Development Of An Expert System Shell For Engineering Design

by

M. L. Maher, P. Longinos

EDRC-12-05-86[3]

September 1986

Development Of An Expert System Shell For Engineering Design

by
Mary Lou Maher[1] and Panayiotis Longinos[2]
Department of Civil Engineering
Carnegie-Mellon University
Pittsburgh, PA 15213

ABSTRACT

This report describes the development of an expert system shell for the preliminary phase (synthesis) of engineering design. It is during this part of the design process that the creativity and experience of an engineer are mostly needed. The increasing complexity of engineering design problems has made synthesis a very difficult process, even to an experienced designer, if not approached in a structured and organized fashion. The proposed shell adopts principles of the morphological approach to design, incorporating heuristics in the form of design constraints.

---

[1] Assistant Professor

[2] Graduate Assistant

Development Of An Expert System Shell For Engineering Design

Mary Lou Maher[3] and Panayiotis Longinos[4]

Design, a combination of art and science, is perhaps one of the most important and difficult tasks an engineer performs. During the last three decades, substantial activity has grown in studying design in an effort to produce a structured approach to this creative process. As a result, a number of design methodologies have been developed for a wide variety of applications and from the background of differing engineering disciplines. Although strong indications exist of the practical potential of these techniques for the engineering design process, the literature falls short in presenting detailed evidence of successful applications of design methodology.

With the introduction of the computer as a powerful tool for the engineering design process, attention has shifted from pursuing study on design methodology to the development of software to aid engineers in the design process. Traditionally these aids have been limited to the well structured aspects of design such as analysis and graphics. Conventional programming techniques have been unable to automate the less formalized phases and thus adopt an overall design methodology. Today, the intuitive ability of the experienced engineer is still needed to make the decisions guided by the computational results.

Advancements in Artificial Intelligence research and the subsequent emergence of expert systems provide a new powerful tool for the development of computer programs that can be used as aids for the solution of ill-structured phases of the engineering design process. Expert Systems are an ideal environment for studying design methodologies and learning more about the design process.

This report describes the development of an expert system shell for the preliminary phase (synthesis) of engineering design. The proposed shell adopts principles of the morphological approach to design, incorporating heuristics in the form of design constraints.

---

[3] Assistant Professor, Department of Civil Engineering Carnegie-Mellon University Pittsburgh. PA 15213

[4] Graduate Assistant, Department of Civil Engineering Carnegie-Mellon University Pittsburgh, PA 15213

1. Motivation

The preliminary phase of the engineering design process is learned only after years of experience in the field. The development of design methodologies has resulted in a number of promising but unproven techniques in approaching and organizing this unstructured phase of the design process. An expert system environment able to implement such a technique and thus formalize the preliminary design process can be a powerful tool in learning more about the engineering design process. Such a system can also introduce new engineers to the decisions made during the preliminary design process, something that is absent from today's formal engineering education.

2. Engineering Design

Engineering design may be defined as a process in which scientific principles, technical information and creativity are all combined in order to produce an optimum end product which will serve its intended purpose. The engineering design process involves a number of distinctive phases beginning with the definition of a particular problem and ending with the selection of an optimum solution. Various approaches to engineering design have produced different decompositions of this process. Commonly, engineering design is broken into three main phases, as illustrated in Figure 2-1.

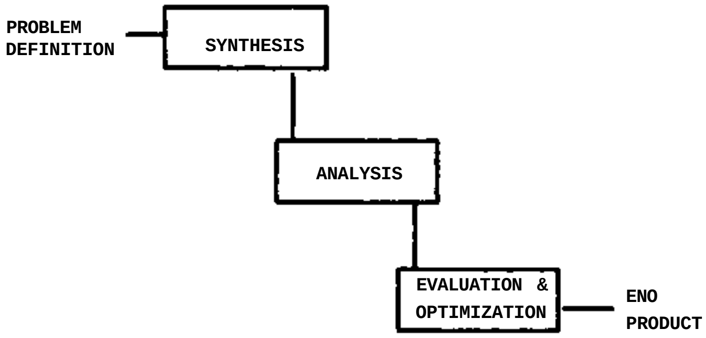


Figure 2-1: Engineering Design Phases

Synthesis (Preliminary design) : An essential feature of all design work, this task deals [illegible] formation of design alternatives. Synthesis involves searching and checking of subsystems [illegible] result of this phase is the selection of one, or at most, a few preliminary design alternatives that [illegible] the key constraints of the particular problem.

Analysis : During this phase a selected design alternative is studied using mathematical arc [illegible] scientific procedures in an effort to determine its response to the intended environment, irrc [illegible] aspects of this phase are: the selection of the proper analysis procedures, the correct use of these procedures and, the appropriate interpretation of the results.

Evaluation & Optimization : This final phase of the engineering design process involves the evaluation of the analyzed designed alternative. At this point, backtracking and repetition of previous design phases is often required to produce a feasible, acceptable or optimal design solution to the specified problem.

The proposed expert system shell addresses the synthesis phase of the design process. Most of the conceptual aspects of engineering design are embodied in this preliminary design phase. At this point, the only information available to the designer are the specifications of the end product. It is during this part of the design process that the creativity and experience of an engineer are mostly needed. The increasing complexity of engineering design problems has made synthesis a very difficult process, even to an experienced designer, if not approached in a structured and organized fashion.

There is no uniformly "best" way in approaching the synthesis process for all designs. A common procedure is to decompose the design problem into the design of independent subsystems. The nature of these subsystems will depend on the nature of the problem at hand. In a similar manner. each subsystem is divided into major components. Alternative candidate designs can be synthesized by considering all possible combinations of the various subsystems that result from combinations of lower level components. This hierarchical approach to the synthesis of a solution enables the designer to consider an exhaustive set of possibilities based upon the manner in which the subsystems and the lower level components are defined. These definitions will depend on the nature of the particular problem as well as the engineer performing the design. A particular problem may justify the decomposition of a problem from an abstract level down to a set of detailed subsystems Another design may be better approached by considering the details first and building up to more general systems. A correct selection of this sequence can increase the efficiency of the synthesis process.

A key consideration in the synthesis of design alternatives is the identification and satisfaction of constraints at the various levels of abstraction. These constraints control the qualification of various components of the design as well as the feasibility of combinations of such components.

The synthesis process can be illustrated through an example of the design of pile foundations The design can be decomposed into four levels of abstraction based on various classification categories

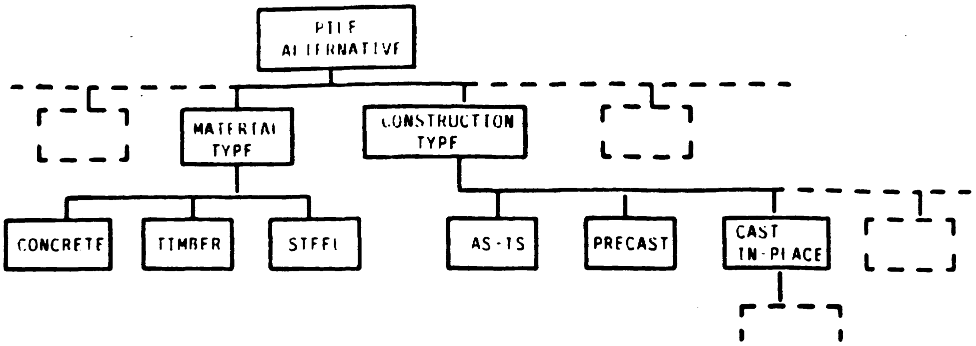


Figure 2-2: Pile Design Example

These categories are: material type, construction type, load-resistance type, and cross-section shape. For this example we consider only two categories as shown in Figure 2-2; material and construction types. Each type has three alternatives; for the material type these alternatives are timber, steel, and concrete and for the construction type the alternatives are cast-in-place, as-is, and precast. Without applying any constraints, nine pile alternatives can be formed but this will include a number of incorrect designs. Introducing the constraint that only concrete can be cast-in-place or precast narrows the solution range and eliminates the incorrect combinations. This type of constraint is inherent to the pile design problem and will always be considered. Another constraint applied to the problem introduces the possibility of obstructions in the soil, a situation that will damage timber piles This constraint is specific for a particular situation and thus more difficult to identify. The identification of both types of constraints is essential to the solution of the pile problem and engineering design in general. It is clear that constraints play an important role in the formation of feasible design alternatives during the synthesis process.

3. Design Methodology

Design methodology can be defined as the science of methods of design. Over the years, a range of specific methodologies have been developed, most of them oriented towards engineering design These techniques are sets of rules, tasks, and procedures for organizing and guiding the design process. As such, design methodologies provide a most useful approach to the design of complex systems, the automation of the design process and the teaching of design. More specifically, design methodologies have been developed to meet the following needs.

Organization of design : for increasingly complex design problems brought about by the rapid technological advancement in the twentieth century. New technologies, production methods and expanding markets have increased the scale and complexity of the design process creating a need for a form of design management.

Teaching of design : based on the fact that formal design education is limited to the analytical aspects of the design process. The more conceptual aspects of engineering design are being taught by example and experience alone. Design methodologies can be used to introduce the general principles of the design process and thus give students a more complete engineering design background.

Designer aids : by providing a more structured approach to design, thus stimulating creativity and increasing the designers' efficiency.

Automation of design : by introducing a method to the design process thus making possible the use of computer technology to automate design.

Of the various applied design methodologies which concentrate on engineering design, one that is relevant to the proposed work is the morphological approach. Morphology is the science associated with the form and structure of a body or system. The morphological approach to the engineering design process involves the visualization of subsystems at various levels of abstraction.

Using this approach, the design process is initiated by visualizing the end product at the highest level of abstraction. This results in widening the possible solution range for a design problem. As the design progresses, the current level of abstraction either has a potential list of solutions or needs to be further decomposed. These abstraction levels are usually referred to as subsystems r*e similarity of the morphological approach to the synthesis phase of engineering design, as disc^sv^ in Section 2, qualifies it as a promising methodology to adapt in this process. Figure 3-1 illustrate •--? decomposition of the system into subsystems, the search for solutions for all the created subs; /^ -s and, finally, the synthesis of subsystem solutions to form design alternatives.

The top-down decomposition followed by the morphological approach is also referred *D IS hierarchical planning. The result of planning is often translated into a chart referred to ii •-•? morphological matrix, shown in Figure 3-2. This table creates a visual representation of the vjr ,s subsystems and their respective solutions in an effort to simplify the process of come ; components to form design alternatives.

As an illustration of the morphological approach, the design of pile foundations is considered. Using this approach, the design process is initiated by visualizing the end-product at the highest [illegible]

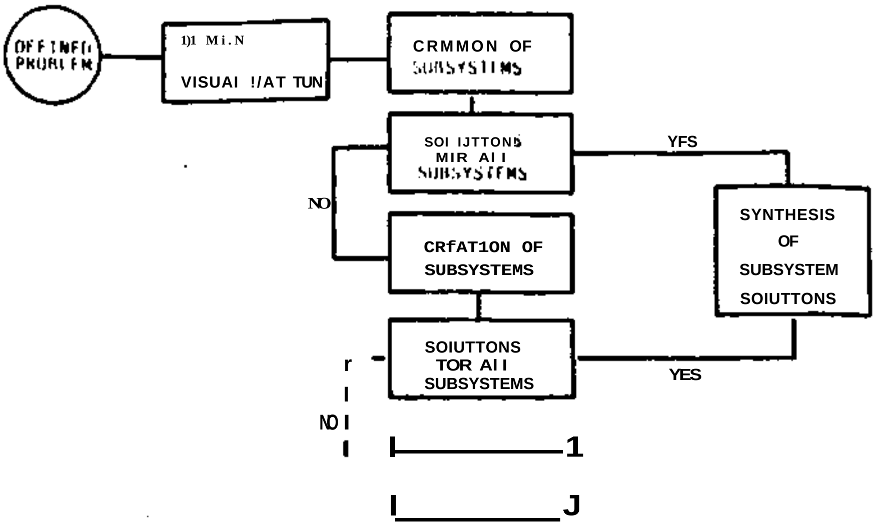


Figure 3-1: A Morphological Approach To Synthesis

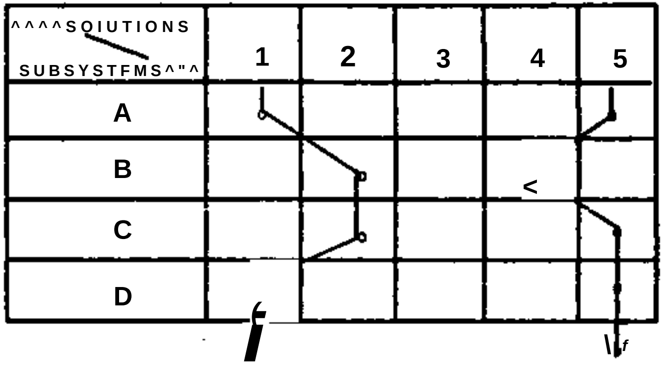
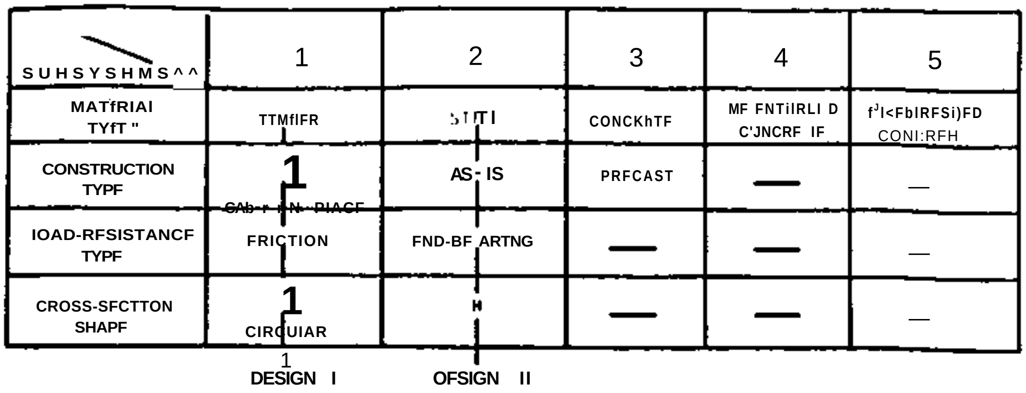


Figure 3-2: General Morphological Matrix

of abstraction. For the pile design problem the end-product is the pile foundation. The problem is then decomposed into a number of subsystems based on various types of pile classifications. Employing the morphological chart shown in Figure 3-2, a matrix of the subsystems and their solutions is created, as shown in Figure 3-3. Various combinations of solutions result in a number of pile design alternatives. Two examples of such combinations are shown in Figure 3-3. The first design alternative (DESIGN I) is a circular timber pile, cast-in-place, transmitting the superstructure loads into the soil through friction. The second design alternative (DESIGN II) is a steel pile transmitting the superstructure loads into the soil through bearing on end. This pile has an H shaped cross section and requires no construction.

Examining these alternatives we notice that the construction type for the first design cannot be correct as timber is not a material that can be cast-in-place. Furthermore, if steel is not available, the



S U H S Y S H M S ^ ^	1	2	3	4	5
MATfRIAI TYfT "	TTMfIFR	5 I IT I	CONCKhTF	MF FNTiIRLI D C'JNCRF IF	f^JI<FbIRFSi)FD CONI:RFH
CONSTRUCTION TYPF	1 CAb-r rN- PIACF	AS - IS	PRFCAST	—	—
IOAD-RFSISTANCF TYPF	FRICTION	FND-BF ARTNG	—	—	—
CROSS-SFCTTON SHAPF	1 CIRCUIAR	H	—	—	—

Figure 3-3: Pile Design Morphological **Matrix**

second design alternative is not valid. These two examples expose one of the most serious drawbacks of the morphological approach.

The morphological approach does not discriminate against infeasible combinations, resulting m the generation of many designs that are not valid. This issue is addressed by the proposed expert system through its constraint handling techniques. Constraints can represent design experience as well as fundamental engineering theory. The use of constraints during the synthesis phase provides a mechanism for eliminating the infeasible design alternatives. In the case of the pile design examples described above we can introduce the constraints that timber cannot be cast-in-place and that not available materials cannot be considered. The ability of expert systems to easily incorporate such constraints makes them an ideal environment for adopting principles of the morphological approach while at the same time, provide remedies for its various drawbacks.

The chosen subsystems for the pile design are essential to the solution of this problem but not sufficient for a complete design. Key quantities such as the length of the pile and its cross sectional area have not been taken into consideration during the decomposition of this problem. Although th.s omission would be easy to detect, this is not the case in most design problems. In fact, the most difficult part of this approach to design is the correct identification of the various subsystems Literature [10,15,23] describing the morphological approach requires that these subsystems should be:

- independent of each other,
- inclusive to all the parts of the problem,
- essential to any solution of the problem, and

- few and simple so that the complexity of the overall problem is decreased.

Satisfaction **of these** rigid requirements is, in most engineering problems, both impractical and unnecessary. The proposed expert system implementation will relax these requirements, particularly the first **two,** in an effort to provide a more intelligent and flexible approach to the preliminary design **process.**

4. Expert Systems Background

Computer programs have become an integral part of engineering. Conventional programming **techniques have been used to create complex and sophisticated software for many aspects** of **the engineering practice. In engineering design the role of the computer has been particularly** limited. **Requirements of completeness, uniqueness, and correctness inherent to the algorithmic approach to computing have made it very difficult to use the computer in formalizing design processes.**

Expert systems, using heuristic approaches to programming, **provide a new tool that** relaxes these **requirements. Such systems can be defined as interactive, knowledge intensive computer** programs **that incorporate judgement, experience and other expertise in order to provide knowledgeable** advise **to the user.**

4.1. Architecture of KBES

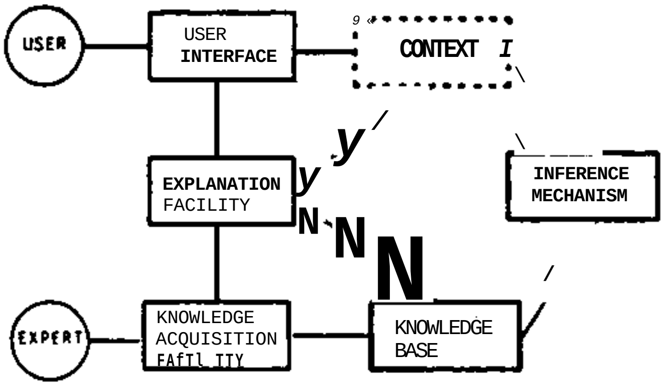


Figure 4-1: Typical KBES Architecture **[19]**

A typical expert system may be composed of six modules, as shown in Figure 4-1. The three main components of an expert system are described below [14,19]:

1. **Knowledge Base:** This module contains the encoded knowledge (facts and rules) for the particular class of problems to be solved.

2. **Context(Working Memory):** This module contains symbols that represent facts and assertions about the current problem. The working memory is a dynamic structure that exists only during consultation sessions hence the dotted line representation shown in Figure 4-1.

3. **Inference** Mechanism(Control): This module contains the problem solving strategies. The inference mechanism manipulates the facts and rules contained in the knowledge base to build or modify the context.

The explicit division between the knowledge base and thei inference mechanism is a primary distinction between expert systems and algorithmic programs. In an expert system environment, the domain knowledge is explicitly manipulated **by** a distinct **and separate control structure.** On the other hand, in an algorithmic approach, the control knowledge appears implicitly within the code of the **program.**

In addition **to** the three main modules presented above, three more parts are included to complete **an expert system as described below:**

1. **User Interface:** The module provides a link between the user **and** the expert system. This interface is responsible for translating the user specified input into a form acceptable by the system as well as for the output presented to the **user.**

2. **Explanation Module:** This module provides explanations of the inferences used by the system.

3. **Knowledge Acquisition Facility:** This module serves as an interface between an expert(s) and the expert system. This interface provides the means for entering and revising knowledge in the knowledge **base.**

The design and development of an expert system is a complex task. One of the key steps ·n simplifying this task is the selection between a growing number of available tools in an ef'crr 'o choose the most appropriate one for a particular problem.

4.2. Tools and Techniques

The tools available for designing and developing expert systems can be divided into three ^a·n classes: general purpose programming languages, general purpose representation languages m.j expert system shells [18] (see Figure **4-2).**

General Purpose Programming Languages: An expert system may be built using a ge^e^i purpose programming language such as LISP [26] or PROLOG [8]. Of the various avanat«e languages, LISP has been the most popular for building expert systems in the United States. This s 3 result of the orientation of this programming language towards symbolic computation.

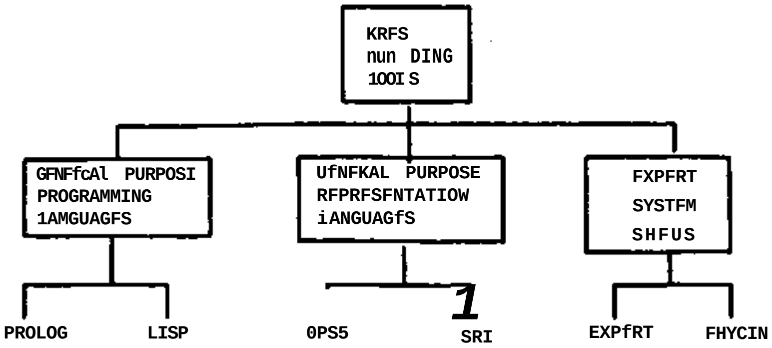


Figure 4-2: KBES Building Tool Classes

General Purpose Representation Languages: This class of tools consists of programming languages developed specifically for use in building expert systems. Such tools are: SRL [29], RLL [13], FRAMEKIT [7], and OPS5 [11]. A general classification of such languages is made in terms of knowledge representation. Two common knowledge representation techniques used are production rules and frames.

Expert System Shells: These are packages that provide an inference engine, knowledge acquisition and explanation modules from which an expert system can be developed by adding the domain knowledge. Such systems include: EMYCIN [27], EXPERT [28], and Deciding Factor [6].

Many commercially available shells have been developed to address the derivational approach to problem solving. This approach assumes that the solutions exist in the knowledge base and the expert system derives the appropriate solution. This approach is not suitable for engineering design because rarely can complete design solutions be enumerated and listed in the knowledge base ^e formation approach to problem solving involves combining and checking parts of the solution >jnv\ feasible solutions are formed. The expert system shell described in this report uses the formation approach to problem solving.

4.3. Expert Systems For Structural Design

During the last few years, a number of expert systems have been developed at the Department of Civil Engineering at Carnegie Mellon University. Many of these systems have been oriented towards structural design in an effort to develop computer programs able to handle the overall engineering design process including the conceptual preliminary stage.

The first expert system developed for structural design at Carnegie Mellon University, *HI-RISE* [18], follows the synthesis, analysis, evaluation methodology for the preliminary structural design of high rise buildings. The system was implemented in PSRL [24], a production system representation language developed at CMU. A hybrid of SRL and OPS5, PSRL provides an environment in which schemas (frames), production rules, and LISP functions are combined to represent knowledge.

The input to *HI-RISE* is a three dimensional grid representing the space planning of the building , the intended use of the building, and the design loads. The output of the system is a number of alternative structural systems, ranked according to their appropriateness for the particular building. The system uses hierarchical planning in its approach to the design process, proceeding from the abstract to the detailed. The synthesis process in *HI-RISE* is composed of a depth-first search through various levels of abstraction. At each level a feasible subsystem is selected and checked for feasibility by a number of elimination constraints. A feasible alternative is one that has not been eliminated at any level.

Three other structural design expert systems were subsequently developed attempting a number of variations to the preliminary design approach introduced by *HI-RISE.* These systems are: *LOW-RISE* [5], implemented in OPS5, which studies the design process for single story building, *ALL-RISE* [25], implemented in SRL, generalizes the synthesis process used in *HI-RISE;* and *FLODER* [17], implemented in OPS5, concentrates on the design of floor systems.

A departure from the group of expert systems described above, *PILE-EXPERT* [17] deals with design in the area of geotechnical engineering. Implemented in OPS5, this KBES was developed for the design of single pile foundations under axial and static loading conditions. The system generates and tests a number of design alternatives based on information provided by the user as well as built m code and standard specifications. The results presented to the user consist of a number of feasible pile designs. *PILE-EXPERT* follows the synthesis, analysis, evaluation methodology discussed m previous sections with special attention paid to the synthesis phase. The primary goal of the system at this phase was to create a general control structure independent of the pile design domain. The development of *PILE-EXPERT* brought into focus the great amount of difficulty involved in attempting such a general approach to engineering design. At the same time the system was able to achieve a certain degree of generality, giving indications that such an approach is possible.

5. An Expert System Shell For Engineering Design

An expert system shell has three basic modules that comprise the kernal of the system, and three additional components that provide a complete environment for building expert systems, as described in Section 4.1. The shell under development initially has the following basic components: knowledge base, context, and inference mechanism as described below.

The **knowledge base** includes the knowledge specific to the class of problems to be solved. The knowledge base is organized into levels of abstraction, where each level contains a list of discrete elements, as shown in Figure 5-1. The knowledge base also contains heuristics in the form of invalid combinations of elements and preconditions. The design expert is responsible for defining the appropriate levels of abstraction and design heuristics to be placed in the knowledge base.

```
LEVEL-1 :     ELEMENTS    A, B, C, D.

LEVEL-2 :     ELEMENTS    A, B

LEVEL-3 :     ELEMENTS    A, B, C, D, E, F

LEVEL-4 :     ELEMENTS    A
   .              .
   .              .
   .              .
LEVEL-i :     ELEMENTS    A, B, ...... j

SAMPLE CONSTRAINT : IF     LEVEL-1 ELEMENT = C
                    AND    LEVEL-2 ELEMENT = A
                    THEN   COMBINATION NOT FEASIBLE
```

Figure 5-1: General Knowledge Base Organization

The **context** contains the information about the design problem currently being solved. This information is defined initially by the user in the form of preconditions and is expanded by the expert system when the inference mechanism searches the knowledge base for feasible solutions.

The **inference mechanism** provides the design strategy. The strategy employed by this shell involves a constraint directed search for feasible combinations representing design solutions. An item is selected from each level in the hierarchy in a depth-first manner. Upon selection, the combination representing the current design under consideration is checked for constraint satisfaction. If a combination is considered feasible, the next level in the hierarchy is considered; otherwise a selection from the same level is tried. This process continues until all feasible combinations are found.

To illustrate the potential use of the proposed expert system shell, the pile design problem introduced in Section 3 is implemented. The three basic components of the shell, described above in general terms, will be reexamined with respect to this particular example.

The knowledge base is organized into various levels of abstraction. Employing the morphological approach and referring to Figure 3-3. these levels are represented by the various subsystems with a number of discrete elements associated with each level. The knowledge base also includes a number of constraints at each level to evaluate elements under consideration and eliminate invalid combinations. The various levels of abstraction, the corresponding discrete elements, as well as examples of constraints are shown in Figure 5-2.

LEVELS	ELEMENTS
MATERIAL-TYPE:	TIMBER. STEEL. CONCRETE. REINFORCED CONCRETE. PRESTRESSED CONCRETE
LOAD-RESISTANCE TYPE:	FRICTION. END-BEARING
CONSTRUCTION-TYPE:	CAST-IN-PLACE. AS-IS. PRECAST
CROSS-SECTION:	CIRCULAR. H
SAMPLE CONSTRAINTS:	IF MATERIAL TYPE IS TIMBER AND CONSTRUCTION TYPE IS NOT NO-CONSTRUCTION THEN THE DESIGN COMBINATION IS NOT FEASIBLE IF MATERIAL IS NOT AVAILABLE THEN THE MATERIAL SELECTION IS NOT FEASIBLE

Figure 5-2: Pile Design Knowledge Base Organization

As described above the **inference** engine provides the design strategy which involves a depth first search at each level of abstraction. Applying this strategy to the pile design example the first selection is the element TIMBER at the MATERIAL-TYPE level. Upon selection, this element is checked for constraint satisfaction. From the constraints shown in Figure 5-2 it follows that if timber is available then it can be considered. Continuing the design, the search follows on to the CONSTRUCTION-TYPE level and selects the CAST-IN-PLACE element. A check of the combination TIMBER, CAST-IN-PLACE shows that it is not feasible. The search continues at this level with the next element being considered. The resulting TIMBER, AS-IS combination satisfies the design constraints and is thus accepted. The search continues until all valid design combinations are found

Conditions for a particular design problem govern the configuration of the end-product. In the case of the pile design, conditions such as the applied load and the surrounding soil play a major role in the selection of the pile classification types as well as the determination of the pile length and cross sectional area. For example, code specifications dictate that timber should not be used as a pile material if the applied load exceeds the value of 270 KNewtons. The applied load is defined by the user and is contained in the context. The context information is a dynamic quantity which is modified and expanded as the design progresses. Using the example described in the previous paragraph the context undergoes the following changes: Initially this component will contain only the user-defined preconditions such as: concrete is not available, or the applied load is 400 KNewtons. As the design progresses the context is expanded to include the alternative pile foundation solutions.

6. Conclusions

The expert system shell under development represents an effort to develop an expert system for engineering design. The prototype expert systems, such as HI-RISE, resulted in a special purpose expert systems in which the control strategy was imbedded in the levels of abstraction and constraint representation. The morphological approach serves to organize the design process but is difficult to use in practice due to its rigid requirements and exhaustive search. The development of a shell for engineering design will draw on the strengths of both expert system techniques and the morphological approach to provide an environment in which an engineer can develop an intelligent and flexible preliminary design aid.

References

1. ALGER, J.R.M. & HAYS, C.V., "Creative Synthesis in Design," Prentice Hall, 1964.
2. ASIMOW, M., "Introduction to Design," Prentice-Hall, 1962.
3. BEITZ, W. & PAHL, G., "Engineering Design," The Design Council, 1984.
4. BUHL, H.R., "Creative Engineering Design," Iowa State University Press, 1960.
5. CAMACHO, G., "LOW-RISE: An Expert System For Structural Planning And Design Of Industrial Buildings," Master's Thesis, Department of Civil Engineering, Carnegie-Mellon University, 1985.
6. CAMPBELL, A. & FITZGERRELL S., "The Deciding Factor," User's Manual, Software Publishing Corporation, 1985.
7. CARBONELL, J. & JOSEPH, R., "FRAMEKIT+: A Knowledge Representation System," Carnegie Mellon University, October 1985.
8. CLOCKSIN, W.F. & MELLISH, C.S., "Programming in PROLOG," Springer-Verlag, Berlin, Heidelberg, New York, 1981.
9. FIKES,R. & KEHLER, T., "The Role of Frame-Based Representation in Reasoning," Communications of the ACM, Vol.28, No.9, September 1985, pp.904-920.
10. FINKELSTEIN, L. & FINKELSTEIN, A.C.W., "Review of Design Methodology," IEE Proceedings, Vol.130, Pt.A, No.4, June 1983, pp. 213-221.
11. FORGY, C.L, "OPS5 User's Manual," Technical Report CMU-CS-8M35, Carnegie Mellon University, July, 1981.
12. FRENCH, M.J., "Engineering Design: The Conceptual Stage," Heinemann Educational Books, 1971.
13. GREINER, R & LENAT, D.D., "A representation language," Proceedings of the First Annual National Conference on Artificial Intelligence, pp. 165-168,1980.
14. HARMON, P. & KING, D., "Expert Systems: Artificial Intelligence in Business," Wiley Press, 1985.
15. JONES, J.C., "Design Methods," John Wiley & Sons, 1980.
16. KARAKATSANIS, A.f "FLODER: A Floor Designer Expert System," Master's Thesis, Department of Civil Engineering, Carnegie Mellon University, 1984.
17. LONGINOS, P., "PILE-EXPERT: A Knowledge Based Expert System for the preliminary design of pile foundations," Department of Civil Engineering, Carnegie Mellon University, 1986.

18. MAHER, M.L., "HIRISE: A Knowledge Based Expert System for the Preliminary Structural Design of High Rise Buildings," PhD Thesis, Department of Civil Engineering, Carnegie Mellon University, 1984.

19. MAHER, M.USRIRAM, D. & FENVES, S.J., 'Tools and Techniques for Knowledge Based Expert Systems for Engineering Design," Adv. Eng. Software (GB), Vol.6, No.4, 1984, pp.178-188.

20. NADLER, G., "An Investigation of Design Methodology," Management Sci., 13(10), 1962.

21. NORRIS, K.W., 'The Morphological Approach to Engineering Design," in Jones, J.C. & Thornley, D.G., (Editors): Conference on Design Methods, Pergamon Press, 1963.

22. OSTROFSKY, B., "Design, Planning, & Development Methodology," Prentice-Hall, 1977.

23. PEARSON, D.S., "Creativeness for Engineers," Edwards **Bros, 1961.**

24. RYCHENER, M.D., "PSRL: An SRL-Based Production Rule System." Reference Manual.

25. SRIRAM, D., "Knowledge-based Approach To Integrated Structural Design,[1]' Department of Civil Engineering, Carnegie Mellon University.

26. STEELE, G.L, "COMMON LISP: the language," Digital Corporation Press, **1984.**

27. VAN MELLE, W.A., "A domain independent production-rule system for consultation programs," Proceedings Sixth IJCAI, pp.923-925, August 1979.

23. WEISS, S.M. & KULIKOWSKI, C.A, "EXPERT: a system for developing consultation models," Proceedings Sixth IJCAI, pp.942-947,1979.

29. WRIGHT, J.M. & FOX, M.S., "SRL/1.5 User Manual," Technical Report, CMU Robotics Institute, June 1983.

30. ZWICKY, F., "The Tasks We Face," J. Am. Rocket Soc., 1951, 21-22,1984, pp.3-20.