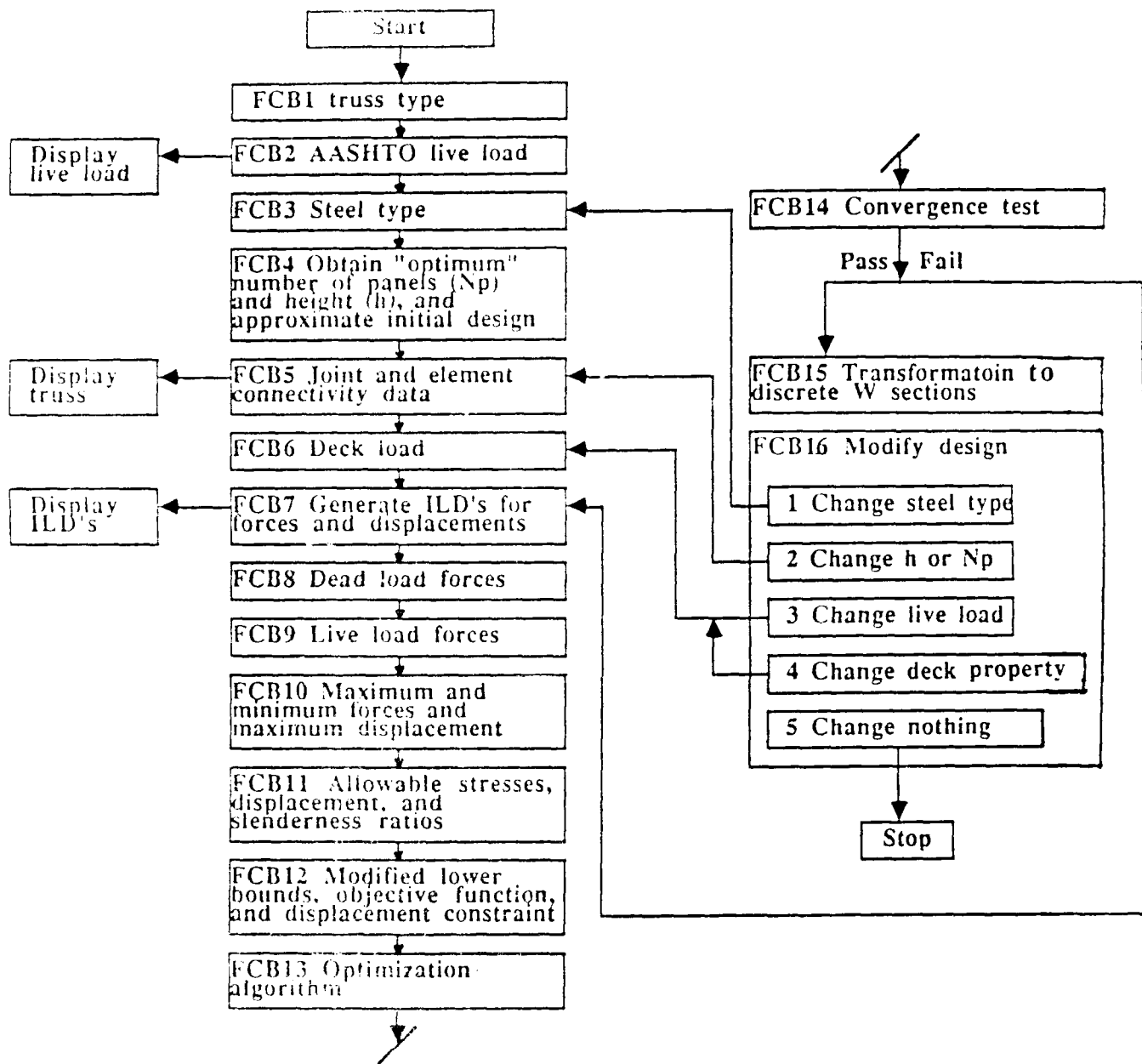


Fig. 2.11 Structure of FCB's (Adeli, 1988)

2.4.5.8 Mathematical optimization

The optimum design of a bridge truss consists of selecting the right combination of the cross-sectional areas of the truss members so as to satisfy all the design constraints and produce a least-weight truss. The allowable compressive stresses and the slenderness limitations provided by AASHTO specification involve the minimum radius of gyration of the cross-section. Using these optimum cross-sectional areas obtained from BTEXPERT and heuristic rules wide flange sections are selected for truss members from a database containing the W-Sections given in the AISC manual (AISC, 1980).

BTEXPERT is currently being extended to the optimum overall design of steel truss and plate girder bridges. Heuristic rules and procedures are being developed to improve the efficiency and accuracy of the optimization process, and for classification of constraints into inactive, partially active, active and violated constraints (Adeli, 1988).

2.4.6 Retaining Wall Design: RETWALL

2.4.6.1 Introduction

The RETWALL expert system was developed to provide expertise in the specific area of retaining wall structures. Its capabilities include consulting on the choice of retaining structures for a given set of user input and performing the preliminary design. The choices of retaining structures in RETWALL are brick, blockwork, gabion, gravity, reinforced

earth, reinforced concrete, and sheet pile.

The ES control lies in the existing expert system shell BUILD, developed by the Department of Architectural Science, University of Sydney, Australia; it uses a backward-chaining production rule system written in Quintus Prolog. The system employs graphics procedures, written in C, to display preliminary designs as well as displays to enhance the input process.

The system consists of two main modules, the high end and the low end. The primary function of the high end of the system which contains the rule base and inference mechanism is to select the particular retaining structure to be used. The lower end module consists of the routines that perform the preliminary design of the different retaining wall options. Presently, the lower end routine has the capability to design only blockwork walls. Design in the lower end routines is performed using design tables within the knowledge base of that module. The major limitation of the system is the lack of an evaluation of design alternatives (Gero, 1986; Hutchinson, 1985). Fig. 2.12 shows the overall concepts of the system.

2.4.6.2 The selection module

The selection module contains the higher level knowledge obtained from the literature review and interviews of experts which is concerned with the selection of the various types of earth retaining structure. Its rules are formulated in such a way as to control the firing of the lower level blockwork module, only when it has been determined that a blockwork wall is suitable for the given application. Currently if a type of

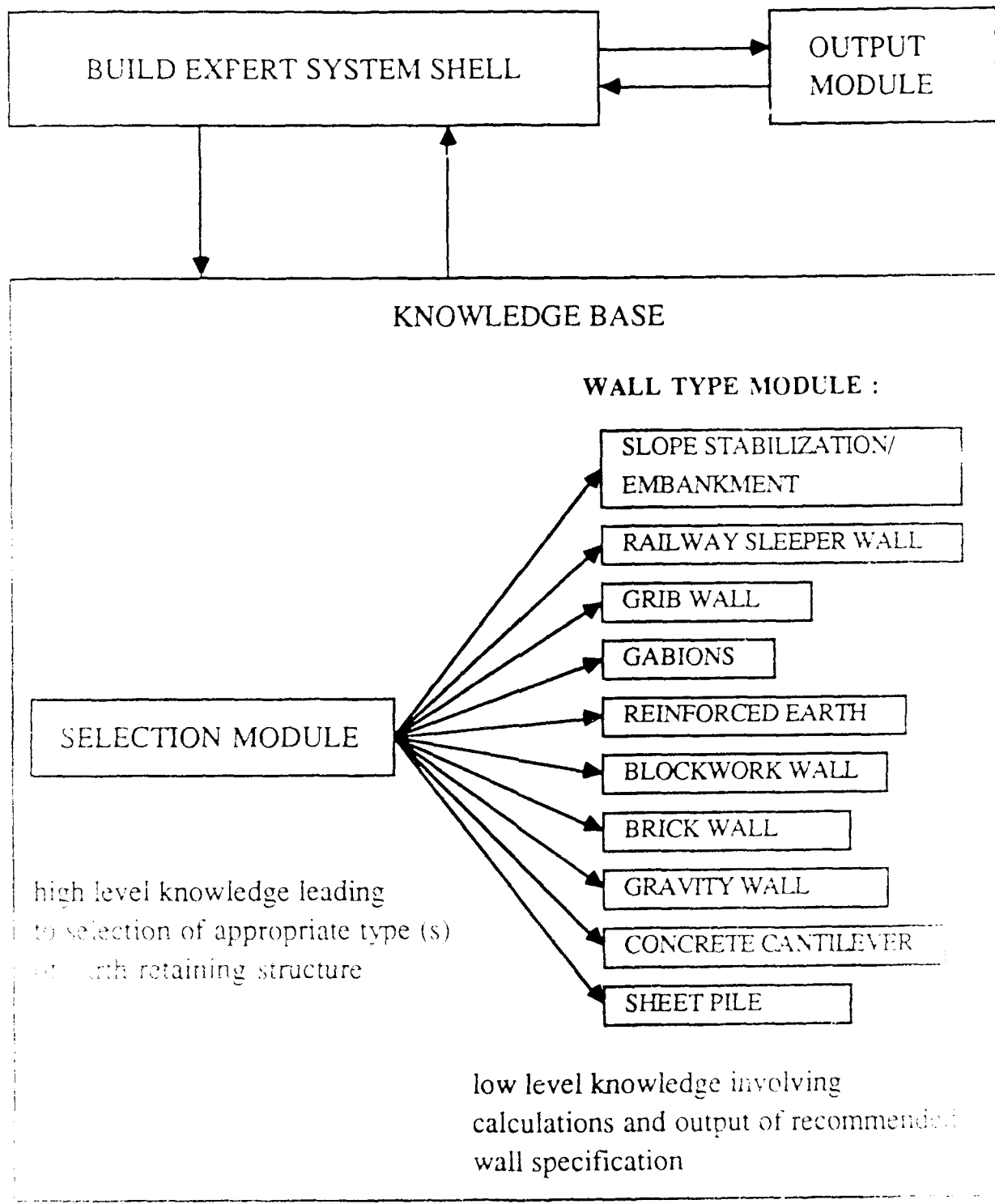


Fig. 2.12 Outline of an expert system for the selection of earth retaining structures (Hutchinson, 1985)

structure other than blockwork wall is determined as being suitable, a message is output that it is suitable and no further investigation of that type is conducted because the relevant lower level modules have not been written.

The rules in the selection module can be divided into a number of blocks which provide knowledge on:

- a. typical site conditions and geometric parameters of the site for the various applications where an earth retaining structure may be required;
- b. whether an earth retaining structure is required or not;
- c. the types of structure which should be investigated for a given applications;
- d. each of the individual types of structure considered and the factors which affect the selection of that type;
- e. various other considerations which affect wall selection such as terracing, surcharge loading and soil properties.

A schematic layout of all the knowledge block in the higher module is given in Fig. 2.13.

Knowledge on all the individual types of structures (brick wall, blockwork wall, crib wall, gabions, gravity wall, railway sleeper wall, reinforced earth, reinforced concrete wall and sheet piling) is included in the system although the amount of knowledge on each structure type reflects the amount of knowledge available from both the literature and the human experts. Hence there is more knowledge in the rules on reinforced earth, which is rapidly gaining popularity than in the rules on gravity walls, which are hardly used now.

The knowledge on typical site conditions is provided not

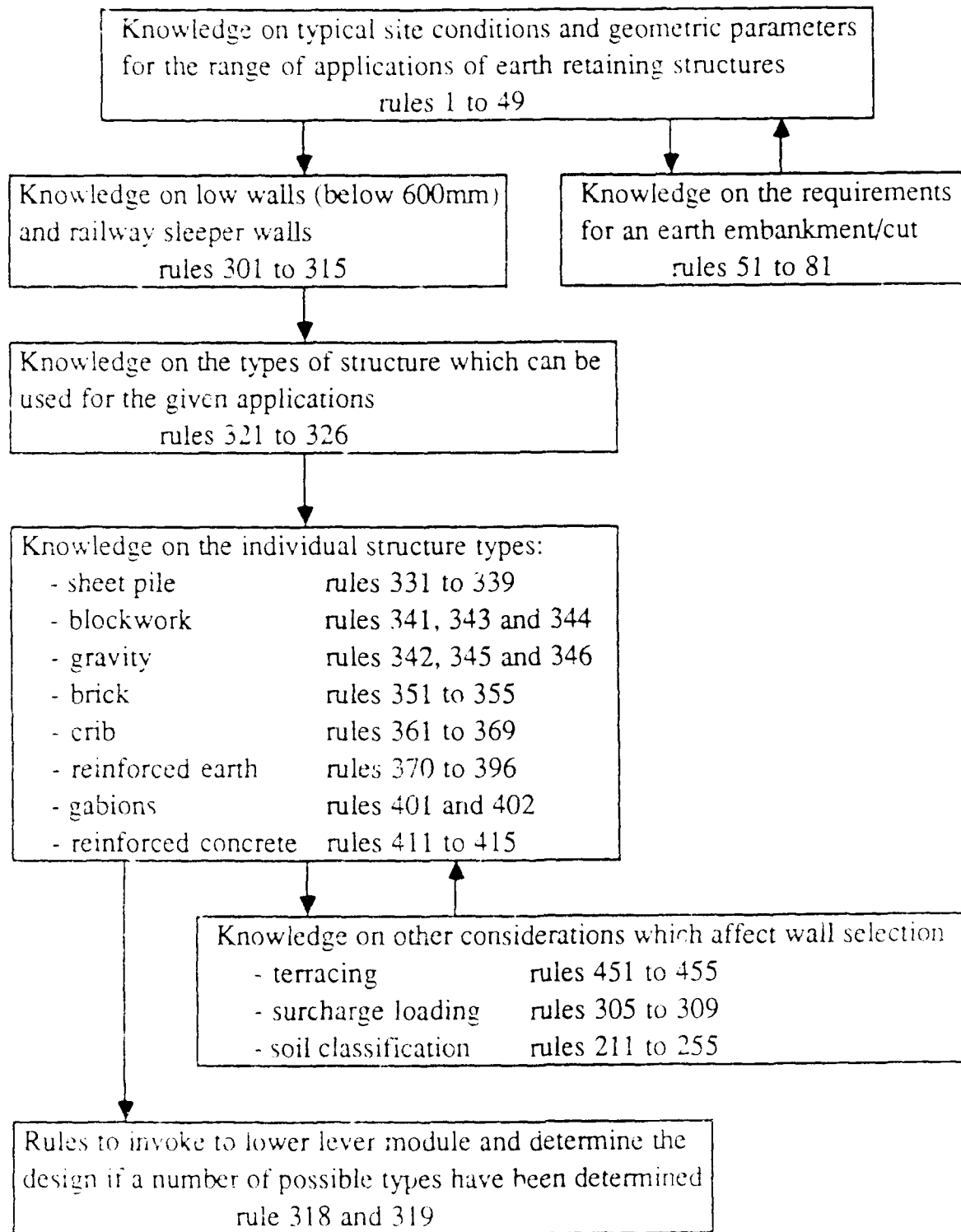


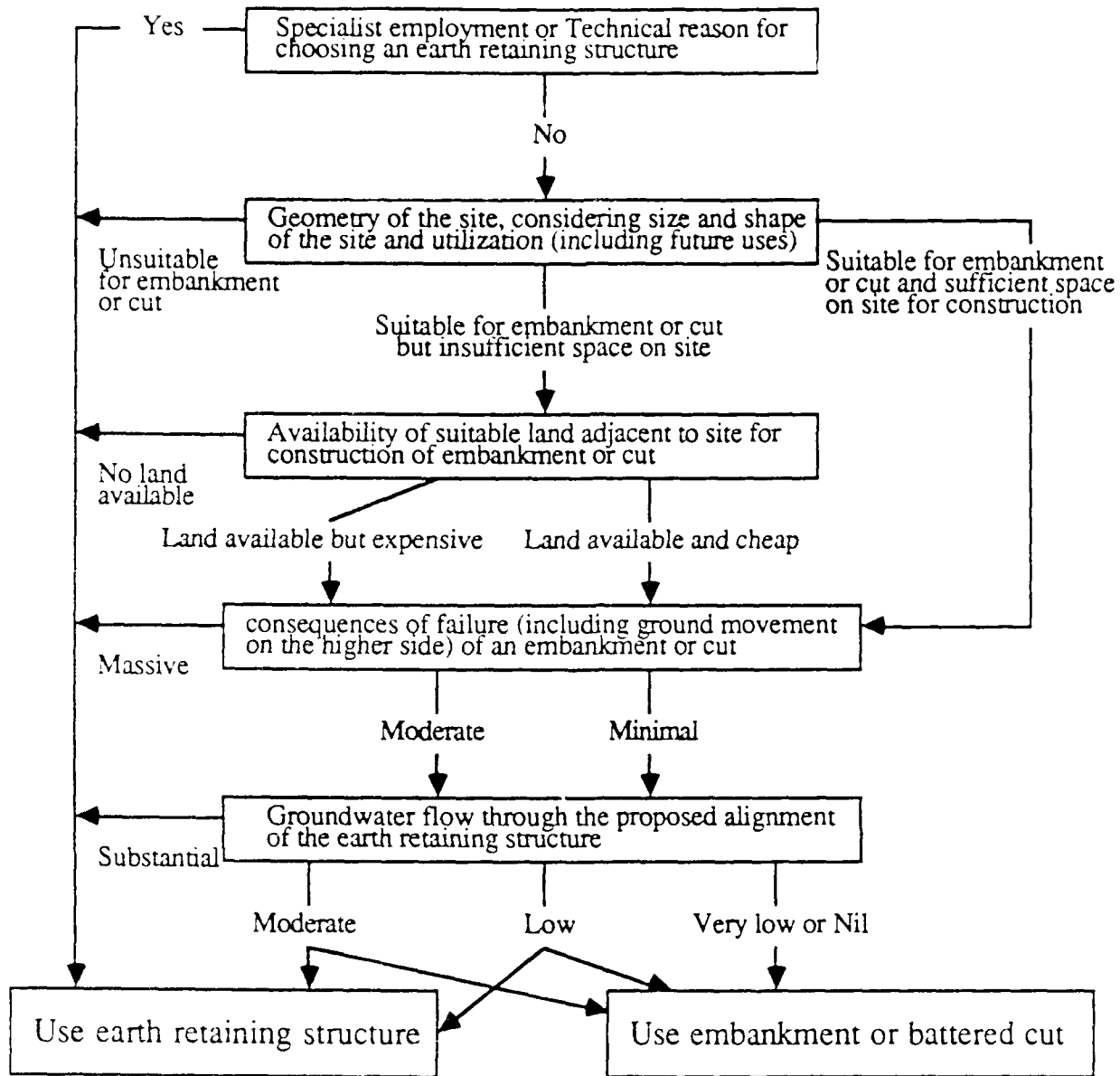
Fig. 2.13 Schematic layout of all the knowledge blocks in the higher level module (Hutchinson, 1985)

only in the rules of the system, but also is displayed by simple drawings produced by a C language procedure and called from within the expert system at the appropriate time. Three different drawings, depending on the application given by the user, can be produced by the procedure, each showing a number of possible alternative site conditions. The user is then asked to indicate the site case most applicable and provide the physical dimension data shown on the diagram.

Fig. 2.14 shows the flow chart for knowledge which determines whether or not an earth retaining structure is required. One of the main points to emerge from the interviews of experts was that an earth retaining structure should only be employed if an embankment or cut could not be used, or if there was some general reason for not wanting an embankment or cut. The knowledge block on whether an earth retaining structure is required attempts to establish if an embankment or cut could be constructed. If not, then it is determined by default that an earth retaining structure is required.

The knowledge on the types of structure suitable for a given wall application provides a higher level control on the search and determines the order in which the various wall types are considered, and which types are considered for every application. If the types considered by these rules prove to be infeasible, then the system will determine that the design is beyond its knowledge and stop execution of all the other possible but not feasible rules for evaluating a design.

The knowledge used in this block is formulated as rules such as:



NOTE: Where there are two possible courses indicated, the course chosen will depend on the results of earlier courses taken. Generally if an earlier decision involved a course on the left of the center of the page, further decisions on the left of the page will result in an earth retaining structure being employed.

Fig. 2.14 Flow chart for knowledge on the requirement for earth retaining structures (Hutchinson, 1985)

```

r325(if
  'earth retaining structure' is_ required and
  'type of application for wall' is_ A and
  'type of application for wall' is_ marine and
  evaluated(
    'Sheet pile is suitable for this application' and
    'Reinforced concrete wall is suitable for this application' and
    'Reinforced earth is suitable for this application') and
  not('Sheet pile is suitable for this application') and
  not('Reinforced concrete wall is suitable for this application') and
  not('Reinforced earth is suitable for this application')
  then
  'design of earth retaining structure' is_
    'beyond knowledge of this system')

```

The rules on the individual types of structure vary with the amount of knowledge obtained on the structures but generally include a range of heights applicable for the structure, the types of application for which the structure may be used, the aesthetic suitability of the structure and the availability of labor and materials for the structure. A typical example is:

```

r351(if
  'height of earth retaining structure (in mm)' is_less_or_equal_to 1500
  'Brick wall is aesthetically acceptable'and
  'Labor and materials are available for brick wall'
  then
  possible('type of earth retaining structure type is_ 'brick wall') and
  'Brick wall is suitable for this application').

```

The final block of rules provide knowledge on such things as terracing, surcharge loading, scale of the project and soil conditions which can then be used by the other rules. Some of these rules may not be required in the case of an experienced user who may give the answers they provide directly. Generally they are employed by the user asking 'how' to the relevant question in one of the selection rules:

Examples of the rules for terracing and related rules are:

```

r369(if
  'type of application for wall' is_ A and
  'type of application for wall' is_ domestic or commercial or
  industrial or road or railway and
  'height of earth retaining structure (in mm)' is_ greater_than
  15000 and
  not ('Reinforced earth is suitable for this application') and
  'slope ratio' is_ greater_ or _ equal_ to 1.83 and
  not ('The number of terraces required, considering aesthetics
  and space' is _ nil or 1 or 2), and
  'Crib wall is aesthetically acceptable' and
  'Labor and materials are available for crib wall'
  then
  possible ('type of earth retaining structure' is_ 'crib wall'
  and
  'crib wall is suitable for this application').

```

```

r453(if
  'slope ratio' is _ greater_or_ equal_ to 1.33 and
  'slope ratio' is _ less_ than 1.5
  then
  'maximum number of terraces allowed' is_ 2).

```

2.4.6.3 The Blockwork module

The blockwork module uses knowledge contained in design charts to produce preliminary designs for reinforced concrete masonry retaining walls from 1.0 to a maximum, depending on the backfill soil used, of 3.2 meters in height. A feature of this module is the output produced which not only gives wall parameters but also gives a scaled, dimensioned drawing showing reinforcing bar requirements.

The design charts used to produce the majority of the rules in this module give footing width, reinforcing bar requirements and wall thickness requirements for given wall height, footing type and backfill soil type. An example of one of the charts used is shown in Table 2.2:

Table 2.2

Example of the design charts used for blockwork walls

Back-Fill Type	Height (m)	Wall Type	Footing Width (mm)	Wall Dimensions	
				V-Bars	X-Bars
1	1.0	150	750	S16 @ 400	S16 @ 400
	1.4	200	900	S16 @ 400	S16 @ 400
	1.8	200	1050	S16 @ 400	S16 @ 400
	2.2	200	1300	S16 @ 400	S16 @ 400
	2.6	and	1550	S16 @ 400	S16 @ 400
	3.0	300	1750	S20 @ 400	S20 @ 400
	3.2	300	1850	S20 @ 200	S20 @ 400
3	1.0	150	1000	S16 @ 400	S16 @ 400
	1.4	200	1150	S16 @ 400	S16 @ 400
	1.8	200	1400	S16 @ 400	S16 @ 400
	2.2	200	1600	S16 @ 400	S16 @ 400
	2.6	and	1850	S20 @ 400	S20 @ 400
	3.0	300	2000	S24 @ 200	S24 @ 400
4	1.0	200	1150	S16 @ 400	S16 @ 400
	1.4	200	1450	S16 @ 400	S20 @ 400
	1.8	and	1750	S20 @ 400	S20 @ 400
	2.2	300	2300	S24 @ 200	S24 @ 400

Note: This chart applies for a base type 1 wall (See Fig. 2.15).

The blockwork module contains knowledge to:

- a. classify the backfill into soil types given by Terzaghi and Peck (1967);
- b. check that the allowable subgrade bearing pressure is not exceeded;
- c. select the most appropriate wall footing type for the given site conditions; and
- d. select the appropriate reinforced concrete masonry (blockwork) wall design parameters for the given conditions.

The effects of backfill soil in exerting pressure on the retaining wall are based on empirical charts for active soil

pressure given by Terzahgi and Peck for walls less than six metres in height. The gradings range from granular soil with little or no fines (backfill type 1) to medium or stiff clay deposited in chunks and protected from water penetration (backfill type 5). The lower the type, the more suitable it is for use as backfill and the smaller the section of wall required to retain it will be due to the lower active soil pressures produced.

The system uses either verbal descriptions of the backfill soil or the Unified Soil Classification of the soil to grade the backfill as type 1 to 5. For example, a 'backfill type' is 'sand or gravel containing some silt' or Unified Soil Classification GP-GM, GW-GM, SW-SM or SP-SM. To obtain the Unified Soil Classification, a module of about 40 rules (adapted from Burnham, et al, 1984) has been included which gives the classification based on the results of sieve analysis and laboratory tests.

Examples of the rules for backfill type are:

```
r.261(if
  'backfill to be used' is 'sand or gravel with little or no fines'
  then
  'backfill type' is_ 1).
```

```
r262(if
  'backfill to be used' is_ other and
  'soil classification of backfill' is X and
  'soil classification of backfill is_ 'GW' or 'GP' or 'SW' or 'SP'
  then
  'backfill type' is_ 1).
```

The first of these two rules is self explanatory. When this rule is 'fired' the user will be asked what the backfill to be used is and given the five options for the five soil types along

with the choice of answering 'other' and having the system determine the Unified Soil Classification. If the user answers 'other' the first rule fails and the second rule is invoked. The first line of this rule will succeed and the system will then attempt to determine the soil classification. The two lines on 'soil classification of backfill' are required for the same reason discussed for the 'type of application for wall' in the selection module section in order to ensure evaluation of this predicate by the BUILD expert system shell.

The allowable subgrade bearing pressure for the walls given by the design charts used must not be below 125 kPa. To ensure that this restriction is compiled with the rules dealing with footing type selection require that the subgrade allowable bearing pressure is first determined. If the user cannot provide a direct answer in kilo pascals, rules giving approximate allowable bearing pressures based on charts given by Carter (1983) will be invoked which match verbal descriptions of the subgrade soil with a minimum approximate bearing pressure.

These rules are self explanatory and take the form:

```
r105(if
    'soil beneath wall footing' is_ 'firm clay'
    then
    'subgrade allowable bearing pressure (kPa)' is_ 130).
```

A note is included with the display of the question on the 'soil beneath wall footing' to give some rules of thumb for estimating the bearing pressure and matching the verbal description.

Four different wall footing types, as shown in Fig. 2.15, are considered by the blockwork module. The most economical and

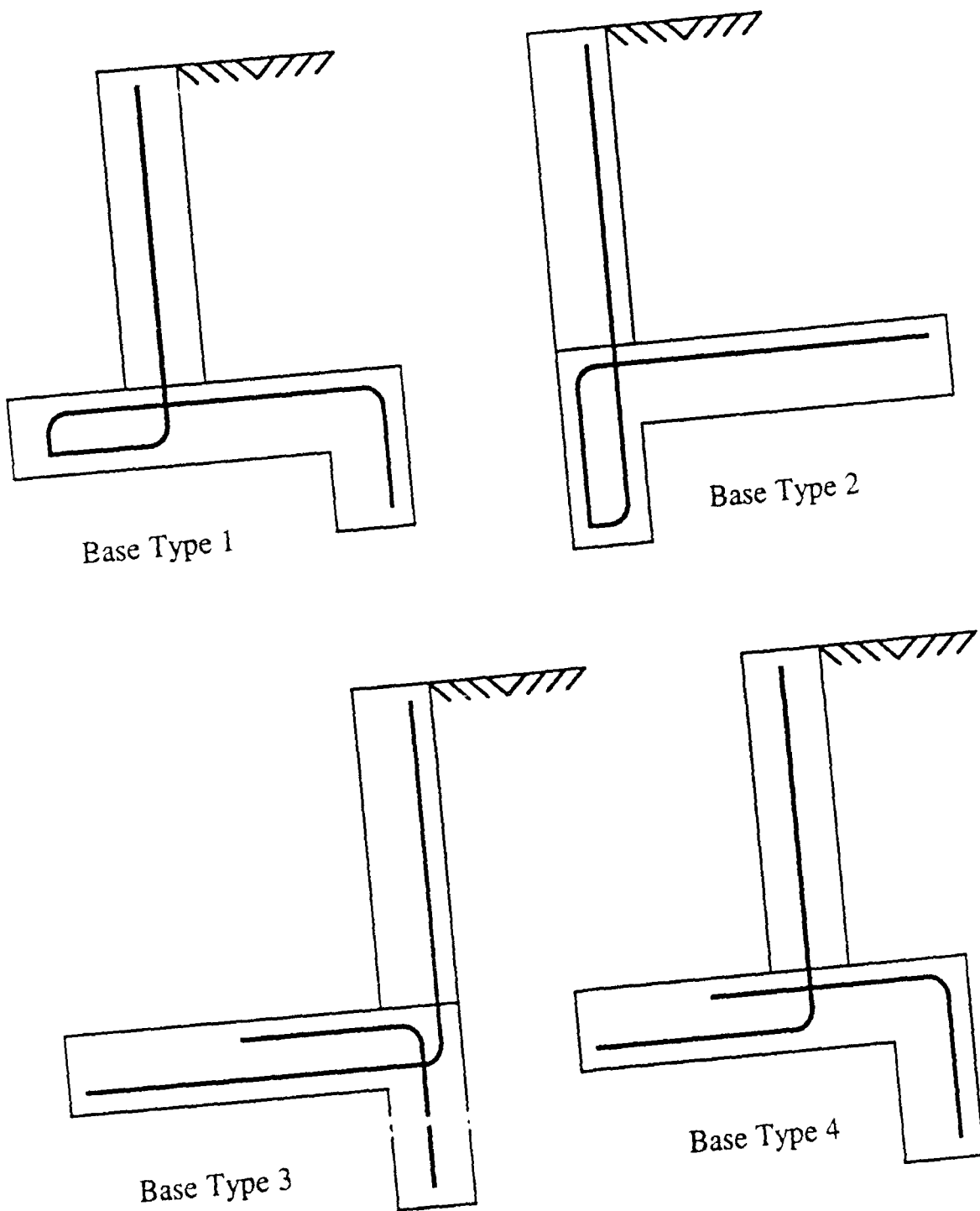


Fig. 2.15 The different wall footing (base) types used
(Hutchinson, 1985)

preferred one is type 1, while type 4 is preferred if only limited space is available for excavation and construction behind the face of the wall. Type 2 and 3 wall footings are applied in boundary wall situations where all the available space on a site is required and the wall footing cannot pass beneath some boundary or site restriction. The knowledge about site geometry and restrictions required by the rules which determine the wall footing type ('base type' is obtained by the selection module and is thus already in the facts base of the expert system. These rules take the form:

```
r271(if
  'subgrade allowable bearing pressure (kPa)' is_greater_than 125 and
  'site case most applicable (as shown in the diagram)' is_1 and
  'horizontal distance shown (d) (in mm)' is_greater_or_equal_to 500
  then
  'base type' is_1).
```

The 'subgrade allowable bearing pressure (kPa)' has already been discussed and these rules ensure that it is instantiated and checked before the design for a blockwork wall can be produced. The 'site case' and 'horizontal distance' refer to a drawing produced by the selection module which the user would already have answered questions on by the time this rule is 'fired'. Hence the user would only have to provide the subgrade allowable bearing pressure and the system would automatically deduce the 'base type'.

The final block of rules in the blockwork module form the major part of the module providing design parameters for the wall and invoking the C language graphics procedure to produce a

scaled, dimensioned drawing showing reinforcing bar requirements.

A typical example of these rules is:

```
r136(if
'height of earth retaining structure (in mm)' is_greater_than 2600 and
'height of earth retaining structure (in mm)' is_less_than_or_equal_to
3000 and
'base type' is_ 1 and
'backfill type' is_ 3
then
'blockwork wall type' is_ 300 and
'footing width' is_ 2000 and
'V-bars' is_ 'S24 at 200' and
'X-bars' is_ 'S24 at 400' and
draw).
```

The blockwork module is invoked by the selection module trying to prove that the 'blockwork wall type' is_ X. In other words, the selection module wants to find a value for the 'blockwork wall type' and that value will be instantiated by the first of the rules of the type shown above which succeeds. In proving the 'blockwork wall type', all of the other predicates in the consequent part of the above rule will also be instantiated and the six design parameters required to describe a blockwork wall will thus be known. These parameters are the height, base type, blockwork wall type, footing width, V-bar and X-bar requirements.

The final predicate in the above rule, 'draw', is recognised by the BUILD expert system shell and a Prolog rule in the shell is 'fired' to call the C language graphics procedure, converting the Prolog form of each of the design parameters into C arguments for the procedure.

Having succeeded in proving that the 'blockwork wall type' is_ X, the control of the expert system returns to the selection

module.

2.5 CONCLUDING REMARKS AND SUGGESTIONS FOR FURTHER WORK

ES applications to structural systems are research oriented rather than commercial oriented and concerned with the representation of design knowledge and design process. The example systems presented here are applications to the structural design of buildings, retaining wall design, bridge design, and design of frameworks. The potential use of expert systems for structural design depends on the complexity of the design problem. The ES approach will aid in the selection process of design problems in which the number of alternative solutions is small.

Knowledge-based expert systems (KBES) deal only with shallow knowledge, i.e., empirical associations. KBES environments could be more closely coupled with algorithmic programs which would contribute the deep, causal knowledge. KBES has the potential to be used not as standalone programs, but as intelligent pre- and post-processors for existing programs such as finite element analyzers. KBES framework would provide increasing user interface, explanation, and knowledge acquisition.

CHAPTER 3

DESIGN STANDARDS PROCESSING

3.1 INTRODUCTION

Design standards play an important role in the design of engineering systems. A design configuration must be checked against all applicable standards to ensure that it is acceptable. Previous research on design standards has been conducted to improve i) the representation and organization of standards, ii) the analysis of standards, and iii) use of standards. Standards are often modeled using three tools: decision tables, information networks and an organization system (Fenves, 1980; Harris and Wright, 1980; Rasdorf and Fenves, 1980).

3.2 GENERIC DESIGN STANDARDS PROCESSING

The processing of design standards in an ES environment was initially investigated by building two knowledge based expert systems: i) Query Monitor addresses the issue of semantics of data retrieval from engineering databases; and ii) Roofload Checker performs design conformance checking utilizing a standard.

3.2.1 Query Monitor

The AISC specification addresses a number of different types of stresses within a structural steel member including tension, shear, compression, bending, and bearing (American Institute of

Steel Construction, Inc., 1978). Depending upon constraints on shape, cross section, loading, etc. any one of a number of equations can be used to determine the allowable stress for a specific structural steel member. A database problem arises when the engineer issues an F_b data retrieval request. Query Monitor was identified as a framework to combine a database with a set of design specification constraints that govern the retrieval of data from engineering databases (Rasdorf and Wang, 1986). Query Monitor architecture was developed using the M.1 expert system building tool (Teknowledge, 1985). The knowledge representation consists of production rules and facts. The inference engine utilizes a goal-driven control strategy. As an example, Fig. 3.1 shows a decision table which is one of the tables from Provision 1.5.1.4 of the AISC specification. The first column of the table was recast in production rule format as follows:

If the axis about which a member is being bent is major and
 the connection of the web and flange is continuous and
 the width thickness ratio for exceptions is ok and
 the depth thickness ratio is ok and
 the laterally unsupported length is ok
Then the allowable bending stress = $0.66 F_y$

A complete program listing as well as several sample execution logs are given in the Query Monitor User's Guide (Wang and Rasdorf, 1985).

3.2.2 Roofload Checker

The Roofload Checker was developed to study the performance of a production system based on a data-driven control strategy to check designs. It consists of two subprograms, Roof Checker and Roof Reporter. The engineer describes the roof design using datum-value pairs, which are stored in the context. Roof Checker then checks the roof design by matching the input against the production rules converted from the BOCA building code (Building Officials and Code Administrators International, Inc. 1984) to determine whether or not the design conforms to the standards it incorporates. However, it does not provide any feedback after its operation. The result after design checking by Roof Checker is stored in an external file. After Roof Reporter is invoked, the data from the file are then reformatted and displayed on the monitor screen.

Roof Checker and Roof Reporter were written in the OPS5 knowledge engineering language and the knowledge representation scheme consists of production rules. Either the data-driven or the goal-driven control strategy can be implemented in OPS5. As an example, the requirements of Table 910 of the code are directly cast in production rule format in the Roof Checker as follows:

```
If      the shape of the roof is pitched and
         4 ≤ the slope of the roof < 12 in/ft and
         0 ≤ the tributary loaded area for structural member
                                         < 200 ft2 and
         the designed roof load ≥ 16 psf

Then    the roof is OK
```

More details of the Roof Checker as well as the Roof Reporter are reported by Wang (1986).

3.2.3 Generic Standards Processing In a Knowledge-based Expert System Environment: SPIKE

The architecture of SPIKE consists of two functions: i) performing design conformance checking, and ii) determining allowable value ranges for undetermined design datums. Fig. 3.2 shows the typical components of SPIKE architecture. It uses provisional and organizational facts for its knowledge base. Because the knowledge base is implemented in the factual format, it is called the Standards Factbase of SPIKE. As in a typical ES, the standards factbase is used by an inference engine as it manipulates the context. The set of production rules encoded specifically for processing the generic standards factbase is referred to as a Standards Processor. The Transformer which is the knowledge acquisition facility in SPIKE translates the knowledge from the decision table format of a standard to the internal representation of the factbase. The Context is the short term memory containing design-specific information entered by the interfaces (interactive and program) or generated by the inference engine. The Interactive Interface provides a command language to enable the designer to describe a design, or to query the system to obtain information about the design or the governing standards. The Program Interface provides a similar functionality for CAD programs.

The SPIKE has been implemented as a research prototype using

Axis = major	Y	Y	Y	Y	Y	N
ConOK	Y	N	Y	Y	Y	Y
Bf_Tf_RatioOK	Y	Y	N	Y	Y	Y
D_T_RatioOK	Y	Y	Y	N	Y	Y
LcOK	Y	Y	Y	Y	N	I

$F_b = 0.66 \cdot F_y$	X					X
Except_Ten		X	X	X	X	
Except_Comp		X	X	X	X	

Fig. 3.1 Decision table for AISC specification provision 1.5.1.4
(Rasdorf and Wang, 1988)

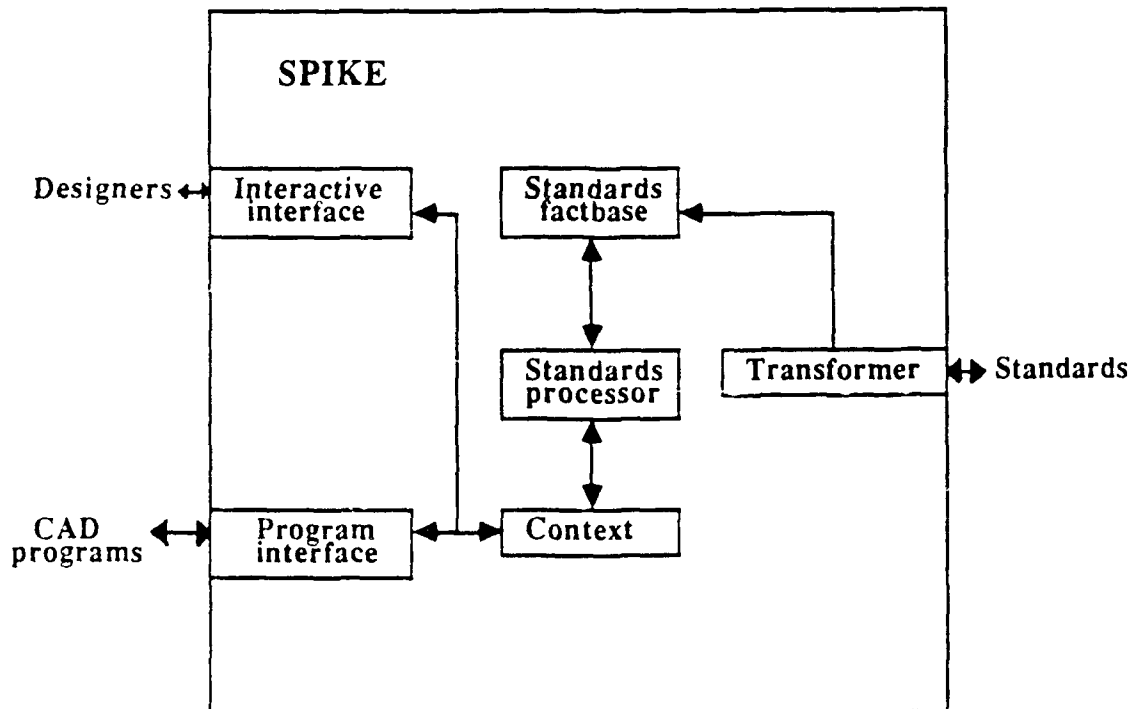


Fig. 3.2 Architecture of SPIKE standards processing system
(Rasdorf and Wang, 1988)

OP55 whose operation is governed by pattern matching. The user enters, as input, sets of datum-value pairs describing the design under review. When the user indicates there is no additional input, SPIKE performs data generation, analysis, design conformance checking and the results are displayed on the screen. The user can then elect to quit or continue, revising the design by entering updated datums or new datum-value pairs and the cycle can be repeated as many times as necessary until a design is derived that completely conforms to the governing standard. The detailed implementation of SPIKE is described by Rasdorf and Wang (1986, 1988).

3.3 AASHTO BRIDGE RATING SYSTEM

An ES that carries out the rating of simply supported highway bridges with reinforced concrete decks and prestressed concrete I-beams is under development at Lehigh University. Effects of vehicular or overloaded vehicular traffic are taken into account. The expert knowledge stored in the database includes AASHTO bridge rating provisions, extensive data on overload of prestressed concrete highway bridges and heuristics essential to decision making strategies. The database is structured in two-dimensional spreadsheet format. The basic approach involves a forward-chaining search of the database for a bridge rating (i.e., AASHTO, past case histories, Grillage Analogy). At the exhaustion of the database, if rating quality is unsatisfactory the finite element algorithms are triggered and the bridge is treated as a new design problem. The system is operational type and written in structured FORTRAN (Kostem, 1986).

3.4 AUSTRALIAN MODEL UNIFORM BUILDING CODE: AMUBC

Design codes contain a large amount of causal and experiential knowledge. Typically, the amount of information in a code is large and represents the best effort on the part of the writers to organize it in a clear fashion. Even with this effort, codes tend to be unstructured and complex and difficult to interpret by many engineers. AISC, ACI, BOCA, etc. are examples of codes that could appear unstructured and are hard to follow. Ability to use a code to its full potential relies on the experience and expertise of the individual using it. The primary motivation for the development of design codes as expert systems is to produce computer systems that will aid not only the engineer and designer but also the local authorities in administration of these codes.

The prototype ES representing only a part of the entire AMUBC is run on an expert system shell written in Prolog 1 on an 8088/8086 microcomputer in a MS-DOS environment which needs a minimum of 128K bytes. A production system approach has been used for knowledge-base development since this rule-based approach facilitates the modeling of the information as it is typically presented in building codes. The system is capable of both forward-and backward-chaining through the rule base. The domain independent metaknowledge which is an important feature of the ES provides the user with the capability of determining the scope of the information relevant to the problem and nature of the knowledge in the domain of the system. One disadvantage of the ES is its lack of interrupt capabilities for explanation

facilities. Work is now in progress in determining better representations and expansion to include more of the AMUBC (Rosemann 1985, Rosemann, 1986).

3.5 KNOWLEDGE-BASED STANDARDS PROCESSOR: SPEX

SPEX is a knowledge-based structural component design system which basically selects requirements, generates constraints, and then satisfies those constraints to find a set of values for the properties of the component. The system is knowledge-based because designer expertise is used to select behavior limitations for detailed design in which the properties of all structural components are determined subject to the satisfaction of structural integrity and functionality constraints.

It is implemented as a blackboard system because the blackboard architecture facilitates the integration of knowledge-based and algorithmic subprocesses in the component design process. The architecture of SPEX is shown in Fig. 3.3. Task specification, design focus hypothesis, standard requirements, constraints and solution form the five levels of abstraction in the blackboard. The knowledge base in SPEX is divided into the design process modules and design knowledge. The design focus module generates a design focus hypothesis using a set of expert rules. The requirement retrieval module generates i) a list of requirements that must be checked and ii) a list of requirements that are translations of the behavior limitations within the design requirements. The constraint set generation module generates a set of constraints from the design requirements. The

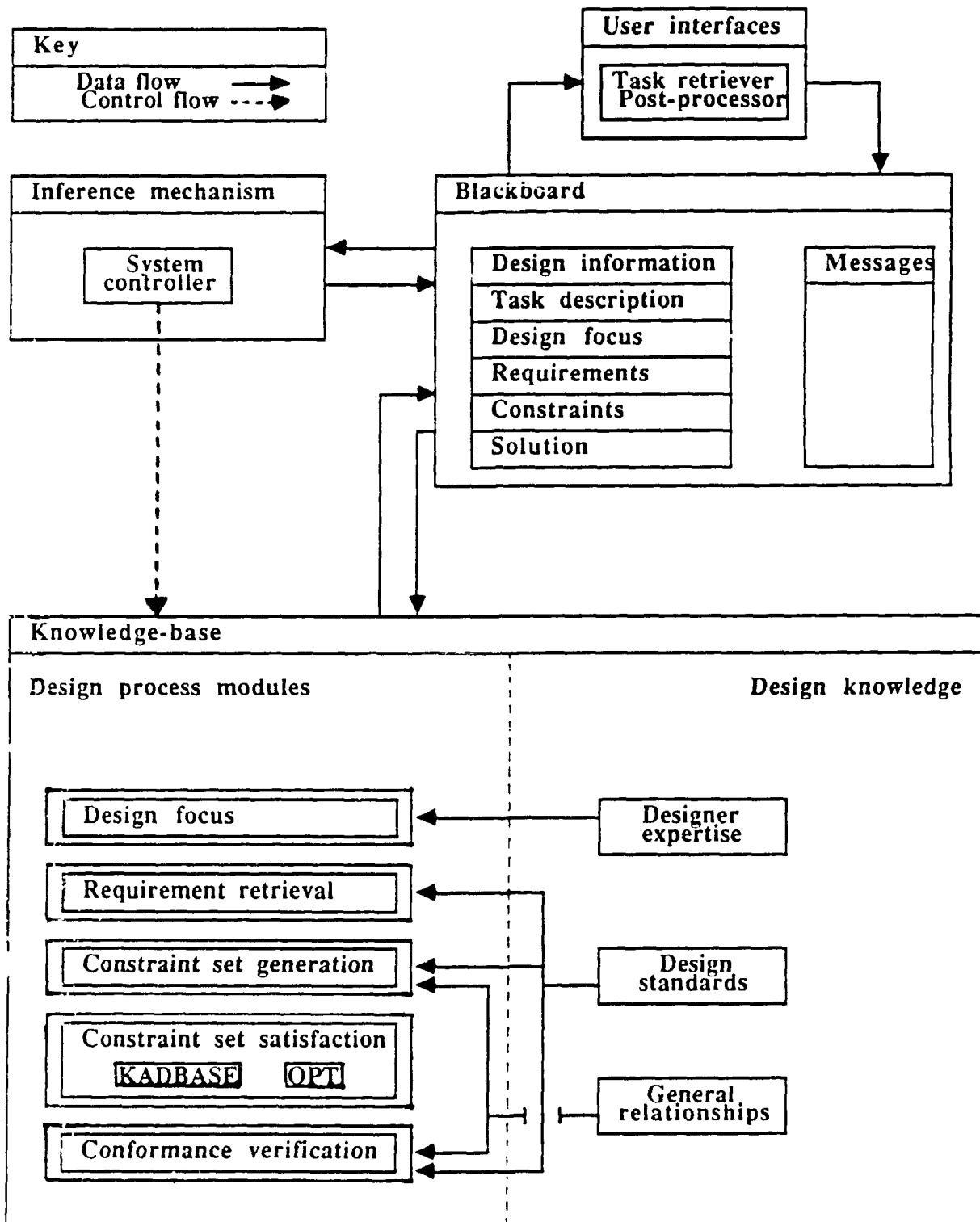


Fig. 3.3 The functional modules of SPEX (Garrett, 1986)

constraint set satisfaction module determines the optimal component design within the solution space defined by the constraint set using either a nonlinear constraint satisfaction routine, OPT (Biegler and Cuthrell, 1985) or a knowledge-based database interface, KADBASE (Howard, 1986). The conformance verification module checks the resulting component design not only for conformance with design requirements, but for conformance with all applicable standard requirements. If violated requirements are found, backtracking situation would occur and design focus module should be invoked to alter its hypothesis such that the violated requirements become design requirements.

The design knowledge in the knowledge base consists of i) designer expertise for the generation, completion and modification of design focus hypotheses, ii) design standards, and iii) general relationships including structural, material and geometric definitions of data items in the design standard. The design knowledge sources are used by various design process modules.

The task specification user interface assists the user in defining the component type, the governing standard, the design method, the design stage, etc. whereas the postprocessor provides the user with commands for displaying information regarding task description, component properties, the constraint set and requirements that were checked, the design requirements, etc. The modules in the knowledge base are invoked by the system controller based on the current design state which is represented by messages on the message blackboard and design

information on the design information blackboard. A set of control rules is used by the system controller to specify modules to be invoked based on information present on the message and design information blackboards (Garrett, 1986).

3.6 A PC BASED ES FOR AUTOMATED REINFORCED CONCRETE DESIGN CHECKING (Saouma, Jones, and Doshi, 1987)

3.6.1 Introduction

This ES checks reinforced concrete designs based on the ACI 318-83. Several software tools including M.I expert system shell (version 2.1 cos), Microsoft Fortran (version 3.30), Microsoft C (version 3.00), and spreadsheet Lotus/123 (Release 2) were used in the development of the system. The overall system architecture is shown in Fig. 3.4.

The system consists of two distribution disks; a "user" disk containing only those files necessary for system operation and a "maintenance" disk containing additional files used in system implementation.

3.6.2 The Spreadsheet

The developed spreadsheet (ACI. WK1) is LOTUS/123 compatible and contains the three columns of interest to the user (variable description column A, data entry column B and legal value column C) and a small data writing macro in protected cells (column I-O, rows 9-14). This macro is used to send a spreadsheet data to a datafile for subsequent input to the expert system.

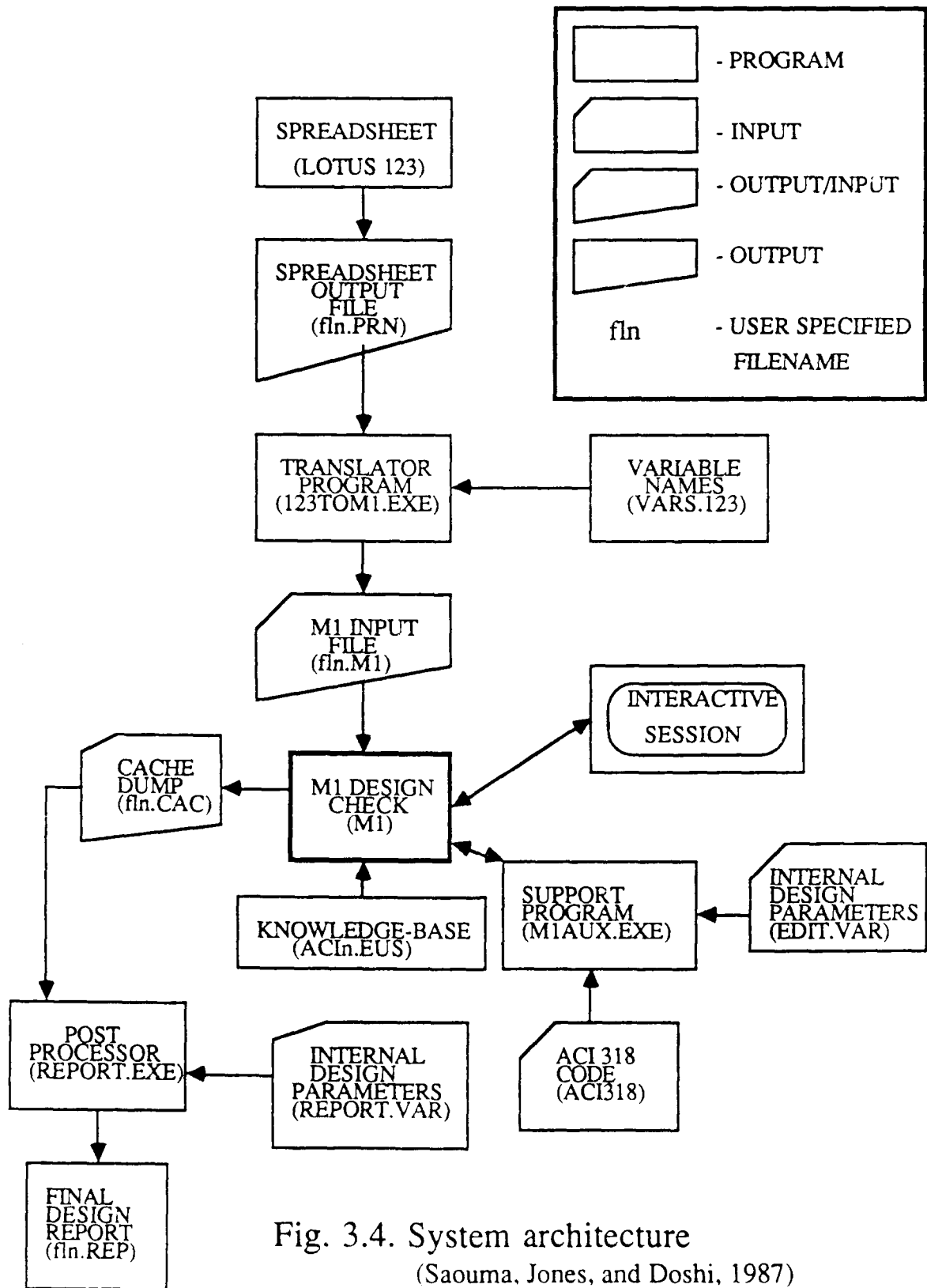


Fig. 3.4. System architecture
(Saouma, Jones, and Doshi, 1987)

3.6.3 Spread Sheet Conversion to M.1 Data

An auxiliary file containing internal variable names is used to take output from the spreadsheet and generate input file for M.1 in a format compatible with the expert system input. The sequence for conversion is as follows:

- i) Fetch a value from the 123 output file
- ii) Fetch the variable name used internally by M.1 for this value.
- iii) Write the variable and value using M.1 cache format.

The match between a value in the 123 output file and the internal name used by M.1 is made by using a variable name file. This file contains the internal M.1 variable names in the order in which the values are written from the spreadsheet.

3.6.4 Expert System

3.6.4.1 The ACI knowledge base

The system consists of ten KB files which contains rules pertaining to user interface operation, top level duties, knowledge base representing Chapters 8 through 12 of the ACI Code (each file has knowledge for individual chapter, and one file contains knowledge for all the chapters and user interface rules.), knowledge base for beam "quick-check", and knowledge base for column "quick-check". The design checker will use a different combination of these files based upon the checking task.

3.6.4.2 Interface configuration

M.1 provides a menu-driven interface for user interaction,

providing display of system output and manu-driven input. User can use several ALT keys to issue common commands.

3.6.4.3 External Function

External functions are needed to perform various duties such as numerical calculation, display of ACI provision text, cache editing, which cannot be easily done by rules.

3.6.4.3.1 External code

The source code for the externals is contained in three files. STUFF.C contains the external functions for sine cosine, cube root, and display of provision text. Data are taken from KB rule via an import statement, calculated in C library function and returned to the knowledge base using export statement Provision display is obtained by imposing the desired specification number (ex. 10-3-1), and searching a file containing the ACI code text CACI 318) for the provision label. Once found, all text up to the first occurrence of "#" (used as a delimiter) is copied to the display. Should the specified label not be found, a message stating such is displayed and control is returned to the knowledge base.

This file invokes the column strength external and the editor. The source code for the editor is in EDITOR. FOR whereas both the files are written in FORTRAN. If current menu, screen is saved and cleared by the M.1 function "savescreen", then control is passed to the editor subroutine. Here, the current cache is read, variables specified in EDIT.VAR are extracted and displayed using the corresponding "user name". The editor now

enters an interactive loop allowing the user to change the rules of the given variables. Upon exit, an updated cache is written by the editor for subsequent input by the KB on external exit. The required input data are imported from the KB, necessary computations are made, and the results stored in separate file. Control then returns to the C code which reads these values and exports them to the KB and returns execution to the expert system.

3.6.4.3.2 Rule partner

Some rules are necessary in knowledge base to invoke the external code via an "external" statement. In the case of sine cosine, and cube root, the rule is one of the ACI provisions. The remaining externals are invoked through rules associated with the M.1 command mapped into the appropriate ALT key sequence.

3.6.5 Report Generator

The function of the report generator is to extract essential information from a construction cache dump and arrange this information in an aesthetic manner in a report file.

3.7 CONCLUDING REMARKS AND SUGGESTIONS FOR FURTHER WORK

Expert system applications to generic design standards processing provide an opportunity to represent and make use of requirements and standards in a concise and unambiguous manner and provide allowable value ranges for undetermined data. The use of codes forms a mandatory requirement in almost all areas of

structural design and hence this is a particularly important application area. Emphasis of earlier work in design standards was to improve i) the representation and organization of standards, ii) the analysis of standards and iii) the use of standards. The synthesis of standards is a promising new area for further work.

CHAPTER 4

CONSTRUCTION ENGINEERING AND MANAGEMENT

4.1 INTRODUCTION

Construction engineering and management can be divided into three major areas: 1) engineering of temporary facilities for construction, ii) management of the construction process, and iii) rehabilitation, repair and maintenance of facilities. The construction engineering and management involve all the planning and design decisions related to the equipment and physical facilities (e.g. cofferdams, access roads, etc.) involved in the construction process. Expert system techniques might profitably be applied to a) design of construction methods, b) manufacturing and placing concrete, c) excavations for construction, d) constructibility evaluation, e) site layout, and f) surveying associated with the precise location of permanent facilities.

The construction management consists of administrative, legal, and financial elements of the construction process. Project planning, scheduling and control are now widely supported by the use of network-based project scheduling techniques for analysis and by database management systems for reporting. Decisions in contract management include selection of overall contracting strategy and contract clauses, identification of project financing, selection of prospective contractors or designers, evaluation of progress payments and potential claims and project organization design. The construction company

management comprises of marketing strategy decisions, personnel management decisions, company organization design, financial planning, construction equipment policy decisions, and safety management. A number of problems in the construction engineering, which has an ill-defined and ill-structured environment are not amenable to satisfactory solution by procedural, algorithmic computer techniques. The complex nature of the problem requires the knowledge and experience of a recognized expert and several expert systems have been developed to capture this expertise.

The possible range of expert systems in construction include the following (Rehak and Fenves, 1984):

- .interpretation of signals and data from exploratory devices and sensors,
- .monitoring performance of equipment and processes,
- .diagnosis of equipment failures and process deficiencies,
- .recommendations for corrective actions in case of malfunctions and shortages,
- .planning of construction activities and equipment functions,
- .design of construction schedules.

Typical representative expert system applications in construction are described in the following subsections:

4.2 EXPERT SYSTEMS IN CONSTRUCTION ENGINEERING

4.2.1 Soil Exploration Consultant: SOILCON

The condition of the soil below the surface of the ground is one of the biggest uncertainties in construction projects. The correct assessment of subsurface risk at an early stage of the project can contribute significantly to the overall success of the construction effort. SOILCON eliminates to some extent the uncertainty involved in subsurface exploration by evaluating known conditions of the site and recommending appropriate methods to continue exploration, if required. The system is designed to be used by the user in order to include subsurface considerations into contract design, thereby reducing contractor contingencies. The output of SILCON includes a list of recommended investigation procedures ranked by certainty, display of their descriptions and cost estimates for the methods. The system uses backward chaining from the knowledge base of rules encoded in a PROLOG-like syntax and runs on IBM PC class computers. It is a developmental expert system which does not have the capability to handle quantitative information (Ashley and Wharry, 1985).

4.2.2 Layout of Temporary Construction Facilities: SIGHTPLAN

(Tommelein, Levitt and Hayes - Roth, 1987)

4.2.2.1 Introduction

Selection of construction methods and equipment and the design of the site layout are given attention at the bidding stage and at the startup of construction of the project, but continuous advance planning is seldom carried out. Inappropriate site layout can lead to considerable lost time in the form of

excessive travel time of workers and equipment and inefficiencies. The quality of temporary facilities layout on site has a significant effect on the efficiency, safety, productivity and cost of construction. The expert system, SIGHTPLAN is designed to assist project managers in their complex task of designing site layouts and updating the plan continually as project time progresses.

During construction of a project, a number of different temporary facilities are located and removed from the site. Determination of their location is a spatial arrangement problem dealing with positioning objects under constraints. A blackboard expert system shell, BB1 has been chosen to apply varying problem solving strategies and construct the layout incrementally (Fig. 4.1). It is particularly well suited for reasoning about alternative objects, simultaneously searching for multiple hypothetical solutions, and dealing with time. ACCORD is a specialization language which provides a vocabulary to express relationships in spatial arrangements. Objects are assigned roles based on the constraints and with the site.

SIGHTPLAN contains all construction management domain knowledge necessary to design site layouts. Objects are described by their type, dimensions, geometry, mobility, possible zoning requirement and duration on site (Fig. 4.2). Objects inherit properties from the class to which they belong (Fig. 4.3). The planning mechanism of BB1 allows SIGHTPLAN to propose two or three alternative possibilities to the user, which satisfy all or most of the given constraints. The evaluation of a

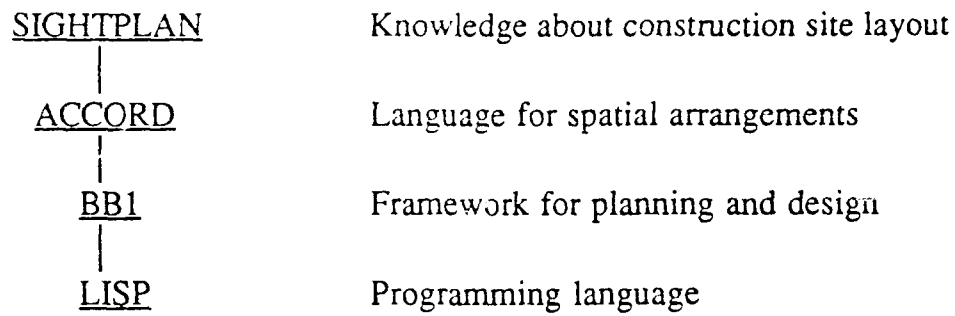


Fig. 4.1. SIGHTPLAN software environment
(Tommelein, Levitt, Hayes-Roth, 1987)

WHELL LOADER	
Type:	930
Length:	5.9 m
Bucket rated capacity:	1 m3
Geometry:	fixed
Mobility:	mobile
Zoning:	any
Duration from:	10Jan88
Duration to:	25Jun88

Fig. 4.2. Object description
(Tommelein, Levitt, Hayes-Roth, 1987)

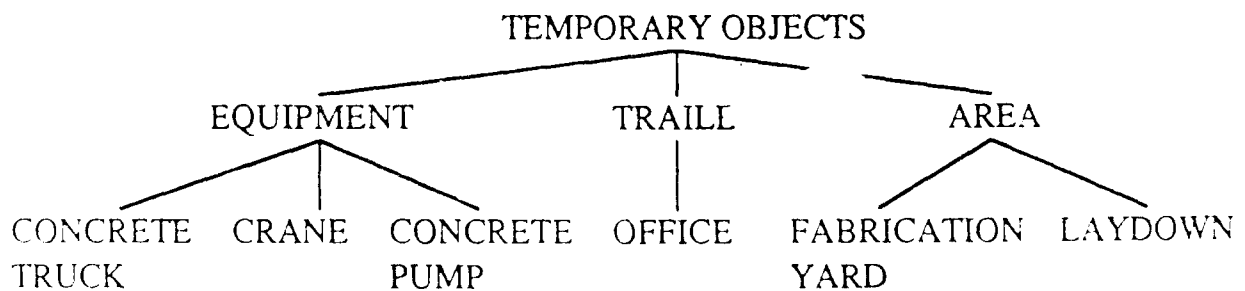


Fig. 4.3. Objects in the site layout domain
(Tommelein, Levitt, Hayes-Roth, 1987)

particular design can then be rated by means of a checklist of shortcomings.

4.2.2.2 Example of SIGHTPLAN's design actions

A reinforced concrete wall has to be constructed in an excavated area 7 meters deep. A crane in the pit lifts reinforcement bars and formwork into place. The current activity is to perform concrete placement by means of crane and bucket. Two subcontractors (subs) are to be involved with the concrete placement: one places reinforcement bars (rebar), the second one sets the formwork. Both subcontractors want to be in the secondary zone (i.e., the zone surrounding the excavation), and within crane reach. Two strategies could apply: i) place rebar first, then place the formwork around it; or ii) alternate placing one and then the other. Two state families are generated: one for the rebar sub location, and one for the formwork sub.

SIGHTPLAN would reason in the following way:

Strategy i):

GOAL: locate sub1 on site

DATA: the area required by sub1 is 900 square feet (100 m²)

CONSTRAINT: the area for sub1 needs to be within crane reach

ACTION: locate the crane on site; find area of crane reach

ACTION: find possible areas for sub1 - the system finds five possible locations

This set of five possible locations is called a "State Family"

ACTION: locate sub1 on site - the system decides on one location

Figs. 4.4 and 4.5 illustrate the five possible locations of sub1 on site. When SIGHTPLAN designs the site at a later stage of construction, the same process will be repeated to locate sub2.

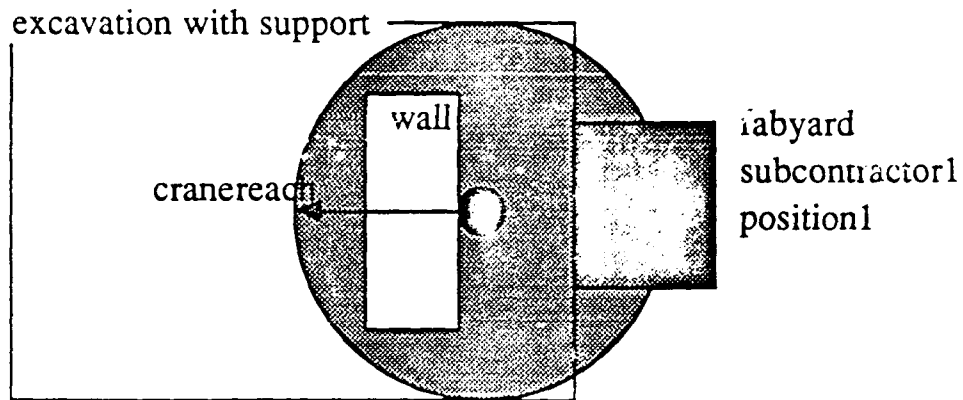


Fig. 4.4 Example of reasoning about site layout
(Tommelein, Levitt, Hayes-Roth, 1987)

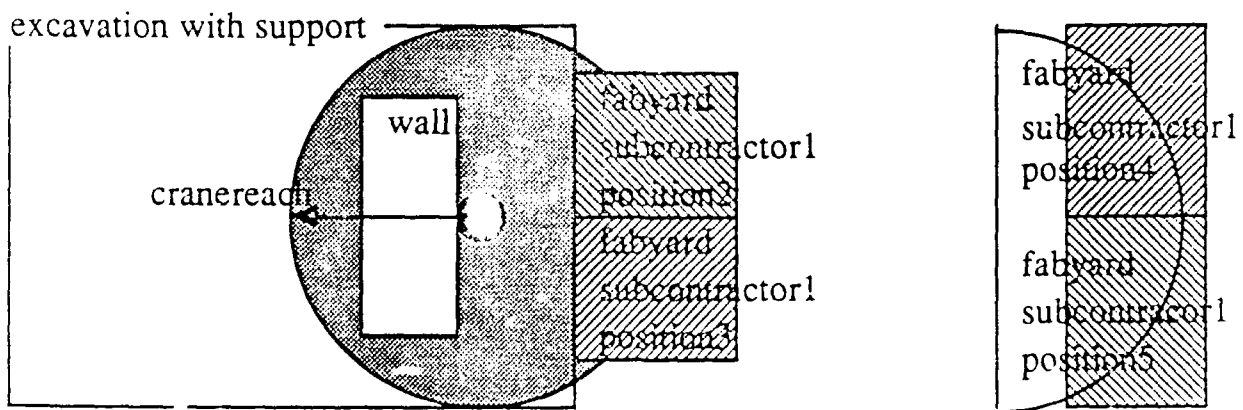


Fig. 4.5. Both subcontractors are located on site
(Tommelein, Levitt, Hayes-Roth, 1987)

Sub1 may decide to take position 1 and when he finishes his work, position 1 is then available for use by sub2.

Strategy ii):

GOAL: locate both sub1 and sub2 on site
SUBGOAL: locate sub1 on site
DATA: the area required by sub1 is 900 square feet (100 m²)
CONSTRAINT: the area for sub1 needs to be within crane reach
ACTION: locate the crane on site; find area of crane reach
ACTION: find possible areas for sub1 - the system finds five locations
SUBGOAL: locate sub2 on site
DATA: the area required by sub2 is 900 square feet (100 m²)
CONSTRAINT: the area for sub2 needs to be within crane reach
ACTION: locate the crane on site; find area of crane reach
ACTION: find possible areas for sub1 - the system finds five locations
ACTION: locate sub1 and sub2 simultaneously on site

If the five same possible locations are generated for each sub, several combinations (e.g., sub1 in position 2 and sub2 in position 3) are possible, which satisfy their location constraints.

SIGHTPLAN prototype is built based on a fictitious project and contains intuitive knowledge. It lays out rectangular objects with given geometry and dimensions on a orthogonal site grid. At a given instant of the construction schedule, a limited number of objects is on site. It is proposed to build on extended system and prototype implementation would be refined and elaborated using an existing construction project, field data and site expertise.

4.2.3 Brickwork Expert: BERT

(Bowen, Cornick, and Bull, 1986)

BERT is an interactive design aid which evaluates proposed designs for the brickwork cladding of a building. It critically reviews a submitted design from an AUTOCAD system and suggests improvements to the user for editing the drawing.

Methodology

The user supplies the design of the brickwork cladding as an input through an IBM PC CAD program called AUTOCAD. This input is then restructured by AUTOCAD's attribute file generator into a text file which describes symbolically the face of the building in question. A graphical representation processor examines the text file and then computes the spatial relationships between the features of the building. Rules about proper location of the movement joints are incorporated in the knowledge base of the system, which are then mapped into LUCIFER programming language rules. The main architecture of LUCIFER is based on forward chaining, although there are provisions for backward chaining and a blackboard type architecture, enabling the knowledge from LUCIFER to be shared by other expert systems. BERT has also a brick database which stores details about the parameters of each of the different types of bricks. Upon completion of analysis of the design, BERT will recommend changes in the design which may be incorporated by the user into the original design. The user may then resubmit it to BERT for another cycle, or exit the program. BERT is an operational prototype expert system designed in conjunction with a major brick manufacturer in order to standardize design advice to architects.

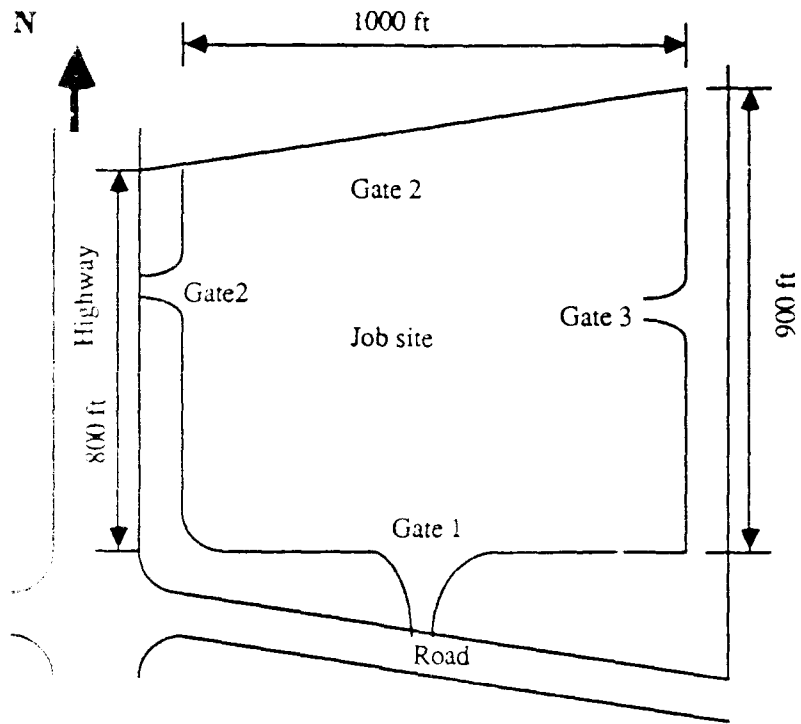
4.2.4 Site Layout Expert System: CONSITE (Hamiam and Popescir, 1988)

4.2.4.1 Introduction

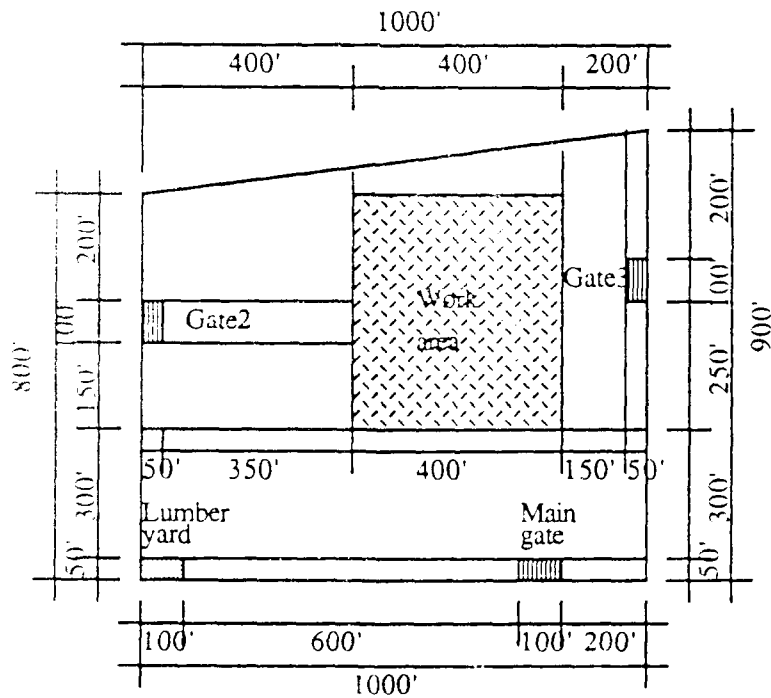
CONSITE has been developed to demonstrate the viability of the knowledge based expert system approach to the jobsite layout problem. Its knowledge base contains representations of the site and the temporary facilities to be located and also embodies the design knowledge of the expert. It manipulates the facilities, extracts information from the actual site layout, generates alternative locations for the facilities, tests constraints, selects a location and updates the layout. CONSITE uses a representation which is a mixture of rules, frames and object-oriented programming in the KEE environment.

4.2.4.2 Facility and site representation

The site is divided into a set of convex polygons of three possible types: open, closed or access. The open space type is the space available for facilities location whereas the closed space is that already used up by any kind of obstruction such as trees, existing buildings, etc. The access space is the space needed by workers and equipment at the site. Each of the polygons is further made up of a set of sides, each side being unique and part of only one polygon. Fig.4.6 shows the representation job site of an office building in CONSITE. A convex polygon representation of a job office is illustrated in Fig. 4.7.

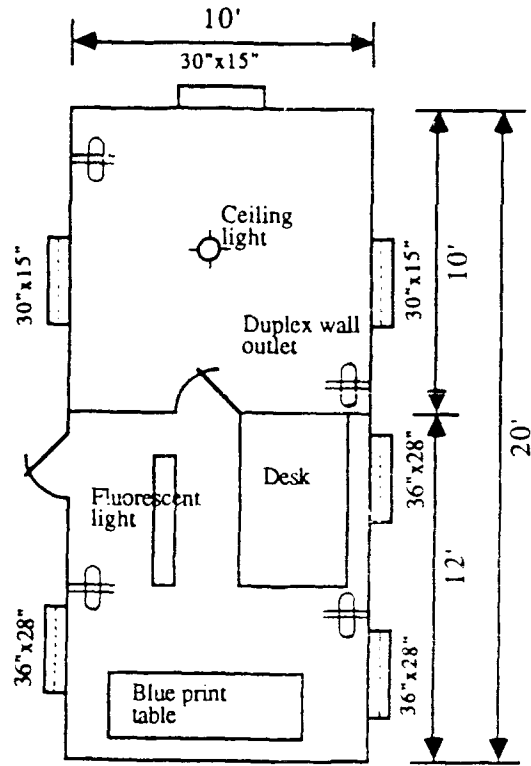


Location description of test case

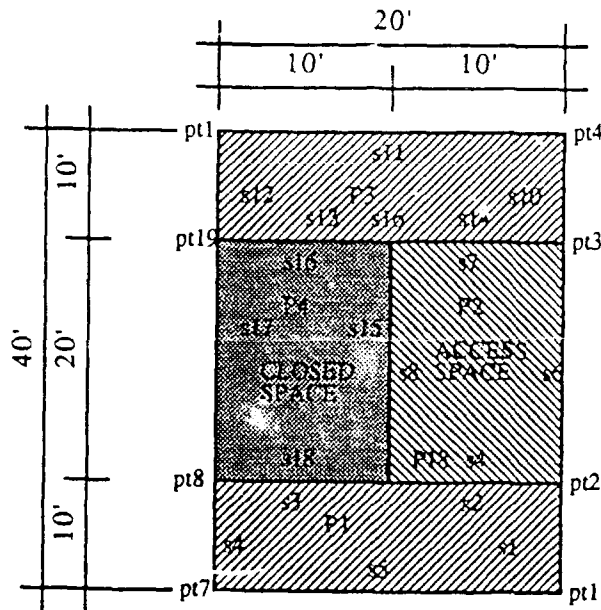


Convex polygon representation of the site of test case

Fig. 4.6 Convex polygon representation of a job office (Hamiani and Popescu, 1988)



Engineering drawing of job office



Spatial requirement of job office

Fig. 4.7 Representation of the job site of an office building (Mamiani and Popescu, 1988)

4.2.4.3 Design knowledge and design status representations

The expert's design knowledge consisting of heuristic and rules of thumb acquired through years of experience is represented in CONSITE as a set of rules. These rules recognize the commonly occurring patterns of layout by identifying the facility, extracting information from the actual layout, activating methods that generate possible locations for the facility at hand and updating the layout representation.

Design status and related information are monitored by CONSITE using a frame named Design that has attributes whose values change in time to represent the different states of the layout process. This unit keeps track of the facility being located, the alternative locations generated and the alternative selected at the previous level of the design process. It also keeps a list of the polygons that represent the site at the current stage of the design.

4.2.4.4 Alternative representation

During the design process, CONSITE generates alternative locations for the facility to be entered into the layout. These alternatives are generated as frames with attributes that allow their identification and evaluation. The set of constraints represented in CONSITE forms an important set of attributes.

4.2.4.5 Constraints representation

Constraints in CONSITE are desired qualities of the layout due to relationships between the facilities and the work area, the facility and the outside world or between the facilities

themselves. Interaction of facilities with other facilities, the work area or the region outside the site boundaries affects their location. The constraints implemented in CONSITE are: i) adjacency constraint, ii) distance constraint, iii) access constraint, iv) spatial constraint, v) position constraint and vii) view constraint.

4.2.4.6 Knowledge base organization

The knowledge base organization is shown in Fig. 4.8 and the knowledge is represented in frames and rules. The frames define the static knowledge that represents objects in the layout and their attributes which are either descriptive or procedural in form and allow a description of real objects such as the site and the facilities and of abstract objects such as the polygons, the sides and the points. Rules represent both heuristic and judgemental reasoning knowledge whereas the object-oriented programming describes procedural language, such as numerical processing, overlay checking, translation and rotation of the facilities. This data-dependent programming is attached to slots in frames describing specific objects.

4.2.4.7 Problem solving strategy

CONSITE uses a plan-generate-and-test strategy. The planning phase takes advantage of the order in which the expert enters the facilities into the layout. This ordering is implemented through task sequencing. At every level of the search tree, the task is to find a location for the actual

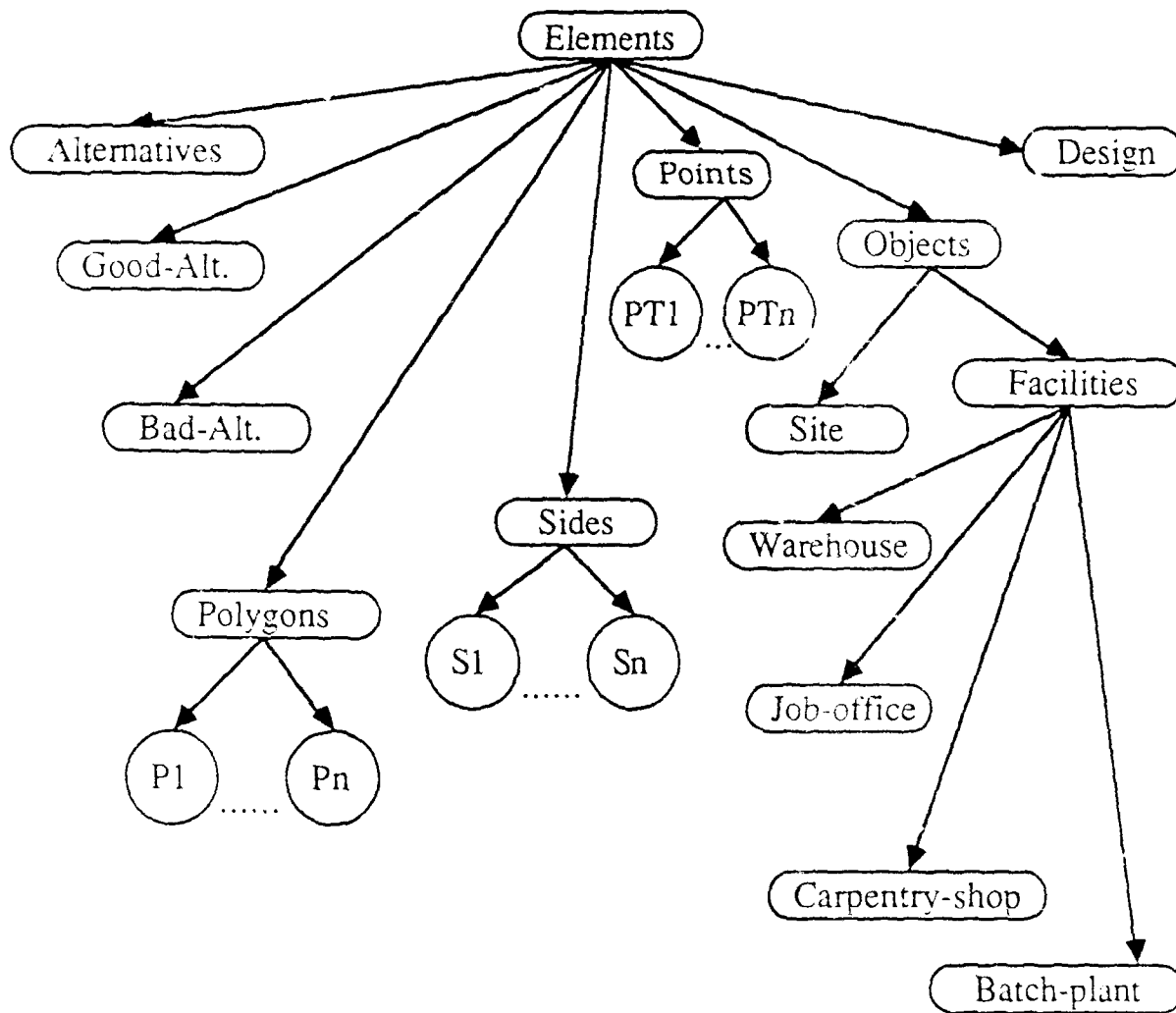


Fig. 4.8. Organization of the knowledge base
(Mamiani and Popescu, 1988)

context (facility). Once the context is known, CONSITE activates the generator which is a set of LISP functions that manipulate the space representation and generate alternate locations. The selection of the alternative is done during the testing phase after checking the facility location for the constraints and transferring them to the good-alternative or bad-alternative category. The final location is selected from the good alternatives and implemented through an update of the list of polygons representing the layout in the frame design. The output is displayed graphically on the screen, indicating the location of each facility on the site. The output from CONSITE after solving the office building problem is shown in Fig. 4.9.

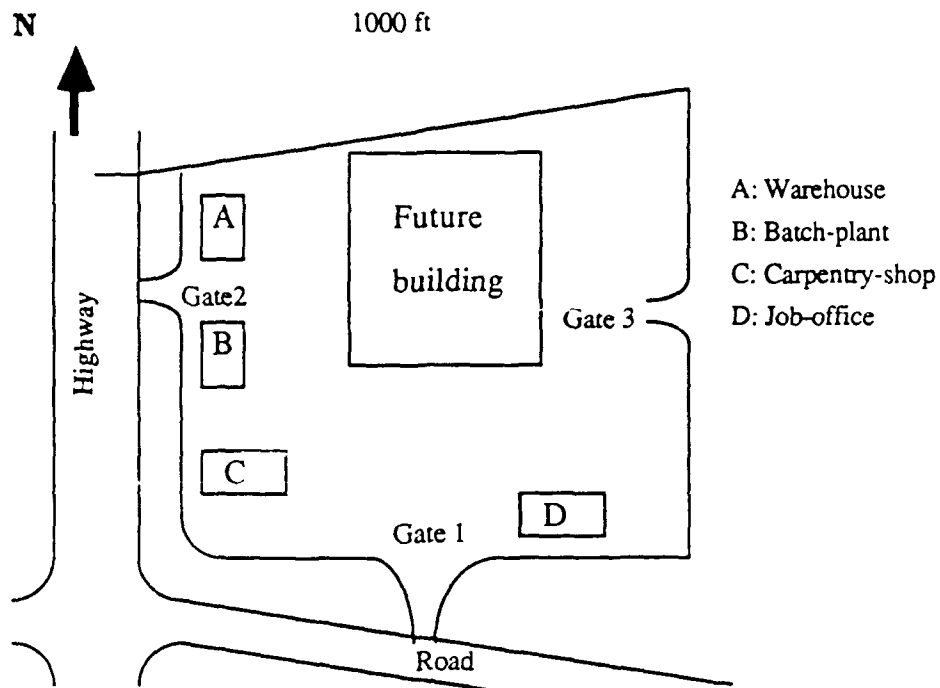
4.2.5 ES for Contractor Prequalification (Russell and Skibniewski, 1988)

4.2.5.1 Introduction

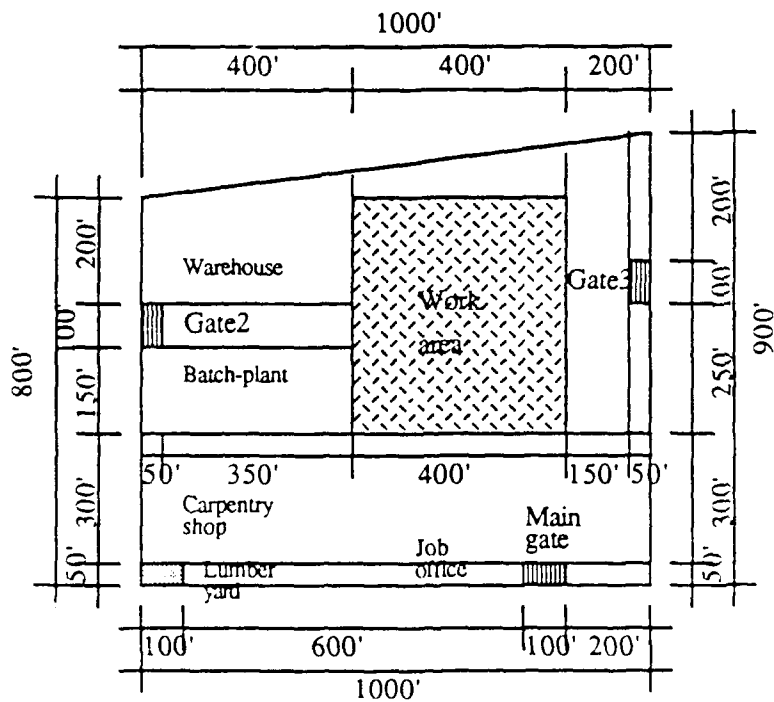
A prototype rule-based expert system is being developed to aid in the contractor prequalification decision-making process from an owner's perspective. The task of selecting the 'right' bidder for a particular project is one of the most challenging tasks performed by an owner or contract administrator. Contractor prequalification is a decision-making process involving a wide range of criteria for which information is often qualitative and subjective.

4.2.5.2 Knowledge acquisition strategy

The knowledge acquisition process involved the following three steps: i) gathering general information (viz.



Actual layout produced by the expert



Final layout produced by CONSITE

Fig. 4.9 Output of CONSITE after solving the office building problem (Mamiani and Popescu, 1988)

identification of decision factors and subfactors, peculiarities and biases in the process) on prequalification process, ii) development of a questionnaire on the impact of major decision factors and subfactors on the prequalification decision-making process, and iii) structuring the subfactors into sub-subfactors, and extracting, formalizing and developing qualitative and quantitative rules.

4.2.5.3 Knowledge base design

The structure of the knowledge base presented in Fig. 4.10 consists of two modules:

- i) Decision-Maker Module (Owner): This module represents the characteristics of the decision maker (owner) which impact the selection of the decision strategy and the development of the prequalification criteria;
- ii) Contractor Module: This module is used to store appropriate characteristics of the contractors being prequalified.

The characteristics of the decision maker include, among others, items such as type of owner (e.g. public or private), owner objectives, type of construction and contracting strategy. The decision strategy selected can include dimensional weighting, two step prequalification process, and subjective judgement. Table 4.1 presents the major composite factors relevant in the decision-making process for public owners and private

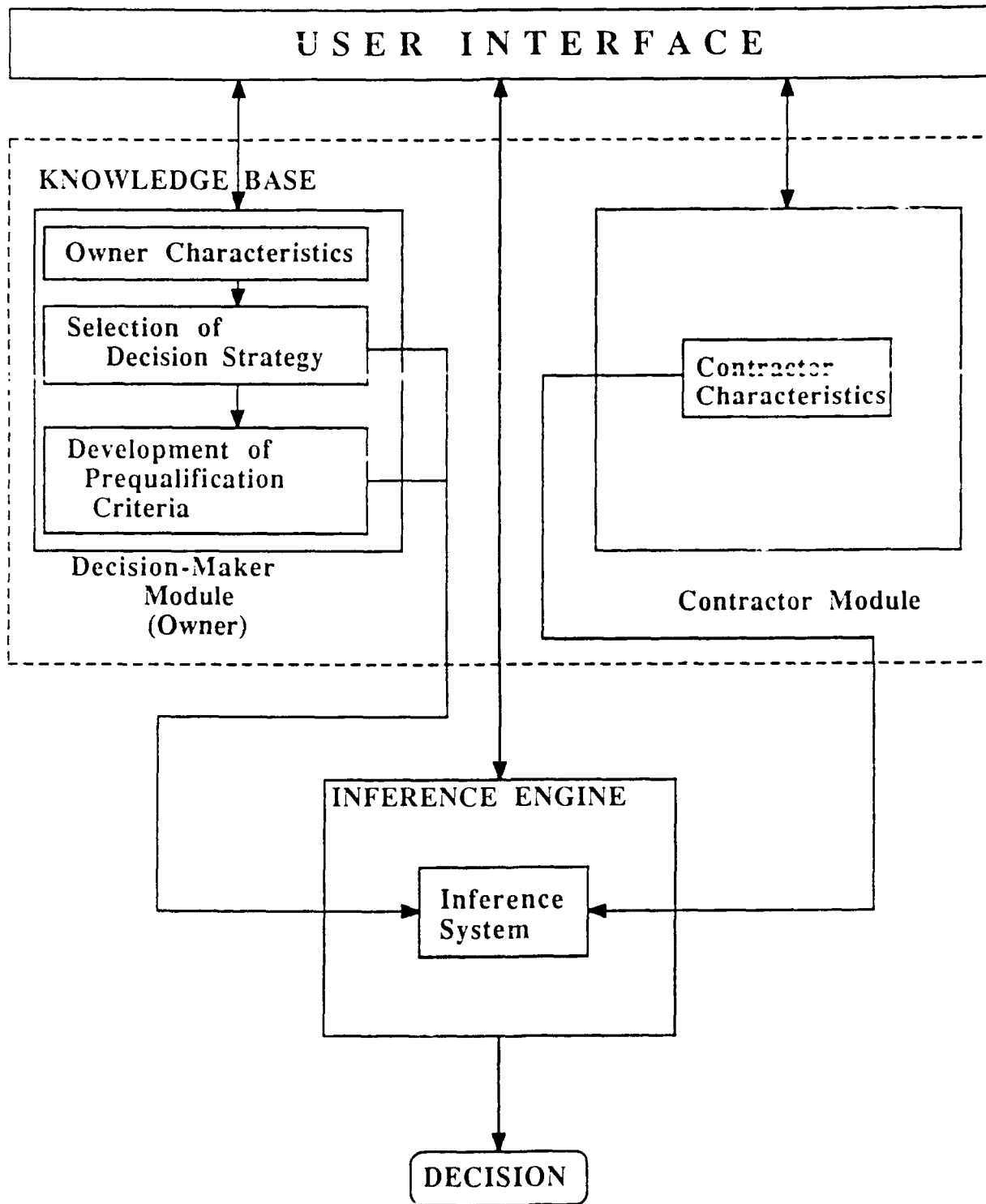


Fig. 4.10 Structure of a rule-based expert system for contractor prequalification (Russell and Skibniewski, 1988)

owners/construction managers. Each of the composite factors listed in Table 4.1 can be further characterized by the factors that make up the given factor.

TABLE 4.1 Major factors utilized in the contractor prequalification process

GROUP	
Public Owners (1)	Private Owners and Construction Managers (2)
Performance	Management
Type of Contractor	Safety
Capacity for New York	Location
Location	Performance
Worked Performed	Resources
Percentage	Financial Stability & Experience
Third Party Evaluation	Failed Performance
Financial Stability	Bonding Capacity
	Capacity for New York

For example, the "management" composite factor for private owners/construction managers consists of:

- i) Project control procedures;
- ii) Project management capabilities;
- iii) Staff available;
- iv) Company organization

The factors can be further characterized by subfactors. For example, 'company organization' consists of:

- i) Type of ownership (e.g. partnership, corporation,

- sole owner);
- ii) Number of years in construction;
- iii) Contractor's licenses held (by state and/or by type of work);
- iv) Number of times a contractor has failed to complete a contract.
- v) Appropriateness of company organizational structure.

The factor 'financial stability' can be broken down into four subfactors:

- i) Credit rating;
- ii) Banking arrangements;
- iii) Bonding;
- iv) Balance sheet

The 'balance sheet' subfactor can be further reduced into the following parameters:

- i) Net worth (shareholder's equity);
- ii) Working capital;
- iii) Debt/net worth ratio

The knowledge will be represented by self-contained pieces of knowledge in the form of "if .. then" production rules. The standard syntax adopted for a production rule is:

IF (condition)
THEN (action)

At each level of the hierarchy production rules must be

formulated; Table 4.2 shows sample production for the subfactor 'balance sheet'.

Table 4.2 Sample rules for the subfactor "Balance Sheet"

IF WORKING CAPITAL < \$0.000
(Current Assets - Current Liabilities)

THEN Contractor is experiencing cash flow difficulties and the contractor's banking arrangements should be checked.

IF DEBIT/NET WORTH RATIO > 3 TO 1
(Shareholders' Equity)

THEN Contractor is highly leveraged and is not carrying a majority of the financial risk (the bank and/or material suppliers and/or equipment suppliers are carrying the risk).

IF NET WORTH < \$10,000
(Shareholders Equity)

THEN Contractor does not have enough financial risk. In the event of an unforeseen situation (e.g. loss of money on the project) it is highly likely the contractor will not stay and complete the project.

IF THE AMOUNT (\$) IN COMMON CAPITAL STOCKS < 30%
OF NET WORTH (Shareholders' Equity)

THEN Shareholders' have little equity in the business.

IF WORKING CAPITAL < \$0.00 and NET WORTH <
\$100,000 and BANKING ARRANGEMENTS = NO

THEN Contractor can not pay his bills.

IF DEBT/NET WORTH RATIO > 3 TO 1 and
NET WORTH < \$10,000

THEN Contractor currently does not have substantial financial risk to guarantee the completion of the project.

The contractor prequalification ES will be developed utilizing a backward-chaining inference mechanism. The procedure will be invoked to determine whether each contractor is acceptable to submit a bid for the project.

4.2.6 Expert Systems in Real-Time Construction Operations

(Paulson, and Sotoodeh-Khoo, 1987)

4.2.6.1 Introduction

The optimum loading time for an earth moving scraper varies with the length of the haul, as well as with changes in variables such as the current soil grain size, moisture content, cohesion, and density. Less experienced operators may overload or underload their scrapers under rapidly varying conditions. The expert systems are designed to specify a fleet of equipment for a given project, aid new operators to understand optimum loading times for each machine and optimize fleet production by communicating between machines in real time.

Real-time data collection via electronic instrumentation of construction field operations can be joined with knowledge based expert systems to implement analytical modeling procedures such as simulation and non-linear production optimization. The real-time instrumentation and monitoring of earthmoving scraper operations have been interfaced with the EXSYS expert system shell running on a IBM-PC/AT computer for implementing the non-linear optimization method.

4.2.6.2 Methodology

4.2.6.2.1 Single-scraper expert system

The scraper operator inputs the type of machine, the type of soil (e.g. sand, clay, etc.) and the working conditions (e.g. wet ground, low traction, etc.) to be on-board expert system. Based on the input information, the knowledge base would inform the operator of a range of load-times where he is most likely to achieve maximum production. A load-time from the specified range (the middle value of the range, for instance) would be selected by the on-board monitoring system. It warns the operator to stop loading and start hauling as soon as that specified load-time has elapsed. A few seconds into the haul cycle, the load cells mounted on the machine could determine the average payload based on several samples of the payload. The balance of cycle time (i.e. haul, dump and return times) for the first run may be provided from the knowledge-base and that in subsequent runs be computed accurately by recording data from load sensors, strain gages, gravity mass sensors, optimal volume sensing, inertia sensing, gearbox sensing and speedometer readings. The machine production in volume units per hour for a specific load-time and a known cycle time and payload could be computed and stored in memory for later reference and comparison.

The system would pick a different load-time (either higher or lower than the first one) on subsequent loadings and production per hour could again be calculated based on the payload during the haul cycle for this load-time and compared to the previous one. An increase in production would mean that

load-time is approaching the optimum value. On the other hand, with a decrease in production the system would then try a different load-time in the opposite direction. Within a few iterations, the system would ultimately converge on a load-time close to the optimum based on the inference of the appropriate rules; the operator would be advised to operate at that load-time until a different value was obtained by the system due to a change in one of the factors affecting production (i.e., an increase or decrease in the cycle time, a change in material properties affecting load-time, a change in the equipment fleet, in haul road conditions, etc.) Fig. 4.11 shows a succession of such points that define a portion of the actual load-growth curve.

4.2.6.2.2 Fleet management expert system

The coordination of a fleet of earthmoving machines consisting of the same size and type or of differing sizes becomes more complex and challenging to minimize wait times for pushers and scrapers during their respective cycles. The fleet optimization problem is carried out using rule-based logic in the EXSYS environment. This software not only allows deduction using rule-based logic in achieving a theoretical optimum fleet balance and load-time for each scraper. This theoretical value will then be checked and adjusted accordingly, based on real field data collected through the on-board sensors.

The knowledge base would compute the correct number of machines to achieve the completion goal based on user input information about the project duration and the earthmoving

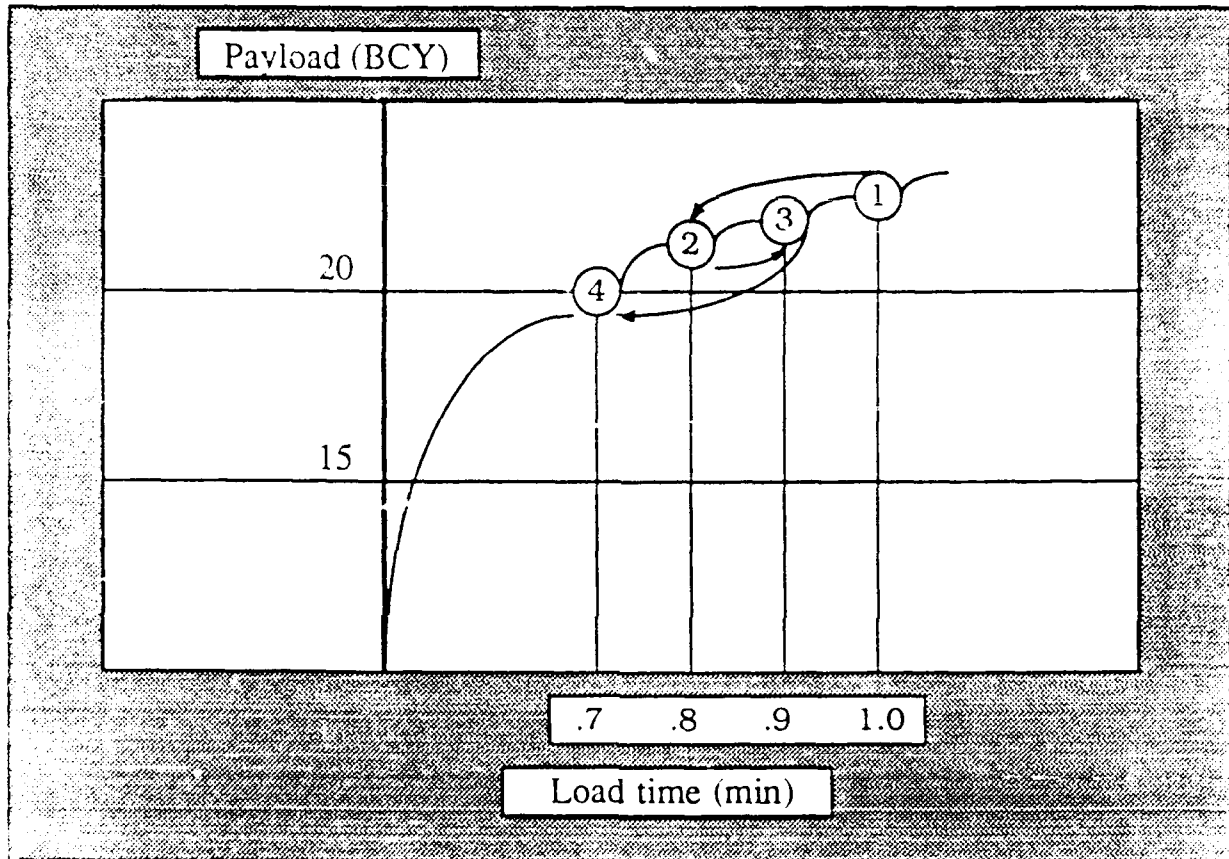


Fig. 4.11 Succession of points defining a portion of the load-growth curve (Paulson and Sotoodeh-Kooh, 1987)

volume. Knowing the haul and return road lengths, grades and rolling resistance, the knowledge base would access external data bases to calculate the scraper travel speeds (when loaded and empty) and determine a theoretical time for the balance of the cycle. Using additional logic based on the fleet theory, a load-time is then selected such that scraper and pusher times are optimally balanced. The selected theoretical load-time is then transmitted to the field for validation in field conditions. The communication between the ES and the real-time data acquisition program for data validation would eventually enable the logic based program to learn from its past suggestions and make better decisions in the future under similar job and equipment conditions.

4.3 EXPERT SYSTEMS IN CONSTRUCTION MANAGEMENT

4.3.1 Introduction

Construction management includes planning, scheduling, and control of construction activities as well as the design of legal, behavioral and other elements of the construction process. Potential applications of expert systems in the area of construction project monitoring involves checking, regulating and controlling the performance and execution of the project. Only selective ES applications in construction management are presented in the following:

4.3.2 ES Architecture for Construction Planning: CONSTRUCTION PLANEX (Fenves, Flemming, Hendrickson, Maher, and Schmitt, 1988)

Construction planning involves the choice of construction technologies, definition of work tasks, estimation of required resources and durations, estimation of costs and preparation of project schedules. CONSTRUCTION PLANEX is a knowledge-based expert system which synthesizes activity networks, diagnosis resource needs and predicts durations and costs. The system will either generate a plan automatically or a planner can review and modify decisions during the planning process. The system has three essential parts as illustrated in Fig. 4.12. The Context stores information on the particular project being considered including the design, site characteristics, planning decisions made, and the current project plan. The Operator Module contains operators which create, delete or modify the information stored in the context. Operators are of two types: i) Specialized and ii) Control. Specialized operators are used for tasks such as technology choice, activity synthesis, duration estimation, etc. The order in which specialized operators are executed is determined by control operators. Interaction between the two types of operators occurs by means of a message interface representing the role of a blackboard. The Knowledge-base contains distinct knowledge sources of tables and rules specific to particular technology choices, activity durations, or other considerations. Each knowledge source is used by a particular operator. A user interface with an explanation module is included in addition to the central components.

The following variety of objects storing information in the Context are available (Hendrickson, Zozaya-Gorostiza, Rehak, Baracco-Miller and Lim, 1987):

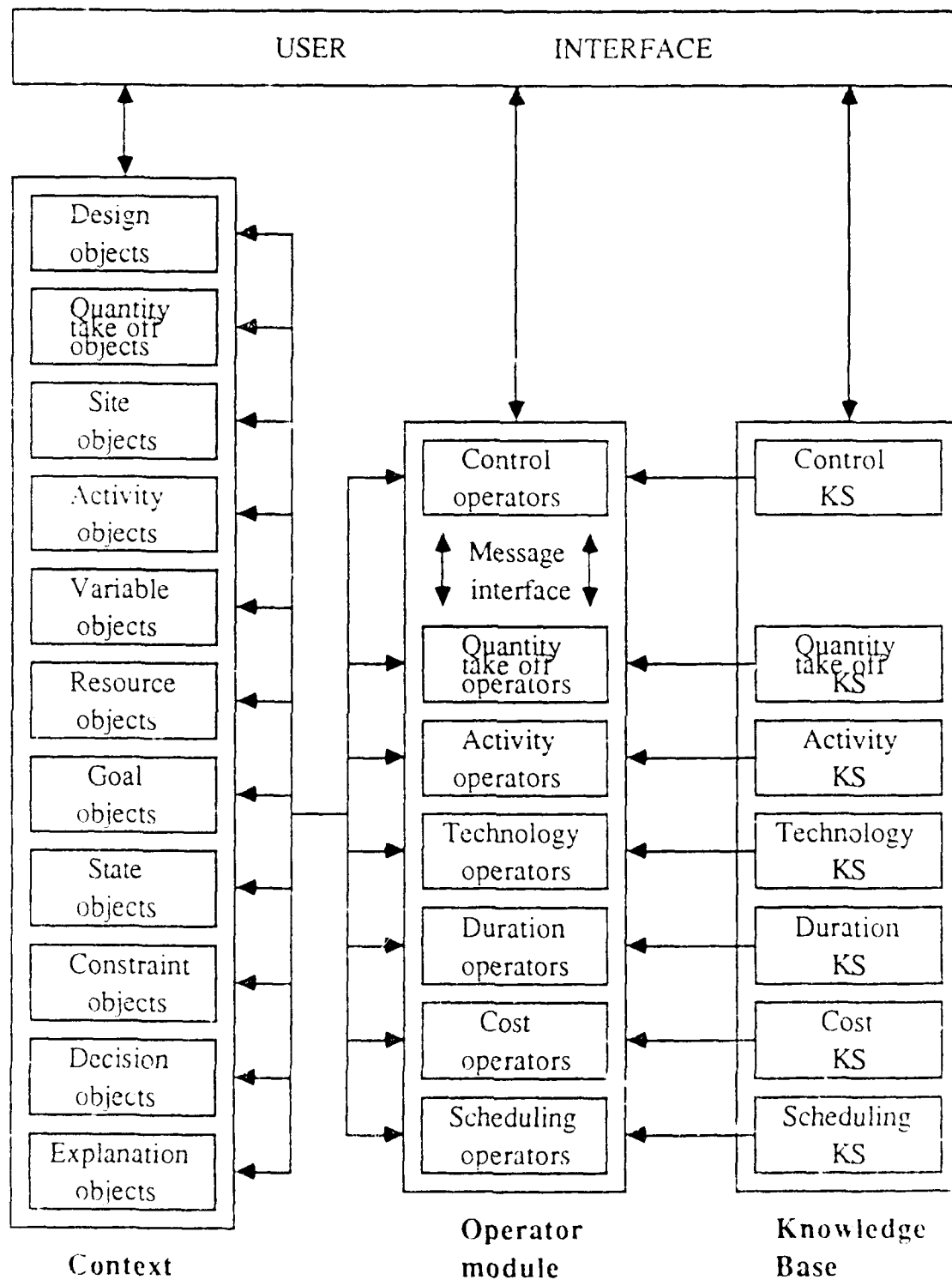


Fig. 4 12 Overview of CONSTRUCTION PLANEX
 (Fenves, Flemming, Hendrickson, Maher, and Schmitt, 1988)

- .Design Element objects that store information about design components,
 - .Quantity-Take-Off objects that store information about elements of work,
 - .Site-Characteristics objects that store information about different conditions on the site,
 - .Activity objects that represent construction tasks at different levels of aggregation,
 - .Resource objects indicating the characteristics of equipment, labor or materials,
 - .Goal objects that define different stages in the planning process,
 - .State objects used dynamically to describe the characteristics of the planning process,
 - .Constraint objects to represent required relationships among states and variables,
 - .Decision objects for representing points in the planning process which are affected by technology choice, resource allocation or other decisions made by the user or CONSTRUCTION PLANEX,
- and
- .Explanation objects to store information or pointers to information about the construction plan.

The above mentioned objects are related by a network of relations representing the current project plan, decisions made during the planning process and different joining schemes. The set of activities thus form a project network whereas the system context contains a more extensive network which also registers the planning process and other information. The generation of elements of work defined by the user in the prototype is automated by the insertion of design element objects in the context. Typical modules contained in the operator module are

the following:

- .QTO operators to create elements of work based on design element information,
- .Activity operators to create, elaborate, expand, link or aggregate activities,
- .Technology operators to suggest appropriate equipment or technology,
- .Duration operators to perform estimation,

and

- .Scheduling operators to provide a project schedule including critical path identification and any required resource allocation.

All operators are generic, so that a single operator can be used for all activities. For example, the duration estimation operator would be called for each element activity and a knowledge source specific to that activity consulted to obtain a duration estimate.

The operation of CONSTRUCTION PLANEX relies heavily on a number of distinct knowledge sources (Zozaya, 1987). An example knowledge source applied to a small task in the overall planning process is illustrated as a decision table in Fig. 4.13. Different sets of activities required for the construction of a footing are suggested in the knowledge source depending upon soil characteristics. CONSTRUCTION PLANEX knowledge sources perform as small expert systems themselves by supporting numerical functions, calls to other knowledge sources and binding.

PLANEX performs the following sequence of operations in the initial creation of a construction plan:

- .Create element activities for design elements. A set

Name : KS-Example		Type : first				
Object	Slot	Op	Value	RULES		
current-object	type-element	is	cast-in-place concrete column-footing	t	t	f
soil-characteristics	backfill	is	yes	t	f	i
excavate-column-footing				x	x	i
dispose-excavation-column-footing				x	x	i
pile-up-excavation-column-footing				x	i	i
boorow-material-column-footing				i	x	i
place-forms-column-footing				x	x	i
reinforce-column-footing				x	x	i
pour-concrete-column-footing				x	x	i
remove-forms-column-footing				x	x	i
KS-other-elements				i	i	x

Fig. 4.13 Illustration of a CONSTRUCTION PLANEX knowledge source (Fenves, Flemming, Hendrickson, Maher, and Schitt, 1988)

of element activities required to construct each design element about precedences among activities, technologies to employ, required resources, etc. are made by other operators.

- . Group element activities of common characteristics in order to have a hierarchy of element activities similar to that of MASTERFORMAT. Thus, element activities are associated with particular physical design elements (such as a column or a beam) and aggregations of activities called project activities and project activity groups
- . Determine amounts of work for element activities. Geometric information for the quantity take off is inherited from design element frames in the central data store.
- . Select units of measure for element activities. Crew productivities or material quantities may be expressed in different units (e.g. days instead of hours).
- . Determine material packages for element activities based on design specifications.
- . Create project activities that aggregate element activities and provide summary information on the underlying element activities.
- . Determine precedences for project activities. Scheduling is performed at the project activity level, reflecting the homogeneity of resource use and the small granularity of detail contained in the underlying element activities in CONSTRUCTION PLANEX.
- . Compute lags for project activities. Element activities of several project activities are structured into an element activity subnetwork. Relevant lags among project activities based on this subnetwork is determined using critical path algorithm.
- . Select technologies for project activities. Technologies are chosen at a macro-sopic or project level since consistency in this regard will reduce costs.
- . Estimate durations for project and element activities.
- . Schedule project activities using CPM, resource allocation and constraint satisfaction.
- . Estimate costs by computing activity costs and project costs using unit costs and scheduling information.

The CONSTRUCTION PLANEX system could be applied to different types of projects, although each type of project would require different knowledge sources. The system is now being implemented in the KNOWLEDGE CRAFT expert system environment for application domain of office building construction.

4.3.3 Analysis of Contingencies in Project Plans: PLATFORM-II (Kunz, Bonura and Stezlner, 1986)

PLATFORM II is an expert system developed to illustrate the use of the Artificial Intelligence (AI) technique of "multiple worlds" in making project feasibility decisions under uncertainty. This technique assists the project manager in making a decision involving multiple uncertainties by generating "worlds" which describe all the possible combinations of choices available to the project manager together with the implications of those choices and their outcome probabilities and values based on user-specified evaluation criteria.

Methodology

PLATFORM-II was developed using the Intelli Corp Knowledge Engineering Environment (KEE) and employs the frames, rules and graphics which are integrated in KEE. The use of the assumption-based truth maintenance system (ATMS) of KEE, Version 3 is a significant feature of PLATFORM-II. The user is allowed to make assumptions regarding a decision (e.g. whether to choose to build the graving dock for construction of the concrete base of a platform in Norway or Scotland). Project cost and duration are dependent upon decisions which must be made by the project

manager. Fig. 4.14 illustrates part of the frame structure of the application knowledge base (KB), named PLATFORM-II. The PLATFORM-II KB uses units to represent diverse objects such as individual activities in the project schedule, rules to update the schedule, and graphical images on display panels. GEOLOGICAL.ALTERNATIVES is a unit which heads a subtree that describes the geological conditions likely to be encountered in the building of the graving dock. The LABOR.PRODUCTIVITY.ALTERNATIVES unit forms a subtree describing labor productivity which must be taken into account in building the platform. The "facts" for each scenario or world combine knowledge about the probabilities for each geological and labor productivity alternative at that site.

Rule premises indentifying the project and its location as issues are shown in Fig. 4.15. Alternatives are specified in the rule premises as the project and location units belonging to the referenced class units, viz. the DRILLING.PROJECTS and POSSIBLE.DOCK.LOCATIONS units. The THEN portion of the rule specifies the appropriate conclusion to make for a given set of issues and alternatives. The rule conclusion records the site and a set of likelihoods for each different location in which the project might be constructed.

Fig. 4.16 shows the rule which stipulates the problem to be analyzed. The issues are recognized in the premises-drilling projects, geology, labor productivity, and siting. The CREATE.WORLD operator in the conclusion forms a new world for each situation in which the premises are valid, and it specifies

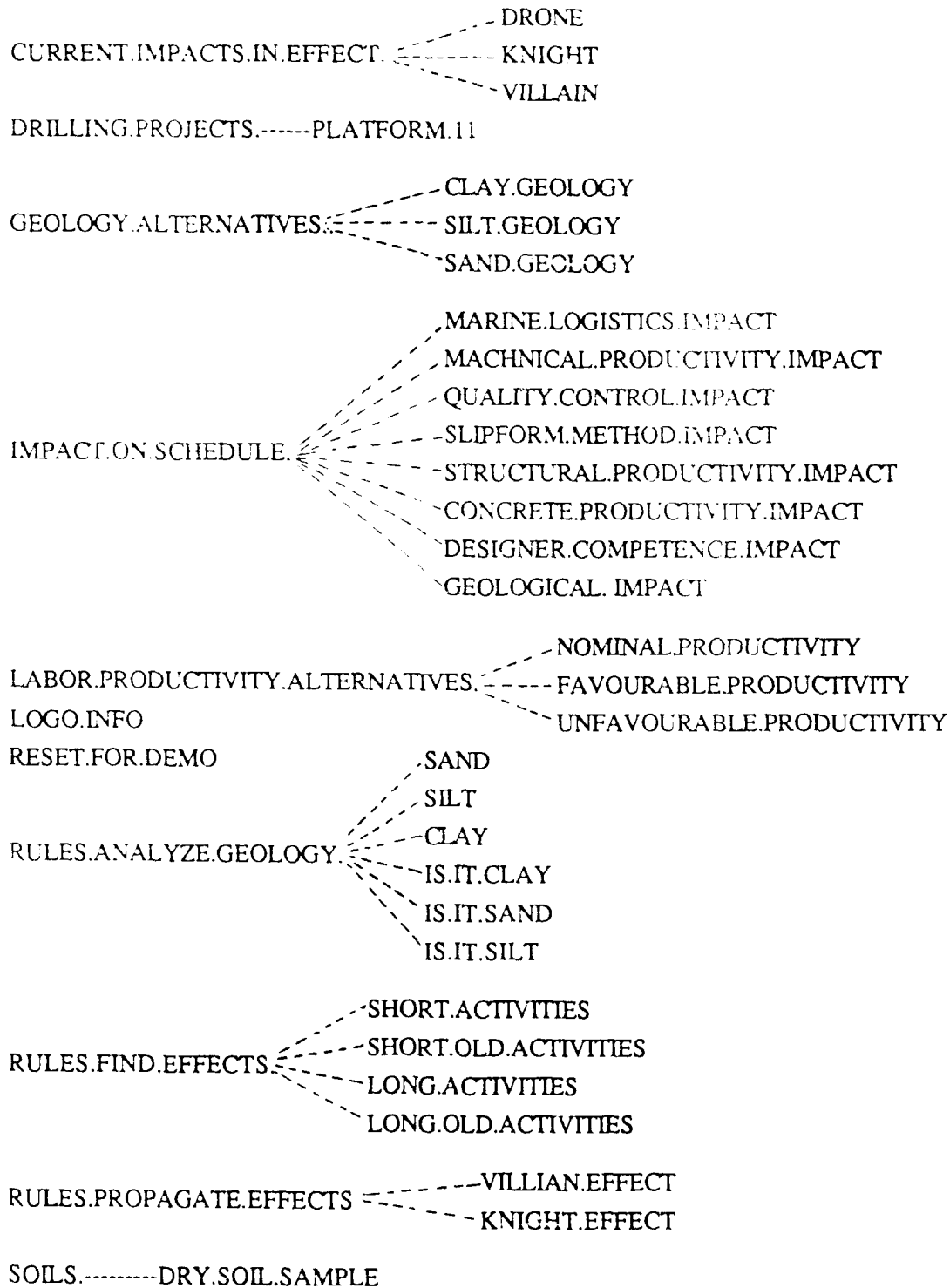


FIG. 4.14 A portion of the platform-II project knowledge base
(Kunz, Bonura, and Stezner, 1986)

```

(IF (?PROJECT IS IN CLASS DRILLING.PROJECTS) AND
  (?LOCATION IS IN CLASS POSSIBLE.DOCK.LOCATIONS))
THEN CREATE.WORLD
  (THE SITE OF ?PROJECT IS ?LOCATION)
  (THE LIKELIHOOD OF NOMINAL.PRODUCTIVITY OF ?PROJECT
    IS (THE LIKELIHOOD OF NOMINAL.PRODUCTIVITY OF ?LOCATION))
  (THE LIKELIHOOD OF FAVORABLE.PRODUCTIVITY OF ?PROJECT IS
    (THE LIKELIHOOD OF FAVORABLE.PRODUCTIVITY OF ?LOCATION))
  (THE LIKELIHOOD OF UNFAVORABLE.PRODUCTIVITY OF ?PROJECT IS
    (THE LIKELIHOOD OF UNFAVORABLE.PRODUCTIVITY OF ?LOCATION))
  (THE LIKELIHOOD OF SAND.GEOLOGY OF ?PROJECT IS
    (THE LIKELIHOOD OF SAND.GEOLOGY OF ?LOCATION))
  (THE LIKELIHOOD OF SILT.GEOLOGY OF ?PROJECT IS
    (THE LIKELIHOOD OF SILT.GEOLOGY OF ?LOCATION))
  (THE LIKELIHOOD OF CLAY.GEOLOGY OF ?PROJECT IS
    (THE LIKELIHOOD OF CLAY.GEOLOGY OF ?LOCATION))

```

Fig. 4.15 Rule identifying the project and its location as issues
(Kunz, Bonura, and Stezlner, 1986)

```

(IF (?PROJECT IS IN CLASS DRILLING.PROJECTS)
  (?SOME.GEOLOGY IS IN CLASS GEOLOGY.ALTERNATIVES)
  (?LABOR.PRODUTIVITY IS IN CLASS
    LABOR.PRODUCTIVITY.ALTERNATIVES)
  (?SELECTED.LOCATION IS IN CLASS DOCK.LOCATION.ALTERNATIVES))
THEN CREATED.WORLD
  (THE RESULT.OF.GEOLOGICAL.EXPLORATION OF ?PROJECT IS
    ?SOME.GEOLOGY)
  (THE LABOR.PRODUCTIVITY OF ?PROJECT IS
    ?LABOR.PRODUCTIVITY)
  (THE LOCATION OF ?PROJECT IS ?SELECTED.LOCATION)
  (THE COST OF ?PROJECT IS '(COMPUTE.PROJECT.COST $WORLD$))
  (THE DURATION OF ?PROJECT IS
    '(COMPUTE.PROJECT.DURATION $WORLD$)))

```

Fig. 4.16 Rule used to identify the issues and alternatives
in the problem analysis (Kunz, Bonura, and Stezlner, 1986)

the conclusion to make new world. In this example of building the dock, there is one drilling project, three geological alternatives, three labor productivity alternatives and two location alternatives and hence an exclusive set of eighteen different worlds is created by the CREATE.WORLD operator. The conclusion part of the rule asserts values of the named attributes in each world, such as LOCATION and COST. The location is determined from the premises, and the cost computed by a cost function created by the user. Each world is available for inspection by the reasoning rules and by the interactive explanation system. If a line of reasoning becomes inconsistent with earlier assumptions, PLATFORM-II backtracks until it can find an appropriate place to modify the search tree. The user may modify assumptions at any time and let the system generate new worlds. Multiple worlds permits rapid computation of outcome values and allows users easily to create new worlds with slightly different facts and examine their impact on the decision or to indicate that certain worlds are inconsistent with specified criteria. It analyzes cost and time outcomes for each of the worlds generated using a complex PERT model with 50 to 100 activities and a realistic cost function which takes into account direct and indirect costs including time-related bonus/penalty amounts. PLATFORM-II which is an operational expert system is currently used to demonstrate the ATM capabilities of KEE.

4.3.4 Know-How Transfer Method (Niwa, and Okuma, 1982)

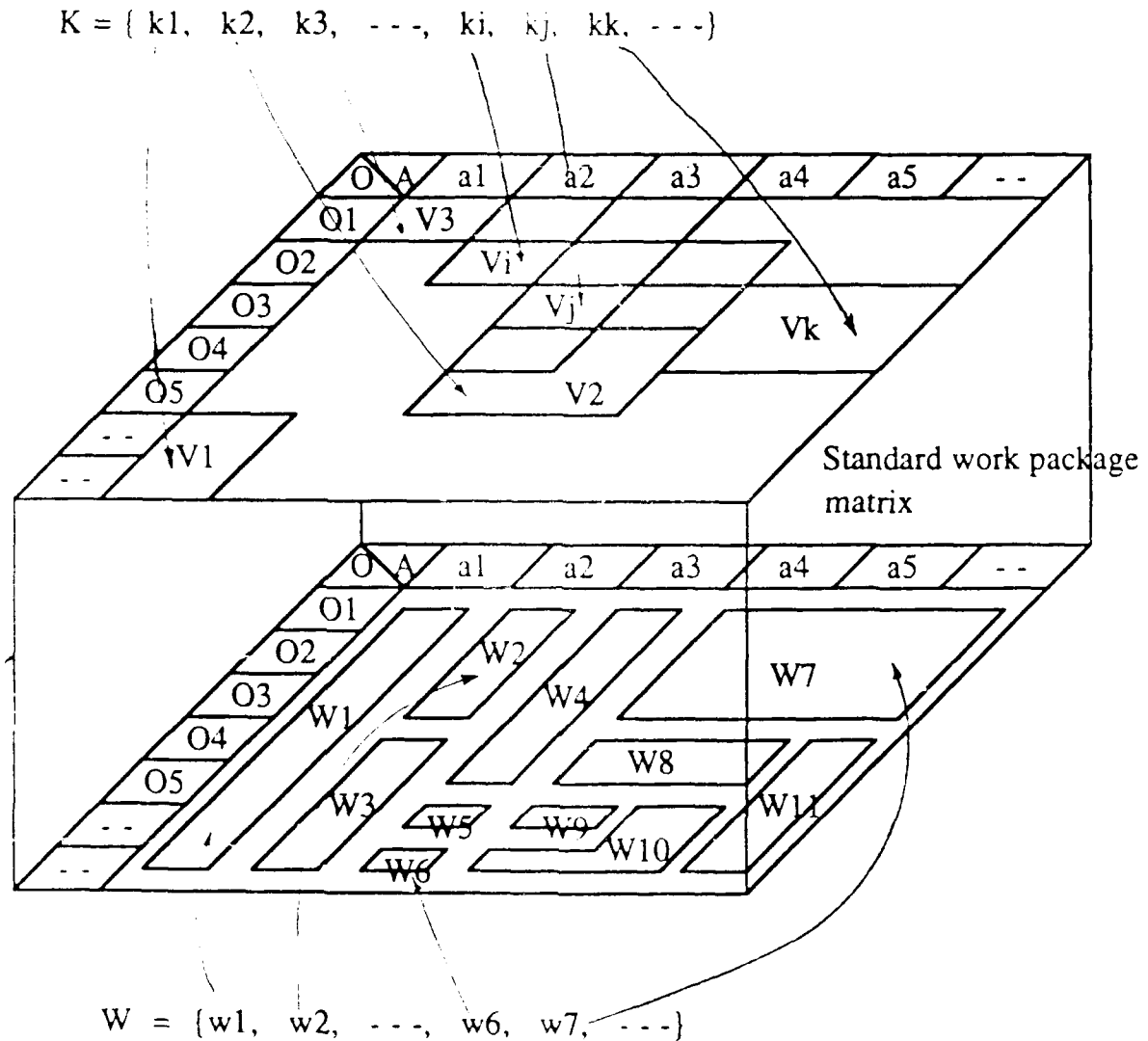
The Know-How Transfer Method is intended to improve engineering or project management. The dramatic changes in the

world economic balance in the 1970s led to many large construction projects in the Middle East. These projects faced long delays in implementation resulting from problems associated with working within a different culture, with different social and religious values. The Know-How Transfer Method was designed to help project managers with risk management at the project execution stage and its main focus is risk identification.

Methodology

The basic feature of this ES is the development of the "know-how" transfer method of acquiring knowledge for the system to use. Multidisciplinary knowledge in the different areas of managerial, technical, economic, financial, social, scientific, legal and political skills constitutes the know-how. The system stores the risk know-how onto a standard work package matrix (Fig. 4.17). The standard work package matrix consists of columns indicating activities and rows indicating objects. Each job in the project is an intersection of an activity and an object. Know-how acquired on a project is also related to an activity and an object and then placed onto the grid. This "know-how grid" is subsequently mapped on to the standard work package matrix so the knowledge may be related to the work packages as a suitable index of knowledge.

Fig. 4.18 shows the total framework of risk management system and examples of use of the ES are illustrated in Fig. 4.19. For instance, the project manager may specify a work package and the output data could be risk-reducing strategies



$K = \{ k_i \}$: Know-how ; $v_i = f(k_i)$: The domain to which k_i corresponds ;
 $A = \{ a_i \}$: Activity ; $O = \{ o_i \}$: Object ; and $W = \{ w_i \}$: Work package

Fig. 4.17 Storage of know-how "standard work package matrix" method (Niwa and Okuma, 1982)

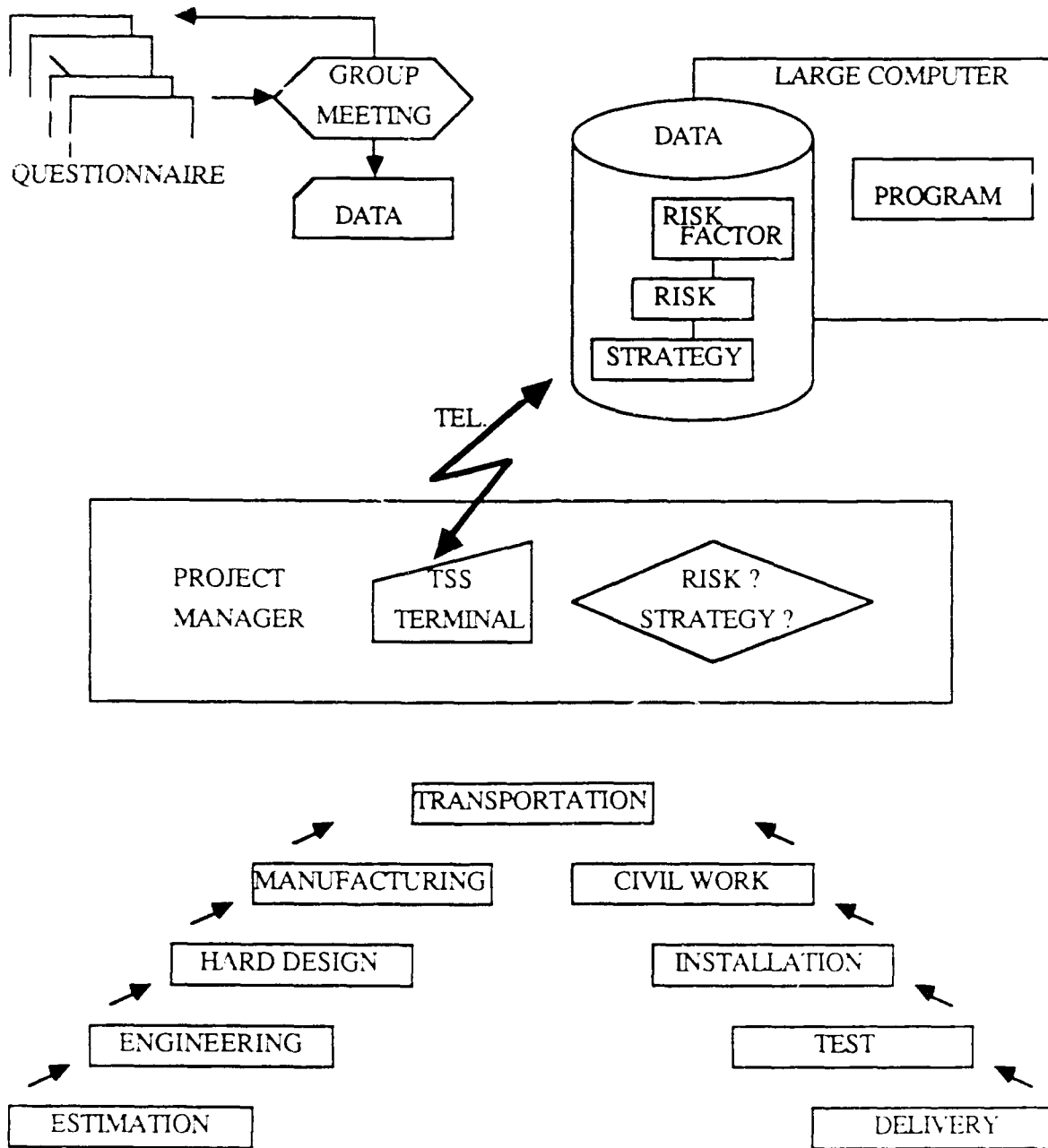


FIG. 4.18 Total framework of risk management system
(Niwa and Okuma, 1982)

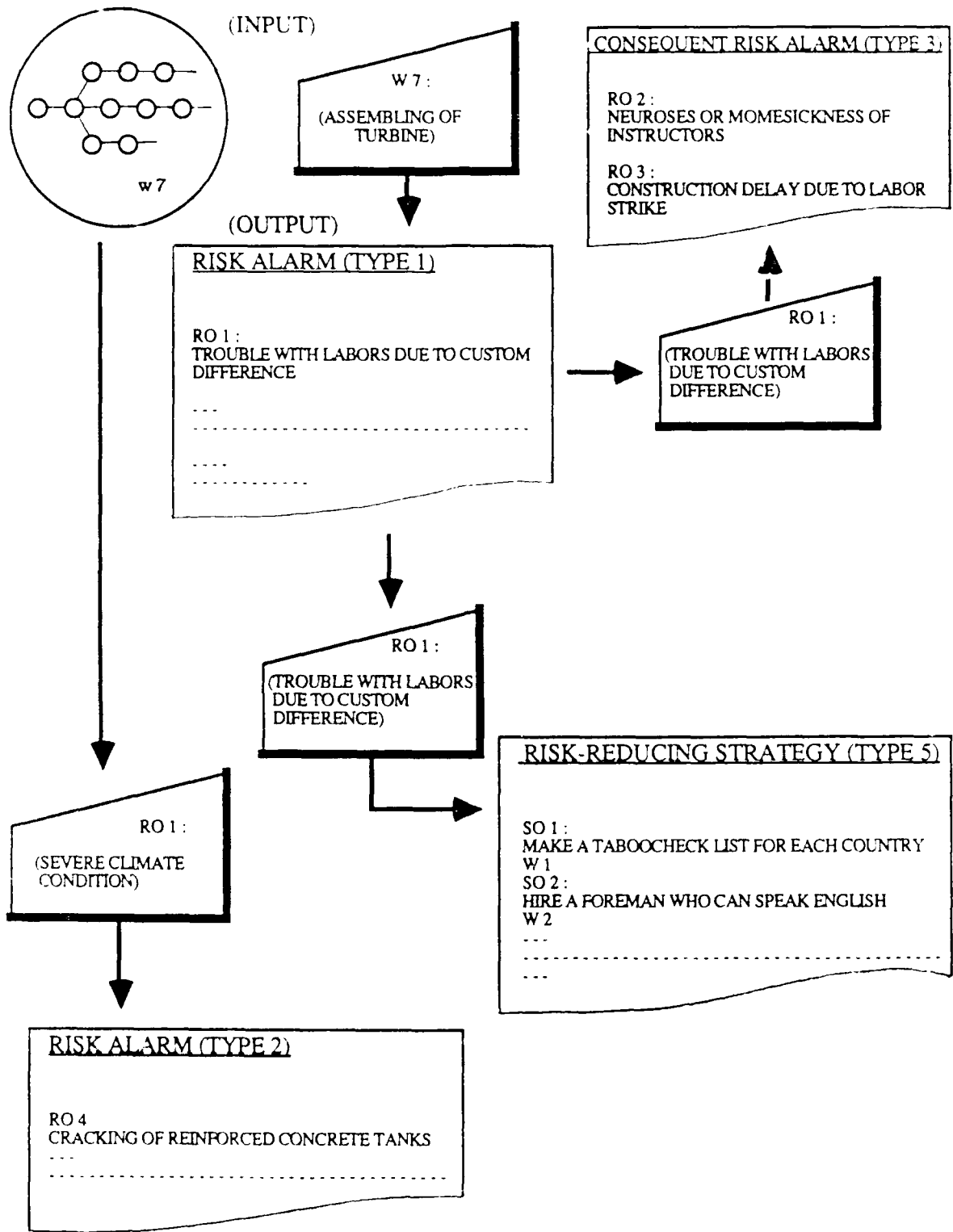


Fig. 4.19 Examples of use of know-how transfer method ES
 (Niwa and Okuma, 1982)

which should be followed for that activity. Another example would be to specify a risk as an input and receive as output the risk factors involved together with other possible risks resulting from the original risk factors.

This knowledge-based risk management system for large project execution was developed at the Advanced Research Laboratory, Hitachi, Ltd. Japan on a Hitachi Computer (HITAC M-200) and this has been in use for over seven years and is the most mature operational expert system in the construction industry.

4.3.5 Microcomputer-Based ES for Safety Evaluation: HOWSAFE

(Levitt, 1986)

Stanford's Construction Engineering and Management Program has been involved in construction safety research since 1969. The inadequacy of knowledge dissemination through journal articles and technical reports to jobsite managers motivated the development of HOWSAFE as a convenient means of knowledge transfer to field construction managers.

Methodology

HOWSAFE is intended as a diagnostic tool to assist the chief executive of a construction firm in determining the adequacy of the firm's safety programs. It is developed and runs on an IBM Personal Computer using The Deciding Factor expert system shell and deals with diagnosis of an organization's structure and operating procedures. The knowledge to be represented in HOWSAFE starts with a top-level hypothesis, "This construction firm has

the required organization and procedures to promote safe construction". A series of intermediate goals such as "Top management truly cares about safety", "Managers at each level are held accountable for the safety of all of their subordinates", etc. lead to the inference of the top-level hypothesis. Each of these intermediate goals is then itself treated as an hypothesis with lower level evidence to determine its truth value. The knowledge is structured like an inverted tree, with the top-level diagnosis on the top supported by lower level inferences, whose validity can be evaluated by the user at the bottom end of each branch. This approach to structuring knowledge is essentially equivalent to a production rule system with certainty factors in which rules should be organized hierarchically. Fig. 4.20 shows a portion of the inference net for HOWSAFE.

The Deciding Factor provides the control structure with backward chaining. KILL Values and CONDITIONAL Logic which are extensively used in HOWSAFE permit the system to be tailored so that the user's responses are sought only when needed and consultations have an easy and logical flow. The Deciding Factor has an attractive feature which permits a user to backtrack in a consultation and change a response previously entered. Starting from the top level hypothesis, the program attempts to satisfy the first goal at the next level. It then chains down, through the first piece of evidence listed at each level, to the bottom or "leaf nodes" of the tree which have no supporting evidence from which their belief can be inferred. This form of knowledge representation was derived from the PROSPECTOR expert system

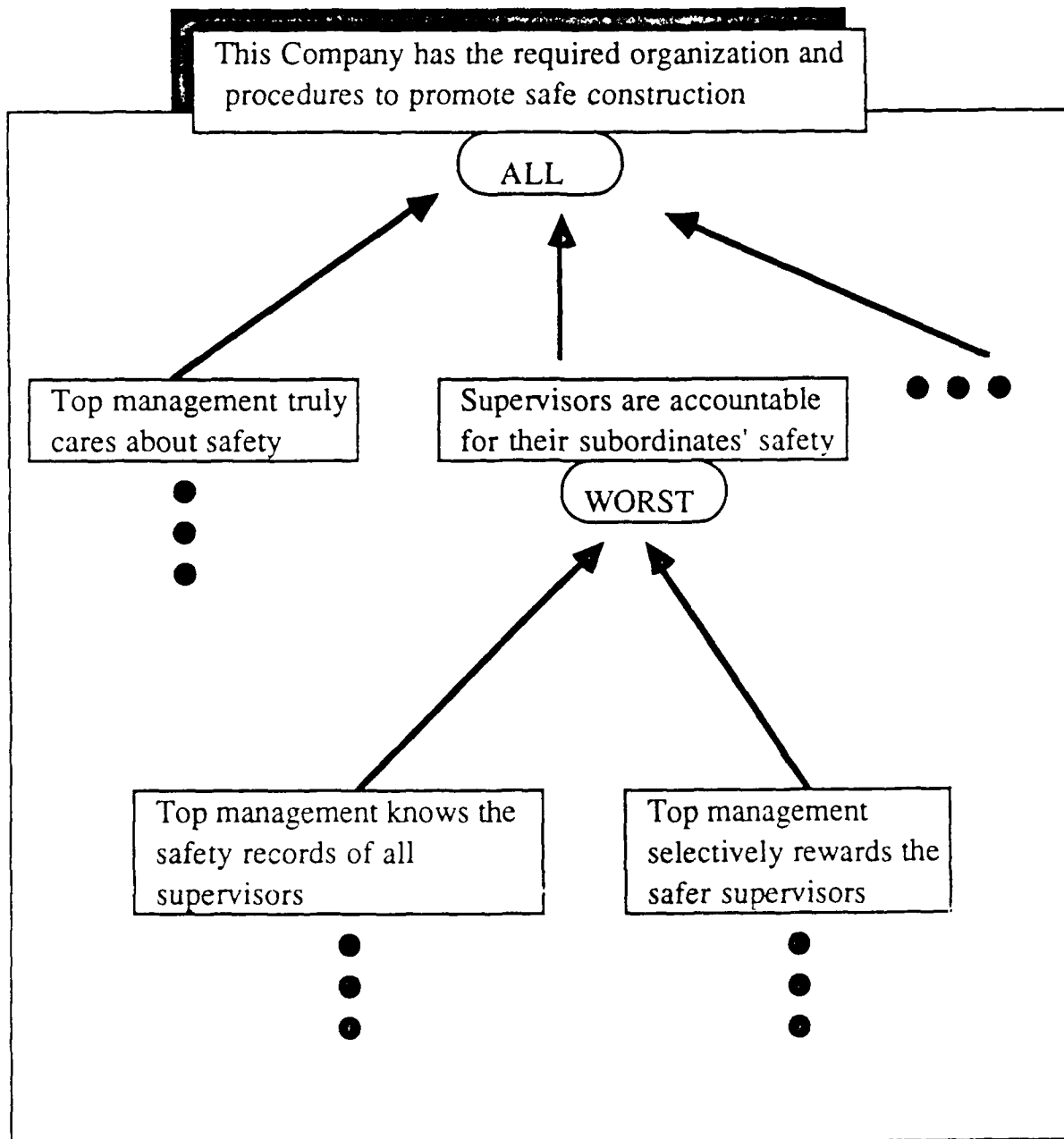


Fig. 4.20 A portion of the HOWSAFE inference net
(Levitt, 1986)

developed at SRI in the mid-1970s. A final degree of belief in the top level hypothesis is reached by combining and weighting the user's responses to leaf node questions.

HOWSAFE has undergone limited external validation and is an operation prototype expert system. A comparison package SAFEQUAL underwent field testing resulting in some minor refinements and is an operational expert system.

SAFEQUAL, also developed using the The Deciding Factor aids construction managers to select contractors based upon their past safety performance and current safety management practices.

4.3.6 Construction Scheduling Knowledge Representation: CONSAES (O'Connor, De La Garza and Ibbs, 1986 De La Giarza and Ibbs, 1987, and Adeli, 1983)

4.3.6.1 Introduction

Construction scheduling together with estimation, cost-control and quality assurance is an essential ingredient of effective project control. The delivery of a completed facility on time is often more important to a client than cost, especially for revenue-generating projects. One of the primary concerns of the present-day claims-conscious construction industry is the ability to forecast the likelihood of project disputes and analyze their origins to assign liability . The U. S. Army Corps of Engineers is very keen in the development of an ES that will aid Army resident engineers to forecast construction schedule variations, the reasons for those deviations and the parties

responsible. Under a multi-year research contract, the University of Illinois Construction Engineering Expert Systems Laboratory (CEESL) and the Corps' Construction Engineering Research Laboratory (CERL) are working in collaboration to develop a PC-based ES for analysis of construction schedules.

The development of a knowledge based ES for construction scheduling is an evolutionary process. The knowledge architecture schemes of semantic net, frames and object-oriented programming provided significant improvements in the representation of heuristic information. With further progress in research, a general knowledge categorization scheme has been developed to divide scheduling analysis and evaluation into two areas, viz. an Initial scheduling analysis module and an In-Progress scheduling analysis module. Fig. 4.21 represents the knowledge structure with Initial and In-Progress scheduling analysis modules based upon major subcategories: i) cost, ii) time, iii) logic, and iv) general requirements. The Initial schedule analysis module provide the type of information that contractors present owners for verification before the commencement of the project. Typical information would comprise of inclusion of owner's approval activities, participation of major subcontractors in the formulation of the plan, etc. The In-Progress scheduling evaluation module allows project managers to examine questions such as delay and duration modification concerns.

4.3.6.2 Methodology

CONSAES (CONstruction Scheduling Analysis Expert System)

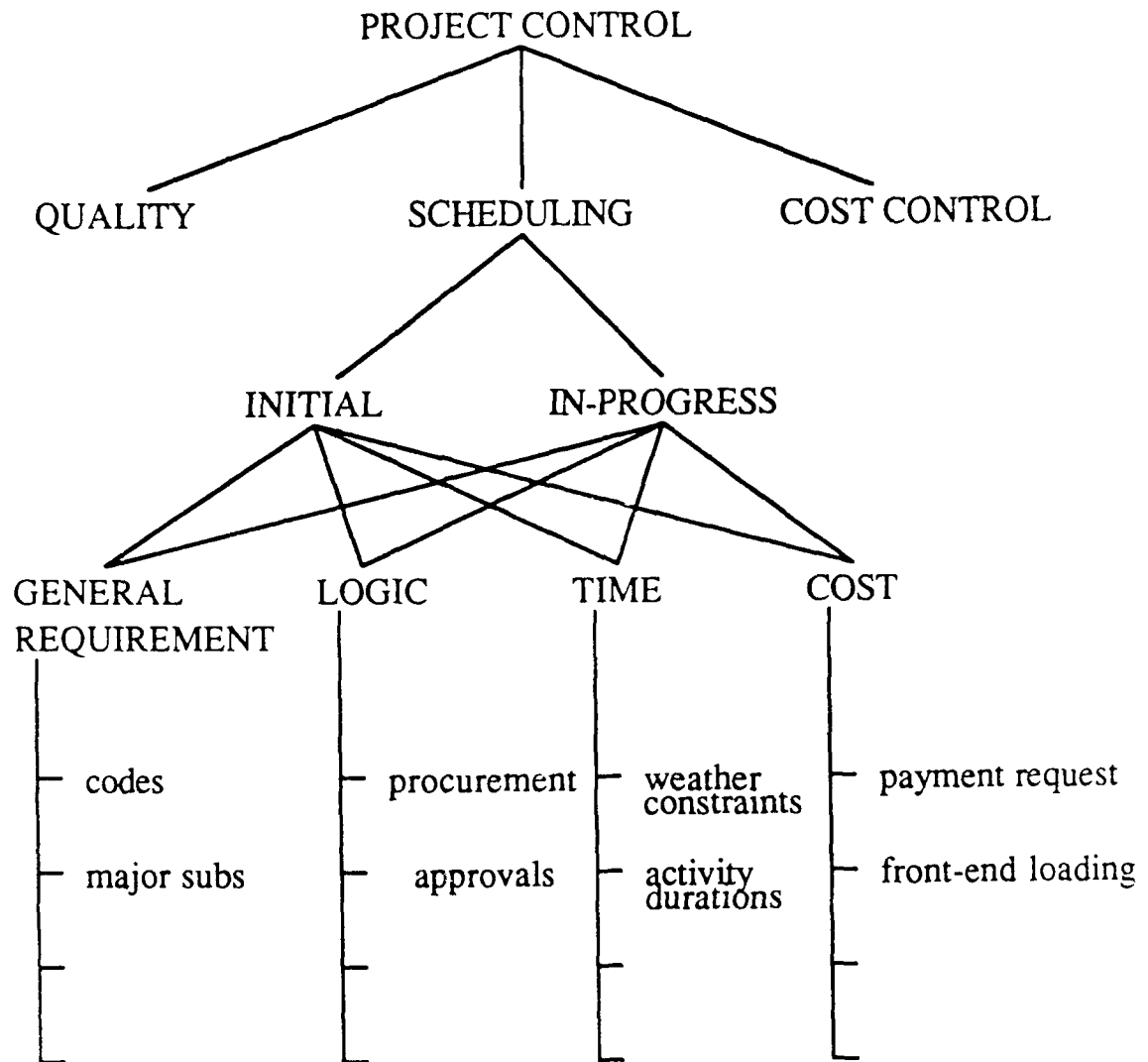


Fig. 4.21 Knowledge structure (O'connor, De La Garza, and Ibbs, 1986)

relies upon existing project control system software to a) identify and capture expressions of similar form in the "paper" knowledge base, b) determine the specific target inference engine, c) decide how the "paper" knowledge base is to be represented in the inference engine and d) develop a mapping technique to adapt the concepts, facts, and rules to the corresponding engine syntax.

4.3.6.3 Knowledge organization

As the "paper" knowledge base became larger, it exhibited some regularity (i.e. expressions of similar form frequently reappeared). These regularities were then captured by building an English-like knowledge acquisition grammar. The facts, rules, and concepts of the construction schedule analysis domain are expressed using this grammar. For example, the syntax for the rule and condition categories is:

```
<rule>           ::= IF <conditions> THEN <conclusions>
<condition>      ::= <frame> HAS <parameter> OF <value>
<condition>      ::= <frame> IS IN CLASS <frame>
```

As a specific example, RULE-111 within the Look-Ahead rule group can be represented by the following English and English-like grammars:

"Paper" knowledge base format:

```
Make projections based on what has happened versus
what was planned.
```

Knowledge acquisition format:

```
IF  ((?some-activity IS IN CLASS activities) AND
      (?Some-activity IS IN CLASS concrete) AND
      (?some-activity HAS status OF finished / in-progress) AND
      (?some-activity HAS assessment of slow-progress) AND
      (concrete HAS lagged OF ( > 5 )))
```

```
THEN ((?activities IS IN CLASS activities) AND
      (?activities IS IN CLASS concrete) AND
      (?activities HAS status OF unfinished) AND
      (set (?activities HAS new-duration OF (* old delay ))))
```

Previous job experience with a particular class of work activities is surveyed for a realistic delay factor. If found, that modifier is then related to all subsequent activities in that class to develop a new anticipated schedule duration. Fig. 4.22 shows the evolution of the knowledge formalization and the advantage of utilizing this generic, intermediate knowledge representation language as a gateway.

4.3.6.4 Knowledge representation

The Automated Reasoning Tool (ART) TM programming environment has been selected as the inference engine to process the knowledge base. It develops "hypothetical worlds" using the technique for generating, representing and evaluating static/dynamic alternatives.

Object-oriented programming provides the facilities, e.g., objects, to structure information which describes a physical item, a concept, or an activity. Each object is represented as a frame, containing declarative, procedural, and structural information associated with the project. A collection of facts representing an object or class of objects having same properties constitutes a frame. Using the object-oriented programming feature, ART permits information of common nature to be stored declaratively in the frames, where it is easily accessible and modifiable.

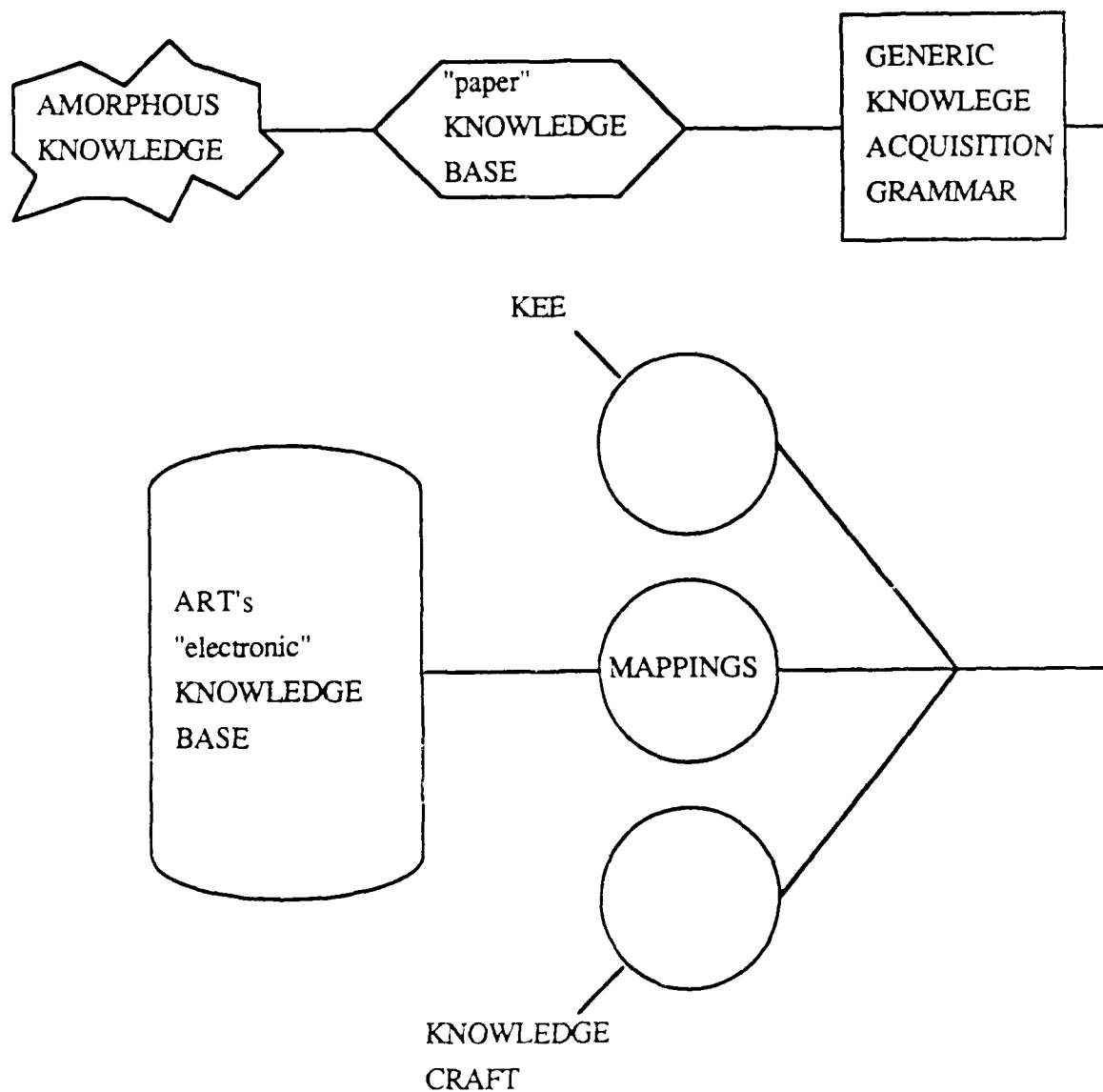


Fig. 4.22 Knowledge metamorphosis (O'connor, De La Garza, and Ibbs, 1986)

4.3.6.5 Knowledge implementation

During the construction planning phase, a work analysis structure is defined based on project phases, goals and organization. Traditionally, milestone descriptions and codes are defined in such a way that they denote both a building and a construction process, e.g., "cast in place 2nd floor slab". Fig. 4.23 illustrate the hierarchical relationship as well as the inheritance path of a typical milestone. The inclusion of one or more relations in a scheme serves to establish it as a node in a hierarchy. The arrows shown in the diagram have significance in that they originate with the object being defined.

A semantic interpretation of every milestone in the construction schedule is provided by CONSAES semantic network. For example, when an activity like "cast in place 2nd floor slab" is found in a schedule, CONSAES immediately deduces a series of facts and compilations about it. This activity for example, i) contains all basic schedule parameters, (e.g., early start, percent complete, etc.) ii) represents a slab in the superstructure, iii) is made of cast in place concrete, iv) consists of formwork, reinforcing steel, and concrete placing, curing and stripping, v) is sensitive to cold temperatures, snow, rain, labor productivity, etc., and so forth.

4.3.6.6 Mappings

A mapping technique adapted to met ART's specifications relates the English-like knowledge acquisition grammar with ART's knowledge representation language. A different mapping technique

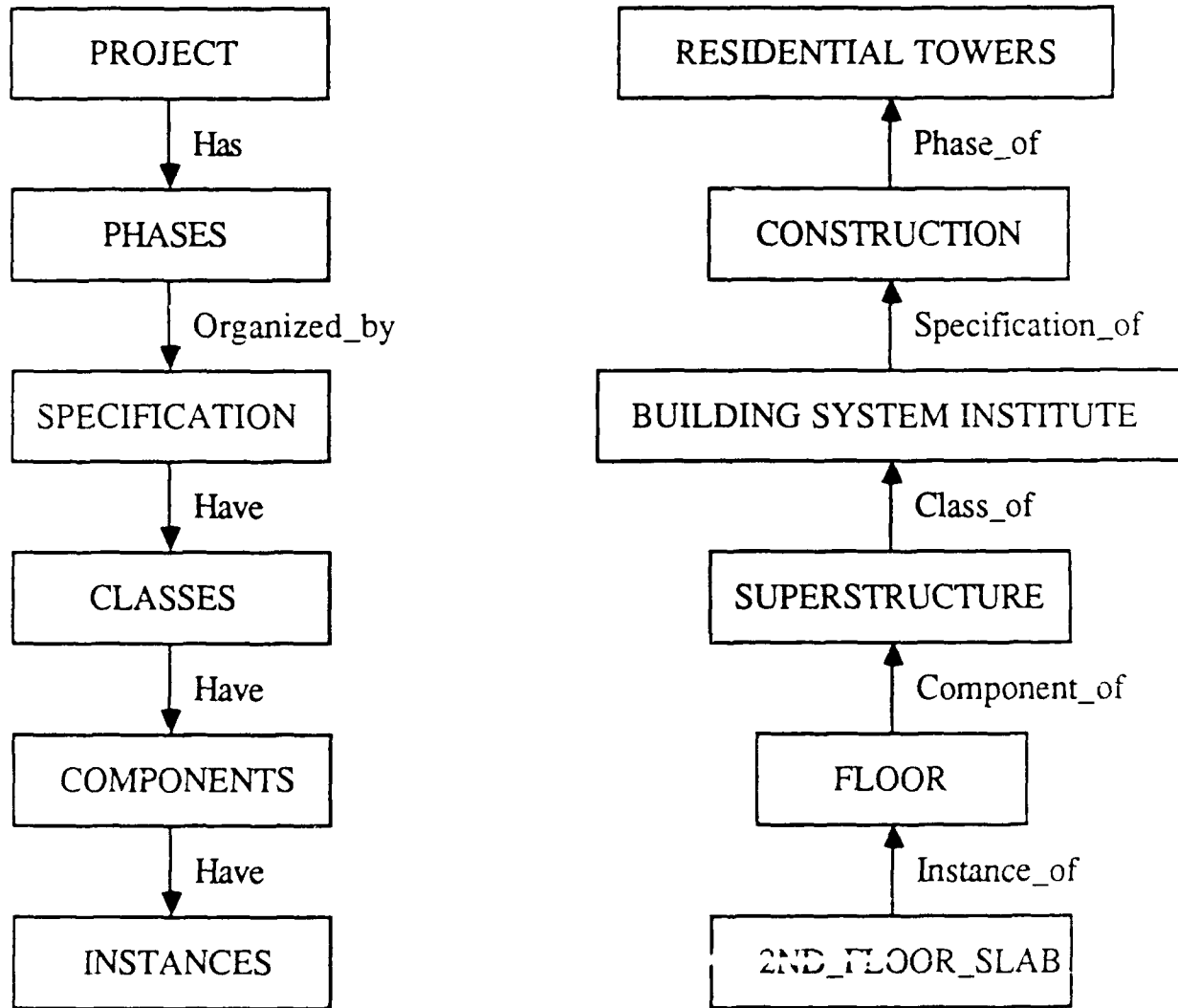


Fig. 4.23 Knowledge base taxonomy (O'connor, De La Garza, and Ibbs, 1986)

needs to be designed for every different inference engine, e.g., ART, KEE, Knowledge Craft (other proprietary, trademarked systems)

4.3.6.7 Relevance of CONSAES

The prototype development demonstrates that this new approach is satisfactory for accelerating and improving the current analyses and computations typical of routine scheduling. CONSAES identifies and organizes the knowledge (analytic and heuristic) useful to the construction engineers to schedule analysis and project management. This ES is ideal for a body of knowledge like scheduling which is partly quantitative and partly subjective.

4.4 EXPERT SYSTEMS IN MAINTENANCE

4.4.1 Introduction

The mechanical equipment maintenance is a critical function in the operation of plants and facilities. With the component failure or malfunction, the consequences could be quite serious and hence evaluations are made regarding the conditions and operating performance of machinery on a periodic time interval. Many variables (pressure, temperature, flow rates, etc.) vibration measurements, and other relevant information are determined and the interpretation of the data requires expertise. The maintenance personnel in the plant are often not experienced enough to interpret the symptoms of problems and determine a remedial course of action. Expert systems are developed to help the less experienced people to resolve problems with

malfunctioning or failed equipment.

4.4.2 Centrifugal Pump Failure Diagnosis: PUMP PRO TM (Finn and Reinschmidt, 1986)

Reliable operation of pumps at most power and process plants is critical. Correction of pump failures often necessitates the use of expensive and time consuming consultants. Although PUMP PRO TM is principally intended to diagnose pump failures at power and process plants, it can also be used to diagnose pump problems by on-site personnel during the start-up phase. Mechanics, technicians, etc. can avail themselves of expert knowledge in the program without the necessity of calling in a human expert consultant.

Methodology

The program is written in MAIDS TM, Microcomputer Artificial Intelligence Diagnostic Service which is a proprietary expert system shell developed at Stone and Webster Engineering Corporation (SWEC). The MAIDS inference mechanism is a forward-chaining, rule-based program that uses a subset of the English language for representing rules. The program has two modules, a rule compiler and an execution module.

The operation of the program is separated into four major phases:

- i) Identification of the symptoms: This is accomplished by means of the MAIDS TM user interface, which consists of text displays, and a question/answer input format. A typical question and associated text display are illustrated in Fig. 4.24.

ii) Identification of the causes: The program uses its forward-chaining inference procedure to apply the heuristic rules to the observed symptoms for identification of the causes.

iii) Provision of tutorials: The program includes a series of optional tutorials, aimed at helping the user understand terminology and procedures. These tutorials are invoked at the user's request, so that users who are familiar with the terminology may proceed directly with the program. In this manner, different levels of user groups can be accommodated without compromising the efficiency or accuracy of the program's operation.

iv) Suggestion of remedies: After identification of probable causes, the program will instruct the user on appropriate remedial action. If the solution of the problem is beyond the user's capabilities, he will be advised to call in a technical specialist.

PUMP PRO TM diagnoses problems by means of twenty-two possible symptom classes and a summarized pump history. It allows input of multiple symptoms and provides seven extensive tutorials and many minor tutorials with approximately three hundred and fifty problem identification rules. A total of approximately seventy rules deal with appropriate remedial strategies and actions. Fig. 4.25 illustrates an example rule using the MAIDS English-like format extracted from PUMP PRO.

PUMP PRO is a mature operational system and is one of a family of similar systems offered by SWEC, accessible by modem using an IBM-PC class computer. Users are assessed a charge based on connect time to the SWEC IBM-AT computer in Boston,

Stone and Webster Engineering Corporation
P U M P PRO (w/monitor)
Microcomputer Artificial Intelligence Diagnostic Service

WHICH OF THE FOLLOWNG DESCRIBE THE PUMP CAPACITY

1. PUMP CAPACITY IS ZERO
2. PUMP CAPACITY IS INADEQUATE
3. PUMP CAPACITY IS ADEQUATE

ENTER THE NUMBER CORRESPONDING TO YOUR CHOICE ? 3

Fig. 4.24 Typical question and associated text display in pump failure diagnosis (Finn and Reinschmidt, 1986)

```
BEGIN RULE
CATEGORY      : 16
AUTHOR        : T.J.FRITSCH
DATE          : 3-29-1985
REASON        : EMPIRICAL
CONDITIONS    : PUMPED LIQUID IS CLEAN
ACTIONS       : CLEAR SCREEN
                DISPLAY BLOCK TEXT
                'CHECK SHAFT SLEEVES AT PACKING
                END BLOCK TEXT
                ASK IS SHAFT/SHAFT SLEEVE WORN
END RULE
```

Fig. 4.25 Example rule using the MAIDS English-like format
(Finn and Reinschmidt, 1986)

Massachusetts. The present configuration for on-line access by users enable the user's PC act as a terminal to SWEC'S IBM PC AT, hosting both the expert system shell and the knowledge bases. Communication through the modem makes the program run rather slowly, especially with the large quantity of text which this system must send to the user's screen. The concept of a large consulting firm acting as a dial-up "knowledge utility" for many kinds of routine consulting services is unique and challenging.

4.4.3 Vibration Analysis Interpretation

The process of diagnosing problems in rotating machinery is dependent, to a large extent, on two factors: i) the data required in order to make a diagnosis and ii) the expertise of the diagnostician in interpreting the data. Vibration monitoring and measuring is an important art in routine maintenance. Experts in this field can identify causes of vibration after examination of very few typical data. This program was developed at SWEC in order to improve the performance of engineers who are assigned the task of vibration diagnosis.

Methodology

The program is an operational ES, which is developed by Stone and Webster Engineering Corporation (SWEC) using the expert system shell EXSYS. It is designed to run on standard, IBM-PC class microcomputers. The inference mechanism uses subroutines for the purpose of analyzing the output of a data collection device and for presenting graphic displays of the analysis

results. A VAX-based version has also been implemented, using the inference mechanism installed on the SWEC VAX. The program operates in an interactive question and answer format and acquires most of its required information from the user, or from the output of its own frequency analysis software. The system is rule based, containing over one hundred rules and is able to diagnose eighteen separate causes of vibration. The program presents the user with a ranked list of probable causes of vibration and provides fairly detailed explanations of each.

4.4.4 Field Diagnosis of Welding Defects (Finn and Reinschmidt, 1986)

Welding defects which are common on most construction sites can drastically impair construction schedules and escalate project costs. Weld repairs are extremely expensive and in certain cases can have more adverse effects than the defect itself. SWEC has developed an ES to identify the causes of defects and recommend procedures for ensuring welds free from defects. This interactive system allows field personnel, welders, supervisors, or quality control personnel to determine probable causes of weld defects. The program takes into account different welding procedures, code requirements, site conditions and observations. It enable more rapid repair of welding defects, thus reducing repair costs.

Methodology

The system is an operational ES and requires the welding supervisor to answer specific questions about observations made

at the site of the weld, the condition of the materials and the environment and details about the welding procedure employed. The system uses a backward-chaining mechanism to reason about likely causes of the defects. A ranked list of possible factors responsible for the defect is presented to the user together with methods for improving the welding operation.

Parts of the program are implemented, while other modules are still under development. The weld diagnosis program is written using the expert system shell EXSYS for use on an IBM-PC class of microcomputer.

4.4.5 ES for Concrete Pavement Evaluation and Rehabilitation (Hall, Connor, Darter, and Carpenter, 1988)

Concrete pavement evaluation and rehabilitation is a complex engineering problem in view of the large number of interacting factors and the lack of adequate analytical models to solve all aspects of the problem. Successful concrete pavement evaluation and rehabilitation currently relies heavily on the knowledge and experience of authorities in the pavement field for diagnosis of the causes of distress and selection of feasible rehabilitation techniques which cost-effectively correct the deterioration. A practical and comprehensive ES has been developed to assist practicing engineers in concrete pavement evaluation and rehabilitation; it uses a new and innovative approach that combines human knowledge and analytical techniques into a user-friendly personal computer program.

Methodology

The ES consists of computer programs, one for each of three concrete pavement types-jointed reinforced concrete (JRCP), jointed plain concrete (JPCP), and continuously reinforced concrete (CRCP). The following are the steps in evaluation and rehabilitation design:

i) Project data collection: The engineer collects key inventory (office) and monitoring (field) data for the project. Inventory data including design, traffic, materials, soils and climate and monitoring data consisting of distress, drainage characteristics, rideability and other items collected during a field visit to the project are entered into a personal computer using a full-screen editor. The overall condition of the project is extrapolated by the system from the sample unit monitoring data.

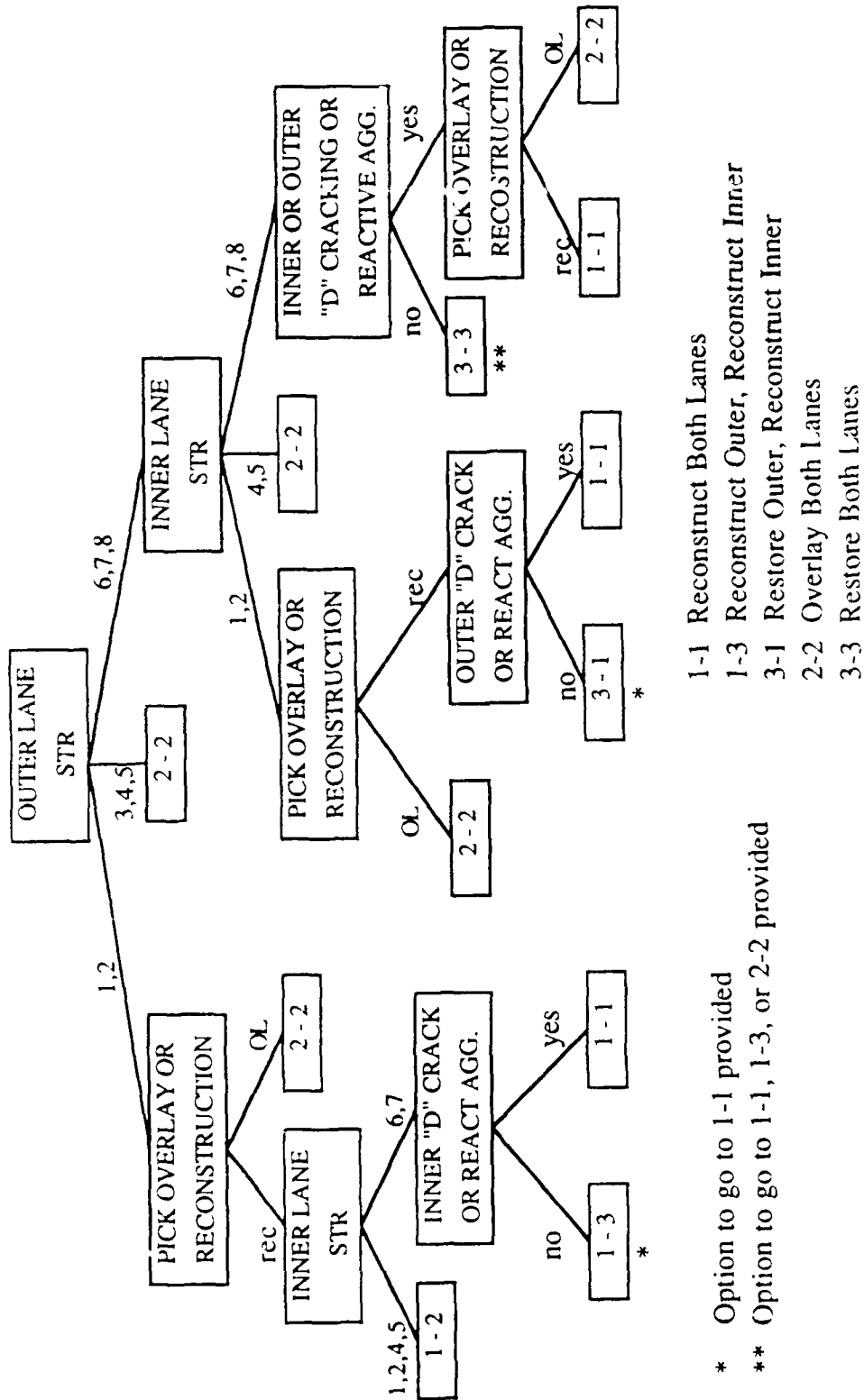
ii) Evaluation of present condition: All the data are analyzed using the evaluation decision trees and major problem areas including roughness, structural adequacy, drainage, foundation stability, concrete durability, skid resistance and shoulders are identified and evaluated; five additional problem areas consisting of transverse and longitudinal joint construction, transverse joint sealant condition, loss of support, load transfer and joint deterioration are evaluated in the case of JRC and JPC pavements. Two additional problem areas, viz. longitudinal joint construction and construction joints/terminal treatments are evaluated in the case of CRC pavements.

iii) Prediction of future condition without rehabilitation: The condition of pavement for twenty years into the future is projected by means of predictive models, the current traffic level and the anticipated growth rate. Performance prediction is carried out in terms of serviceability and distress types, viz. faulting, cracking, joint deterioration, and failures (punchouts, steel ruptures, and full-depth repairs) for CRCP.

iv) Physical testing: The system recommends specific physical tests to verify the evaluation recommendations and provide data needed for rehabilitation design. Recommended types of testing include nondestructive deflection testing, destructive testing (coring and boring) and roughness and friction measurement. Certain types of deficiencies viz. structural inadequacy, poor rideability, surface friction, drainage conditions and concrete durability (cracking or reactive aggregate distress), foundation movement (due to swelling soil or frost heave), loss of load transfer at joints, loss of slab support, joint deterioration and evidence of poor joint construction may justify physical testing.

v) Selection of main rehabilitation approach: The most appropriate main rehabilitation approach for each traffic lane and shoulder is selected by the engineer based upon the evaluation results and subsequent interaction with the system. The options consist of reconstruction (including recycling), resurfacing (with concrete or asphalt), or restoration. A decision tree has been developed for each pavement type to assist the engineer in selecting the most suitable rehabilitation approach. Fig. 4.26 shows the decision tree for JPCP.

Main Rehabilitation Approach for JPCP



* Option to go to 1-1 provided
 ** Option to go to 1-1, 1-3, or 2-2 provided

- 1-1 Reconstruct Both Lanes
- 1-3 Reconstruct Outer, Reconstruct Inner
- 3-1 Restore Outer, Reconstruct Inner
- 2-2 Overlay Both Lanes
- 3-3 Restore Both Lanes

Fig. 4.26 Decision tree for selection of rehabilitation approach for JPCP
 (Hall, Connor, Darter, and Carpenter, 1988)

vi) Development of detailed rehabilitation strategy: After selection of a suitable rehabilitation approach, the engineer proceeds to develop the detailed rehabilitation alternative for each traffic lane and shoulder by selecting a feasible set of individual rehabilitation techniques to correct the deficiencies present. This may include such items as subdrainage, shoulder repair, full-depth repairs, joint resealing, etc. A set of decision trees has been developed to guide the rehabilitation strategy development process.

vii) Prediction of rehabilitation strategy performance: The future performance of the developed rehabilitation strategy is then predicted in terms of key distress types for twenty years into the future based upon assumed traffic growth. The engineer must evaluate the results and determine whether or not the strategy provides an acceptable life with an optimum cost.

viii) Cost analysis of alternatives: The engineer computes the cost for each item in each rehabilitation technique included in the alternative strategy and determines the total and annual costs for the strategy.

ix) Selection of preferred rehabilitation strategy alternative: The engineer considers the life-cycle cost together with constraints that exist for the project such as traffic control, construction time, available funding, etc. in the selection of the preferred alternative. Based upon estimated initial and annual costs, expected life and performance and various constraints, the user selects the preferred rehabilitation strategy from among the feasible alternatives

available.

The shell used was Insight 2+, developed by Level V Research, Inc. Insight 2+ is a production-rule-based system shell in which knowledge is expressed in terms of "if-then" rules. The decision trees are incorporated into the Insight 2+ shell by programming each path down each tree (a path being composed of a set of nodes and connecting branches terminating at a conclusion as a single rule. The system has been developed in both manual and computerized form. The programs operate on any IBM-compatible personal computer.

4.5 CONCLUDING REMARKS AND FUTURE TRENDS

The extent and breadth of work already completed, under way, or in the early conceptual stages indicates that many researchers and practitioners in the construction industry consider expert systems as offering new and potentially valuable capabilities to support decision making in the industry. The software tools available for building expert system applications in construction have improved dramatically over the last five years. Systems that can run on IBM PC computers offer outstanding ease of use (The Deciding Factor), the capability to interface with external data and programs (Insight+), and even support of frames (Personal Consultant Plus).

Future research and development of expert systems in construction will involve hybrid systems combining expert systems with database management systems and computational systems. The use of expert systems for integrating between design and construction decision making is likely to be one of the areas for

fundamental research and development on expert systems in construction. Expert system programming approaches can be used in such hybrid systems to develop individual expert system modules, as well as to communicate between these multiple "knowledge sources" and other expert systems, databases and application programs. Expert systems in construction can be interfaced with CAD systems which can attach nongraphical attributes to their graphical objects. The areas of diagnostics for inspection, maintenance and repair appears to be promising where small expert systems could be developed for use in desktop or portable personal computers.

REFERENCES

1. AASHTO (1983) "Standard Specifications for Highway Bridges", 13th edn., American Association of State Highway and Transportation Officials, Washington.
2. Adeli, H. (1988) "Expert Systems in Construction and Structural Engineering", Chapman and Hall, New York.
3. Adeli, H. and Balasubramanyam, K. V. (1988), "A Knowledge-Based System for Design of Bridge Trusses", J. Computing in Civil Engineering, Vol. 2, No. 1, pp. 1-20.
4. Allwood, R. J., Stewart, D.J., and Trimble, E.G. (1985), "Some Experiences From Evaluating Expert System Shell Programs and Some Potential Applications", Proc. Second International Conference on Civil and Structural Engineering Computing, Vol. 2, Civil-Comp. Press, London, pp. 415-420.
5. American Institute of Steel Construction, Inc., (1978) Manual of Steel Construction, 8th Ed., AISC, Chicago, Ill.
6. Ashley, D. B., and Wharry, M. B., (1985), "Prototype Expert System For Subsurface Risk", NSF Grant CEE-8352354, 1985.
7. Bennett, J.S. and Engelmere, R.S. (1979), "SACON: A Knowledge Based Consultant For Structural Analysis", Proc. Sixth International Joint Conference on Artificial Intelligence, Tokyo, pp. 47-49.
8. Biegler, L.T. and Cuthrell, J.E. (1985), "Improved Infeasible Path Optimization For Sequential Modular Simulators: the Optimization Algorithm", Computers and Chemical Engineering 9, No. 1, 257-267.
9. Bowen, J., Cornick, T. C., and Bull, S. P. (1986), "BERT- An Expert System for Brickwork Design", working paper, University of Reading, England, Departments of Computer Science and Construction Management.
10. Building Officials and Code Administrators International, Inc. (1984), "The BOCA Basic Building Code, 9th Ed., Country Club Hills, Ill.
11. Camacho, G. (1985), "LOW-RISE: An Expert System For Structural Planning and Design of Industrial Buildings", M.S. thesis, Department of Civil Engineering, Carnegie Mellon University, Pittsburgh, PA.
12. De La Garza, J. M. and Ibbs, C. W., (1987), "Issues in Construction Scheduling Knowledge Representation", Managing Construction Worldwide, Volume 1, Systems for Managing Construction, Ed. Lansley, P. R. and Harlow, P. R., E & F. N. Spon, New York., pp. 543-553.

13. Fenves, S.J. and Norabhoompipat, T. (1978), "Potential For Artificial Intelligence Applications In Structural Design And Detailing", Artificial Intelligence and Pattern Recognition in Computer-Aided Design (ed J.D. Latombe), North Holland, Amsterdam.
14. Fenves, S.J. (1980), "Software For Analysis of Standards", Proc. Second Conference on Computing in Civil Engineering, ASCE, Baltimore, Md., June, pp. 82-91.
15. Fenves, S.J., Maher, M.L. and Sriram, D. (1984), "Expert Systems: C.E. Potential", Civil Engineering/ASCE, October '84', pp. 45-47.
16. Fenves, S.J. (1986) "What Is An Expert System", Proc. of Symposium on Expert Systems in Civil Engineering, ASCE Convention, Seattle, Washington, pp. 1-6.
17. Finn, G. A. and Reinschmidt, K. F. (1986), "Expert Systems in an Engineering - Construction Firm", Expert Systems in Civil Engineering, Proc. Symposium, ASCE, Seattle, Washington, Ed. Kostem, C.N. and Maher, M. L., pp. 40-54.
18. Firlej, M. (1985) "Training and Education for Knowledge Engineers", Proceedings of 1st International Expert Systems Conference, London, October 1-3, pp. 209-216.
19. Forgy, C.L. (1981) OPS5 User's Manual, Technical Report, CMU-CS-81-135, Carnegie-Mellon University.
20. Garrett, Jr., J.H. (1986) "SPEX- A Knowledge-Based Standard Processor For Structural Component Design", Ph.D. thesis, Carnegie-Mellon University.
21. Gaschnig, J, Reboh, R. and Reiter, J. (1981), "Development of Knowledge-Based System for Water Resources Problems", SRI Project 1619, SRI International.
22. Gero, J.S. and Coyne, R.D. (1986), "Developments in Expert Systems For Design Synthesis", Expert Systems in Civil Engineering, ASCE, New York.
23. Hamiani, A. and Popesen, C., (1988), "CONSIT: A Knowledge Based Expert System for Site Layout", Proceedings of the Fifth Conference, Computing in Civil Engineering: Microcomputers, ASCE, Ed. K. M. Will, pp. 248-256.
24. Harris, J.R. and Wright, R.N. (1980), "Computer Aids For The Organization of Standards", Proc. Second Conference On Computing in Civil Engineering, ASCE, Baltimore, Md., June, pp. 92-101.
25. Hendrickson, C. T., Zozaya - Gorostiza, C. A., Rehak, D., Baracco-Miller, E. G. and Lim, P. S. (1987), "An Expert

System for Construction Planning," ASCE Journal of Computing, October, 1987.

25. Howard, H.C. (1986), "Interfacing Databases and Knowledge Based Systems For Structural Engineering Applications", Ph.D. thesis, Carnegie-Mellon University.
27. Hutchinson, P. (1985), "An Expert System For The Selection Of Earth Retaining Structures", M.S. Thesis, Department of Architectural Science, University of Sydney, Australia.
28. IBM (1984), "Graphical Data Display Manager: Application Programming Guide", Program 5748- XXH, Release 4, 3rd edn., IBM Corporation, Cary, North Carolina.
29. Kostem, C.N. (1986), "Design Of An Expert System For The Rating of Highway Bridges", Expert Systems in Civil Engineering, ASCE, New York.
30. Kunz, J. C., Bonura, T. Stezlner, M. J. (1986), "Applications of Artificial Intelligence in Engineering Problems", Proc. First International Conference, Southampton University, U. K. Vol. II, Ed. Sriram, D., and Adey, R. A. Computational Mechanics Publication, Springer-Verlag, New York.
31. Levitt, R. E., (1986 , "HOWSAFE: A Microcomputer-Based Expert System to Evaluate the Safety of a Construction Firm", Expert Systems in Civil Engineering, Proc. Symp. Sponsored by the Tech. Council on Computer Practices of ASCE, Seattle, Washington, Ed. Kostem C. N. and Maher, M. L., pp. 55-66.
32. Lin, T.Y. and Stotesbury, S.D., (1981), "Structural Concepts and Systems For Architects and Engineers", John Wiley, New York.
33. Maher, M.L. (1984), "HI-RISE: A Knowledge-Based Expert System For The Preliminary Design of High Rise Buildings", Ph.D Thesis presented to Carnegie-Mellon University, Pittsburgh, PA.
34. Maher, M.L. (1986), "Problem Solving Using Expert System Techniques", Expert Systems in Civil Engineering, ASCE, pp. 7-17.
35. Maher, M.L. (1987), "Expert System Components", ASCE Expert Systems Committee Report on Expert Systems for Civil Engineers: Technology and Application.
36. Niwa, K. and Okuma, M., (1982), "Know-How Transfer Method and its Application to Risk Management for Large Construction Projects", IEEE Transactions on Engineering

Management, Vol. EM-29, No. 4, pp. 146-153.

37. O'Connor, M.J., De La Garza, J. M. and Ibbs, Jr. C. W., (1986), "An Expert System for Construction Schedule Analysis", Expert Systems in Civil Engineering, Proc. Symp. on Computer Practices of ASCE, ASCE convention, Seattle, Washington Ed. Kostem, C. N. and Maher, M. L., pp. 67-76.
38. Paulson, B. C. Jr. and Sotoodeh - Khoo, H. (1987), "Expert Systems in Real-Time Construction Operations", Managing Construction Worldwide, Vol. 1, Systems for Managing Construction, Ed. Lansley, P. R. and Harlow, P. A., E & F N. Spon Ltd.
39. Rasdorf, W.J. and Fenves, S.J. (1980), "Design Specifications Representation and Analysis", Proc. Second Conference on Computing in Civil Engineering, ASCE, Baltimore, Md., June, pp. 102-111.
40. Rasdorf, W.J. and Wang, T. (1986), "Expert System Integrity Maintenance For The Retrieval of Data From Engineering Databases", Proc. Fourth Conference on Computing in Civil Engineering, ASCE, Boston, Mass., October, pp. 654-668.
41. Rasdorf, W.J. and Wang, T.E. (1988), "Generic Design Standards Processing In An Expert System Environment", Journal of Computing in Civil Engineering, Vol. 2, No. 1, January. pp. 68-87.
42. Reboh, R. (1981), "A Knowledge Engineering Techniques and Tools In the PROSPECTOR Environment", SRI Technical Note 243, Stanford Research Institute, Menlo Park, California.
43. Reddy, D.R., Erman, L.D. and Neely, R.B. (1973), "The HEARSAY Speech Understanding System: An Example of the Recognition Process, Proc. International Joint Conf. on AI, Tokyo, pp. 185-193.
44. Rehak, D. R. and Fenves, S. J. (1984), "Expert Systems in Construction", Proceedings of the 1984 International Computers in Engineering Conference and Exhibit, Las Vegas, Nevada, ASME, Vol. I, Ed. Gruver, W. A. pp. 228-235.
45. Rooney, M. and Smith, S.E. (1983), "Artificial Intelligence In Engineering Design", Computers and Structures, Vol. 16, pp. 279-88.
46. Russell, J. S. and Skibniewski, M. J., (1988), "An Expert System for Contractor Prequalification", Proceedings of the Fifth Conference, Computing in Civil Engineering: Microcomputers to Supercomputers, ASCE, Ed. K. M. Will, pp. 239-247.
47. Saouma, V. E., Jones, M. S., and Doshi, S. M. (1987), "A PC Based Expert System for Automated Reinforced Concrete Design

- Checking", Report of Contract No. DACA39-86-K-0011, Information Technology Laboratory, US Army Engineer Waterways Experiment Center, Vicksburg, Mississippi
48. Stefik, M. (1980), "Planning With Constraints", Technical Report STAN-CS-80-784, Computer Science Department, Stanford University.
 49. Teknowledge, Inc. (1985a), M.1 training materials, M.1 reference manual, version 1.3, Teknowledge, Inc., Palo Alto, California.
 50. Terzaghi, K. and Peck, R. B. (1967), Soil Mechanics in Engineering Practice, Wiley, New York
 51. Tommelein, I. D. Levitt, R. E. and Hayes-Roth, B., (1987), "Using Expert Systems for the Layout of Temporary Facilities on Construction Sites", Managing Construction Worldwide, Vol. 1, Systems for Managing Construction, Ed. Lansley, P.R. and Harlow, P. A., E & F N. Spoon Ltd, New York, pp. 566-577.
 52. VanMelle, W. (1979), "A Domain Independent Production-Rule System For Consultation Programs", Proc. Sixth IJCAI, Aug., pp. 923-925.
 53. VanMelle, W. (1979), "A Domain-Dependent Production Rule System For Consultation Programs", Proc. Sixth International Joint Conference on Artificial Intelligence, Tokyo, Japan, pp. 923-925.
 54. Wang, T. and Rasdorf, W.J. (1985) Query Watcher User's Guide, Version 1.0, Computer Studies Program, North Carolina State University, Raleigh, North Carolina.
 55. Wang, T. (1986), Generic Design Standards Processing In A Knowledge-Based Expert System Environment", M.S. thesis presented to North Carolina State University at Raleigh, North Carolina.
 56. Wang, T. and Rasdorf, W.J. (1986), SPIKE User's Guide, Version 1.0, Computer Studies Program, North Carolina State University, Raleigh, North Carolina.
 57. Welch, J., and Biswas, M. (1986), "Application Of Expert Systems In The Design of Bridges", Technical Report.
 58. Zozaya-Gorostiza, C., Hendrickson, C., Baracco, E., and Lim, P. (1987), "Decisions Tables for Knowledge Representation and Acquisition", Technical Report, Carnegie Mellon University, Department of Civil Engineering, Pittsburgh, PA.