A Knowledge-Based Approach to Preliminary Design Synthesis

by

M. L. Maher

EDRC-12-14-87 3

Table of Contents

List of Figures

A Knowledge-based Approach To Preliminary Design Synthesis[1]

Mary Lou Maher
Department of Civil Engineering
. Carnegie Mellon University

ABSTRACT

Design synthesis involves producing *one* or more design solutions. During preliminary design, the form of the design solution is identified such that a few key constraints are satisfed. This form is refined during detail*design such that all relevant constraints are satisfied. This paper describes a knowledge-based approach to preliminary design synthesis using constraint-directed search through levels of abstraction to construct a design solution. The approach is implemented in a knowledge-based framework, EDESYN.

1. Introduction

Design, a combination of art and science, is perhaps one of the most important and difficult tasks an engineer performs. Substantial activity has grown in studying design theory and methodology in an effort to produce a structured approach, for example, those documented in [4, 2] and the existence of an NSF program for research in design theory and methodology. A number of approaches have been developed for a wide variety of applications and from the background of differing engineering disciplines.

Most recently, the approaches have been implemented as knowledge-based systems or tools for building knowledge-based systems. Arciszewski[1] uses morphological analysis as a basis for generating design solutions by combining feasible states in a qualitative decisions table. Mitchell [7] decomposes the design process into modules and uses rules to identify the appropriate design solution for each module in a program called VEXED. Tong [10] represents the design process as a set of steps where each step represents a level of abstraction and the solution is a combination of the results of each step. Tong's approach is implemented in a program called DONTE. McDermott [6] has implemented a problem solving method in SALT that operates on a decomposition of the design process by making an acceptable decision at each level and using rules to recover from violations or contradictions, ""he common theme among all approaches cited is problem decomposition and constraint

[1]Sub.T.ilied to AI EDAM, May 1937

satisfaction. The approach described in this paper uses similar concepts in 2n effort to provide an environment for developing a knowledge-based system for preliminary design synthesis.

In this paper, a distinction is made between preliminary and detail design. During preliminary design, the form of the design solution is identified such that a few key constraints are satisfied. During detail design, the design solution is specified for construction or manufacturing such that all relevant constraints are satisfied. 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 knowledge-based system environment able to support 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. Synthesis

Design is a process by which design intentions are transformed into design descriptions. The design process has identifiable phases within it, synthesis being one of the phases. Although the phases may not be addressed hierarchically for the entire design cycle and are often carried out recursively, there is an inherent order in which designers approach a design problem. The following represents one decomposition.

- *Design formulation* involves identifying the goals, requirements and possibly the vocabulary relevant to the needs or intentions of the designer.
- *Design synthesis* involves the identification of one or more design solutions within the design space elaborated during formulation.
- *Design evaluation* involves interpreting a partially or completely specified design description for conformance with goals and/or expected performances. This phase of the design process often includes engineering analysis.

This paper is concerned with design synthesis, during which the form of the design solution is identified. Typically, this phase of the design process is ill-structured. Experienced designers resort to trial and error less often than novice designers when they synthesize designs, suggesting that the use of knowledge-based systems to represent 'experience' may improve design synthesis.

Design synthesis can be considered as prototype refinement where a prototype is selected as part of the formulation process or .constructed during the synthesis process from other prototypes. This effectively locates the designer at a specific state in the design space and constrains movements away from that state into a narrow set defined by values for decision variables. The knowledge needed to select an appropriate prototype is not well articulated or fully understood. The approach discussed in this paper involves constructing a prototype from predefined levels of abstraction representing partial prototypes.

3. Synthesis Using Constructive Search

There is no uniformly "best" way to approach the synthesis process for all designs. A common procedure is to decompose the design problem into 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. The morphological analysis approach [1] proposes that alternative candidate designs can be synthesized by considering possible combinations of the various subsystems that result from combinations of lower level components. This approach to the synthesis enables the designer to consider a random set of possibilities based upon the manner in which the subsystems and the lower level components are defined. Other approaches to synthesis propose that either a decision is made at each subsystem level before considering the next level [6] or each decision be deferred until enough information is available to evaluate constraints on that decision [9].

The approach considered here makes use of the concepts of problem decomposition and constraint directed search. Using this approach, a design problem is decomposed into levels of subsystems or decisions. The decomposition into levels and the order in which the levels are considered depends on the nature of the particular problem as well as the engineer performing the design. A particular problem may justify the consideration of more general decisions before lower level details while another problem 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.

Synthesis using constructive search involves organizing the design decisions and associated alternatives, compiling the constraints on invalid combinations of decisions, and systematically searching for valid combinations for a specific problem. The valid combinations represent feasible solutions to the synthesis problem and are then further evaluated in order to identify the solution or solutions to be pursued further.

DECISIONS	ALTERNATIVES
3D-SYSTEMS	Core Tube Bundled Tube 2D-Orthogonal
2D-SYSTEMS	Rigid Frame Braced Frame Shear Wall
MATERIAL	Steel Reinforced Concrete Prestressed Concrete Prefabricated Concrete

Figure 3-1: Structural System Design Decisions

This approach to synthesis can be illustrated through an example of the preliminary design of the structural system for a building. This problem has been addressed in a knowledge based system called HI-RISE [5], in which the control knowledge used during synthesis was embedded in the knowledge about structural systems. The potential combinatorial explosion during the synthesis of structural systems has been recognized by designers, as documented in [8]. The synthesis of structural systems can be decomposed into three levels of abstraction: 3D-systems, 2D-systems, and materials. The alternatives for each level are illustrated in Figure 3-1. The consideration of all combinations of alternatives would result in 4x3x4, or 43, preliminary design solutions, many of which are not acceptable. This decomposition does not include alternative placement schemes for the structural systems, which would greatly increase the number of alternative solutions.

The introduction of constraints is necessary to guide the search for feasible combinations. Introducing the constraint that braced frames are only constructed from steel narrows the solution range and eliminates incorrect combinations. This type of constraint is inherent to the structural system design problem and will always be considered. Another constraint limits the use of 2D-orthogonal systems to buildings under 40 stories. This constraint is specific for a particular situation and thus more difficult to identify. The identification of both types of constraints is essential to a structured approach to structural system synthesis.

A particular design problem is identifed by values for preconditions. The preconditions required for the synthesis of structural systems include the number of stories, bay sizes, total perimeter dimensions, and intended use of building. These preconditions are used in the elimination constraints to guide the search for feasible alternatives.

The synthesis of feasible solutions is generated.by visiting the levels of abstraction, selecting one element from the list of alternatives, and checking the applicable constraints, in a depth first manner. For example, the 3D system *core* is selected and checked for compatability with the input conditions. If no constraints are violated, the next level, 2D-systems is considered. The 2D system *rigid frame* is selected and checked, continuing in this manner until a feasible combination of one element from each level is found and stored in a solution list. The depth first search continues until all feasible combinations are found. Each feasible combination represents a prototype to be further refined during the remainder of the design process.

4. EDESYN: An Implementation

EDESYN is a knowledge-based framework that provides an inference mechanism suitable for synthesis using constructive search, as described above. EDESYN is implemented in a frame based language, FRAMEKIT[3], and runs on a MICROVAX II. The framework provides for a separation of control knowledge and domain knowledge by providing an *inference mechanism* that operates on the *knowledge base.* The preconditions and resulting solutions are stored in the *context* A *knowledge acquistion* facility is provided to translate a description of the knowledge base into frames that the inference mechanism can recognize. Each of these components are described below.

The *knowledge base* includes the knowledge specific to the class of problems to be solved. EDESYN's knowledge base is organized into levels of abstraction, where each level contains a list of discrete elements, as shown in Figure 4-1. Each level of abstraction represents a design decision and the elements at each level represent alternative solutions to a decision. The knowledge base also contains heuristics in the form of invalid combinations of decisions and preconditions. The design expert is responsible for defining the appropriate levels of abstraction and design heuristics to be placed in the knowledge base.

Tfie knowledge-base is implemented using two classes of frames: level-frames and constraint-frames. A level-frame **has attributes: name,** element-list, parent-frame, constraint-list and current-list. The name is assigned by the designer, e.g. 3D-systems. The element-list is **a** complete list of alternatives for the current level, e.g. core tube bund!ed-tube 2D-orlhogonal. The parent-frame is a relationship between the current level and the level considered previously, represents the order in which the levels are considered as defined by the designer. The constrajnt-Iist contains constraint-frames that must be checked when the current level is considered. The current-list attribute contains the part of the element list that has not yet been considered. The constraint-frame has attributes: condition-levels, condition-elements. Condition-levels and condition-elements attributes represent the left hand side of the constraint rule.

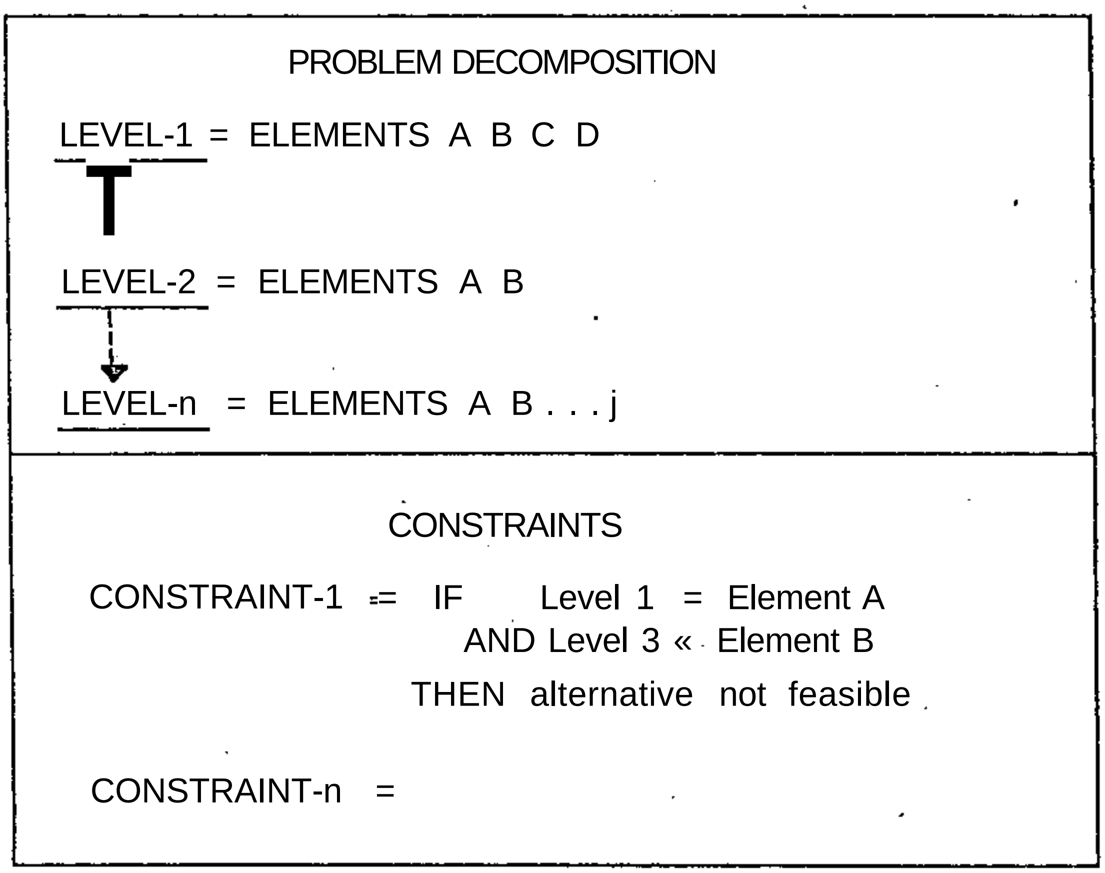

Figure 4-1: General Knowledge Base Organization

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 EDESYN when searching the knowledge base for feasible solutions. The context takes the form of a tree, where a node in the tree represents a single design decision and links between nodes represent valid combinations.

The context is implemented by one class of frames: element-frames. An element-frame has attributes: name, parent-element. The name attribute is the same name that appears in the element-list of the level-frame. The parent-element is a relationship between elements.

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 of abstraction and the levels are visited 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 of abstraction is considered, otherwise a selection from the same level is tried. This process continues until all feasible combinations are found. The inference mechanism is implemented as a Lisp function.

To illustrate the use of EDESYN, the structural system design problem introduced in the previous section 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. These levels represent various decisions such as 3D-systems, 2D-systems, and material, with a number of discrete elements associated with each level. The knowledge base also includes a number of constraints to eliminate invalid combinations, such as:

```
IF 2D-systems=braced fraine
AND materialOsteel
THEN alternative not feasible
```

The various levels of abstraction, the corresponding discrete elements, as well as examples of constraints are the same as those described in Section 3.

As described above, the inference engine provides the design strategy which involves a depth-first search at each level of abstraction. The resulting context tree, illustrated in Figure 4-1, has four alternative prototypes for a 30 story office building. The first alternative is a concrete shear wall around the core. The second is concrete rigid frames in each direction. The third is steel rigid frames, and the fourth is steel braced frames. These alternatives only identify the form of the solution, the details are

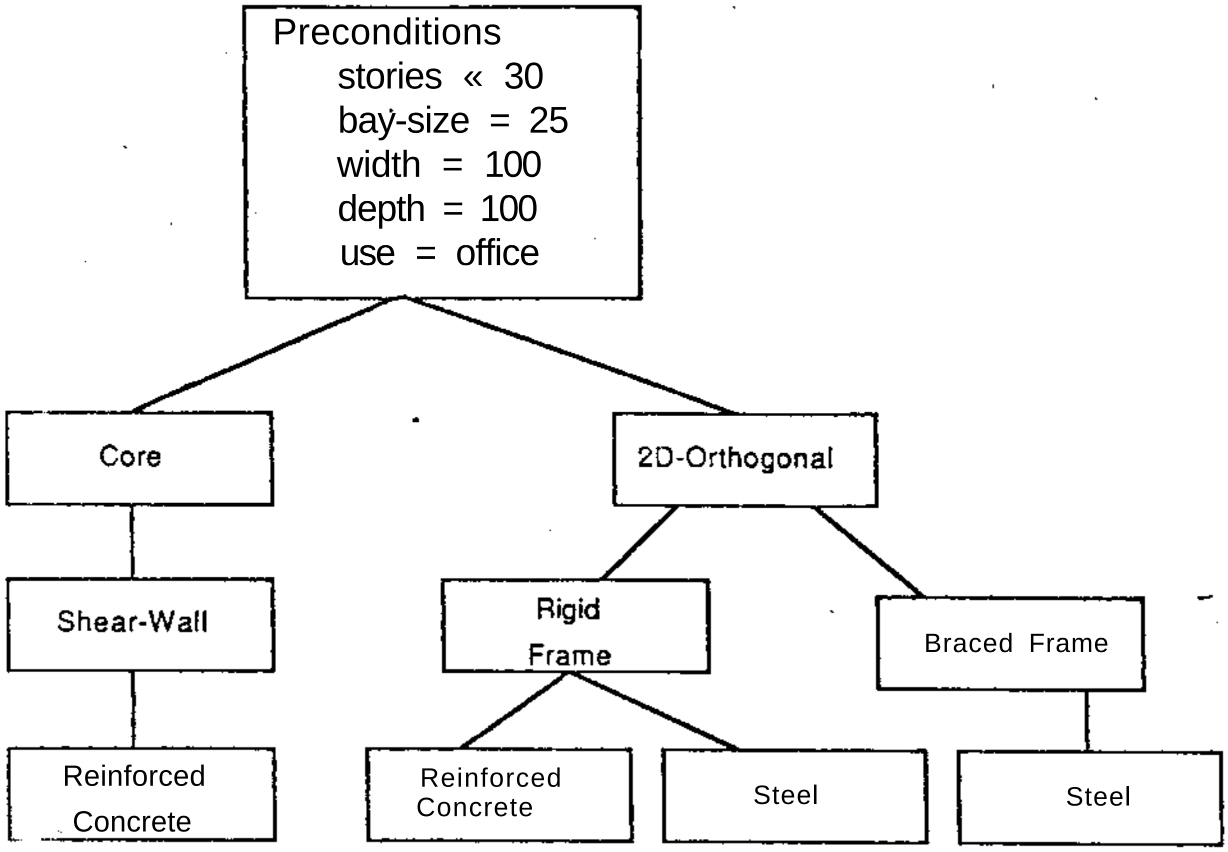

Figure 4-2: Context Tree For Structural System Design

defined after synthesis.

5. Extensions

EDESYN is currently limited to design problems that can be decomposed into a linear set of feveis of abstraction and can only reason about discrete alternatives for each level. The extensions under way address both of these limitations. In order to deal with more realistic design situations, EDESYN is being extended to allow each level of abstraction to be considered *as* a set of discrete alternatives (as in the implementation described above), as a set of levels to be synthesized into a set of alternative prototypes, or as a function which returns a single value consistent with previous decisions made at other levels. These extensions allow the design to be decomposed into a *tree* of levels of abstraction where the current version only allows a list. The introduction of a function for determining the appropriate alternative at a given level provides a mechanism for reasoning about continuous values, as well as discrete values.

A more difficult extension to EDESYN includes providing a mechanism for evaluating the alternative prototypes that are constructed during the search process. Currently, EDESYN leaves the evaluation to the designer. The designer is presented with the prototypes and selects among them for further refinement. Automatic evaluation involves identifying criteria and methods for measuring the value of each prototype **according** to the criteria.

In order to **make EDESYN** more interesting to designers, graphics are necessary. To extend EDESYN to include graphics requires that the frames representing levels and elements include geometric information and procedures for display. Currently EDESYN describes the feasible alternatives as a list of eiements.

6. Conclusion

The systematic search for alternative design solutions is only one approach to the design process. This approach appears to be useful for preliminary design of engineering systems. The implementation of constraint directed depth first search through levels of abstraction to construct prototypes for further refinement has been described as a framework for building knowledge-based systems. The framework, EDESYN, has been applied to structural system design, as described in this paper, and to foundation design. EDESYN identifies a control strategy and knowledge base organization for reasoning about preliminary design situations.

7. Acknowledgements

This work has been supported by the NSF sponsored Engineering Design Research Center at Carnegie Mellon University. Panayiotis Longinos and Suneet Shukla are acknowledged for their contributions to the implementation of EDESYN.

References

[1] Arciszewski, T.
Mathematical Modelling of Morphological Analysis.
In *Proceedings of Fifth ICMM$_t$* pages 52-56. ICMM, 1986.

[2] Arciszewski, T.
ARIZ 77: An Innovative Design Method
Technical Report, Department of Civil Engineering, Wayne State University, 1S87.

[3] Carbonell, J.G. & Joseph R.
FRAMEKIT+: A Knowledge Representation System.
Technical Report,, Pittsburgh, Pennsylvania, June, 1985.

[4] Gordon, W.
SYNECTICS The Development of Creative Capacity.
HarperS Row, 1961.

[5] Maher, M.L
HI-RISE: A Knowledge-Based Expert System For The Preliminary Structural Design of High Rise Buildings.
PhD thesis, Department of Civil Engineering, Carnegie Mellon University, 1984.

[6] McDermott, J.
Making Expert System Explicit.
In *Proceedings of IFIP Congress .* IFIP, 1986.

[7] Mitchell, T., Steinberg, L, Shulman, J.
A Knowledge-based Approach To Design.
IEEE Magazine :27-34, 1984.

[8] Rush, R.
Mixed framing systems: one-way ticket to complexity?
Building Design & Construction , March, 1987.

[9] Stefik, M.
Planning With Constraints.
Technical Report STAN-CS-80-784, Computer Science Department, Stanford University, January, 1980.

[10] Tong.C.
Knowledge-Based Circuit Design.
PhD thesis, Stanford University, 1987.
forthcoming.