A Structural Design Process Philosophy and Methodology

Becker, James

1973-09-01

REPORT NO.
UC SESM 73-13

STRUCTURES AND MATERIALS RESEARCH
DEPARTMENT OF CIVIL ENGINEERING

A STRUCTURAL DESIGN PROCESS
Philosophy and Methodology

by

JAMES M. BECKER

SEPTEMBER 1973

STRUCTURAL ENGINEERING LABORATORY
UNIVERSITY OF CALIFORNIA
BERKELEY CALIFORNIA

A STRUCTURAL DESIGN PROCESS

PHILOSOPHY AND METHODOLOGY

BY

JAMES M. BECKER

Submitted in partial satisfaction of the requirement for the Ph.D. Degree, June, 1973. Research carried out under supervision of Professor B. Bresler.

Report No. UC SESM 73-13

STRUCTURAL ENGINEERING AND STRUCTURAL MECHANICS
DEPARTMENT OF CIVIL ENGINEERING
UNIVERSITY OF CALIFORNIA, BERKELEY

September, 1973

A STRUCTURAL DESIGN PROCESS

Philosophy and Methodology

by

James M. Becker

Abstract

A model descriptive of a structural design process is the focus of this dissertation. The model developed describes an information process that transforms a recognized need into a design capable of fulfilling that need. The model is extended into MAGID (The Manipulation and Generation of Information for Design), a method intended to aid a designer in the conceptual aspects of a design process.

The model of a structural design process is based upon the postulate that design can be interpreted as the acquisition, manipulation and generation of information. From the moment of need recognition, the design process is seen to spiral upward in a hierarchical structure of information that ends with the development of an acceptable solution. This spiraling process is symbolized by the cyclic interaction of sets of 'fuzzily' defined information that correspond to a problem solving cycle of synthesis-evaluation-decision. These 'fuzzy' sets of information are further defined as collections of BIT's (Basic Information Terms).

A design method called MAGID is developed as a logical extension of the above model. MAGID is a primitive information control system intended to be a partner for a structural designer aiding him in the initial identification of design kernels (the conceptual design). MAGID is a multi-level interactive process that through a sequence of requests to an expanding problem oriented library guides the designer in the construction of a design space. The construction of this design space is achieved by an inverse process of first defining desired attributes, then determining how these attributes are to be evaluated, and finally, identifying what design descriptors are needed for evaluation. The design space is then uncoupled into subspaces relevant to particular attributes, so that the designer can judgmentally evaluate the related subspace vectors. Upon completion of these evaluations, MAGID recouples the subspaces and presents the designer with an ordered set of acceptable initial design kernels. Sample problems have been worked with MAGID to explore its flaws and potential usefulness. The results of one of these sample problems is presented in order to help clarify the concept of both Magid and the model of a structural design process.

ACKNOWLEDGMENTS

This research was carried out during the author's graduate study for the degree of Doctor of Philosophy in Engineering at the University of California, Berkeley.

The author wishes to express his sincere appreciation to Professor B. Bresler for his patience and guidance during the course of this research. He would also like to acknowledge the assistance of Professors N. Distefano and E. L. Wilson while serving as members of the thesis committee and the suggestions offered by Professor L. A. Zadeh. In addition, he would like to thank visiting Professors G. G. Goble of Case-Western University and C. A. Cornell of M.I.T., and colleagues Stephen Mahin, Weston Hester, J. Athol Lennie, and Ashok Vaish for their many helpful discussions. The assistance of Patricia Klitgaard in editing and Diane DeSmidt, Kay Jessee and Barbara Matthews in typing is gratefully acknowledged.

The author would also like to acknowledge the support of the University of California Computer Center which subsidized the computer time necessary for this research and the National Science Foundation which supplied the author with financial support through his appointment to a NSF Traineeship.

A particularly heartfelt thanks are offered to the authors parents for their continual encouragement and support. And, finally, the author wishes to thank Joanna for her patience and Joshua for his omnipresent companionship.

INTRODUCTION

Design is the primary focus of the diverse areas of inquiry of the structural engineer. It is the process by which he determines how resources are to be converted into structural systems. The structural engineer obtains fundamental tools as a student and, in the course of acquiring professional experience, evolves into a designer. However, as analytical techniques continue to advance, materials become more numerous and complex in their behavior, and experiences multiply, an increasing body of information becomes available to the structural engineer. Structures have always existed in complex physical environments and today the impact of a changing social context must be acknowledged. Thus, it is becoming increasingly difficult and important for a designer to bring this expanding body of information to bear on the task of structural design. This dissertation examines structural design with the hope of establishing a basis from which structural engineers may begin to develop a deeper understanding of design. For design is not just a collection of tools and experiences but a human endeavor, an entity, unto itself.

At this point it would normally seem appropriate to offer a definition of what is meant by design; however, none is forthcoming as there can be no truly concise meaning of design. This dissertation, in itself, is intended as a definition for it conveys the author's conception of and feelings toward structural design. How then is the structural design process to be explored? "A Structural Design Process - Philosophy and Methodology" is the title of this dissertation and within the title is found the basic framework for this exploration - a division into philosophy and methodology. The first four chapters present a philosophy of structural design through the development of a descriptive model of a structural design process, while the fifth and sixth chapters use that model as the basis for a methodology intended to aid a designer in conceptual aspects of design.

The first chapter presents a perspective for structural design by postulating that design can be viewed as a form of complex problem solving. This leads to the examination of relevant work in psychology, a short historical review of structural engineering and the development of a concept of problem solving as it might apply to structural design. The second chapter reviews different apporaches to problem solving and design from the fields of operations research, architecture and engineering. The third chapter developes a concept of information as it relates to a design process. Finally, a descriptive model of a structural design process is constructed in the fourth chapter. The model is intended to be flexible enough to allow for the description of design processes presently being used and at the same time provide a specific enough framework to suggest possible ways in which design may be further explored and improved.

To help show the potential for such a model, a design method is developed in the fifth chapter. This design method is an information control system that is intended to interact with a designer to help stimulate and expand his ability in the conceptual aspects of a design process. Chapter Six then presents a sample problem that has been worked with this design method. The dissertation is summarized with the observations and conclusions of the seventh chapter.

INDEX

Chapter 1 - Perspective on Structural Design

Chapter 2 - Approaches to Design

Chapter 3 - Information in Design

Chapter 4 - The Structural Design Process - A Model

Chapter 5 - The Structural Design Process - A Methodology

Chapter 6 - Parkmerced - A Sample Problem

Chapter 7 - Summary, Suggestions and Conclusions

CHAPTER ONE

PERSPECTIVE ON STRUCTURAL DESIGN

1.1 Approach

Developing an approach for the study of structural design is probably one of the most difficult questions raised in this dissertation. The available literature relating specifically to structural design processes is scarce; generally it appears in the form of case studies of particular designs, historical essays, or as brief introductions to more technical articles. Since structural design processes presently constitute no prescribed area of study, it becomes a necessary part of this research to explore and bring forward information that will help to delineate such an area.

It is only necessary to remove the word, "structural," and to concentrate on design processes in order to discover a vast body of information ranging from vague concepts to well documented theories. The question now becomes one of identifying ideas relevant to structural design. The majority of the work in this area postdates World War II, for in many ways much of it has been influenced by the introduction of the digital computer and the early struggles of operations research. Ideas relating to design appear in such diverse disciplines as psychology, information science, operations research, business administration, environmental design, and engineering. In much of the related literature the term, "design," seldom has a clear or distinct meaning.

The areas that come closest to relating to the question of structural design are environmental design and engineering design. The last fifteen years have seen a great deal said and written about design from within these fields. Many valid and exciting ideas about design have been brought forward, but they never seem to provide the fundamental basis from which one can truly understand the many processes that are called design. The contributions of these fields are important and will be reviewed in the next chapter. The question still remains as to how one can begin to understand structural design.

Every person, either consciously or unconsciously, continually solves problems. Structural design at its most fundamental level poses just another problem to be solved. This seemingly simple transformation from structural design to problem solving provides the nucleus from which this research will grow. Psychologists have traditionally concerned themselves with problem solving; within their work can be found basic concepts helpful in developing an understanding of structural design.

1.2 Problem Solving and Psychology

The work done in the psychology of problem solving, and of immediate interest to this research, is divided into two categories: phase models and mechanistic models. The phase models are generally the product of subjective reasoning and attempt to identify different phases in a problem solving situation. The mechanistic approach includes models in which problem solving is seen as being constructed out of many basic operations. The differences between these two areas might be understood by considering the phase models as presenting a broad concept of problem solving and the mechanistic approach as one which tries to specify an internalized process.

1.2.1 Multiphase Models of Problem Solving

One of the first and most often cited of the multiphase models is John Dewey's model of reflexive thought. Primarily an educationalist, Dewey, in his book, How We Think [1]* published in 1910, divides the thought process into five parts:

> (i) a felt difficulty;
> (ii) its location and definition;
> (iii) suggestion of possible solution;
> (iv) development by reasoning of the bearings of the suggestion;
> (v) further observation and experiment leading to its acceptance or rejection; that is the conclusion of belief or disbelief [1, pg. 72].

Dewey then notes that the first and second steps may be considered a single, initial observation and the last step a final observation. He observes that,

> Between those two termini of observation, we find the more distinctly mental aspects of the entire thought cycle: (i) inferences, the suggestion of an explanation or solution; and (ii) reasoning, the development of the bearings or implications of the suggestion [1, pg. 72].

It is in this observation that two very important aspects of problem solving first appear: the generation or synthesis of a trial solution and the reasoning or evaluation of that solution.

In The Psychology of the Inventor [2], written by J. Rossman

---

*Numbers in brackets refer to references and when necessary appropriate page numbers.

and published in 1931, the results of a study of over seven hundred inventors is presented. This study identifies seven different phases in the process associated with invention:

1. need or difficulty observed,
2. problem formulated,
3. available information surveyed,
4. solution formulated,
5. solution critically examined,
6. new ideas formulated, and
7. new ideas tested and accepted.

In Rossman's model two additional ideas, implied by Dewey, are more clearly identified: the idea of reformulation or iteration (6) and the separation of evaluation (5) and the determination of a course of action or decision process (7).

The terms, "creativity" or "productive thought," are often associated with problem solving: this association generally takes the form of directly equating these ideas with problem solving. The trouble here appears to be semantic in many ways and will be resolved by associating creativity and productive thought with the synthetic aspects of problem solving. It is entirely possible, as for example in the case of an artist, that creativity (synthesis) may so dominate a design process that all other aspects become insignificant.

It is within this concept of creativity and productive thought that the famous incubation and illumination of Graham Wallas can be presented. In 1926 Wallas published his book, <u>The Art of Thought</u> [3], in which he presented a four step model for creative thought:

1. Preparation,
2. Incubation,
3. Illumination, and
4. Verification.

The first and fourth steps are quite obvious in their meaing, and it is for the second and third steps that this model is most widely known. Wallas observed that often after the initial preparation had been completed a period of seeming indifference might pass before a synthesis would occur. Thus, a trial solution might occur from what would appear to have been nowhere (illumination). Wallas concluded that the period of seeming indifference was actually a time for quiet internal thought (incubation). While incubation-illumination is an essential type of synthesis, it by no means is the only identifiable form.

Donald Johnson, in his book, The Psychology of Thought and Judgment [4], identifies three types of productive thought: trial and error, insight, and gradual analysis. In addition, a three phase problem solving model is presented:

1. Preparation,
2. Production, and
3. Judgment.

The similarity with Wallas' model is apparent with the difference that Johnson accounts for two additional types of production, trial and error and gradual analysis. It is possible that Johnson's two additional types of production might appear to be less creative and thus justifies Wallas' identification of only incubation-illumination (insight), or it might be equally well interpreted that all three of Johnson's processes could occur within the incubation period. While of interest, this debate does not obershadow the obvious agreement between all the aformentioned psychologists that problem solving is a process that includes some form of synthesis and evaluation.

The models discussed have all attempted to identify different phases in a problem solving process; however, it seems unlikely that any human process can be so definitively categorized. A typical criticism is offered by E. Vinacke:

> The real weakness in the view that creative thought [problem solving] consists of a sequence of fairly well defined phases is not that these stages do not exist but that they are regarded as universal, clearly recognizable, successive, and distinct from each other. In actuality, it would be better to conceive of creative thinking in more holistic terms, a total pattern of behavior [5, pp. 247-248].

The work of Köhler, Duncker [6] and Wertheimer [7] of the Gestalt School of Psychology has led to a more 'holistic approach' to productive thinking or problem solving. The Gestalt concept is of a unitary process in which the solution derives from the problem as a "series of events leading from one state to another of a self-regulating system under stress" [8, pg. 154]. Duncker sees these "successive solution phases" coming from an "insistent analysis of the situation...," [6, pg. 21] accompanied by a continual "analysis of the goals" [6, pg. 23]. The path between problem and solution is seen as one of "conflict" in which the stress created by the problem must be resolved. To quote Wertheimer:

> When one grasps a problem situation, its structural features and requirements set up certain strains, stresses, tensions in the thinker. What happens in real thinking is that these strains and stresses are followed up, yield vectors in the direction of improvement of the situation, and change it accordingly [7, pg. 195].

The idea that problem solving is the resolution of conflict arising between a desired and existing state quite aptly states one important part of any designer's function, for example, a designer often finds himself resolving the contrasting needs of his client and the requirements of a safe structure. An even more important concept is that of continual (insistent) analysis in the problem solving process. This is an important break with phase models in which the analysis of the situation or goals is considered just another stage in a process. It seems far more acceptable to consider the designer as one who is always considering (analyzing) the problem, whether consciously or unconsciously, no matter what else is transpiring.

Duncker also states that a solution develops from "the general or 'essential' properties..[to]..specific properties" [6, pg. 8]. This statement provides the basis from which the evolution of a design must be followed in a temporal sense. In structural design this is analogous to the transition from a conceptual design to detailed working drawings.

While providing many valid and usable concepts, the Gestalt model of problem solving does not seem well suited to providing a complete understanding of structural design. It does, however, have a strong effect on the concept of problem solving accepted in this dissertation. A multiphase model seems appropriate for structural design, but an important modification will be the continual state of problem analysis. The model must furnish flexibility to problem solving, for, to quote Johnson, it "is a fluid enterprise...[where]... like raindrops coursing down a windowpane, intellectual operations coalesce and separate and run together again." [4, pg. 23].

1.2.2 Mechanistic Models of Problem Solving

Models of the problem solving process that are aptly termed multiphase were reviewed in the previous section; however, each phase in one of these models may in turn be considered a problem. In this manner 'synthesis' becomes the problem of generating alternative solutions and 'evaluation' the problem of assessing the value of alternative solutions. Continuing this type of subdivision suggests the dissection of a problem solving process into some form of basic transforms. It is the attempt to work with these basic types of processes and their implied applications (specifically using digital computers) that has suggested the use of 'mechanistic' in the title of this section. This type of approach to both human behavior and that of other complete systems led to the development of Cybernetics [9]. Another development in this general area is Information Processing or Artificial Intelligence in which an attempt is made to simulate human behavior (see Computers and Thought edited by Feigenbaum and Feldman [10]).

"Elements of a Theory of Human Proglem Solving" [11] was one of the earlier (1958) works in the area of Information Processing. The authors, Newell, Shaw and Simon, introduced concepts in which problem solving was considered as a process "compounded out of elementary information processes." In its "simplest aspects" problem solving can be viewed as "a search for a solution in a very large space of possible solutions. The possible solutions must be examined in some particular sequence and if they are, then certain possible solutions are to be examined before others."

[11, pg. 159]. This implies some form of direct enumeration coupled to an efficient search algorithm; it is in this algorithm that the heuristics which control human processes must be simulated.

A goal of Information Processing is the creation of working models that can simulate human problem solving behavior; three postulates are given by Newell, Shaw and Simon [11, pg. 151] for the construction of such models:

> 1. A control system consists of a number of memories, which contain symbolized information and are interconnected by various ordering situations.
> 2. A number of primitive information processes, which operate on the information in the memories.
> 3. A perfectly definitive set of rules for combining the processes into whole programs of processes.
>
> In general, the processes that compose the program are familiar from everyday experiences and from research on human problem solving: searching for possible solutions, generating these possibilities out of other elements, and evaluating partial solutions and cues [11, pg. 152].

The idea "that the free behavior of a reasonabl intelligent human can be understood as the product of a complex but finite and determinate set of laws" [12, pg. 293] is the contribution of information processing to this discussion, not the hope that it may lead to a working model of a designer. Newell and Simon [12] see information processing as a concept in which the conflicting views of different schools of psychology can be resolved. It is this suggested neutrality of problem solving as an information process which eventually will provide the means of bringing the many diverse ideas reviewed in this dissertation into focus and provide the foundations for a general model of the structural design process.

While not ideally conforming to a mechanist approach to problem solving, J. P. Guilford's conception of problem solving, offered in her book, The Nature of Human Intelligence [13], adds significantly to this discussion. Problem solving is developed on the basis of a morphological model of the 'Structure-of-Intellect.' Guilford is essentially an educational psychologist whose motivation for developing this model is the desire to identify and classify different aspects of human intelligence. She views thought processes as being constructed of operational and informational aspects where information is classified as to its "content" and "product." These three aspects of human intelligence, operations, content, and product then go to make up the three axes of Guilford's morphological model (Figure 1.1). It is the operational aspect of this model that is of concern in this section of the dissertation; information will be discussed in the third chapter. Guilford defined five possible operators for her model:

> Cognition--awareness, immediate discovery or rediscovery, or recognition of information in various forms; comprehension or understanding....information right now [13, pg. 203].
>
> Memory-- it is retention or storage, with some degree of availability, of information in the same form in which it was committed to storage and in connection with the cues with which it was learned [13, gp. 211].
>
> Divergent Production--generation of information from given information, where the emphasis is upon variety and quantity of output from the same source; likely to involve transfer [13, pg. 213].
>
> Convergent Production--the problem can be rigorously structured and is so structured, and an answer is forthcoming without much hesitation [13, pg. 214].

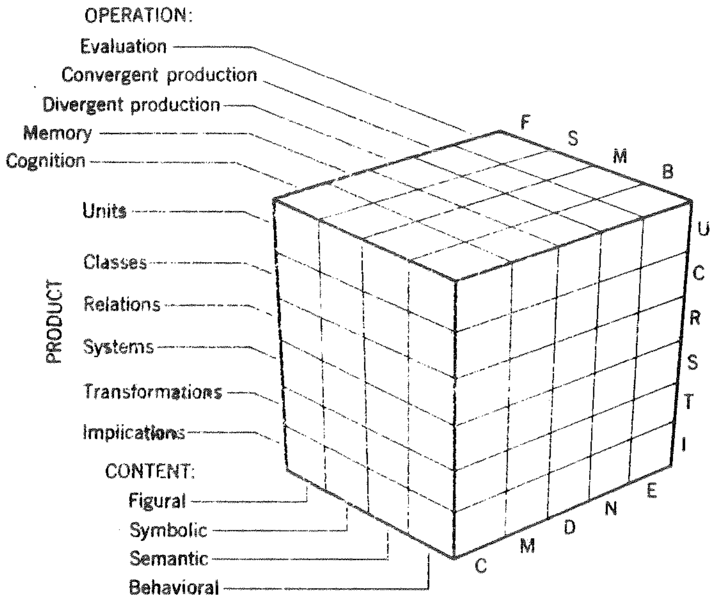


Figure 1.1: The Structure-of-Intellect Model [13, pg. 63]

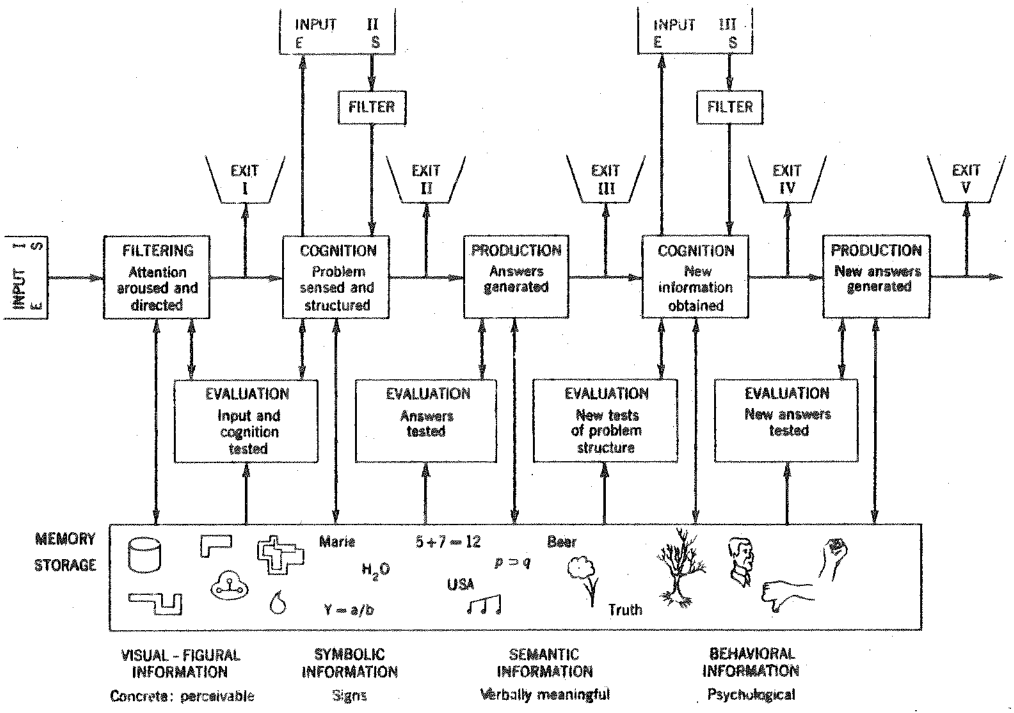


Figure 1.2: Model of Problem Solving [13, pg. 315]

> Evaluation--a process of comparing a product of information with known information according to a logical criteria, reaching a decision concerning criterion satisfaction [13, pg. 217].

These operators are then used, along with generalized information storage (memory), to develop an operational model of problem solving (graphically presented in Figure 1.2).

> The model is considered to be a communication system with inputs from the environment (E) and the soma (S). The latter are concerned with the behavioral information regarding the individual's own disposition, his motivational and emotional condition [13, pg. 313].

The most significant difference between this model and the ones previously discussed is the attempt to develop the operational model with respect to the possible sources of information, particularly the environment, the soma and the memory. These links are intended "to take care of the individual's active search for information....and also to take care of any incidental new input as the operations of cognition continue." [13,pg.314-316]. Multiple exits exist to account for possible rejection upon submission or rejection if the problem is found to be impossible to solve. Several synthesis-evaluation cycles exist, with the operation of evaluation "quite generally distributed, for there can be testing of information at any step of the way....One of the most important features of the model is the liberal allowance for looping phenomena, with the involvement of feedback information." [13,pg.314-316].

It is in Guilford's model that the connection between what have been termed the multiphase and mechanistic models can be observed. The identification of basic operators and their information content is one of the primary contributions of the mechanists.

The construction of a multiphase model from these basic building blocks is then well suggested by Guilford. The concept that problem solving is hierarchical is a direct outgrowth of the work of the mechanists along with the development of the medium in which any problem solving process must occur (information). An interesting example of this type of problem solving philosophy being directly applied to structural engineering (and therefore design) is found in a paper by A. Wong and G. Bugliarello, "Artificial Intelligence in Continuum Mechanics." [14].

1.2.3 A Problem Solving Model for Structural Design

A summary of the approaches to problem solving just reviewed is given in Figure 1.3. From this summary it is now possible to develop a model of the problem solving process as it might apply to structural design. This model then will serve as a focal point for the discussion of other approaches to problem solving and eventually act as a building block for a model of a structural design process. It seems appropriate here to remind the reader that the previous discussion was not intended to be a comprehensive survey but rather the presentation of selected concepts intended to provide a perspective of design as a human thought process. The key to the model to be developed comes from D. Johnson:

> The solution of complex problesm,...., is largely a process of producing and manipulating complex patterns: patterns of data at hand, response patterns to fit these, and, between these two, various instrumental or heuristic patterns. [4, pg. 228].

Here again is the ideal expression of Guilford's work, the identification of an operational process (production and manipulation) and

Proposed Section of Spiral	Dewey	Wallas	Rossman	Gestalt	Johnson	Information Processing (Simon)	Guilford
Problem Recognition NEED	Felt Difficulty		Difficulty Observed	Stimulation by the Situation Seeing the Problem			Problem Sensed
Continual Cognition (Insistent Analysis)	Location and Definition	Preparation	Problem Formulated Information Surveyed	Analysis of the Situation Resolution of Intrinsic Stresses Analysis of Goals	Preparation	Searching for Possible Solutions	Filtering Cognition
(Production) Synthesis	Suggestion of Possible Solution	Incubation and Illumination	Solution Formulated New Ideas Formulated		Production	Generating These Possibilities	Production
Evaluation	Development by Reasoning	Verification	Solution Critically Examined		Judgment	Evaluating Partial Solutions	Evaluation
Decision			New Ideas Tested Accepted				Exit
Design				Solution			

Figure 1.3: Summary Table-Models of Problem Solving

a medium (information or complex patterns) in which operations take place.

Problem solving is to be viewed as a process resembling a spiral that rises from a large body of undifferentiated, general, information to the specifics of a detailed solution. The process is initiated by the recognition of a need and terminated by the acceptance of a solution; for a structural designer, this is analogous to the transformation of a client's requirements into an acceptable design. Any segment of this spiral would be, in itself, a grouping of subproblems and in turn those subproblems could be partitioned until they become amenable to a basic information process. A section of this spiral would reveal a process at work very similar to the multiphase models. Figure 1.4 presents a graphic illustration of this problem solving spiral.

After the implanting of a need into the undifferentiated information in which this process is embedded, a cognitive operation is initiated. The product of this operation is the gradual differentiation of the information into groupings relevant to the designer. As the process moves upward, it becomes more definitive as is shown by a typical section in Figure 1.4. Out of this cognitive activity, referred to as "location" and "definition" by Dewey, "preparation" by Wallas and Johnson and "problem formulation and information surveying" by Rossman, comes the first primitive synthesis. Unlike many models, cognition is to be considered a persistant activity that cannot be isolated as a separate phase with the exception of the very early stages of the process when it is the only observable activity. This continual cognition is suggested by Duncker's "insistent analysis of the situation."

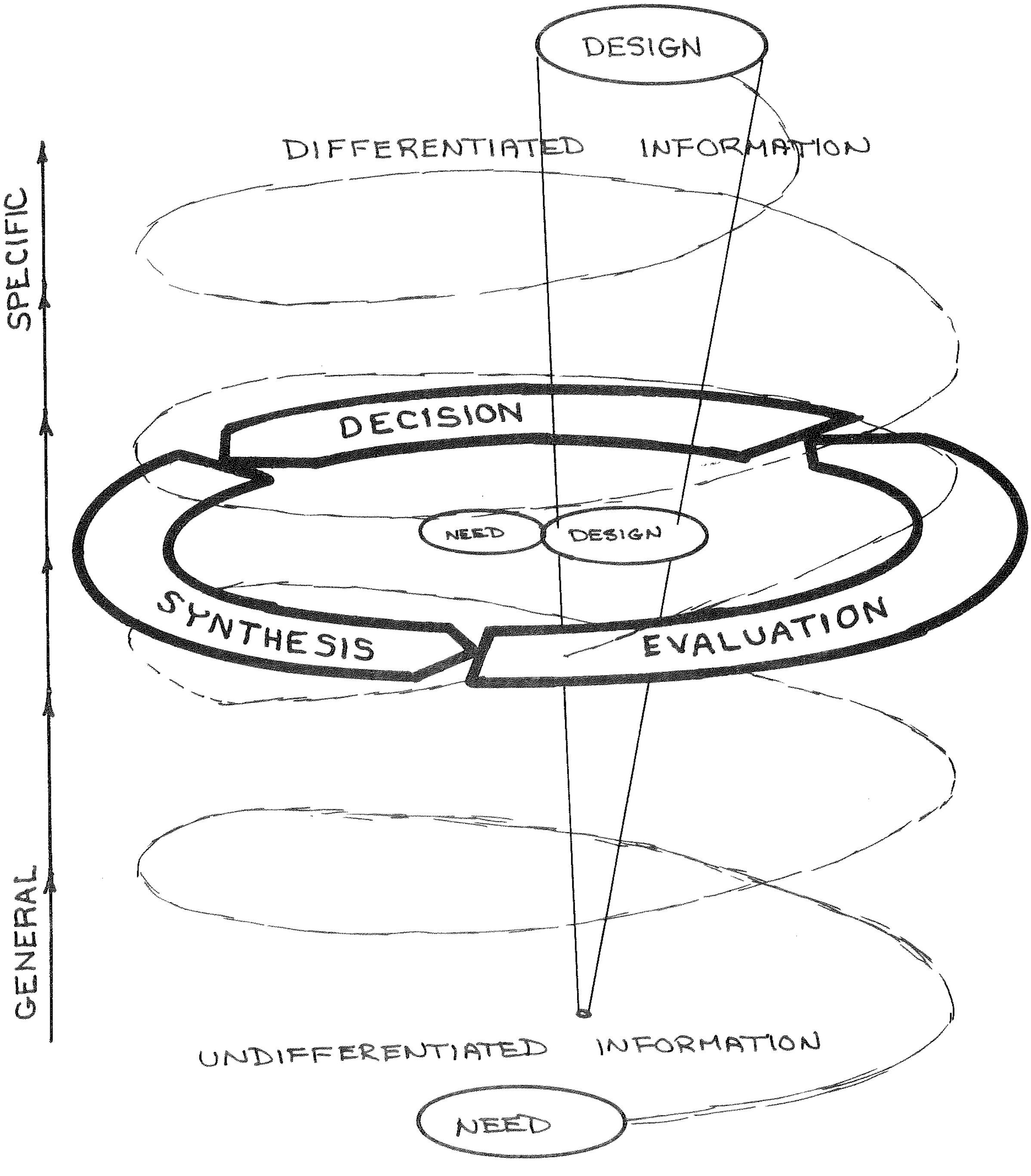


FIGURE 1.4 THE PROBLEM SOLVING SPIRAL

The process revealed in a section of the spiral is essentially iterative in nature with identifiable operations of synthesis and evaluation. These operations encircle areas labeled "need" and "design." This is to symbolize that the information provided by the initial need remains active throughout the process while the design (solution) is undergoing continual evolution. It may be that the need already contains aspects of the design and therefore that the two bodies, need and design, may overlap.

The evaluation process is seen as one concerning information dealing with both fact and value. This distinction is thoroughly discussed by Herbert Simon in his book, Administrative Behavior [15]. When a course of action must be chosen concerning a particular alternative solution (a product of a synthesis process), the structural designer may determine such factual information as a structure's behavioral response to an earthquake or its potential cost. This factual information does not conclude the process, for a decision requires an expression of value be associated with these facts. Evaluation for the designer then consists of two aspects: the attempt to ascertain facts about a design and the association of value with these facts. While the distinction made here is not always apparent in actual design processes, careful examination should reveal its presence.

Structural analysis and structural mechanics are classical examples of evaluative processes used by a structural designer, for here an attempt is made to predict the behavior of an alternative design in a simulated environment. Predicted behavior does not end the designer's evaluative task, for a decision must be made as to the acceptability of the behavior (e.g., does the predicted stress exceed some limiting allowable stress). Factual propositions (e.g., the resutls of structural analysis) are, according to Simon, something that theoretically can be "tested to determine whether they are true or false," but in real processes are often the products of human judgment, an educated or conditioned conjecture. To quote Simon:

> In making...decisions it is continually necessary to choose factual premises whose truth or falsehood is not definitely known and cannot be determined with certainty with the information and time available for reaching the decision [15, pg. 51].

The evaluation process identified in the previous review is now seen as bipartite: the "factual evaluation" and the "ethical (value) decision" (henceforth referred to as evaluation and decision).

This distinction between evaluation and decision is important to the structural designer in his understanding of the design process. An example of this is the current use of the term, "rational design process." Rationality as defined by Simon is "roughly speaking,...concerned with the selection of preferred behavior alternatives in terms of some system of values whereby the consequences of behavior can be evaluated." [15, pg. 72]. It is in this context that structural engineers also interpret retionality, and, therefore, the subject matter that the phrase, "rational design process" refers to is more appropriately labeled "rational decision processes in structural design."

The labeling of particular processes as either evaluation or decision may often depend upon one's perspective. An example of this labeling would be the response of a contractor queried by a structural designer as to the constructional aspects of a design, for the contractor would be making a decision which would serve as an evaluative fact for the designer. Here can be seen the hierarchical nature of a problem solving process in which the contractor must solve a problem which is only part of the designer's process. The designer, in turn, is really part of a larger process symbolized by the client.

In summary, problem solving is to be considered the transformation of a need, embedded in an undifferentiated body of information, into a solution that will satisfy that need. This transformation is a spiraling process in the presence of continuing cognition with distinct phases of synthesis, evaluation and decision. It now becomes possible to view the structural design process as the acquisition, manipulation and generation of information.

1.3 A Historical Perspective

Throughout history, structures have served as symbols of the progress of man. The ability to conceive and then construct a shelter or a bridge was man's first attempt, albeit unconsciously, at designing a structure. In contemporary society the structural aspects of an environment are generally the responsibility of an engineer. It is the problem solving process of this modern engineer that is often called structural design. Using a historical perspective, this section discusses those features of engineering design (particularly that of a structural engineer) that make it a unique form of problem solving.

1.3.1 The Builder

> No trace of the use of the word "Engineer" or any of the words from which it is derived in describing the building of any of the ancient structures had been found...engineers of great skill existed...but they were designated otherwise, as for example:...levelers,...bulders,...land surveyors, architects.... [16, pg. 5].

The introduction of the term, "engineer," was more than a semantic change. It is in the differentiation of the processes used by the ancient builder and the modern engineer that can be found a basic understanding of the structural design process.

Man probably discovered different structural forms by observing nature, a fallen tree acting as a beam or a rock fall creating a natural arch. By attempting to duplicate these observations for some conscious use, man began the painful process of learning.

> Stones placed one on top of another are maintained in place by exerting upon each other [a gravitational] force and the requirements of equilibrium are determined by the act of placing the stones. This is the empirical approach and by experiment, observation and deduction man's experience has been extended and stable structural forms have been developed [17, pg. 13].

It is in this empirical and intuitive sense that the ancient builder's carried out their tasks. The science of builders is found in empirical rules constructed from the continual accumlation of trial and error experiences such as the Roman arch span to pier width ratio [17, pg. 18]. There were exceptions to this use of empirical rule such as Archimedes and the school of Greek mechanists, but their influence was negligible. Rather it is the use of empirical rules reinforced by the concepts of Aristotelian metaphysics that was to influence the builder for nearly two thousand years.

> Aristotle had maintained as against the Pythagorean theory of Plato, that mathematics, though useful in defining relations between certain events, could not express the 'essential nature' of physical things and processes, for it was an abstraction excluding from consideration irreducible qualitative differences which, nevertheless, existed. According to Aristotle, the study of physical bodies and events was the proper object not of mathematics but physics (meta-physics) [18, pg. 130].

Thus, until the Age of Reason and the Industrial Revolution, structures remained the product of builders and architects operating with empirical rules and relying on intuition. One of the reasons the builders were unable to extend the work of the Greek mechanists was the admonition of Aristotle against the use of mathematics. Even such structures as the Gothic cathedrals which often exhibit a fine statical sense appear to be the product of builders with only an intuitive grasp of statics.

What was the design process of this period reaching from antiquity to the early industrial revolution? One in which the emphasis was on creation, upon synthesis. While this period was by no means homogeneous, empirical rules and intuition remained the keys to the builder's processes. And empirical rules and intuition are mechanisms for conceiving form and proportion, not evaluation. The evaluation function was one of feedback or experience, implied or stated in the rules of synthesis. The builder then was a true creator whose concepts were the total of man's creative experience. In terms of the problem solving process the builder operated essentially as a synthesizer for whom the rules of synthesis included experiential evaluation; thus the synthesis-evaluation cycle was dominated by the creative act.

1.3.2 The Engineer

> Nearly two thousand years before Galileo, we find, the Greeks were familiar with the principle of the lever, the pulley, the inclined plane, and the screw. Over those twenty centuries nothing further of interest was contributed in the development of modern structural engineering....Certainly to Galileo, who broke in method and in philosophy with all that before, must go our gratitude and our admiration....[19, pp. 822-825].

Thus wrote S. C. Hollister in his 1938 article, "Three Centuries of Structural Analysis," commemorating the three hundreth anniversary of the publication of Galileo's The Two New Sciences. Galileo did indeed examine problems in the realm of mechanics (e.g., the cantilever beam, since to bear his name), but this is not his real contribution to modern engineering. It was his break with Aristotelian physics and methodology, indeed it was the introduction of the scientific method, for which engineering remains indebted. Quoting A. C. Crombie from his Medieval and Early Modern Science:

> It was by his insistence on measurement and mathematics that Galileo combined his strictly experimental method with the second main characteristic of his new approach to science. This was to try to express the observed regularities in terms of a mathematical abstraction, of concepts of which no exemplaries need actually be observed but from which the observations could be deduced [18, pg. 140].

The idea that physical observations and mathematics could be combined to "deduce future observations" lays the foundation for the analytical evaluative procedures today associated with modern structural engineering. It required almost two additional centuries after the publication of The Two New Sciences before modern engineering practices began to be observed in the design of structures. Quoting S.

Timoshenko from his History of Strength of Materials:

> During the seventeenth century, scientific investigation developed principally in the hands of men working in academies of science. Few people were interested in the mechanics of elastic bodies...scientific curiosity was the principal motive power....During the eighteenth century, the scientific results of the preceeding hundred years found practical application and scientific methods were gradually introduced in various fields of engineering. New developments in military and structural engineering required not only experience and practical knowledge, but also the ability to analyze new problems rationally. The first engineering schools were founded, and the first books on structural engineering published [20, pg. 41].

As can be seen in the above quote, the transition from the builder to the engineer was gradual, where dates and events are at best symbolic. The word, "engineer," appears to have reached general usage by the early eighteenth century as is seen with the founding in France of the Corps des ingenieurs des ponts et chaussees in 1720. In 1742 the first chronicled use of structural analysis is found in the attempt "to ascertain the cause of cracking and damage" in the dome of St. Peters in Rome [21, pg. 111]. The establishment in 1747 of the Ecole des ponts et chaussees signifies the recognition of a need for engineered structures and a body of knowledge associated with fulfilling that need.

> The introduction of iron construction brought with it the necessary transformation of building from an empirical and pragmatic craft into a branch of scientific technology. This change began around 1750 and was one of the most pervasive economic and material consequences of the scientific revolution that occurred in the sixteenth and seventeenth centuries [22, pg. 76].

This observation from C. Condit's American Building acknowledges the transition from builder to engineer and suggests some of the complex influences that stimulated this change. While Condit suggests a date around 1750, it seems more appropriate to wait until the turn of the century to pick a symbolic date for the birth of modern engineering. An appropriate choice would seem to be 1794 and the founding of the Ecole Polytechnique. The introduction of a scientific curriculum-intended for application in meeting the needs of a society-is symbolic of the transition from the empirical, intuitive builder to the applied scientist, the modern engineer. It should be remembered that engineering is not only applied science but also that science serves as the core of engineering. It is through the understanding of science that the engineer becomes able to deduce future observations. In terms of the problem solving model, the modern structural engineer's processes will include an active evaluation function; thus, synthesis must now be confirmed through evaluation.

> From the dawn of time up to the eighteenth century, the master builder had remained a craftsman who, even in the design of important structures, was mainly guided by intuition....The advent of building statics as a science, signified the birth of modern structural design [21, pp. xiv-xvii].

1.2.3 The Influence of Evaluation-The Truss

The early nineteenth century found the works of such men as Euler, Lagrange, Musschenbroek and Coulomb beginning to coalesce into what is today structural engineering, an attempt to have a systematic and scientific understanding of the behavior of structures. An engineer could now attempt to predict the behavior of a postulated structure (a product of synthesis) and check for its theoretical acceptability, replacing the need for an entirely pragmatic trial and error approach. The design process became one in which evaluation could take an active role along with synthesis. It should be understood that many complex influences work upon the evolution of structural design, economics and changing materials for example, but the primary concern here is the influence of structural analysis and mechanics. The early development of trusses is briefly discussed to highlight the impact of the new evaluative capacity upon the structural design process.

The truss as a structural form has existed since antiquity; however, its form, generally of a king or queen post type, was limited by the nature of the available material-wood. In the early Renaissaince, Pallidio constructed several truss bridged which showed his clear grasp of their basic behavior. In the eighteenth century Swiss and German builders constructed complex bridges, generally of a truss-arch combination [20, pp. 182-183]. But, it is not until the end of the eighteenth century that the proper circumstances come to bear for the true development of the truss. Europeans had continually used masonry construction in an attempt to build lasting structures, but the expanding United States needed structures immediately; impermanence was not a concern. Thus, with an abundant supply of an appropriate material, wood, the truss provided a good solution.

Beginning with Palmer's bridging of the Merrimack River with a Palladian trussed arch in 1792 [22, pg. 63], many truss forms utilizing wood as a primary material were introduced (e.g., the Town, Burr and Long trusses). Many of these forms were highly indeterminate, often combining arch and truss action; however, these products of highly intuitive and inventive men met the needs of a growing America at the beginning of the nineteenth century. It was with the coming of the railroad and the availability of iron that the intuitive American builder began to falter. To quote Condit:

> As the engineer was faced with the constantly increasing demands for higher buildings and heavier, longer bridges, he was increasingly compelled to turn to science in order to solve the structural problems thrust upon him. Building could no longer be treated as an art or a craft; it had to become a branch of theoretical and applied science [22, pp. 76-77].

While the concept of static equilibrium expressed in the form of a funicular polygon had been mentioned as early as the sixteenth century by Simon Stevin [21, pg. 60], and Gauthey had described the proper design process for a truss by 1813 [22, pg. 60], it was not until Squire Whipple, with his publication of <u>An Essay on Bridge Building</u> in 1847 that a correct static analysis of a determinate truss was developed [20, pg. 85]. As this 'scientific' approach was applied to design as symbolized by Whipple's work and attempts were made to make optimal use of available materials, the indeterminate trusses of earlier periods yielded to more determinate forms such as the Pratt, Warren and Whipple trusses.

The structural designer now had a definitive means of predicting behavior and this initially hinged upon the concept of determinacy. This dependence was reflected in the trusses that were developed and it is this influence on conception that made the evaluative function of the engineer so pervasive. The ability to systematically evaluate replaced the trial and error function of the builder, and, in so doing, the synthesizing process began to reflect this new scientific understanding.

1.4 Summary

This chapter has endeavored to establish a perspective from which to observe and interpret a structural design process. Structural design was identified as a particular type of problem solving process. Using concepts from the field of psychology, a problem solving model was developed for application to a structural design process. It should be noted that the discussion of problem solving dealt with individual processes. Where needed it will be assumed that the related observations can be transformed to group activities.

Structural design was depicted as a spiraling process in which a need was transformed into a solution-the design. A section of this spiral revealed definite functions of synthesis, evaluation and decision surrounded by a continual cognitive function suggestive of the insistent analysis of the Gestalt school. A brief historical review established the importance of the evaluative function in modern structural design.

The problem solving process described was embedded in a medium of information. This led to the possibility of interpreting structural design as the acquisition, manipulation and generation of information.

The next chapter will use the perspective developed here to examine contemporary work in areas related to design processes. The third chapter will further extend the idea of information into an operational concept. Finally, as a product of these first three chapters, a descriptive model of the structural design process will be presented in chapter four.

CHAPTER TWO

APPROACHES TO DESIGN

2.1 Introduction

The conversion of resources and ideas into physical and spatial environments, objects and devices, or mechanisms for guiding behavior encompasses a broad spectrum of activities often called design. In the period following World War II, several factors caused people interested in design to examine their professions and to explore new approaches to the related problems. The common denominator was design and the assumption that an act of design was independent of what was being designed.

The reasons for this type of self examination are complex but seem to focus on certain common realizations:

1. The faltering of current approaches in the face of problems of increasing complexity.
2. The awareness that previous designs had not consistently solved problems as well as had been desired.
3. The introduction of Operations Research (OR) in solving problems dealing with complex systems having definite objectives.

This chapter looks at some of the concepts and methodologies resulting from this exploration of design. First, however, some of the basic ideas of OR are examined to help establish a perspective for this evolution of design concepts. The problems being solved in OR are seldom of the same type that concern designers; thus, OR is best understood as a problem solving process parallel to that of design. This differentiation was made by Joseph Esherick in his article "Problems of the Design of a Design System" [23]:

> ...OR problems are not design problems...What we can obtain from OR is not the direct example of immediately applicable methodologies but the indirect example of how they solved their own problems. [23,pg.76]

After the discussion about OR, approaches to design are divided into Design Methods, as generally related to environmental design, and engineering design. This separation is somewhat artificial being based on the nature of the evaluative functions available to the designer with the engineer associated with the more rigorous evaluative ability provided by science.

2.2 Operations Research

> OR is the application of scientific methods, techniques, and tools to problems involving the operations of a system so as to provide those in control of the system with optimum solutions to the problem. [24,pg.18]

This definition of OR was tentatively offered by Churchman, Ackoff and Arnoff in their 1957 book, Introduction to Operations Research [24]. This definition highlights three important aspects of OR: the systematic (i.e., scientific) approach, optimality, and control. One of the major differences between OR and design is found in the concept of control, for while the designer creates an entity, the operations researcher is involved in developing an optimal condition through the control of a system. The idea of optimality has also changed such that the concern may be more for a quasi-optimal state, one in which a system shows definite improvement. This quasi-optimal state is reflective of the realities of scientific modeling; that is, optimality implies a certain level of credibility that is often not possible.

2.2.1 The Approach

A generally accepted view of the OR process was presented by R. Ackoff in "The Development of Operations Research as a Science" [25] and contains six distinct phases:

1. Formulating the problem
2. Constructing a mathematical model to represent the system under study
3. Deriving a solution from the model
4. Testing the model and the solution derived from it
5. Establishing controls over the solution
6. Putting the solution to work, implementation

In some respects this process resembles the ideas previously developed in the general problem solving model. In the formulation phase, there are attempts to gather and differentiate information into a consistent concept of the system being studied. This formulation is similar to the effect of a need on a body of undifferentiated information at the beginning of a design process (often called preparation). The concept of the system developed in the formulation leads to the construction of a model that will simulate the behavior of the system.

The model is then used to establish the optimum configuration of the system. Because of the nature of model building, idealization and simplification, it is necessary to test both the model and the related results for their reliability and acceptability. The spiral started in the formulation has led through the model and its projected solution and reaches completion with the establishment of controls and implementation. The process is then a transition from a general situation involving some system to a specific course of action related to that system.

The model is the heart of the OR process and the basis for the systematic approach so often associated with OR. In some respects, the model may seem trivial in its actual functioning, but it provides the focus necessary for a successful study. The use of a simulation model is an integral part of the process and can be interpreted as an automated part of a problem solving spiral. In many ways the process of deriving a solution from the model is analogous to a section of the spiral, as will be shown later.

Since the modeling of a system is a primary focus of OR, the idea of a model and its related solution techniques will be examined further. It is the purpose of a model to represent the "essence of a problem;" however, it must always be kept in mind that it is only a representation of a reality and not the reality. A symbolic format for "any problem situation" is presented by Ackoff in his book, The Scientific Method [26]:

$$V = f(X_i, Y_j)$$

where V = the measure of performance or accomplishment that we seek to maximize or minimize.

X_i = the aspects of the situation we can control: the 'decisions' or 'choice' or 'control' variables.

Y_j = the aspects of the situation (environment of the problem) over which we have no control. [26,pg.28]

From this representation, certain major aspects of modeling become apparent. The researcher must define (or identify) certain "decision" variables which represent the aspects of the system being studied; that is, the variables that can be manipulated in determining the makeup of a system or control the functioning of a system. Only a finite number of these "decision" variables can be used, and it is their identification which becomes a primary concern of the researcher. The aggregate of these variables is often called a decision or solution space, an n-dimensional space in which a vector represents a possible configuration of the system.

In addition to the "decision variables," the researcher must also determine what conditions may influence the system and which cannot be controlled or for which there is no desire to control. These conditions outside of the influence of the researcher are called the context or environment of the problem. In order to judge the goodness of a possible configuration of the system, the measure "V" must be determined. This measure is usually the product of what is called an objective function, which is intended to reflect the values which are associated with the system. While it is not stated in the function presented by Ackoff, it must be assumed that the values of "X_i" are considered to be feasible; that is, their postulated values must be potentially viable choices. This question of feasibility, as opposed to value, provides one of the major questions to be resolved in actual operational models. Coupled with analytical techniques, the model becomes a tool which allows the operations researcher to find an optimal set of decision variables that represent a feasible alternative for a systems configuration.

There are many solution techniques available that influence the structuring of a model. Linear Programming [27] is probably the most widely known and well used of these techniques. These solution techniques are generally classified as mathematical programming and vary in their nature from linear to non-linear, from deterministic to probablistic, from discrete to continuous, and from single-stage to multi-stage. To better understand these techniques and to see their similarity to a section of the problem solving spiral, generalized, non-linear programming will be discussed.

2.2.2 Generalized Non-Linear Programming

> The general programming problem can be formulated as follows. It is desired to determine values for n variables $X_1,\ldots,X_n$ which satisfy the m inequalities or equations
>
> $$g_i(X_1,\ldots,X_n)\ [\geq,=,\leq]\ b_i \quad i = 1,\ldots,m$$

and, in addition, maximize or minimize the function

$$z = f(X_1,\ldots.,X_n). \quad [28,\text{pg}.1]$$

Here in this simple statement by G. Hadley, from his book Nonlinear and Dynamic Programming [28], is the essence of the generalized or non-linear programming problem. A solution is defined by a vector $\overline{X}$ in a n-dimensional space. The functional relationship suggested by Ackoff has been replaced by a series of functions: constraint functions (g_is) which delineate the feasible region of the solution space and the objective function (f) which supplies a value "z" (replacing "V").

These functions, "g_is and f," are completely general in nature. Often restrictions, such as continuity and convexity, are placed upon them in attempting to develop rigorous solution techniques. The severest of these limitations is linearity which leads to the classical linear programming problem. The objective function is generally thought of as representing a single value (e.g., maximum profit) but can just as easily be a weighted multivariate function. The question of combining values into a single measure is one of great difficulty; its detailed discussion is beyond the scope of the dissertation, but will be briefly discussed in later chapters.

Solution techniques can now be considered as searching for an optimal (as defined by "f") vector $\overline{X}$ in a constrained (by g_is) n-dimensional solution space. In linear programming this search is formalized in the simplex algorithm and an optimal solution can be found for "bounded problems." Easing the linearity condition to that of continuity and convexity still allows the development of algorithms that will guarantee optimal solutions. But in the general programming problems, there never can be any real guarantee of optimality. Here then is the first break with rigor, the introduction of the possibility of localized optimums. In all ill-conditioned solution space, an algorithm may lead to a seemingly optimal solution which, when compared to a solution from a different region of the space, may prove to have been only a local optimum, thus the distinction between local and global optimums. It is this loss of rigor that often makes mathematical programming as much of an art as a science.

Given a starting point, it is possible to look upon all search techniques as attempting to answer three questions: Have any constraints been violated by the proposed solution? If not, what is the value of the solution, and is it optimal? And if not, what is the next solution that can be proposed? Assuming an initial synthesis, often a random choice, mathematical programming follows some form of iterative cycle. The values of the constraint and objective functions are calculated, an evaluation process. Using the values of these functions, decisions are made dealing with feasibility and optimality. If the solution is infeasible or feasible and not optimal, a new solution must be generated, an act of synthesis. Thus, mathematical programming can follow a cycle similar to that observed as a section of a problem solving spiral, synthesis-evaluation-decision. Examples of synthesis in mathematical programming are the "Rosen Gradient Technique" or the "Random Walk" method. The modeling of the system, the constraint and value functions, provide the ability to evaluate. Decisions about constraints are often made by direct comparison, while those dealing with optimality may require such approaches as the "Kuhn Tucker Conditions."

A variation on the programming siutation just described occurs when there are no constraints, unconstrained optimization. For this situation there are algorithms that may be more efficient in obtaining a solution than those for the constrained case. Because of this, "penalty function" techniques have been developed. The penalty function technique provides a means by which constraints are eliminated by their conversion to supplemental values for inclusion in the objective function. This then allows the solution of a constrained optimization problem by the use of unconstrained optimization techniques. This conversion may be achieved by modifying the objective function in the following fashion.

$$g_i'(\overline{X}) = \begin{cases} 0 & \text{for } g_i(\overline{X}) \; [<, \neq, >] \; b_i \\ \infty & \text{for } g_i(\overline{X}) \; [\geq, =, \leq] \; b_i \end{cases} \quad \text{for } i = 1, \ldots, m$$

$$f'(\overline{X}) = f(\overline{X}) + \sum_{i=1}^{m} g_i'(\overline{X})$$

While the penalty function technique is of mathematical interest, its philosophical import is of more immediate concern. Here the combination of constraints and values poses an interesting question: Can constraints have real value? Constraints that are violated obviously have a highly negative value in the sense that they negate any possible value of a solution; however, do constraints met and exceeded have more than the passive value of feasibility? The answer to this question may depend upon the level of solution development, but it seems logical, for example, that if a designer found that his design would achieve greater earthquake resistance than originally required he might have a more positive feeling toward its choice as a good solution. The division into feasibility and value is a normal procedure in OR, but in looking for clues as to how a design process might function, the possibility of a more subtle relationship between constraints and objective functions in a decision function is highly appealing.

By looking at generalized nonlinear programming, its parallel nature to a section of the design spiral has been observed. In a search for an optimal or good solution, processes of synthesis, evaluation and decision are seen. In OR, solution techniques other than nonlinear programming are used (e.g., dynamic programming) which are not as directly analogous to the problem solving spiral, but all in some ways resemble human thought processes. Thus, what on the surface appears to be a very systematic problem solving process is really only a formalization of human processes which have existed for a long time.

2.2.3 Application in Structural Design

There is no intention here to review the area of cross-application between OR and structural design often called structural optimization or automated design; rather, certain observations are made to establish ideas helpful to the understanding of structural design processes. Structural optimization provides tools, tools for the designer, subprocesses often capable of replacing portions of the problem solving spiral. It is important to note that as the model is only a portion of the OR process so structural optimization is only a portion of a design process. As the operations researcher, the structural designer must define his solution space and the constraint and objective functions active in that space before optimization can take place; thus, this formulation becomes the design task.

The formulation of a simplified structural optimization model will help illustrate how a portion of a design process can be similar to an OR model. The example used is taken from a course presented by Professor G. G. Goble at the University of California in 1968. The problem is to design a beam of length "L" able to carry a uniform load "w" (see Figure 2.1). Using "engineering judgement," the designer decides that the beam is to be a rectangular wooden member that will be simply supported. The length, L, and the requirement of the load,

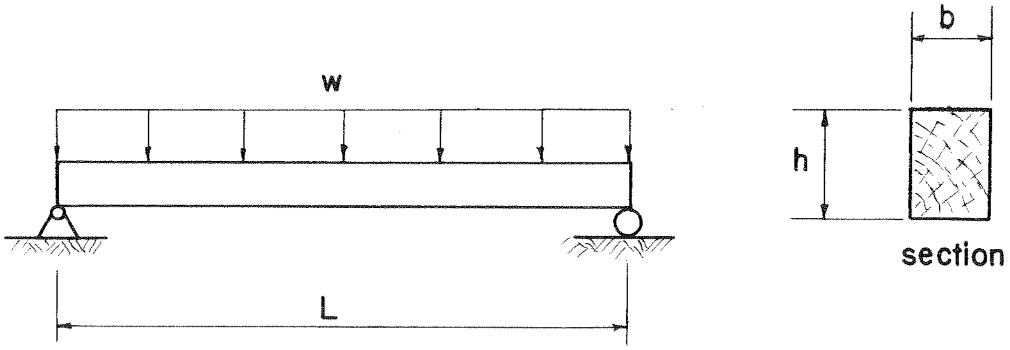


Figure 2.1: Beam Design Problem

w, express a need that the engineer desires to fulfill. The material (wood), the cross-section (rectangular), and the nature of the support conditions (pin and roller) are preliminary or conceptual engineering decisions. The combination of these quantities constitute the "design parameters," decisions that are no longer under question. The design parameters are similar to the problem context, variables which are out of the designer's control.

The questions left to be decided are the dimensions of the rectangular cross-section: height, h, and width, b. These variables, height and width, are called "design or decision variables." Two types of constraints are defined: "behavioral constraints" and "side constraints." Allowable stresses, f_b and f_v, and allowable deflections, δ_{all}, are examples of behavioral constraints, while side constraints usually reflect limitation imposed by prior design decisions (e.g., $h < H$).

"Behavioral functions" relate the design parameters and design variables to the behavior constraints. These behavioral functions come from structural analysis and structural mechanics, the areas that provide the foundations for modeling by structural engineers. To complete the model, an objective function must be identified, in this example minimum weight is used. The tabulation presented below shows this example problem in the formulation suggested by both Ackoff and Hadley.

General model used by Ackoff, $f(X_i, Y_j)$ = weight

X_1 = height, h

X_2 = width, b

Y_1 - length L

Y_2 = material, wood

Y_3 = cross-section, rectangular

Y_4 = load, w

Y_5 = support conditions, pin and roller

Generalized programming model - Hadley

$$g_1(b,h) < f_{b,\text{allowable}} \qquad \text{where} \quad g_1 = \frac{3wL^2}{4bh^2}$$

$$g_2(b,h) < f_{v,\text{allowable}} \qquad \text{where} \quad g_2 = \frac{3w}{2bh}\left(\frac{L}{2}\right)$$

$$g_3(b,h) < \delta_{\text{allowable}} \qquad \text{where} \quad g_3 = \frac{5wL^4}{32Ebh^4}$$

$$g_4(b) < B \qquad \text{where} \quad g_4 = 1b$$

$$g_5(h) < H \qquad \text{where} \quad g_5 = 1h$$

$$g_6(b,h) < A \qquad \text{where} \quad g_6 = b/h$$

$$f(b,h) = z(\text{weight}) \qquad \text{where} \quad f = \rho bhL$$

With this formulation, an appropriate mathematical programming technique can now be used to find the optimum configuration for the beam. The process of modeling and optimization represents only a small portion of the actual design process. Before the designer could construct the model, many major decisions had to be made; for example, deciding what material to use or what load should be used. The question now must be raised as to how these earlier decisions have been made. Is it possible that a process similar to that observed in the solution of the model has occurred previous to the modeling? That is, is a section lower on the design spiral similar to that automated in the optimization process?

Once the optimal cross-section has been determined, should it be immediately accepted or maybe checked against the designer's intuitive sense--his reality? Again, would this process be any different? In addition, the design must now be detailed (e.g., support conditions) before implementation; again, what is the process to be used here? At best, structural optimization is only a portion of a design process, but this automated process is analogous to the synthesis-evaluation-decision cycle of the designer regardless of his position in the problem solving spiral.

This brief discussion of model development has illustrated one cross application of OR to structural design; in addition, it has raised the questions of how the designer develops the problem to the modeling stage and what the designer does with the solution derived from the model. One last question must be asked in relationship to modeling: Is it possible that the drive to construct a model controls the designer's ability to approach a problem with an open mind? Or rather, does it provide a means by which a designer can crystalize his thoughts and develop a more thorough understanding of the problem?

2.3 Environmental (Architectural) Design

Creating environments which are complimentary to human activities has become one of the major problems facing technological society. Architecture, the profession traditionally charged with this responsibility, has had to expand its perspectives in face of this growing problem, thus the emphasis on environmental not architectural design. As this design task has increased in complexity, new approaches to the design of environments have been sought. The seeming success of Operations Research and its off-shoots, system engineering [29] and decision-making [30,31], have found an extension into architecture called Design Methodology (D.M.).

Design Methodology is a term equally well applied to areas other than environmental design, such as engineering, but as mentioned earlier, a distinction, although somewhat artificial, is being made between architectural and engineered design. It has been stated that design at best is a parallel process to OR, and in the case of environmental design, the system involved is far too complex for any direct modeling; in addition, OR generally looks for an optimal state of a system, while design attempts to create an acceptable entity capable of operating in a system (environment). The overwhelming complexity of environmental design has given rise to it being referred to as a "wicked" problem.

This "wicked" problem has been observed, dissected and restructured in search of new approaches. These new approaches in environmental design and related areas can be found in conference proceedings and compendiums: *Conference on Design Methods* [32], *The Design Method* [33], *Design Methods in Architecture* [34], *Emerging Methods in Environmental Design and Planning* [35], and *Design Methods--Seeds of Human Futures* [36]. (In addition, references 33 and 35 contain extensive bibliographies.) The last of these books, J. Christopher Jones' *Design Methods--Seeds of Human Futures*, is the most complete survey on D.M. presently available; because of this and Jones' excellent discussions on design, this work will serve as a primary source for the ensuing discussion.

2.3.1 Design Methods

In discussing D.M., the first problem encountered is lack of coherent usage of the term design. The only real accord found is that design is a process which requires a creative effort. The most primitive use of design is in reference to only the creative aspect of a problem solving process. A slightly more inclusive application of design is to the preliminary or conceptual phase of the process. Finally, the widest usage of the term "design" is in the description of an entire process from need recognition to need fulfillment. It is the last of these that is accepted as the scope of design in this discussion; therefore, it is in this context of the total process that methods are to be considered. Design methods are building blocks that, when combined with a designer's experience and intuition, can help construct an entire design process. In some cases, methods attempt to account for the complete design process; that is, they offer an entire strategy for the designer.

Jones, in his "State of the Art" presentations at the Portsmouth [34] and MIT [35] conferences and in his book [36], discusses design "methods from three points of view; that of creativity, that of rationality, and that of control over the design process!"

> From the creative viewpoint the designer is a black box out of which comes a mysterious creative leap; from the rational viewpoint, the designer is a glass box inside which can be discerned a completely explicable rational process; from the control viewpoint the designer is a self-organizing system capable of finding short cuts across unknown territory. [36,pg.46]

The black box designer believes that it is "irrational to expect designing to be wholly capable of a rational explanation." From this viewpoint, experience and intuition become the backbone of creativity, and related methods try to stimulate this aspect of design by attempting to eliminate psychological barriers. Jones cites both brainstorming and synectics as examples of methodologies inspired by this concept of the designer. Although considered sacred by some designers, even this idea of a black box designer is the subject of studies in information processing. While not attempting to simulate the physiological mechanisms of the mind, psychologists hope to develop their own "black boxes" which can simulate a designer's behavior; an example of this would be C. M. Eastman's article, "An Analysis of the Intuitive Design Process" [37].

The glass box approach is symbolized by words like "rational" or "systematic" and provides a picture of a designer much like a "human computer, a person who operates only on information that is fed to him and who follows through a planned sequence of analytical, synthetic, and evaluative steps and cycles until he recognizes the best of all possible solutions" [38,pg.6]. Jones lists the common characteristics of a glass box method:

> 1. Objectives, variables, and criteria are fixed in advance.
> 2. Analysis is completed, or at least attempted, before solutions are sought.
> 3. Evaluation is largely linguistic and logical (as opposed to experimental).
> 4. Strategies are fixed in advance; these are predominantly sequential but often include parallel operations, conditional operations, and recycling.
>
> [38,pg.7]

In this list of characteristics can be seen the strong influence of operations research and systems engineering. It is further suggested that glass box methods can be categorized as either splittable or unsplittable problem solvers. This separation is based on the idea that at one stage or another some design problems can be decomposed into logical subproblems such that each new subproblem may be solved and in turn these individual solutions combined for a total solution to the original problem. One of the early methodologies of the splittable type was presented by C. Alexander in his book Notes on the Synthesis of Form [39]. In "Notes" Alexander proposed the formulation of a problem as a graph where the nodes are called "misfit variables," statements which expressed the conflicts that had to be resolved by the design, and the linkages indicated the relationships between the "misfit variables." This graph was then decomposed into meaningful subgraphs and therefore subproblems. Variation on this idea is the work of Donald Stewart on "Partitioning and Tearing" of graphs [40,41] which attempts to tear variables or criteria into partitioned groups of highly related elements. Unsplittable processes attempt to handle the entire problem at one time; an example of this is Archer's "The Structure of the Design Process" [42] which will be discussed in the next section.

The last of Jones' classifications is the self-organizing system. This approach to design may be needed when the black box (alternative generating systems) or glass box (fixed often rigid approaches) designer is confronted by "a universe of unfamiliar alternatives too large to explore." The self-organizing designer is one who can develop a strategy for searching this "universe." In addition, it is necessary to have a good model of the "external situation that design is intended to fit" so that the solution and the strategy may be constantly evaluated.

> The way out of the dilemma of having too much novelty to evaluate all at once is to divide the available design effort into two parts:
>
> 1. that which carries out the search for a suitable design
> 2. that which controls and evaluates the pattern of search (strategy control). [36,pg.55]

One of the manifestations of this type of approach is the "creation of a meta-languate of terms which are sufficiently general to describe relationships between a strategy and the design situation" [36,pg.55]. The introduction of a "Pattern Language" by C. Alexander is this type of approach and will be discussed in the next section.

While Jones' black box, glass box and self-organizing designers provide a reference point from which to understand different methods, they are not intended as a means of classification. One categorization has been suggested by W. E. Eder: "(1) experience, (2) modification and running redesign, (3) check-lists, (4) design trees, (5) the fully systematic method, and (6) the system search" [43,pg.23]. Jones' offers two different types of categorizations for design methods: one considers how methods function in a design process and the other is based upon input/output couplets. Design is defined as having three basic phases: divergence, transformation and convergence. Divergence is a phase in which the boundaries of a design situation are extended "so as to have a large enough and fruitful enough search space in which to seek a solution."

Transformation "is the stage of pattern-making, fun, high-level creativity, flashes of insight, changes of set, inspired guesswork; everything that makes design a delight"[36,pg.66]. Convergence is the reduction of a range of options to a single chosen design as quickly and cheaply as can be managed and without the need for unforeseen retreats. Convergence is seen as the role of the traditional designer.

From this model of design, Jones identifies six types of methodologies: prefabricated strategies (convergence), strategy control methods, methods of exploring design situations (divergence), methods of searching for ideas (divergence and transformation), methods of exploring problem structure (transformation), and methods of evaluation (convergence). In the other method of classification, six types of design situations are identified and in turn methods are classified on the basis of which type of design sitation acts as input and what is desired as output. It is interesting to observe the similarity between Jones' concept of design, divergence-transformation-convergence, and that of the information processors, particularly Guilford, mentioned in the previous chapter. Seldom does a method fit neatly into a category, and in some cases, a method will fit into several categories depending upon its use. The simple fact is that D.M. presents to the designer new approaches in handling the problems of today's technological society. In this sense, methods are building blocks and not complete structures.

One last distinction should be made about methods. This distinction stems from the observation by Jones "that all design methods...are attempts to make public hitherto private thinking of designers, to externalize the design process" [35]. Design methods are considered operational concepts, but is that necessary for externalization? That is to say, if one of the important functions of design methods is to provide a means of "externalizing" design processes for observation, dissection and discussion, could this function not be equally well carried out by a descriptive methodology, a model of the design process? An example of such a model is provided in G. Best's article "Method and Intention in Architectural Design" [44], which will be discussed in the next section. Thus, the final distinction is between operational and descriptive methods which need not be mutually exclusive.

2.3.2 Examples of Design Methods

This section takes a brief look at four different methods. This examination serves to acquaint the reader with some typical methods, to establish ideas to be used later in the dissertation, and to reinforce and amplify ideas already presented. The first three methods are intended as operational, while the fourth is descriptive in nature. The first method is an example of a glass box, prefabricated strategy (convergence), from L. Bruce Archer's "Structure of the Design Process" [42,45]. A morphological approach is then examined [30,46,47], an example of a combination of glass box and black box strategies which Jones calls a "Method of Searching for New Ideas (divergent and transformation). The third methodology is a self-organizing system, both divergent and convergent, from C. Alexander's "Pattern Language" [48,49,50]. Finally, G. Best's article "Methods and Intention in Architectural Design" [44] is reviewed, in the course of which is offered a model descriptive of the information flow in a design process.

2.3.2.1 The Structure of the Design Process (Archer)

Archer is one of the most systematic of the methodologists. In his suggested approach can be observed the strong influence operations research has had on methodology. Of particular interest is the attempt at softening the nature of the OR model and the employment of a network formulation for developing a temporal strategy.

> There can be no solution without a problem; and no problem without constraints; and no constraints without a pressure or need. Thus, design begins with a need. [51,pg.4]

To satisfy this need, Archer envisions a design process structured out of a sequence of operational models coordinated by a design (project) program with a total effect of a "reiterative problem solving routine." The operational models are constructed from systematic and analogue models. The systematic model is a sequence of transforms and relationships that can evaluate a given design. Each operational model either handles a particular subproblem or some combination of subproblems. The synthesis portion of the model is provided by the insertion from the "real world" of suggested values for decision variables and design goals. The values are supplied by the "decision-makers" who control the problem. These values are determined only after the construction of the entire operational model. Four sets of variables are defined in the operational model:

1. Design variables which describe a design "i"
2. Context variables which describe the context "k" of the problem (i.e., the portion of the problem situation not controlled by the designer)
3. Properties which describe the performance "x" of a given design "i" in a context "k"
4. Objectives which are combined into a merit "y" that establishes the "value" of a given design.

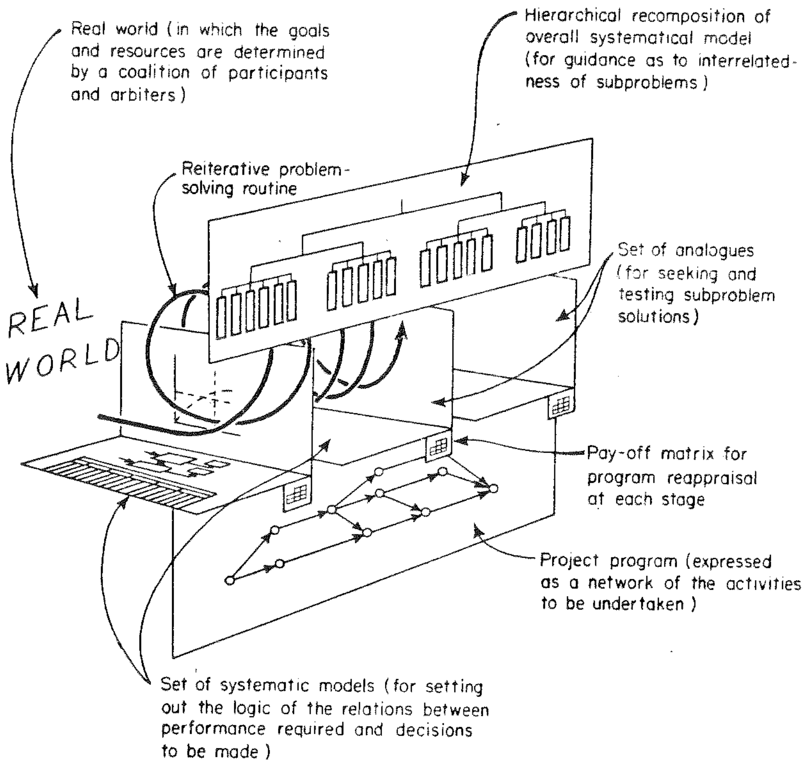
The operational model includes transforms that convert the input of design and context variables to an output of properties, convert the design properties into objective values, and convert objective values into a single measure of merit. The first of these transforms is the analogue model that combines with the systematic model to form the operational model. These analogues can be in the form of mathematical models of behavior, graphs or any other reasonable means of predicting behavior. The transforms relating performance to objectives are presented by Archer as graphical; however, there is no reason why these transforms could not have the same scope as the analogue models. The objectives are converted by a subjective weighting system to a single merity of "y."

Archer points out that the systematic model is iterative, "repeat...as often as necessary or as often as time and money will permit, until the merit...of overall performance is as high as possible" [45,pg.298]. The general sequencing of all these operations is controlled by a design program. In an early publication [51], Archer gave a program with 299 activities in 9 basic categories. A newer "plan of work," adapted in part from a publication of the Royal Institute of British Architects, is given below:

Stage A.	Inception
Stage B.	Feasibility
Stage C.	Outline Proposals
Stage D.	Scheme Design
Stage E.	Detail Design
Stage F.	Production Information
Stage G.	Bills of Quantities
Stage H.	Tender Action
Stage I.	Project Planning
Stage J.	Operations on Site
Stage K.	Completion
Stage L.	Feedback

The design program is then combined with the operational models in a graphical presentation (Figure 2.2).

> The design process may, therefore, be thought of as having three main components.
>
> 1. The advance through the project and through time indicated by the design programme and accomplished with the aid of various analogues.
> 2. The branching of the problem into its logical parts independent of time, indicated by the systematic model.
> 3. The cyclical movement through the subproblem, occupying man-hours but perhaps coexisting in time, connecting the real world, the systematic model, various analogues and the design programme as described by the reiterative routine... [42,pg.101]



Design procedure consists in applying a reiterative problem-solving routine to a complex of goal-decision systems in accordance with a project program.
The expediency of continuing, and of alternative courses of action, must usually be appraised at each stage.

Real world (in which the goals and resources are determined by a coalition of participants and arbiters)

Hierarchical recomposition of overall systematical model (for guidance as to interrelatedness of subproblems)

Reiterative problem-solving routine

Set of analogues (for seeking and testing subproblem solutions)

Pay-off matrix for program reappraisal at each stage

Project program (expressed as a network of the activities to be undertaken)

Set of systematic models (for setting out the logic of the relations between performance required and decisions to be made)

Figure 2.2: The Overall Structure of Design [45,pg.306]

"Design models tend to be concerned with the handling and arrangement of information rather than its content" [42,pg.237]. After presenting a highly systematic procedure for the designer, Archer tries to caution the designer not to get lost in the system. "What I am suggesting is that the information used and what you want to do with it may be more significant than the methods of arranging or handling the information" [42,pg.138]. The first step in a design should be the "conscious decision as to which [facts/information] are considered to be relevant." These warnings should be sufficient reminder that, no matter what method the designer uses, the ultimate decisions are his and his alone.

2.3.2.2 A Morphological Approach (The Zwicky Box)

The idea of a morphological box is usually credited to Professor F. Zwicky, thus the name "Zwicky Box." The morphological box is basically a combinatoric approach to identifying possible solutions to a defined problem. In this sense it is a divergent process that identifies all possible vectors existing in a design space. Zwicky outlines a five stage approach:

> First step. The problem which is to be solved must be exactly formulated.
>
> Second step. All of the parameters which might enter into the solution of the given problem must be localized and characterized.
>
> Third step. The morphological box or multidimensional matrix which contains all of the solutions of the given problem is constructed.
>
> Fourth step. All of the solutions which are contained in the morphological box are closely analyzed and evaluated with respect to the purposes which are to be achieved.
>
> Fifth step. The best solutions are being selected and are carried out, provided that the necessary means are available. This practical application requires an additional morphological study. [46,pg.285]

The basic idea is that the relevant parameters, the dimensions of a design space, can be identified and possible values assigned. With the definition of the design space, all possible vectors, alternative solutions, can be enumerated and evaluated.

In a space of any sizeable dimensionality, the number of possible combinations becomes prohibitively large, a combinatorial explosion. A. Kaufman [30] has suggested a partial solution by an ordering of the values of each dimension, so that basic morphologies (the design vector) with lower sums of their indices would be preferable and thus fully evaluated. This approach, while helpful, amplifies one of the problems of this method which is created when an alternative is not generated because a dimension or a value of a dimension has been neglected. Another attempt at eliminating this combinatorial explosion is the elimination of obviously incompatible morphologies in the enumeration process. This has been suggested in the AIDA Method reviewed in Jones' book [36,pg.310]; however, in a logical programming sense, this also can become prohibitive in a problem of any size. Combinatorial explosion is not the only problem of this obviously divergent technique. The solutions are very sensitive to the defined space, and no logical process has been suggested for defining this space. Even with these deficits, the morphological box can be helpful to designers, for in a well defined problem, it may stimulate the exploration of design alternatives which might have otherwise remained dormant.

2.3.2.3 Pattern Language (C. Alexander)

The recent work of C. Alexander and his associates at the Center for Environmental Structure in Berkeley, California, is directed toward formulating a new approach, Pattern Language. While Alexander now considers himself an anti-methodologist [52], his development of the Pattern Language is in reality another method. The Pattern Language does not find its origins in OR, as so many methodologies do, but rather it is better understood from the study of the psychology of problem solving. It is because of this relationship to problem solving that the Pattern Language is of particular interest.

> ...each pattern has two parts: the PATTERN statement itself, and a PROBLEM statement. The PATTERN statement is itself broken down into two further parts, an IF part and a THEN part. In full, the statement of each pattern reads like this:
>
> IF:X THEN:Z / PROBLEM:Y
>
> X defines a set of conditions. Y defines some problem which is always liable to occur under the condition X. Z defines some abstract spatial relation which needs to be presented under the conditions X, in order to solve the problem Y. [48,pg.15]

Patterns are defined in the resolution of conflict, a problem, existing between the present design situation and the desired solution, the design. These conflicts are similar to the stresses and strains postulated by Gestalt psychologists as the forces that guide a person into the resolution of a problem, the solution. The patterns are related by a "meta-language" which allows them to coalesce into designs. The language is viewed much as natural language exists.

> In order to design with the language, you must internalize the structure of the language; once you have it in your head, and it has become automatic, then you can use it to design. [48,pg.51]

It is this language that provides the process through which these patterns are brought together into designs. This idea of patterns and a related language seems analogous in concept to Johnson's "producing and manipulating complex patterns."

The Pattern Language is not anti-methodologist; it is different only because it has rejected OR as a foundation and found more primitive roots from which to grow. It is a serious effort at externalizing and improving the previously internalized patterns from which designers have traditionally synthesized.

2.3.2.4 Method and Intention in Architectural Design (G. Best)

In "Method and Intention in Architectural Design," G. Best develops the idea that "designing can be usefully interpreted as a variety-reducing process." In order to review different approaches to design, a designing model is presented:

> The designing model chosen is a distortion of reality. It is not a psychological model; it is a logical picture of the way in which information flows in a designing situation; .... [44,pg.151]

While not intended as a methodology, it is this model (see Figure 2.3), in reality a descriptive method, which can be used in externalizing design processes.

There are four main components to this model: input, output, process, and control. "Input is meant to stand for the external classes of information that are specifically fed into a design situation." Outputs are the information that a designer produces: drawings, specifications, reports, etc. These inputs

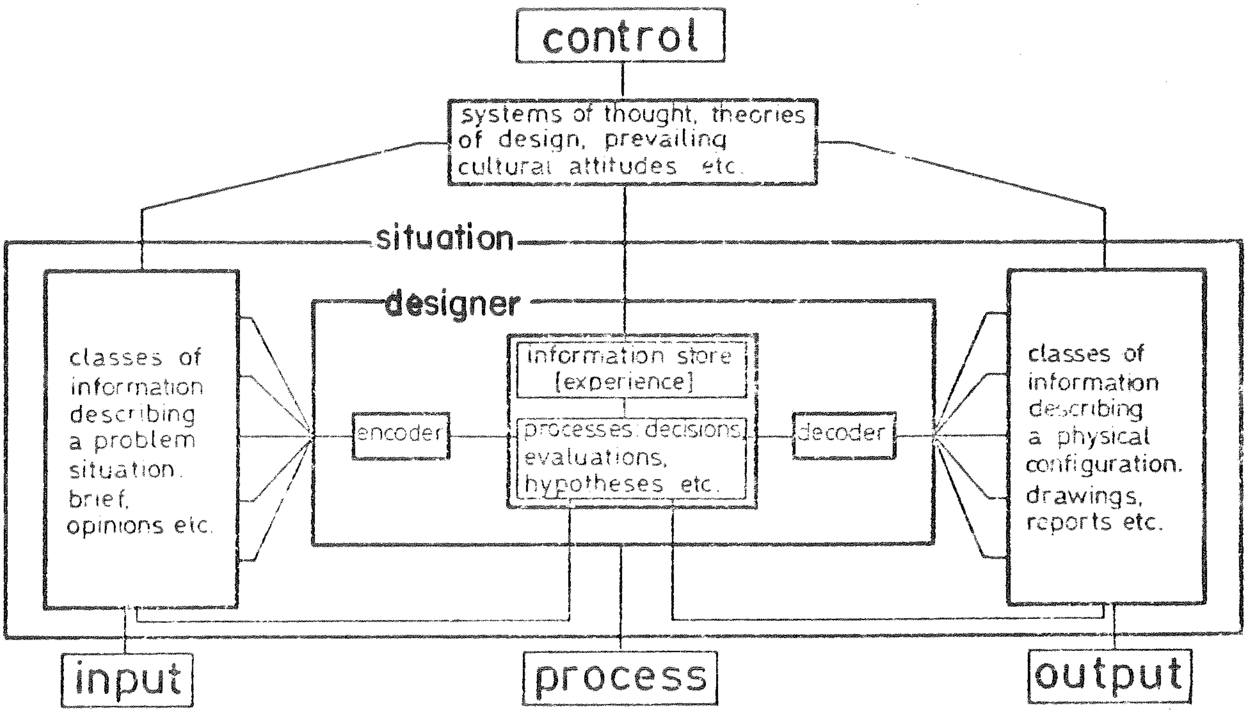


Figure 2.3: Information in Design [44,pg.151]

are transformed into outputs by the designer who is envisioned as a three part model. The encoder transforms information into the designer's processing unit, and the decoder transfers information from the processing unit back into the external world. The processor has a memory store and processing instructions (e.g., instructions for evaluating and deciding). This concept of the designer as a processor is very similar to Guilford's model of the intellect.

The input, designer, and output make up the design situation. External to the design situation is control, the influences "outside the specific design situation..., it is meant to include systems of thought, current architectural or other design theory, cultural attitudes, and so on...It is important to think of each box as changing state with time...eventually a design configuration emerges" [44,pg.152].

Best concludes that the design approaches he has reviewed are variety restricting processes (situation-interpretation-realization) and in turn develops the concept of design as a variety regulating process (situation-interpretations-variety regulation-realization). Regardless of the viewpoint, restricting or regulating, Best sees design as a process initiating in "unstructured information" which is gradually reduced to structured information and then to a solution. This view of design is very similar to the transformation of undifferentiated information into a design described in the problem solving model for design developed in the first chapter.

2.3.3 Observations on Design Methods

Architectural design has traditionally been a synthesis oriented process, similar in many ways to the processes earlier associated with the builder. Design methodology is an attempt to break with this traditional approach, the internalized black box. Part of this reformation is found in the development of evaluative techniques that deal with human behavior; basically, the developments of techniques for evaluating the effect of environments on people. Unfortunately, or maybe fortunately, behavioral models are much more complex than physical models. The Pattern Language is the only one of the three operational models reviewed that attempts to come to grips with this behavioral aspect of design. Both Archer's design process and the morphological approach are rigid and only applicable in very specific situations.

Every designer must accept or determine his own working processes, which may or may not include design methods. But it is in the externalization of the process that progress is critical. A designer that escapes into the black box of security, to be free of criticisms and blame for failure, contributes nothing to the furthering of better environments. Thus, while operational methods are helpful to the designer, the impact of description (externalization) may eventually be the most important influence of design methodology.

2.4 Engineering Design

Methods in engineering design have a great deal in common with those associated with environmental design. While there is this commonalty between architecture and engineering, they differ in one important aspect--the evaluative models used in engineering generally deal with physical behavior. This does not negate the role of human factors in engineering, rather it recognizes that there are viable models available to answer many of the questions asked by engineers. Because of this availability of models for predicting physical behavior, engineers often approach design in a more rigorous manner. The rest of this section is divided into two portions: the first covering some general models of engineering design and the second looking at specific applications of new methods to structural design.

2.4.1 The Engineering Design Process

While engineering has not yet found itself in the same problematic situation as architecture, the sputnik age, both in its impact and need for systems engineering, has rekindled inquiry into the meaning of engineering design. This challenge and the introduction of the digital computer have drastically reshaped the scope and capabilities of the engineering profession. The ensuing interest in the design process can be seen in conferences on engineering education which have included discussions on design [53], the participation of engineers in the field of design methodology and in the publication of books on engineering design. Some typical books on engineering design are H. Buhl's Creative Engineering Design [54] in 1960, M. Asimow's Introduction to Design [55] in 1962, J. Alger and C. Hay's Creative Synthesis in Design [56] in 1964, and T. Woodson's Introduction to Engineering Design [57] in 1966.

Asimow's work contains a widely accepted representation of the design process, often cited in related literature and discussed later in this section. Buhl describes design as a seven stage process: recognition, definition, preparation, analysis, synthesis, evaluation and presentation. Alger and Hays offer a six stage description: recognizing, specifying, proposing solutions, evaluating alternatives, deciding on a solution and implementing. These two models of engineering design, while not in total accord, are typical in their interpretation and worth noting here for their inclusion of a period of preparation or analysis and concluding the models with implementation.

The problem solving model offered for the design process in the first chapter included only three specific functions, synthesis-evaluation-decision; however, both Buhl and Alger and Hays identify some period that can be called preparation or analysis. It is the contention of the problem solving model that preparation or analysis are continuing processes that cannot be isolated. While an engineer may spend time involved in preparing or analyzing data, in a case where an active state of synthesis-evaluation-decision is not observable, it becomes possible that this cognitive function is then the sole activity. The identification of an active function (implementation) as the end of a design process refers to the fact that a solution, in itself, seldom ends such a process. Rather, a solution transformed into need satisfaction is a more appropriate ending.

The general model of the design process offered by Asimow and another model proposed by D. Ramstrom and E. Rhenman [57] is the subject of the rest of this section.

2.4.1.1 The Engineering Design Process (Asimow)

The basic principles from which Asimow builds his design model are found in his "philosophy of design" (graphically represented in Figure 2.4).

> A philosophy of engineering design comprises three major parts; namely, a set of consistent principles and their logical derivatives, an operational discipline which leads to action, and finally a critical feedback apparatus which measures the advantages, detects shortcomings, and illuminates the directions of improvement. [55,pp.45]

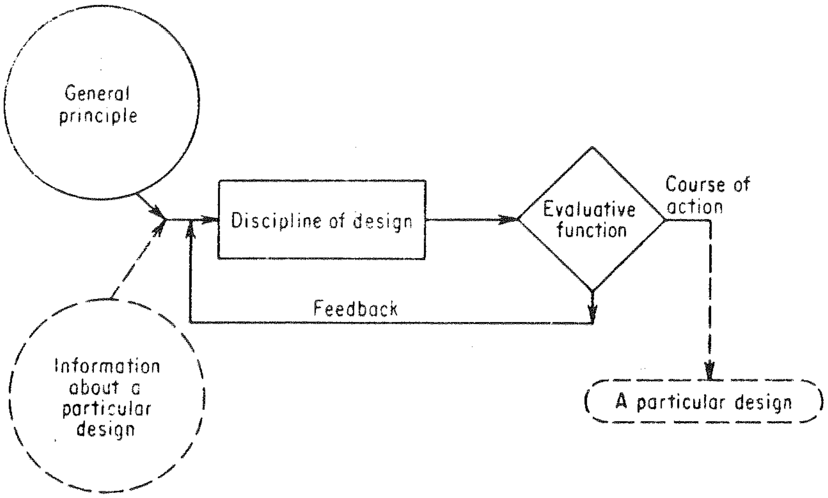


Figure 2.4: Philosophy of Design. The feedback becomes operable when a solution is judged to be inadequate and requires improvement. The dotted elements represent a particular application. [55,pg.5]

The design process resulting from the above design philosophy is divided into a vertical and a horizontal structure. The vertical structure is temporal in its nature and called the design "morphology"; it progresses from the "abstract to the concrete." The morphological structure of the design is rooted in a "primitive need" from which is initiated a process divided into seven phases: feasibility study, preliminary study, detailed design, planning for production, planning for distribution, planning for consumption, and planning for retirement. The first three of these phases encompasses the development of a design while the last four deal with the "production-consumption cycle" of the design. Asimow's audience for his book appears to have been mechanical and not structural engineers which would account for the emphasis on the production-consumption cycle; thus, for structural design, it would seem appropriate to consider these four phases as the implementation of the design.

The horizontal structure of design is called the "design process" and is defined as "an iterative problem solving process."

> The design process resembles the general process of problem solving in the main features, but it uses sharper, and for the most part, more analytical tools, which have been especially shaped and sharpened for the problems of engineering. It carries the process through *analysis*, *synthesis*, and *evaluation* and *decision*, and extends it into the realms of *optimization*, *revision*, and *implementation*. [55,pg.44]

This concept of a design process is almost identical to the section of the problem solving spiral with the exception of the analysis as a particular function. What does "analysis" consist of? A "statement of the problem" is the objective of the analysis. This "statement makes clear what goals are to be achieved, what difficulties must be overcome, what resources are available, what constraints will circumscribe any acceptable solution, and finally, what criterion should be used to judge the goodness of a possible solution" [55,pg.45]. It is just such a function that has been ascribed to the continual cognition in the problem solving process, the function of differentiating information so that the processes of synthesis-evaluation-decision may exist. This is a function of continual change and evolution in which the rest of the process is embedded.

Asimow also attributes three additional qualities to design: optimization, revision, and implementation. Optimization is the search for a "best" solution reminisent of an OR approach. This desire for optimality is a legitimate function for an engineer, but it must be understood that, in real engineering problems, optimal solutions do not exist--only a "good" solution can exist. In this sense, when the constraints of real and not ideal problem solving exist, the separation between "best" and "good" solutions becomes negligible, and optimization then does not truly differentiate engineering design. The need to revise, that is, to return to an earlier point in the process, is something often associated with design, but again, it is not unique. Lastly, implementation, while of great importance to the engineering designer, is really no different than the average person responding to a realized solution of a problem with which he is confronted. Asimow's inclusion of these three points is reasonable, but they are not attributes that delineate modern engineering design as a unique form of problem solving.

The design morphology and the design process can now be combined to give the same type of spiraling process envisioned in the first chapter. The design morphology functions as an axis for the problem solving spiral and in its first three phases, symbolizes a transition from the general (abstract) to the specific (concrete). It can be concluded that this model is very close in philosophy to that of the problem solving process presented in the first chapter.

2.4.1.2 "A Method of Describing the Development of an Engineering Project" (Ramström and Rhenman) [58]

The intent of the work of Ramström and Rhenman was the development of a method for describing and analyzing the progress of an engineering project. The motivation for this study was to arrive at an understanding of how engineering projects are administered; thus, the scope of the proposed model extends somewhat beyond the usual model of design into the control of design by sources outside of the engineer. To achieve their goal, the authors felt it imperative to develop a descriptive model of an engineering design process. With this model, they were then able to develop the protocol of an actual design project which they felt confirmed their basic model. In the model, design is considered as a problem solving process and was based upon the identification of the "dimensions of the project."

> The process of problem-solving in engineering work can be described formally as a transformation of the problem defined in the space formed by the need dimensions to the solution given in the space formed by the product dimensions. That is to say, problem-solving involves
> (1) choosing relevant dimensions
> (2) assigning a value to these dimensions.
> An essential characteristic of the problem-solving process is thus the transformation of need to product dimensions, a transformation which normally takes place via the utilization of control and engineering dimensions.
> [58,pg.81]

The need dimensions are the description of the project supplied by the customer somewhat similar to Asimow's primitive need or the need that initiates the problem solving process. The product dimensions are what delineate the design space, essentially decision variables. The possibility that a need dimension may overlap into the product space is suggested by Ramström and Rhenman, that is, that the need may already contain a significant part of the product space.

The transformation from the need to the product is accomplished with interim dimensions of control and engineering. Control dimensions are the restraints imposed by the management, an example being a customer's time requirement (need) transformed into a time scale for the engineer's work. It is the engineering dimensions which act as the real guide for the transformation from need to product. These dimensions are an expression of engineering criteria. The authors observed that the engineering dimensions were often an operational expression of non-operational concepts expressed in need dimensions.

> In general, these dimensions (engineering) are utilized by applying the relevant engineering criteria and norms (e.g., safety factors in connection with risk for fatigue failure). The criteria are a convenient way of summarizing previous experience; the usual means of access are engineering handbooks, the unit's aresenal of data from previous experiments, etc.
>
> Control and engineering dimensions refer, in other words, to all those aspects upon which attention is focused during the transformation of a problem (expressed in need dimensions) to the solution (expressed in product dimensions). [58,pg.8]

In a manner similar to Asimow, the authors introduce a sequence of concepts to temporally relate the design's progress. "Limitation" is seen as the process "ultimately leading to a limited number of solutions and often a unique solution."

"Generalization" is a "step backwards" when a limitation on a dimension is no longer deemed appropriate. "Change" is the discovery of a new sequence of solutions as yet explored, in a sense the discovery of a new portion of the design space. Finally, "reformulation" is the redevelopment of previously defined dimensions. These concepts are not as well related to a time line as Asimow's morphology but do provide a basis for following the progress of a design.

"In conclusion, we may thus say that engineering work consists simultaneously of transforming values from one set of dimensions to another and limiting the alternative courses of action." [58,pg.82] This descriptive model differs in concept from that of the problem solving model in that it is based on the identification of the products of processes and not the processes themselves. In this sense one can see the engineering dimensions as the focus of evalution, the product dimensions as the product of synthesis and the progress of the project (i.e., limitation, generaliztion, change and reformulation) as the result of decisions. Ramström and Rhenman have presented a model that identifies the descriptive and not operational content of engineering design, and in this sense, they have developed a model parallel to the problem solving model.

2.4.2 Applications in Structural Design

This section is an inquiry into the current development of concepts and methodologies for structural design. Improving evaluative modeling (the understanding and predicting of structural behavior) is an essential ingredient for better design; however, it is the way that modeling functions in the design process, not the modeling, that is of interest here. Like other areas with a developing consciousness toward design, much of the work being done is oriented toward making use of digital computers. Indeed, one of the few non-computer oriented manners in which design is discussed is in the occasional documentation of case studies, project histories and failure reports found in professional journals and magazines. This recording of past experiences and their evaluation (feedback) is similar in many ways to Hopkin's "stones placed on top of another."

The research cited here is not intended as a survey but rather, as in previous sections, is included to give the reader an idea of what is occurring in the field. The majority of this work can be classified in one of the following categories: structural optimization, rational decision theory, information storage and retrieval systems, information control systems, man-machine interface and artificial intelligence. The areas of structural optimization and rational decision theory have already been mentioned and require no further discussion. While of interest, work done on man-machine interface problems (e.g., computer graphics) will not be covered. It is worth noting that an improvement in the man-machine interface may well prove to be an effective way to expand the capabilities of the designer.

One of the most basic uses of a digital computer is in the area of information storage and retrieval (a sophisticated library). A common example of this type of approach is in the computerization of design specifications, two different examples being the BUILD and STORE systems. The BUILD system [59], developed at MIT, is intended to aid the designer by storing "the current state of the design." This "state of the design" is a description of the design as a physical object. A typical advantage of this type of system is found in the existence of a common information base for all designers working on the same project. BUILD is also conceived of as more than a simple information storage and retrieval system in that it is given certain operational capabilities such as helping the designer generate the description or providing quantity estimations based upon the stored description. This type of operational capability provides the nucleus for an information control system.

The STORE system [60] "presents a method for the development, exchange and use of computer programs and information in structural mechanics, analysis, design and research" [60,pg.IV-4]. This library of programs and information elicits interaction from its users by requesting commentary, additions and corrections. STORE is envisioned...

> as a convenient tool for increasing the usefulness of key engineers and scientists by providing them with essentially a desk capability of using a continuously updated library of current research results and methods in structural mechanics. [60,pg.IV-4-2]

The information space, that is, the basis for indexing the library, has three dimensions: Type of Structure, Load/Environment, and Analytical Model. By choosing values for these dimensions, the designer or research is able to obtain the latest available information relevant to his research or evaluation problem.

A more sophisticated approach toward the information handling capabilities of the computer is found in the area of information control systems. In this approach, a set of operational capabilities is provided that allows the generation and/or manipulation of information. The development of "Constraint Processing" [61] and "Decision Logic Tables" [62], at the University of Illinois, falls into this category. "Constraint Processing" uses a topological approach to examine the sensitivity of a design, after evaluation, to its related constraints. An interesting conclusion of this research is the suggestion that constraints should be considered an attribute, that is, associated with a value. The work with decision logic tables establishes a logical structure for design specifications (e.g., the 1969 AISC Code [63]) so that they are amenable to ideas such as "Constraint Processing."

Artificial intelligence, previously called information processing in the first chapter, has been the basis of some research in structural engineering and mechanics. The work of Wong and Bugliarello, "Artificial Intelligence in Continuum Mechanics" [14], resulted in the development of a computer program capable of simulating certain capabilities associated with a mechanist. A program, called CONFORM, was developed with the capability of...

> the recognition (by their pattern)of formulas and expressions, and their logical manipulation, complemented by a capacity to step directly, when desired from symbolic expression to the performance of numerical evaluation. [14,pg.1239]

The basis of this program is a library of "knowledge cells" which are packages of operational information dealing with fluid mechanics. The program can retrieve and expand these cells for the solution of problems in mechanics.

Another example of this type of approach is seen in Spiller's and Friedland's work on "Adaptive Structural Design" [64]. The concept of adaptive design is to provide an automated technique for improving on an initial design. The problem simulated by this research was the addition of joints and members to a preliminary truss design. This technique is directed at the improvement upon a point already identified in a design space.

Most of the work cited covers a narrow range of a design process. Models of the structural design process are presented often in conjunction with this type of research. The presentation of these models is generally an attempt to justify the approach taken to the research (e.g., "The complete design process can be considered consisting of the following three steps: (a) analysis, (b) sizing of components, (c) checking of constraints." [62,pg.2]). In general, the interest shown is in the latter stages of a design process (i.e., Asimow's "detailed design") and not in what may precede or proceed those stages. However, all of this research is symbolic of an interest in design beyond that of behavioral modeling. It seems reasonable that before attempting to provide tools for a designer, an attempt should be made at understanding what that designer is, or should be, doing. It is to this question that this dissertation has addressed itself.

2.5 Observations and Conclusions

This chapter has breifly examined operations research, design methodology, and engineering design in assessing possible approaches to the structural design process. The concepts and methods discussed were compared to the problem solving model developed in the first chapter; none were found that directly conflicted with the basic ideas expressed by that model.

Operations research was defined as a problem solving process parallel to that of design. The process was seen to center around the construction and application of a model describing the system being studied. The process of building and using the model, along with the techniques for deriving optimal system configurations from the model, were found to be consistent with the problem solving model previously presented. The concepts of design spaces, feasible solutions, problem contexts, and objective functions (value of solutions) were introduced. The existence of the penalty function solution technique suggests the possibility of associating a value with a constraint; a reinforcement of this idea was found in the work on constraint processing in engineering design.

The area of design methodology was seen to evolve from a desire to understand and improve design in its modern context.

Several methods were reviewed, of which the morphological approach will be made use of later in the dissertation. While design methodology can be seen as an attempt to supply tools for the designer, it can be equally well interpreted as having two other important functions: first, the externalization of the design process, and second, the creation of a basis from which a dialogue about design may take place. It is in light of this latter function that both Best's and Ramström's and Rhenman's descriptive models were presented to show that methods do not have to be operational in order to contribute to the better understanding of design.

In reviewing engineering design, the ideas of analysis (or preparation) and implementation had to be resolved with the problem solving model. Particularly the idea of preparing of analyzing data had to be equated to the continual cognitive function found in the model. The design model presented by Asimow compared very favorably with the problem solving model. Unlike previous models, the model presented by Ramström and Rhenman dealt with the products of functional relationships (dimensions) and not the functions (transformations). While being of potential importance to the designer, the work done on structural design does not provide for a basic understanding of the process involved.

It is the purpose of this dissertation to explore the structural design process and, in doing so, to provide a basis for the better understanding of structural design. This goal will be pursued by the development of a descriptive model of the structural design process; one general enough to include present approaches and yet detailed enough to suggest possible directions for the future. The problem solving model from the first chapter will act as a basis for this model along with ideas presented in this chapter. The problem solving model at this point consists basically of processes, but not of what the processes and their related input and output are constructed. The next chapter will develop the idea of information in design, for it is information from which processes are built and upon which they operate.

CHAPTER THREE

INFORMATION IN DESIGN

3.1 Information

A conclusion of the first chapter was that design could be interpreted as the "acquisition, manipulation and generation of information"; however, the question of what was meant by information remained unanswered. This term, "information", has been submitted to many definitions from strict mathematical theories to philosophical discourses. This chapter will look at some of these concepts and develop a definition of information that will be applicable to the development of a model of the structural design process.

T. Woodson in his book, Introduction to Engineering Design views design as a series of information transforms where information is "facts, data, unorganized knowledge or intelligence" [57, pg. 41]. This type of vague concept does not provide any meaningful insight into possible meanings of information. The complexities of developing a concept of information can be demonstrated through an example.

If a person were shown the sequence of symbols, 29500000, what meaning might they associate with it? If that person happened to be a structural engineer, he might well associate the sequence with the Modulus of Elasticity of steel, while to anyone else the sequence would probably be meaningless. How is it that a strucutral engineer can find meaning in this sequence of symbols (commonly referred to as a number)? In the answer to that question lies one of the basic ingredients of any concept dealing with information - the ability to discriminate.

It is through association with other information that the structural engineer is able to find meaning in 29500000. So it is the associations made with the number that provide meaning, not the number itself. This may be easily tested by showing the same engineer the sequence of symbols, 1110000100010001001100000, and asking for its possible meaning. The response will probably be that of the non-engineer when shown the original sequence, even though the new sequence is only a binary representation of the same number. What type of associations did the engineer have to make to find meaning in 29500000; what additional information was needed? A unit system was assumed; in this case it was a pound-inch-second system (2,070,000 kg/cm^2 in the metric system). In addition, some physical theory defining a Modulus of Elasticity must have been recalled. It is possible to identify other associations, such as the necessity of recognizing Arabic numerals, and to trace the information that supports these associations in turn, but this is not vital to the point being made.

The previous example suggests two concepts about information: first, the concept of discrimination and, second, a hierarchical structure which aids in this discrimination. This hierarchical nature can be seen in one of the many chains eminating from 29500000: the sequence of symbols led to the modulus of elasticity of steel, which led to a linear elastic theory of material behavior, which would lead to a basic understanding of physical concepts such as force. All of the parts of this chain are information, but they vary from specifics to generalizations. It can be observed that all chains of such inference will go from the general to the specific or from the specific to the general. This type of transition bears a striking resemblance to the transition of information described in the problem solving model.

The concept of discrimination and structure have been introduced but alone they cannot serve as the basis of an informational theory. The question then remains as to how information is to be defined and how it can be used in developing a model of the structural design process.

3.2 Basic Information Terms for Design

Shannon's paper "A Mathematical Theory of Communication" [65] introduced the Theory of Information, which established a mathematically rigorous concept of information. This theory provides for the modeling of the transmission of a message over a communication channel. A message is a grouping, possible sequential, of some primary information unit. The information contained in a message is then defined in the terms of a change in probability of choice given to the recipient of the message. The primary unit in the message is defined as the smallest unit that may be meaningfully referenced. A familiar example of this basic unit is the BInary digiT associated with digital computation.

The ideas expressed in the Theory of Information are too rigorous for a direct application to the development of a concept of information for design; however, some of the basic ideas in this theory can serve as guiding principles. The information content of a message (often called entropy) is a measure of the level of understanding conveyed by the message. This concept of understanding is analogous to the idea of discrimination previously discussed. The designer, however, is more concerned with the message than with its primary components. While existing, the basic message unit as defined by Information Theory is not the conscious unit of information for the designer. What is desired, then, is to define some basic unit of information that is identifiable to the designer, that is, some unit that will allow the designer to discriminate. This unit will be less rigorous than Shannon's, probably closer to a message, and will be called a Basic Information Term (suggesting the mnemonic BIT). At first the use of BIT may seem objectionable to enigneers familiar with the usual association with computers, but it is just this familiarity with the mnemonic, as a basic element, that strongly suggests its continual use in this new context.

Unlike the basic unit in Information Theory, the BIT does in itself provide for discrimination. The term generally associated with discrimination in Information Theory is "uncertainty"; however, "uncertainty" is usually used in a different context by structural engineers and, therefore, will not be used here. Information Theory established a quantitative measure of discrimination, while here it is only necessary to discuss information on the basis of qualitative scales. Thus, a BIT is discussed in terms of richness as opposed to a calculation of the entropy of a message.

The richness of a BIT is then a measure of the designer's ability to discriminate its content; possibly a more fruitful way of stating this is that richness is a measure of a designer's ability to understand. For the structural engineer, a BIT may then be very rich and yet contain a high degree of uncertainty (e.g., the measured of value of material A in test 2 is 12.4). Richness deals with the ability to discriminate, while uncertainty deals with the content itself.

A Basic Information Term for the structural designer has been identified. It is said to allow discrimination and its contents can be discussed in terms of richness. It is now necessary to establish a more complete concept as to what is meant by a BIT and what can be done with it. The rest of this chapter is then in search of a BIT.

3.2.1 Information in "The Structure-of-Intellect" (Guilford)

The first question to be explored is that of the content of a BIT. A possible classification of information is offered by J. P. Guilford as part of the model of "The Structure-of-Intellect" originally mentioned in the first chapter. Guilford's model had three axes; the axis discussed in the first chapter was the operational axis, while the other two axes delineate twenty-four possible categories of information. It should be kept in mind that the motivation for this model was to provide a means for intelligence testing and not a basis for functional theories. Even with this reservation, Guilford's model provides a fundamental insight into the meaning of a BIT.

The two informational axes are defined by product and content. The product axis alludes to the "way or form in which any information occurs"; a synonym would be "conception, which also pertains to ways of knowing or understanding". There are six possible values of the product axis:

1. Units - A unit of information is a thing, each unit has a unique combination of properties. This is Guilford's basic building block.
2. Classes - An abstraction from a set of units that hold class membership by reason of common properties.
3. Relations - Some kind of connection between two things, a kind of bridge or connecting link having its own character.
4. Systems - Complexes, patterns, or organizations or interdependent or interacting parts. A mathematical formula would be considered a system.
5. Transformations - Changes, revisions, redefinitions, or modifications, by which any product of information in one state goes over into another state.
6. Implications - Something anticipated, expected, or predicted from given information.

The content axis delineates the possible forms that the information may take; it has four possible values:

1. Figural - A concrete form, perceived or as recalled in the form of images.
2. Symbolic - Signs, materials, the elements having no significance in and of themselves, such as letters, numbers, and other code elements.
3. Semantic - The form of meaning to which words commonly become attached, hence it is most notible in verbal thinking and communications.
4. Behavioral - Instinctual feelings of an individual, such as the internalized, or gut, reaction to a particular aesthetic object.

It is now possible to combine any one value from each axis to obtain one of twenty-four possible classifications of information. Examples would be a symbolic-system, a mathematical formula, a figural-unit, a designer's recollection of a bridge or a behavioral-relation, the sense of strength associated with steel. These classifications seem to proliferate beyond the scope of what one might associate with structural design. An important observation is that information may appear in many forms (the content axis) from a specific number to a gut feeling about the proportions of a structure. The more interesting aspect of this description is found in the product axis which defines the nature of what is being described. Thus, from a qualitative viewpoint, the concern is with what is described, not how it is described. This distinction diminishes the closer one gets to an operational system.

3.2.2 Knowledge Related to Design

In the structure of human intellect information was defined as the product and the content of the product; another possible method of classification could be based upon the use of information in a design process. Such a classification is used by Professors Rittle and Protzen in their Design Methods course at the University of California, Berkeley. As opposed to classifying information, Rittle and Protzen chose to identify classes of knowledge as they might apply to a design process. Knowledge implies a higher level of organization than is intended for information; that is to say, knowledge is richer in content than normally expected of information. This differentiation does not lessen the value of examining this categorization of knowledge.

Knowledge is divided into five classifications: factual, deontonic, instrumental, explanatory, and expectational. Factual knowledge provides a description of what is, while deontonic knowledge refers to what "ought to be". Instrumental knowledge is functional in nature in that it describes what can be done. Explanatory knowledge explains why something is, and expectational knowledge describes something that is desired. This categorization is obviously intended to be consistent with a particular concept of design. The five categories actually take two possible forms, that of describing a state of being, in the present or the future, or providing the ability to operate upon a particular description. An important observation is the exclusion of any description of the form of knowledge. Unlike Guilford, Rittle and Protzen found it unnecessary to establish the "content" of knowledge.

3.2.3 Information for a Scientific Methodology (Ackoff)

A third approach that can be used to define a BIT is found in the work of Ackoff [26]. The approach is suggested by the definition of what are called "statements". Statements are defined as "answers to questions" and are composed of "expressions" [26, pg. 9]. Three forms of statement are possible: predication-classification, comparative, and functional. This type of classification is consistent in method with that of Rittle and Protzen and similar to Guilford's "products". The predication-classification statement attributes a property to an "object, event, or state"; an example being, steel is a metal. The comparative statement is one that provides an "ordering" to its subjects: an example is "steel is stronger than wood". The functional statement describes the predicative relationship between the subjects. Three types of functional statements are identified: cause-effect (deterministic causality), producer-product (probablistic cauality), and correlation. A typical example of a functional statement would be a mathematical formula.

In addition to the three types of statements, Ackoff classifies the expressions that make up a statement as ranging from qualitative to quantitative. In turn, the scope of a statement is referred to as ranging from the general to the specific. Ackoff has provided a description of a statement similar to Guilford's information in that both a nature of the statement and the content of the statement have been defined. There are similarities between all three of the informational concepts discussed, and it is from these similarities that the definition of a BIT can be further developed.

3.2.4 The BIT

The concept of basing a definition of information upon its potential usage, a concept used by both Rittle and Protzen and Ackoff, forms the basis of understanding for the BIT. The BIT is defined to function in three possible manners: as a descriptor, as an operator, or as a relator. Another way of presenting this concept is that a BIT is the answer to a question:

1. a BIT as a descriptor - What is it?

2. a BIT as an operator - Given A and B what is C?
3. a BIT as a relator - How does A relate to B?

An important differentiation made by this means of presentation is that a BIT cannot be a question but only the answer to a question.

The similarities between this classification of BITs and the previous discussions are found in Figure 3.1. The other aspects of a BIT is its content. As previously stated, the nature of a BIT's content can range from a specific number to a vague feeling. An example of the possible scope of an operational BIT is found in the "Knowledge Cells" of Wong and Bugliarello. [14]

Type of BIT	Guilford's product	Ackoff's statements	Rittle-Protzen knowledge
Descriptor	Units Classes	Predication-classification	factual deontonic explanatory expectations
Relator	Relations	Comparative	Instrumental
Operator	Systems Transformations Implications	Functional	Instrumental

Figure 3.1: Comparison of Information Classifications

A basic information term (BIT) has now been defined. The BIT is the answer to a question providing a description, an operation, or a relation. The content of a BIT can be discussed both as to its richness and as to its nature.

The question of the reliability of information and therefore the reliability of a BIT have not been discussed. Basically the question of reliability is beyond the scope of this dissertation; however, a few brief comments are justified. The question of reliability is one which attempts to establish the legitimacy of the information contained in a BIT. Ideally a BIT would contain within itself an indication of its own validity; however, this is not the case in information normally associated with engineering. It would appear that information used in engineering is generally assumed to be both reliable and deterministic; because of this, it is assumed that information is reliable unless otherwise indicated. It then becomes the responsibility of a designer to review information intended for use to assess its validity before fully incorporating it into the design process.

3.2.5 The Hierarchical Nature of BITs

In the beginning of this chapter it was implied that information had a hierarchical nature. The import of this was in the discrimination of information; of greater interest, however, is a particular aspect of this type of relationship, the transition of information from the general to the specific. As a design progresses from the need to a solution, the information involved in the design process changes from general concepts to detailed descriptions. It is the hierarchical nature of this transition that is of interest in further developing the concept of the BIT.

This hierarchical concept is one of the bases of Marvin Manheim's application of Bayesian decision theory to the highway route location described in his report Hierarchical Structure:

A Model of Design and Planning Processes [66]. Manheim envisions the design process as a layered collection of operations, where a level consists of both SEARCH and SELECTION operations such that "each succeeding operation results in increasing detail and precision of specification" [66,pg.16]. The objective of such a process is "a solution, described in all the detail necessary for implementation in the real world" [66,pg.31]. This process is similar to the problem solving model both in its functions and in its transition from the general to the specific.

As part of his design model Manheim defines both a design space and a measure called a "metric". It is this metric and its relationship to the design space that contributes to the concept of a BIT. A "universal" set of actions is defined in terms of a space constructed by a set of variables capable of describing all possible solutions to the design need. Groupings of possible actions of equal detail may be delineated by the metric. "A metric is a set of exhaustive disjoint subsets of the set of points in the action space" [66,pg.36]. It now becomes possible to speak of a more detailed design (finer metric) being included (derived from) in a less detailed design (coarser metric). This concept of inclusion serves as a basis of a hierarchical structure in which the design process differentiates information into continually finer metrics until a detailed design emerges as a solution--a transition from the general to the specific. Manheim makes use of this concept by defining a seven level process for highway route location and developing operators capable of functioning at each level. Thus, the metric becomes a measure of the level of detail of information and a delineator of the operational nature of the information.

This concept of an operational level associated with information, the metric, is analogous to S. Stevens' work on measurement found in his article "Measurement, Psychophysics, and Utility" [67]. Measurement is defined by Ackoff "in terms of its function: it is a way of obtaining symbols to represent the properties of objects, events, or states, which symbols have the relevant relationship to each other as do the things which they represent" [26,pg.179]. This type of concept implies that measurement would only be relevant to BITs of the descriptor type. Stevens presents the hierarchical relationship of the scales of measurement (Figure 3.2). The cumulative affect of these scales is similar in nature to the idea of increasingly finer metrics; thus, as a design's metric becomes increasingly fine, the information would become more amenable to measurements applicable to higher order operations.

The transition from a coarse to a fine metric and from a nominal to a ratio scale is an expression of the transition from the general to the specific described in the problem solving model for structural design. It is in this development of the BIT in the course of the design that provides the temporal structure to the design model which will be developed in the next chapter. Both Manheim and Stevens coax their concepts in terms of information analogous to a descriptor BIT. In a design process this concept, symbolized by the increasingly fine metric, would seem descriptive of the development of all types of BITs; therefore, by adopting Manheim's use of "metric", one becomes able to describe the transition of all information in the design process.

Scale	Basic Empirical Operations	Mathematical Group Structure	Typical Examples
Nominal	Determination of equality	Permutation group x' = f(x) where f(x) means any one-to-one substitution	"Numbering" of football players, assignment of type or model numbers to classes
Ordinal	Determination of greater or lesser	Isotnoc group x' = f(x) where f(x) means any increasing monotonic function	Hardness of minerals, street numbers, grades of leather, lumber, wool, etc., intelligence test raw scores
Interval	Determination of equality, of intervals, or of differences	Linear or affine group x' = ax+b a>0	Temperature (°F or °C), position, time (calendar), energy (potential), intelligence test "standard score"
Ratio	Determination of the equality of ratios	Similarity group x' = cx c>0	Numerosity, length, density, work, time, inertia, temperature (Rankine or Kelvin) loudness (sones), brightness (brils)

Figure 3.2: A Classification of Scales of Measurement [66,pg.25]

3.3 Sets of Information

By themselves, BIT's can seldom satisfy a need for information. The response to a need for information is more likely met by some collection of BIT's It is just such a collection of BITs, those related to a need, that is to be considered a set of information. The purpose of this section is to extend the concept of a BIT into that of an informational space in which the operational units are sets of information.

3.3.1 Relevance of Information

The basis for establishing the concept of an informational set is the definition of set membership. All BITs of information, whether they are internal or external to the designer, are defined as an informational universe. The normal concept of set membership then becomes one of delineating part of that universe as an informational set; that is, a decision must be made as to what BITs are relevant to a particular need. This concept of membership is best demonstrated in the following example.

A structural engineer is asked to design a beam out of material "A". The engineer's informational universe, M, consists of four BIT's.

m_1 - Material A behaves in a linear elastic fashion having a modulus of elasticity of E and a proportional limit S,

m_2 - Material B is viscoelastic,

m_3 - Material A maintains its primary behavioral characteristics up to a temperature of 500 °F, and

m_4 - The extreme fiber stress of a beam behaving in a linear elastic fashion is given by the bending moment divided by the section modulus.

The engineer is now asked to define a set M_1 that contains the BIT's of information relevant to his problem. The probable answer would be a set including BIT's m_1 and m_4.

$M_1 = [m_1, m_4]$

The concept of membership expressed in the example is deterministic in the sense that it is a yes or no proposition. This binary proposition seems inconsistent with the idea of relevance in that seldom there is the ability to so clearly differentiate information. While BIT m_3 was not given membership in the example, it is entirely possible that such information might be relevant to the design problem. It is this difference in degree between m_1 or m_4 and m_3 in which traditional set theory becomes inadequate. One response to this quandry is probablistic, that is the assignment of a probability that a BIT is relevant; however, this is only a probability of the BIT being completely relevant or completely irrelevant and not a differentiation of degree. An alternative solution to this problem of membership of informational sets is suggested by the concept of "fuzzy sets".

3.3.2 Fuzzy Sets

The expression of degrees of relevance could be handled as the assignment of a weighted membership. It is just such a concept of weighted membership that has led to the development of a theory of "fuzzy sets" by L. A. Zadeh [68,69]. A fuzzy set is a class of sets that "admits of the possibility of partial membership" [69,pg.2]. This concept of partial membership is directly analogous to a weighting of information on the basis of relevance. The mathematical definition of a "fuzzy set A in X is a set of ordered pairs

$$A = [(x, \mu_A(x))], \quad x \in X$$

where $\mu_A(x)$ is termed the grade of membership of x in A" [69,pg.4]. The membership grade is assigned on an arbitrarily determined scale, usually taken from 0 to 1 (with 1 signifying full membership). It is now possible to define a set of information in terms of its relevance (fuzziness), that is, the set's membership function would be determined by the relevance of a BIT to a need for information. A set of information S in an informational universe K is a set of ordered pairs

$$S = [(\text{BIT}, \mu_K(\text{BIT}))], \quad \text{BIT} \in K$$

where $\mu_K(\text{BIT})$ is a function expressing the relevance of a BIT to a need, S. This membership function may well be determined through the subjective judgement of a designer. Returning to the earlier example, it now becomes possible to express M_1 as a fuzzy set.

$$M_1 = [(m_1,0.99), (m_2 0.00), (m_3,0.50), (m_4,0.95)]$$

The concept of exclusion or inclusion in a set becomes meaningless with fuzzy sets because all elements of a domain belong to all fuzzy sets in that domain. It then becomes essential to differentiate, at least in an operational sense, elements of concern within a particular fuzzy set.

R. Ashby has suggested that "the existance of a threshold induces a stage of affairs that can be regarded as a cutting of the whole into temporarily isolated subsystems" [70,pg.66]. The introduction of "α-level-sets" [71] by Zadeh is directly analogous to Ashby's idea of a threshold. An "α-level-set" is a non-fuzzy set derived from a fuzzy set in the following manner

> given a fuzzy set R in X, then there exists and α-level-set R_α such that
>
> $$R_\alpha = [R(x) \mid \mu_R(x) \geq \alpha].$$

The establishment of a threshold "α" during the design process would allow a designer to isolate what information was both active and relevant to the problem. Again returning to the earlier example, a set M_1' can be defined by establishing a threshold value of 0.70.

$$M_1' = [m_1, m_4].$$

It is now possible to speak of sets of information which have membership functions based upon a BIT's relevance to a particular need. In addition, the idea of establishing a threshold provides a means of uncoupling BITs from an informational universe and placing them in an operational mode. As conceived by Zadeh, these fuzzy sets are also amenable to the operations normally associated with Boolean Logic; this aspect will be covered in further detail in Chapter Five.

3.4 Information in Design

A concept of information based on BITs of information, their relational structure and their collection into sets, has been presented in this chapter. The ideas expressed in this concept can now be integrated into a philosophy of information as it relates to a design process. The acquisition, manipulation and generation of information, previously associated with design, now becomes the transition of an informational universe into well defined sets of information. These sets of information are collections of BITs capable of descriptive, operational or relational functions.

This transition of information takes several forms: the expansion of the informational universe, the transition in BIT content, and the definition of relevant sets (assignment of membership). The assignment of membership is a primary function of the continual cognitive process identified in the problem solving model. Thus, the informational universe, K, when exposed to a particular need of the design process, a, provides a set of information, A, relevant to that need; symbolically expressed

$$K:a \rightarrow A$$

The set A will continually evolve in the design process in the hierarchical fashion previously described. Thus, as the design process nears completion, the metrics delineating A in K become increasingly finer. In the design process, informational sets are temporal in nature in that, as the process progresses, BITs are being continually refined and also act as bases for the development of new BITs.

Design then can be seen as the evolution of sets of information which start when a need becomes imbedded in a designer's informational universe. Sets of information evolve in a hierarchical manner as general concepts transcend toward detailed information associated with need fulfillment. Using these concepts of information, along with the problem solving model presented in chapter one and the ideas stemming from the design approaches reviewed in chapter two, it is now possible to construct a descriptive model of a structural design process.

CHAPTER 4

THE STRUCTURAL DESIGN PROCESS - A MODEL

4.1 Introduction

A descriptive model of the structural design process is developed in this chapter from the ideas presented in the first three chapters. A guiding postulate for this model is that structural design can be interpreted as the acquisition, manipulation and generation of information. It is in terms of this postulate that the ideas of the first three chapters are now brought together. A basic framework for the model is provided by the spiraling problem solving process presented in the first chapter. The second chapter refined the concepts of problem solving related to design, established some of the unique features of engineering design, and provided terminology applicable to the model. The Basic Information Terms (BIT) defined in the third chapter are the medium, the fundamental building blocks, of the model along with the concepts of their hierarchical nature and their grouping into sets.

It is from these last two concepts, grouping of information into sets and the hierarchical nature of information, that the model of the structural design process takes its form. Structural design is envisioned as a process that transforms a need into a design, a transition in a hierarchical fashion from general information to the detailed information required of a solution. The transition is the product of a problem solving process which can be modeled by the identification of the sets of information that are active in that process. Thus, the model of the structural design process is developed by the definition of sets of information, both as they relate to the problem solving process and as they evolve toward the solution in a hierarchical fashion.

4.2 The Bounds of the Model

The limits of a structural design porcess are found within the input set, *I*, the output set, *O*, and by the informational universe, K, of the designer or the design team. The need, N, which initiates the design process is contained within (i.e., a subset of)input. This need becomes part of the informational universe of the designer. The eventual response to the stimulation caused by N is an output set containing a design description which then leads to need fulfillment.

The informational universe of a designer is essentially the information stored internally, the memory. As a result of any design situation, and of observations of implemented designs (feedback), the designer's universe continually evolves, a process usually considered the gaining of experience. The informational universe (the set K) is, in reality, the designer; thus, K contains the information necessary to develop the process that will transform input into output. The set, *K*, is an extension of K in that it contains K and the external information directly accessible through K, that is, for practical purposes *K* becomes the informational universe that the designer finds readily available. Structural design, then, is a process where K provides the information used by a designer to transform I into O.

4.2.1 Input (I)

The design process is initiated by the stimulus of a need, N . This set of information is a subset of one of the basic sources of information, the input set, I. I contains the information required by the process that is not initially in K at the instant of need recognition; however, by completion some of I will have become part of an expanded K.

N is an expression of the needs of the structural designer's client. The normal definition of a client is understood in terms of such people as architects, entrepreneurs, or planners. Today, as sensitivities to societal needs are extended, the client is also thought of as such groups as the potential users, future owners, or even society in general. These pseudo-clients are not necessarily the initiators of need in this model; rather, they are more likely a part of what is to be defined as a social and economic context. The client, then, is defined as the decision maker who has expressed a need to whom the designer is to supply a solution. Thus, the client supplies N and also contributes in general to I, for example, the contribution of values by which alternative solutions may be judged.

This seemingly strict definition of the client is not intended to diminish the importance of pseudo-clients, but is, rather, an attempt to maintain a proper perspective. The decision maker, or client, may not be a simple entity. The general public may have a right to vote on the financing of major bridge; in this circumstance, the public has become the client because of their participation in the decision making process. On the other hand, if it is the designer's professional responsibility to consider the public's desires (i.e., the public has no power as a decision maker), then the public's desires become part of the social context of the problem.

The scope of the need can vary greatly from vague desires, coarse metric, to specific details, fine metric. This variation of metric reflects the degree to which the solution may already be expressed in N. The finer N's metric, the higher will be the level at which the designer begins making use of a design process. A highway planner may ask for a bridge to span a river, leaving the designer with a largely undefined problem (coarse metric), while an architect may have already decided upon location and size of concrete members and wishes the structural engineer to determine the size and location of reinforcing bars (fine metric).

Input is broadly defined as the information required in the structural design process and not already in K. Input, then, is best understood in relationship to the design process for which it acts as a buffer between the external world and the designer's problem solving process. *I* then becomes the potential information base for the designer and will contain information that is either acquired or generated. Typical sources of information for are the client, technical journals, engineering handbooks, information systems or consultation with another designer. In addition, it is possible that the designer needs information that may not already exist (e.g., climatic data or material properties) in which case the information must be generated.

I is continually evolving, so that, at the completion of a design process, all information has come from either I or K. Unlike I, N has a static nature in that the designer is not expected to affect a change in N. The designer may feel a need to influence N, but generally, it is in the realm of I in which the designer and the client interact. Only when the client undergoes a basic change in his own desires does N undergo any real modification. However, it is possible that the professional responsibilities of the designer may cause him to encourage the client to re-evaluate his basic statement of need. Thus, it is N which initiates the design process and I, acting as a buffer, which funnels information into the sets of information of the active design process.

4.2.2 Output ($\mathcal{O}$)

Output, $\mathcal{O}$, is the set of information whose development is the goal of the structural design process. This set is not necessarily equivalent to the need fulfillment associated with problem solving or the implementation of engineering design; however, it does contain adequate information for need fulfillment or implementation. Normally thought of as the construction or manufacture of the design, the act of implementing is generally carried out by an agent other than the designer. Thus, the design process produces a set of information, $\mathcal{O}$, which leads to need fulfillment.

Implementation is not a function of the design process, but at best, it is the responsibility of the designer as an overseer. While implementation may not be an integral part of the designer's process, nevertheless, it serves an important role in design. The ability to construct or to implement a design must be one of the problems that is of concern to the designer in his evaluation of alternative design. In addition, the designer should follow the design through to its implementation to provide feedback essential to the evolution of an informational universe. The output set then conveys the intent of the designer (i.e., the design) in a form which a contractor or any other agent of implementation can convert into a physical reality.

$\mathcal{O}$ generally contains a design description, $\underline{d}$, and some type of specification or other mechanism that provides control and, when necessary, additional descriptive information. The design description, $\underline{d}$, can be interpreted as the design vector that is the final product of a structural design process; however, the term vector may convey too harsh a concept at this time, so $\underline{d}$ should be considered a set of information that provides a description of a design at a sufficiently fine metric to allow implementation. A typical example of $\underline{d}$ would be a set of engineering drawings that shows the basic structural configuration of the design along with sufficient examples of the detail to allow the agent of implementation to convert the design to the working drawings necessary for construction. The specifications or other additional information beyond the design description are usually intended as mechanisms of control to help assure both the designer and the client that the agent of implementation will carry out the intent of the designer's solution.

In a sense, O is a summary of all that has occurred previously in the design process. It is the means by which the designer attempts to communicate his conception of how a need is to be fulfilled. It is a communication to the client and to the agent of implementation.

4.2.3 The Transition from Input to Output

A structural design process is the mechanism through which input is transformed into output. This relationship is symbollically shown in Figure 4.1.

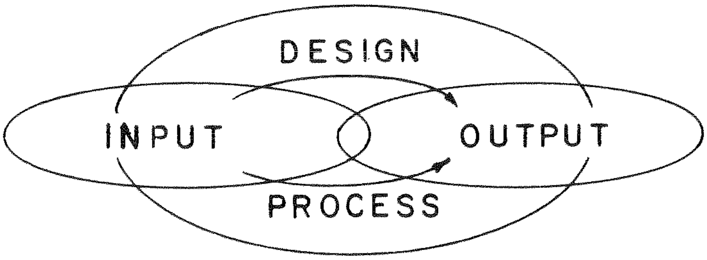


Figure 4.1: Function of the Structural Design Process

The design process is constructed by a continual mapping of undifferentiated information into I and then into sets of information related to specific aspects of the problem solving process (e.g., Context).

$$K{:}N \rightarrow I \rightarrow \text{Context}$$

The information that enters the design process is either functional or descriptive; thus, the functional information evolves into an operational process that transforms the descriptive information into a design contained in $\underline{d}$.

The design process is temporal; that is, any design requires some finite time for its generation. A time line description of the evolution of a design, like Archer's or Asimow's (sections 2.32.1 and 2.4.1.1 respectively), is not to be directly incorporated into the model of the design process, but rather, an anlogous process of metric reduction, similar to Manheim's model, is to be used in describing the transition from the need to the design. It is then the hierarchical development interpreted as metric refinement that is used to describe the evolution of a design in the problem solving process.

The metric of the output set is usually less varied and finer than N, subject to the problem's complexity. The output in a simple problem might be the size of a wooden beam, leaving the choice of wood and type of connection details to a contractor, while in a large building the output set will be presented at a very fine metric in order to make the designer's intent as clear as possible. An example of this type of hierarchical development between input and output is suggested by a hypothetical data structure presented by S. J. Fenves in his article "Design Philosophy of Large Interactive Systems" [72] (see Figure 4.2).

The structural design process, then, is defined as being bound by the sets of information input, I, and output, O, and is seen to develop from the resources found in the extended information universe of the designer, K. The process is a hierarchical transition in which the metric of the output will be less than or equal to that of input.

4.3 The Model

Structural design is to be modeled by the identification of sets of information (BIT's) active in the problem solving process responsible for the transformation of a need into a final solution. In a manner analogous to Asimow's model, the problem solving spiral presented in the first chapter is now divided into the

- Structure
  - Designation
  - Size
  - .
  - .
  - .

    Properties of entire structure
  - Floors
    - Floor Designation
    - Elevation
    - Square Feet
    - .
    - .
    - .

      Properties of a floor
  - Columns
    - Size
    - Weight
    - .
    - .
    - .

      Properties of columns on this floor
  - Slabs
    - Thickness
    - Span
    - .
    - .
    - .

      Properties of slabs on this floor
  - Beams
    - Bmsize
    - Bmspan
    - .
    - .
    - .

      Properties of beams on this floor
    - Connections
      - Location
      - Type
      - .
      - .
      - .

        Properties of connections on this beam on this floor

Figure 4.2: Hierarchical Outline of Hypothetical Data Structure [72,IV-1-19]

hierarchical development of the design and the problem solving cycle, synthesis-evaluation-decision. The sets of information are intended to model the problem solving cycle, while the hierarchical development of these sets is to model the transition toward a final solution. It is assumed that, with the exception of an initial period of differentiation, the problem solving cycle remains the same regardless of the metric associated with the design. The model then describes sets of information, their relationship, and how they evolve into the problem solving process of structural design.

At the instant of need recognition, the only identifiable sets of information are the need, N, representing the client, and the extended informational universe of the designer, *K* . It is again necessary to extend this concept of an informational universe into a new set called the design universe, U . U is introduced in recognition of the need for a designer to reach beyond his immediate bounds, *K*, in the course of a design process; thus, U is the set of all accessible information. The scope of such a set is beyond normal comprehension and can be brought into sharper focus by considering it to be a set of all information relevant to structural design and accessible in some manner by the designer, K . Another way of looking at U would be as the union of all sets, *K* .

The sets of the design process are then subsets of U in the sense that, as the process generates new BIT's, they will immediately be assigned membership in U . The design sets take form through the continual cognitive processes of the designer. Initial cognition identifies the information that will act as the basis for further hierarchical development. The designer maps the information in U into sets relevent to both the problem solving process and the recognized need; thus,

$$U{:}N \rightarrow I \qquad \ldots\ldots \qquad (4.1)$$

and, then

$$I{:}a \rightarrow A \qquad \ldots\ldots \qquad (4.2)$$

where a is a requirement for information to some aspect of the problem solving process, and

A is the information required by a .

The relations expressed in formulas 4.1 and 4.2 can then be combined into the following expression,

$$U{:}N(a) \rightarrow A \qquad \ldots\ldots \qquad (4.3)$$

where N(a) is defined as that aspect of the need that relates to the requirement, a, of the problem solving process.

A then is a fuzzy set of information that is relevant to a need and a function in the design process; this can be expressed as follows:

$$A = [(\mathrm{BIT}, \mu_A(\mathrm{BIT}))] \qquad \mathrm{BIT} \in U$$

where $\mu_A(\mathrm{BIT})$ is the relevance of the BIT to $N \cap a$ (the intersection of N and a).

This mapping process, continual cognition, is generally a human function of information discrimination, a filing of information into a "memory" (broadly defined).

The sets of information constructed from this initial mapping are either descriptive or functional in nature. The descriptive sets contain BIT's that basically are describing something (i.e., answers to "What is it?"). The functional sets are constructed out of operational and relational BIT's (i.e., answers to "Given A and B, what is C?" or "How does A related to B ?"). These are fuzzy sets, and the distinction between descriptive and functional information is by no means absolute; for example, a functional BIT may well have membership in a descriptive set, for it can influence what description may be required. The design model is then created by the identification of operational and functional sets of information as they relate to the problem solving process.

4.3.1 Descriptive Sets

The solution of a structural design problem has been defined as an output set, $\mathcal{O}$. This set contained a subset $\underline{d}$, a detailed design description. The model is focused upon the detailed design description, for $\underline{d}$ is the designer's intended means of need fulfillment. While $\underline{d}$ is the product of some synthetic process, the immediate discussion will assume its existence. At the most fundamental level (a very coarse metric), $\underline{d}$ is normally called a conceptual design; however, in this model such a $\underline{d}$ will be called a design kernel. The design kernel is the root of a hierarchical set that eventually evolves into the detailed set associated with O .

The design, $\underline{d}$, is considered to be a vector in a design space D . The concepts of a space and a vector must be expanded beyond their normal implications in order to encompass the idea of fuzzy sets of information. A vector is usually an expression of the values associated with the prescribed dimensions of a space, in which sense the vector describes a point that has been identified in the related space. The dimensions of an informational space are, in themselves, sets of information; thus, the design space is described by dimension sets, d_i. The design space, D, can then be expressed as a union of all such relevant sets, d_i; assuming an n-dimensional space, this can be expressed as:

$$D = \bigcup_{i=1}^{n} d_i \qquad . \quad . \quad . \quad . \quad . \quad . \qquad (4.4)$$

$\underline{d}$ would then be a union of sets containing BIT's that associated some measure to these dimensions. In a hierarchical development, the value of a dimension at one level may well become a dimension in the next level. While values in $\underline{d}$ are generally of similar metric, it is not unreasonable for a significant difference to exist; for example, the development of a feasible connection detail deemed critical to the acceptance of a basic concept may be examined at an early stage in the design process.

D, then, is a descriptive set associated with the actual description of the developing solution; however, it must be remembered that D is not a mathematical space but a space of fuzzily differentiated information. D is the union of the dimension sets, d_i, and it is in the assignment of values for these dimensions that the design, $\underline{d}$, is expressed. In addition, assuming the ability to develop a design kernel, each subsequent assignment of values to d_i establishes the next dimension necessary in the hierarchical development.

Given the existence of $\underline{d}$, the designer must decide upon a course of action: Should development continue? Should alterations be made? Should the alternative be eliminated? It is in the designer's response to this type of question that Ramström and Rhenman observed "limitation, generalization, change, and reformulation" [58]. The need to decide upon a course of action requires the development of criteria upon which the acceptability of a proposed design may be judged.

As Ramström's and Rhenman's "product dimensions" are analogous to the design dimensions, d_i, so their "engineering dimensions" lead to performance dimensions, p_j. It is these performance dimensions that act as design criterion which, according to Wright, et al., are "intended to assure satisfactory function or response of the system under design" [61,pg.801]. The scope of such criterion as suggested by Woodson "includes the technological-social-economic-legal feasibility, rather than only the technical fraction" [57,pg.57]. These performance dimensions then define a performance space, P, such that for m criterion:

$$P = \bigcup_{j=1}^{m} p_j \quad \ldots\ldots \qquad (4.5)$$

The desired attributes of the design are measures of the performance dimensions, so that the characteristics of an acceptable design can be expressed as a vector, $\underline{p}_d$, in P . Like the design space, P is developed in a hierarchical manner; for example, the transition from the desire for a safe structure to the specification of an allowable stress criterion to the assignment of the actual allowable stress.

In order to make a decision, the designer will attempt to evaluate a design in terms of the performance space. The result of such an evaluation is another vector in P, the predicted performance of a design $\underline{p}_p$. To carry out such an evaluation, the designer must know the context in which the design is to exist. The context of the design, often called the environment, is contained in a set, C. Churchman describes an environment as "not only . . . something that is outside the system's [designer's] control, but it is also something that determines in part how the system performs" [73,pg.36]. Ackoff sees the context as acts of nature or acts of other decision makers (reactions or counter-actions) [26,pg.67]. Woodson expands upon this: "There are the inevitable or environmental inputs, such as the surroundings, atmosphere, climate, gravity and so on" [57,pg.59]. The actual scope of C will be discussed later; however, it is worth noting the connection between P and C . The interrelationship of design sets is hard to sharply define, but it is obvious that a given contextual situation can dictate the existence of a dimension in P (e.g., earthquakes) and that a performance criterion will require certain contextual information (e.g., estimated loads).

Ideally a designer would like to know what the future context for a design will be; however, this can only be predicted at best. The designer attempts to bound the extremes of future reality with a sequence of simulated contexts, sets c_k. Thus, C is the union of all such simulated or predicted contexts, then assuming ℓ such contexts:

$$C = \bigcup_{k=1}^{\ell} c_k \qquad \ldots\ldots \qquad (4.6)$$

Each subset, c_k, will not necessarily affect all the performance criterion, p_j, but the union of all c_k's will hopefully contain, within reason, an accurate description of what will eventually be the real context.

There are then three basic descriptive sets of information in the problem solving cycle. Set D describes the solution. Set P contains the information necessary in judging the design. Set C simulates the context in which the design will eventually exist.

4.3.2 Functional Sets

The functional sets of information related to the structural design process are based upon the problem solving cycle of synthesis-evaluation-decision. This problem solving cycle, called the design process by Asimow, is a continuum with no real beginning and no real end. The continuum extends from the recognition of an initial design kernel to the development of an acceptable solution. Thus, with the exception of the discrimination of information immediately after need recognition and the identification of the initial design kernel, the problem solving spiral is modeled as a sequence of levels, as suggested by Manheim, of the synthesis-evaluation-decision cycle, where each level is of a continually finer metric.

Let it be assumed that a design, $\underline{d}$, has already been identified; in addition, it is assumed that the context, C, and the performance space, P, have already been developed. The first functional set to be defined maps $\underline{d}$ and C into P. This is the evaluation function, H (eta), a process normally associated with an engineer's modeling of physical behavior (e.g., structural analysis and mechanics). The set, $\underline{p}_p$, is the product of H, which can be expressed as follows:

$$H[\underline{d},C] \rightarrow \underline{p}_p \qquad . \ . \ . \ . \ . \qquad (4.7)$$

This function is similar to Ackoff's fundamental expression of modeling presented in the second chapter; however, that expression resulted in a product of 'V' , which is single valued, while no such restriction is intended for $\underline{p}_p$.

H is meant to be more general than these behavioral models; it will contain all physical, social and economical models, associations or relations, whether mathematical, empirical, experimental or intuitive, that the designer may use in evaluating a design. The product of this evaluative function is similar in intent to Simon's determination of fact. Thus, estimating the cost of a design or placing a value on the proportioning of a structure (an aesthetic value) may be as much a part of H as is the determination of strength. It is also possible for H to contain identity relationships as may be the case when there are dimensional criteria in P.

Once $\underline{p}_p$ has been generated by the evaluative function, the designer must decide upon a course of action relative to $\underline{d}$. It is this decision function that is to be modeled by the set Δ (delta). Δ is basically a reflection of the value system active in the design process. The product of this decision function is the merit set, m, to be associated with $\underline{d}$, thus Δ can be expressed as follows:

$$\Delta[\underline{p}_d, \underline{p}_p] \rightarrow m \qquad \ldots\ldots \qquad (4.8)$$

Decision functions usually found in use are single valued; however, decisions are seldom so simple, to quote Woodson:

> "The engineer uses all the analysis and quantification
> he can command; but in the end, the decisions are
> made subjectively; and there is no avoiding it."
>
> [57,gp.204]

The merit, m, of a design is a set that includes some statement of the design's acceptability, a statement that may be relative or absolute, quantitative or qualitative, or feeling or fact. What is important about m is that it allows the designer to act, that is, to continue the design process until completion.

The product of Δ, the merit, can now be combined with $\underline{d}$

and $\underline{p}_p$ into a triplet, a new set, d', where:

$$d' = \{\underline{d} \cup \underline{p}_p \cup m\} \qquad \ldots\ldots \qquad (4.9)$$

A union of all d' and D then defines another new set, D', where:

$$D' = D \cup \bigcup_{i=1}^{n} d'_i \qquad \ldots\ldots \qquad (4.10)$$

This new space, D', is a delineated design space which is being continually restricted with each synthesis-evaluation-decision cycle. It is this product of the problem solving cycle, D', from which the final solution will emerge; thus, as the hierarchical metric continues to decrease and D' becomes well delineated, there emerges a detailed and acceptable design.

After the decision function, the process proceeds to a synthetic or conceptual function. This function, modeled by information set χ (chi), is the least understood and probably most complex function of the problem solving process. The input requirements of χ are, essentially, all previous descriptive information; that is, the delineated design space, the performance space, and the context:

$$\chi[D',P,C] \rightarrow \underline{d} \qquad \ldots\ldots \qquad (4.11)$$

Thus, the product of χ is a new or improved design vector $\underline{d}$, and the process has then come full cycle.

The function of χ is the identification of a design vector, an act often dependent upon the designer's creativity. It is this very human aspect of χ and the requirement that χ be able to identify the original design kernel from very general information and continually identify improvements to that design as the metric decreases that makes it the least understood of all the design sets. This conceptual function can contain BIT's from both H and Δ with high memberships. This strong interaction of the functional sets in χ comes from the designer's desire to identify new or improved design vectors that have sufficient merit for final acceptance where merit is a product of both H and Δ. It is in this sense that at the simplest level χ can be approximated by the inverse of H, that is:

$$\chi \simeq H^{-1} \quad . \quad . \quad . \quad . \quad . \qquad (4.12)$$

The functional sets have been identified: evaluation, H, decision, Δ, and conception, χ. Combined with descriptive sets D, P, and C, these functional sets constitute the heart of the descriptive model of the structural design process.

4.3.3 The Completed Cycle

Structural design is modeled by the identification of sets of information in relationship to their roles in a problem solving process. These sets are both descriptive and functional and combine into a problem solving cycle of synthesis-evaluation-decision (see Figure 4.2). Progress toward a solution is measured by the change in metric such that progress is seen as a transition from the coarse metric of the initial need to the

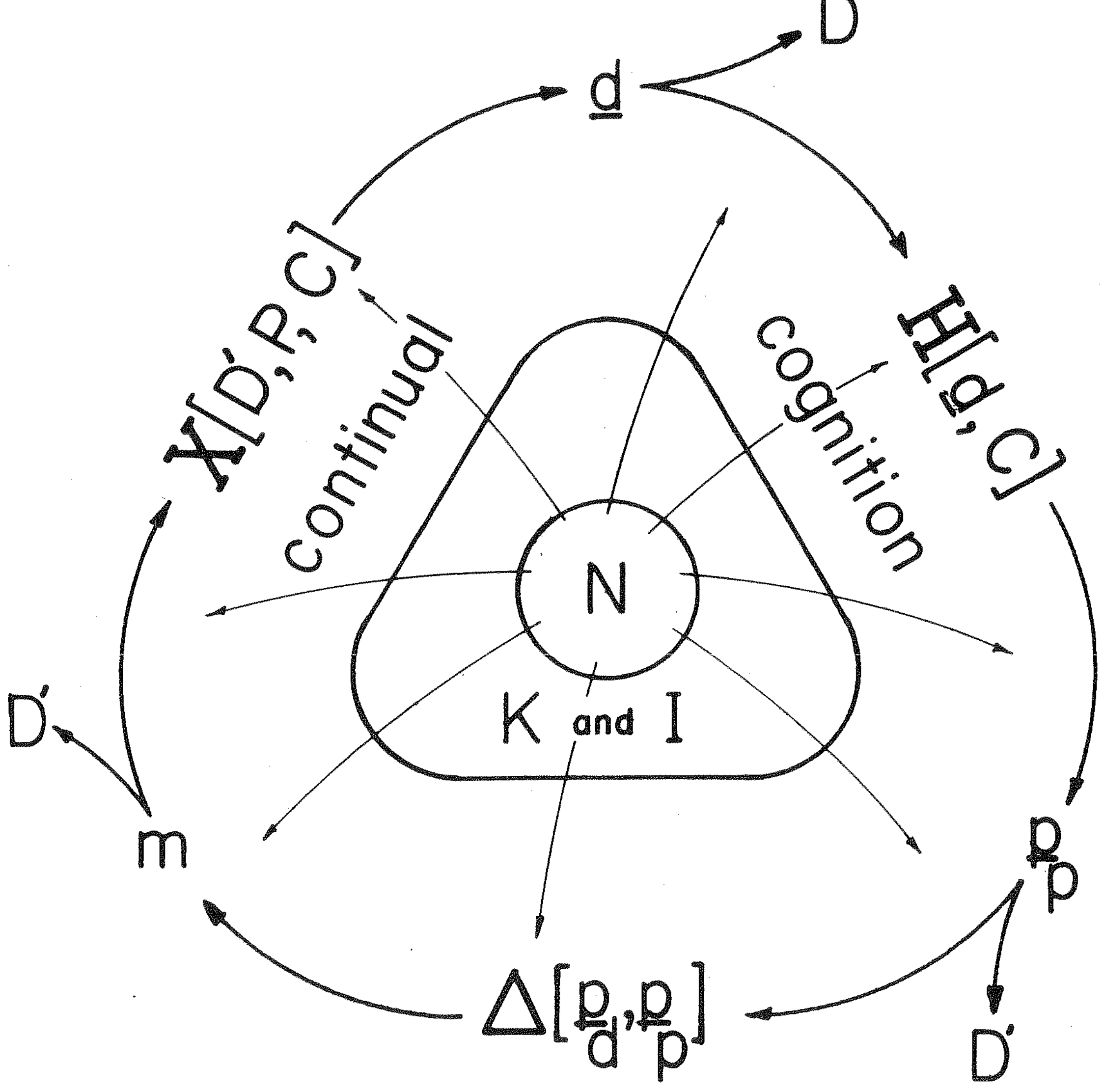


FIGURE 4.2 THE PROBLEM SOLVING CYCLE OF A STRUCTURAL DESIGN PROCESS

refined metric of the solution. In a sense, this transition toward detail is the product of a continual cognitive process, that which differentiates U into I and I into sets related to the design process. Finally, as the metric continues to decrease and the problem solving cycle further delineates the acceptable design space, a design emerges.

4.4 The Design Sets

This section examines the six primary sets of information active in a structural design process: the design space, the performance space, the context, the evaluative function, the decision function, and the conceptual operator. This examination attempts to develop a deeper understanding of the nature of these sets and, when possible, to identify what type of information will receive membership in these sets. It must be remembered that these sets are fuzzily defined in a universe of information (BIT's) and that while notation may imply mathematical manipulation, it also is meant to convey relationships between sets of information.

4.4.1 The Design Space (D)

In this section, "design space" is used as a generic term referring to all sets of information that describe the evolving or the final solution associated with a structural design process. In this sense, D also refers to D' and $\underline{d}$. This combined meaning is necessary because of the hierarchical nature of information, for the values contained in a design vector at one metric become the design dimensions of the next finer metric. In presenting the final design (i.e., the intended solution), it is reasonable to expect the use of BIT's of several different metrics. The need for this type of variation in measure is explained by the need to discriminate; for example, a BIT containing a description of a beam will have little meaning without some connection to BIT's describing the larger scheme to which the beam belongs.

The designer may find himself developing several different design descriptions, even seemingly contradictory ones, because the descriptive requirements for implementation may be different than those for evaluation. The design space then may include disjointed or at best fuzzily related design descriptions demanded by both the need for implementation and the needs of the different evaluative processes required for the prediction of the diverse attributes contained in P. The descriptions dependent upon the evaluative needs of the design process may well be of a finer metric than those associated with the output set. Thus, the designer may have to transform the descriptions generated because of evaluative needs to a description compatible with implementation.

BIT's found in the design space are normally thought of in a static and deterministic manner. While this may be an operational necessity, it is often far removed from reality. A structure is alive, it has a dynamic presence, for it is an entity which was borne by the designer, raised by the implementer (contractor), aged through use, and will eventually return to dust. While the above analogy may seem somewhat melodramatic, it does strongly suggest the temporal nature required of a complete design description. In addition, the intended solution is usually described in a deterministic manner, whereas, at best, the description is only an approximation of what will be the actual physical reality of the structure.

The temporal nature of a design requires an extension of the previously developed idea of the design space to include the possibility of time dependence (e.g., $\underline{d}(t)$). Thus, the intended solution becomes the description at a time, t_o or $\underline{d}(t_o)$. The raising or implementing process (i.e., usually construction) requires a design description at some time prior to t_o; so that, if it is assumed that implementation will start at time t_c, there will exist a design vector undergoing continual change starting as $\underline{d}(t_c)$ and emerging as $\underline{d}(t_o)$. The question of aging is less clear, for here there are two considerations: the possibility of future alternations (e.g., an increase in the number of floors) and time dependent physical process (e.g., weathering or creep). The latter of these is not really in the realm of the design space, for physical processes are not prescribed but evaluated. On the other hand, the consideration of future alterations to the design are part of a potentially changing design description. If the expected life of a design is t_1, then the planned temporal life of the structure will be expressed as a transition from $\underline{d}(t_o)$ to $\underline{d}(t_1)$.

The relationship of the finished physical structure to that of the planned or intended solution is an uncertain one, in that it can only be expressed in a probablistic manner. An example of this type of problem is found in the necessity of producing a concrete with a mean crushing strength of 3.7 ksi to satisfy a design description of 3.0 ksi. The design description then is only an approximation of what will eventually be a reality. Often portions of the output set, $\mathcal{O}$, are concerned with the quality control procedures to help insure that the implemented design closely resembles the intended design. By assigning a subscript, R, (symbolizing the real structure) to a design vector, it becomes possible to represent the real nature of a design description (see Figure 4.3).

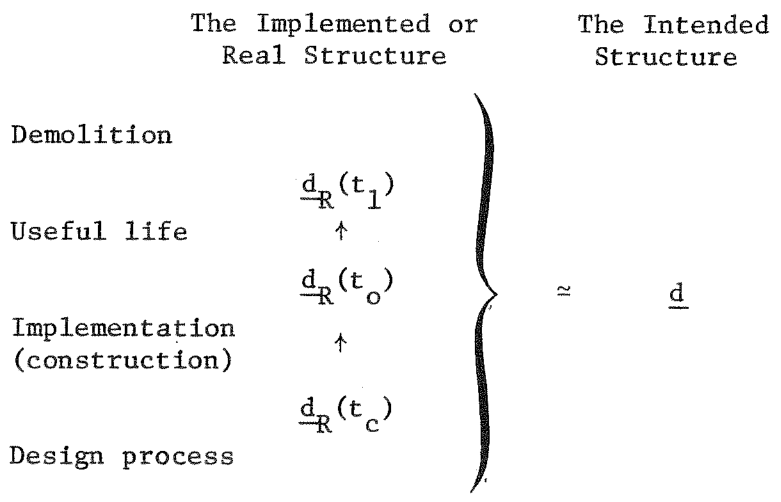


Figure 4.3: The Nature of a Design Description

With the previous clarifications of the temporal and approximate natures of a design description in mind, it is now possible to examine what really constitutes the information in a design space. A physical description could be developed as the cumulative structure of some sequence of primitive particles in a four dimensional space (i.e., the particles' location in time and space). At an atomic level, this concept violates a basic assumption in the concept of a BIT as a primary element of information; that is, that it must be meaningful in some manner to the designer. While in a philosophical sense the location of matter could be considered a description, in the practical world of the designer, a description must be more utilitarian. However, by exchanging building elements for atoms, it is possible to develop a design description. This design description will have the following components:

1. Description of the building element
2. Location and orientation of the building element
3. Relationship between the building element and the surrounding structure (e.g., how it connects)
4. Function of the building element (only if necessary).

This concept of a design description must be understood in the perspective of the hierarchical nature of the design space where it would serve as the basis for the communication of the intended design in the output set. While the solution at a coarser metric may not follow this outline exactly, it will be similar in content.

The first component of a design description is another description, that of the element. The CIB Master Lists [74] gives the following information requirements for a building element:

> 2 Description
>
> This section is designed to contain information relating to the description of the element 'as purchased or initially constructed' . . . .
>
> 2.01 Constituents, parts, type of finish
>
> List the principal materials and components used in the production of the element . . . .
>
> 2.02 Method of construction
>
> 2.03 Accessories
>
> 2.04 Shape
>
> The shape may be identified by words, drawings, photographs, etc.
>
> 2.05 Size, dimensioned elevations, plans and sections
>
> Record here information concerning the size of the element in whatever form is most appropriate, together with specified and actual tolerances.
>
> 2.06 Weight
>
> 2.07 Appearance, including texture, colour, pattern, opacity, lustre; feel; smell.
>
> [74,pg.13]

The CIB Master Lists are intended for applications to all aspects of the building industry and, therefore, are too inclusive for direct application to structural problems. In general consideration, categories 2.01, 2.04, and 2.05 are the most relevant to structural problems. An important distinction is made in category 2.01, the building element's "principal materials" are identified, but no properties are given. This absence of material properties arises because they in themselves are not at the discretion of the designer, for the designer specifies a material that has desired properties (an obvious exception occurs when a material must be specified by its desired properties, e.g., compressive strength of concrete).

The question of non-structural elements in a design description is not easily handled, for they are generally out of the control of the structural designer and, thus, are a part of the context. However, non-structural elements may often have more than a passive influence in the behavior of the structure (e.g., non-load-bearing partitions often add lateral stiffness). In theory, the design space should be under the control of the designer, but the fuzzy nature of BIT's allows the presence in D of non-structural elements that may have direct influence. In reality, the structural designer's design space is only a subset of some larger space where the portions of that larger space not under the control of the structural designer are part of the structural problem's context.

The design space has been shown to have both a temporal and a probablistic nature. The possible need for diverse descriptive sets was discussed, particularly in recognition of evaluative requirements. Four basic characteristics were attributed to a design description: an element's description, location, relation, and function. The final design description associated with implementation will contain BIT's of many metrics, because finer metric BIT's often rely upon BIT's with coarser metrics for their proper interpretation.

4.4.2 The Performance Space (P)

The performance space, P, is the union of all the criteria that must be assessed in order to pass judgement upon a given design. The concept of P is consistent with Simon's idea of dividing decision making into the assessment of factual information and the ethical (value laden) decision, for P delineates those facts required by the decision set, Δ. Like the design space, the performance space is developed in a hierarchical manner so that the BIT's contained in a performance vector, $\underline{P}_d$, at one metric will serve as the dimensions associated with the next finer metric. The performance space provides the linkage between the evaluative set, H, and the decision set, Δ. For example, if cost is to be a consideration in the decision function, then there will be a cost dimension in the performance space, and in turn, there will be some evaluative function capable of estimating the cost.

Because of this type of relationship between H and Δ, the dimensions of P may be dependent upon information from either set. An example of this is when the designer wishes to guarantee that a structure will not fail; this judgement requires the inclusion of a strength of structure dimension in P. At some finer metric, the availability of a linear elastic model of material behavior suggests the hierarchical development of the strength of structure dimension into dimension of allowable stresses. Performance dimensions are then the product of various sources at the different metrics of a given design process.

Ramström and Rhenman have suggested that the "engineering dimensions" (performance dimensions) are the transformation of the "need dimensions" (the need) into a designer's functional processes. In addition, performance dimensions may originate from the designer as a function of his technical knowledge, his professional responsibility, and his personal values, from statutory requirements of society, and as a response of the more general needs of the society at large. The performance space is expressed in BIT's of wide ranging metrics and finds its origins in many diverse value systems. The scope of these performance dimensions is seen in the difference between criteria "primarily concerned with ends not means" [75,pg.125] and building codes that may express criteria as allowable stresses or required load factors (e.g., AISC Code [79]). Performance criteria can only become active when the design process has reached the metric in which they are expressed.

The CIB Master Lists provide a broad spectrum of the possible dimensions of P in its fifth category, "Applications, design." Only part of this list is directly relevant, and that is given below.

> 5 Applications, design
>
> This section is for recording information on the design details and the functional and economical factors which need to be taken into consideration when selecting or designing a building for a particular purpose.
>
> 5.01 Suitability, function
>
> Shape, size and weight
> Appearance
> Strength and stability
> Fire resistance and other factors relating to fire
> Weathertightness
> Effect of matter (gases, liquids and solids)

5.01 (Continued)

Biological agents
Thermal criteria
Optical criteria
Acoustic criteria
Protection against radiation
Durability
Ease of construction, alteration and demolition
Ease of maintenance

5.02 Suitability, economic

This heading should be used for information concerning the costs of the building, both of initial construction or erection and the subsequent operation maintenance, restoration, alteration, and possibly demolition, including salvage value of the materials.

5.03 Suitability, statutory

5.04 to 5.07 eliminated

[74,pg.8]

While this list is not complete in the overall philosophical terms of a performance space (e.g., no mention of social criteria), it does provide a reasonably comprehensive listing of information that is related to P.

The question of strength and stability mentioned in 5.01 is the area that is normally associated with structural engineering and, therefore, structural design. While by no means does this imply a limitation, it does suggest a slightly deeper investigation into what is meant by "strength and stability." The concepts of limit state design provide a method of describing structural performance criteria that is both convenient and consistent with the model. D. E. Allen, in his paper, "Limit State Design," describes these states as "the structural conditions for which the structure is no longer fit for the purpose for which it is intended" [77,pg.1]. Allen suggests two basic types of limit states:

> Serviceability Limit States - Excessive deflection; damage to non-structural elements due to structural deflection; excessive local deformation - cracking, spalling; excessive oscillation or acceleration.
>
> Ultimate Limit State - Collapse due to fracture (fatigue), rupture, primary buckling, overturning, etc.; excessive inelastic deflection (plastic mechanism).
>
> [77,pg.1-3]

A third limit state is proposed in the book Principles of Modern Building:

> (3) Economic considerations in the design of structures shall include full allowance for the need for, and cost of, maintenance during the life of the structure.
>
> [78,pg.4]

Each limit state can be looked upon as a dimension in P that will serve as a root for further hierarchical development, so that the limit state may be assessed in some more quantitative fashion.

The performance space is the middle ground between evaluation and decision; it delineates the facts to be evaluated, and it defines the facts required for a decision. The dimensions of P are generated from many sources reflecting diverse value systems and complex technical questions. It is also possible for a dimension in P to be a simple linear measure required by some geometric constraint. P is really a midway point where the physical description of the design has been transformed and reduced into information that is more directly applicable to the making of a decision.

4.4.3 The Context (C)

The context, C, is that information which effects the design and is in itself outside the control of the designer. If it were possible, a set, C_R, would be developed which would be the real context; that is, a precise and accurate description of the future. Unfortunately, no such set can exist except in a historical sense; thus, the designer must concern himself with the simulation or prediction of the future. To do this, and in response to various evaluative needs, the designer develops a collection of sets, c_k, that attempts to approximate the future. The union of these sets is, then, the context C. By the development of such sets, c_k, the designer hopes to establish the extreme bounds of what will eventually be C_R.

Like the design space, C has both a temporal and probabilistic nature, similar in a sense to a stochastic process. The actual need for a set, c_k, is determined by the evaluative needs of the performance space. In some situations, a contextual set is so dominant in itself in that it is reflected by a particular dimension in P (e.g., earthquakes). The context can be thought of as being an invariant in relation to the design; however, this is only true in a macro sense; for example, the wind pressures felt by a structure may be drastically affected by the turbulence created by that structure. This micro behavior (e.g., the turbulence) is not really part of the context but is, rather, an aspect of the modeling techniques necessary in an evaluative process. The context then is only invariant in a macro sense, for the structure will affect its own local (micro) environment. The loadings normally used by structural engineers in determining the internal forces of a structure are normally not a part of the context; rather they come from the interaction of the context with the design.

What information is then to be considered part of the context? An all inclusive list is not feasible, but it is reasonable to provide a general categorization. In terms of this model of a structural design process, there are four basic categories of contextual information: physical, societal, cliental, and descriptive.

I. The Physical Context - information dealing with the physical realities of the universe - those phenomena which, while possibly controlled by man, are amenable to modeling through physical laws. (Note: The fuzzy nature of information suggests the inclusion of BIT's in more than one set if membership is high enough.)

  A. Climatic - those phenomena normally associated with weather - the movement and state of gaseous, liquid and solid matter in the space directly above the surface of the earth, the atmosphere.

    1. Temperature

    2. Precipitation
      a. Rain
      b. Snow
      c. Hail

    3. Wind

    4. Composition of Air
      a. Chemical content
      b. Moisture content
      c. Particulate content

5. Storm related phenomena
    a. Hurricanes
    b. Tornadoes
    c. Typhoons
    d. Floods
    e. Tidal Waves

B. Geologic - those phenomena associated with the earth

1. Surfical conditions (e.g., type of soil and depth to bedrock)

2. Seismicity
    a. Earth movement
    b. Tsunami

3. Water
    a. Surface water (e.g., rivers and lakes)
    b. Ground water
    c. Floods
    d. Tides

C. Universal - information relating to phenomena that are not understood in the sole contect of the earth.

1. Gravity

2. Radiation
    a. Solar
    b. Cosmic
    c. Nuclear

3. Meteorites

D. Material - information that contains those parameters necessary in predicting the behavior of materials. Material properties are not part of a design description unless they are necessary in specifying that material. The design description requires only the designation of a material (e.g., ASTM A36 Steel). If a given evaluative model requires information relating to that material, then that information is part of the context (e.g., the minimum yield strength of ASTM A36 Steel is 36 ksi). A material's properties are determined by how its behavior is modeled. If the designer were to use an elastic perfectly plastic model of behavior for a steel structure, he would require information dealing with the modulus of elasticity, the Poisson's ratio, the proportional limit, and the ultimate strain; while if the designer were concerned with thermal properties, the necessary material parameters would be the conductivity, the specific heat, and the density. The requirements of evaluative modeling specify the nature of material parameters found in the context. A higher order of material properties will also belong to the material context; these properties are the product of the previous level of properties along with what is to be called the constructural context. An example of this type of higher order property is the allowable bending moment of a laterally supported steel section achieved by combining an allowable stress and the section modulus of an available steel cross-section. These types of material properties are found in manuals, manufacturers product specifications, and so forth.

II. The Societal Context - information dealing with phenomena associated with man and, in that sense, not subject to physical laws. Different categories found within the societal context are listed below:

A. Social - human behavior and values in the general society, in specific interest groups, and on an individual level.

B. Legal - the rules and regulations (e.g., codes) that directly or indirectly control the bounds of the designer's function.

C. Political - the body charged with representing society and enforcing its laws. Unfortunately, such statutory bodies do not necessarily directly fulfill their obligations in a reasonable or prescribed manner.

D. Economic - information describing the facts, values and myths associated with money or other means of barter (e.g., interest rates, importance of cash flow and tax structures).

E. Constructural - a description of the ability to implement a given design. This includes the availability and skill of contractors, the nature of the labor force, accepted construction techniques, and the available building materials. It also would indicate the willingness to initiate new techniques and make use of new materials. This set interacts heavily with all the previously listed societal contexts.

F. Technological - the state of the art with regard to the available technological information for modeling physical behavior (e.g., research reports, technical journals, manuals, and specifications).

III. Cliental Context - the needs of the client as they relate to the context, what is normally called the usage. Examples of this information would be occupancy, items to be stored in a warehouse, or a H20-S16-44 standard truck.

IV. Descriptive Context - that part of an entire design space not associated with the structural aspects of the problem. Very few problems are entirely structural by nature, so that some design space larger than D may exist. D then would be a subset of this larger design space, and that portion of the space not included in D is, therefore, out of the immediate control of the structural designer.

In many ways, at the beginning of the design process, C is equivalent to U (the design universe), and by the end of the design process, the context may have become a well defined sequence of subsets. The isolation of the roots of these subsets from the design universe and their development into meaningful predictions of the future is a critical problem for all designers, for nothing is uncertain in the past except the ability to determine it, and nothing is certain in the future, especially the ability to predict it.

4.4.4 The Evaluative Function (H)

Information relevant to the evaluative set, H, enables the designer to map the design vector, $\underline{d}$, in light of the context, C, into the performance space, P. The product of this mapping, and therefore the evaluative function, is a vector, $\underline{p}_p$, which is the designer's prediction of how $\underline{d}$ will perform. This relationship between H and P is a conditional one, in that for every dimension in the performance space, p_i, there must be at least one related subset in H, thus:

$$U{:}p_i \rightarrow \eta_i \qquad \ldots\ldots \qquad (4.13)$$

where η_i is the set of information capable of mapping a design onto p_i,

and then,

$$H = \bigcup_{i=1}^{m} \eta_i \qquad \ldots\ldots \qquad (4.14)$$

One or more evaluative functions may be present in a given set, η_i. Again, there is a hierarchical development involved; for example, the designer's intuitive feeling of strength toward the structure described by a design kernel will eventually find form in some physical model of the strength of the structure at a much finer metric.

The modeling of physical behavior of materials and structures is considered the normal activity of a structural engineer. However, while the idea of modeling is the heart of contemporary structural engineering, it is by no means the only evaluative function active in a design process. In fact, it is probably accurate to say that the majority of evaluations that a designer makes in the course of a design process are not of this analytical type. What then is meant by an evaluative function?

The first class of evaluative functions can properly be called models, more exactly, scientific or physical models. Ackoff [26] identifies three types of scientific models: iconic, analogue, and symbolic. These models require progressively higher levels of abstraction according to Woodson

[57,pg.156]; that is, refinement of the problem solving process' metric. The iconic model is meant as a scaled representation; the analogue is a functional parallel; and the symbolic model uses symbols to represent quantities (i.e., a mathematical formula).

Physical behavior models are drawn from the technological context of the problem. These tend to be symbolic models based upon physical laws and modified when necessary by experimental results. Examples of such models are theories of the strength of material, the principle of minimum potential energy, graphical analysis, theories of heat transfer, and so forth. In addition to these scientific models, there will often be a need for converting contextual information into the required inputs for these models (e.g., the interaction of the design description and an expected snowfall to develop a roof loading). This type of preprocessing is also to be considered part of the evaluative function.

All problems are not amenable to the type of analytical evaluation associated with the previously mentioned behavioral models. This could be due to the lack of an adequate model, as in questions dealing with human behavior, or to the fact that the design description and context are not yet at a fine enough metric to be handled mathematically. In these cases, the designer may resort to analogue models (e.g., scale model testing), iconic models (e.g., sketching out a concept), or just subjective reasoning.

Symbolic models may help differentiate the modern structural designer from the classical builder, but their existence does not preclude the use of the more traditional empirical laws. These empirical laws are the result of accumulated experience and, in other cases, experimentation. What is important about empirical laws is that they must be stated; that is, they must have membership in the technological context. An example of this would be the data in a table on the fire rating of different structural elements. A fine difference is made here between conducting an experiment as a direct result of the design process (an analogue model) and using the accumulated results of previous testing (empirical laws).

Personal evaluative functions are another class which, unlike the physical laws, come from the designer's internal information universe, K, and not C. These personal functions originate in experience both professional and personal. Probably the single most important aspect of this type of evaluative function is that they are internal to the designer and therefore are not documented. It is this type of personal function that must be highly active in the early and final stages of the design process. The functions are basically experiential; that is, deriving from experience. However, this experience may have been partially obtained from work with theoretical models or in experimental stiuations. Another important source of experiential information is the potential feedback from designs that have already been implemented and in which the designer can observe the product of his own or some else's previous assumptions.

As was mentioned earlier, identity functions can also exist in H. This can be attributed to the possible equality between dimensions in D and P. For example, if P requires a limitation on a building's height, then the design dimension of height need only be equated with that particular dimension in P.

While structural designers are often familiar with economic models required in design evaluation, they seldom are concerned with other societal behavior. The prediction of human behavior is the subject of psychology and normally not associated with structural design. However, the assessment of human behavior can play an important role in a design process; for example, in establishing the response of occupants to a building's movement. As designers are forced to extend the context in which their problems exist, there will be an increased need to understand certain aspects of human behavior.

Evaluative functions take many forms, but they are all motivated by a desire to generate enough factual information to make a decision. Engineers today recognize the importance of physical modelling; however, it is the values reflected by personal and societal models that often control a design. The decisions made in the formative stages of a design process (e.g., to accept a particular design kernel) are often controlled by personal evaluation, and the final decisions may well rest in such societal values as economic justification.

Evaluative information, H, is that which takes a design description, $\underline{d}$, and maps it from a physically descriptive space, D, into a space, P, descriptive of attributes. It is then the description of attributes, $\underset{\sim}{p}_p$, that finally enables a designer to make a decision.

4.4.5 The Decision Function (Δ)

> "To associate a value function to a set of objectives is one of the most difficult problems of the science of action; the methods used finally amount to the introduction of a 'measure' of the set of objectives."
>
> [30,gp.57]

This assignment of a measure, the merit, to a set of objectives, the predicted performance vector, is the function of the decision set, Δ . As the above quote from A. Kaufmann in his book The Science of Decision Making [30] has suggested, the function of the decision set is often difficult. Within this decision set are the values of the client, the designer, and other possible values that might be active in the determination of a design's merit.

Δ, then, is a decision making function that reflects the values of the decision makers active in the design process. In addition, it may also reflect value systems previously expressed in the societal context or as a dimension in P. These additional values may be active in the different sense than those of the client or the designer, for they are values to which the decision maker will reassign his own values. For example, if some members of a group were queried about their aesthetic appreciation of a particular design, their response would be a reflection of their values, but it would only become another set of information in $\underline{p}_p$ which then will be used in the decision process. The previous example shows how, within the concept of this design model, the assessment of some other value may really be an evaluative function which is generating information for the model's decision function.

Simon points out in his book, The Science of the Artificial [79], that the "traditional" engineering decisions are made on the basis of "satisficing." "Satisficing" is a term that had been previously introduced by Simon to describe the process of assuring that a design will at least meet some minimal level of acceptability. Essentially, it becomes the assignment of a measure of good or bad to each performance dimension. Optimization reflects this idea of satisficing in the determination of feasibility and value. Merit is assessed by some objective function and can only be assigned if all constraints have not been violated.

An objective function is normally associated with a single measure of value (e.g., profit or weight). This concept of a unique value may provide a tractable analytical model for decision making, but it will seldom reflect reality. The question then can be expanded by the acceptance of a multi-valued concept of merit. Two basic questions must be considered in such a multi-valued concept: How can active values be related to a unique measure? and How can these various measures be combined into a single concept of merit?

From a pragmatic viewpoint, the idea suggested by Archer [42,45] in his work on the design process is a reasonable solution. He suggested the development of a function, graphical in his example, which would transform an objective to a value; each value would then be assigned a weight and the weighted values summed to achieve the design's overall merit. Another approach can be formulated in the framework of Bayesian decision theory or decision making in the presence of uncertainty. An example of this approach is found in C. Turkstra's book Theory of Structural Design Decisions [80].

Decision making in the ultimate context of engineering design is subjective and heavily ladened with the engineer's own values [57,pg.204]. A definition of Δ is needed which can reconcile both this reality and the mathematical exactness observed in optimization problems. The combination of satisficing and the concept of penalty functions (see section 2.2.2) provides the basis for such description of the decision function.

Every performance criterion, a dimension in P, has some value associated with it, even if it is only a binary type response of feasibility or infeasibility. If p^i is defined as a subset of $\underline{p}$ that contains the information related to a performance dimension p_i, then, for each dimension, a value is derived from the relationship of p_d^i and p_p^i. Let this relationship be called the difference between these sets where this difference will be symbolized by a minus sign. In this manner, the decision function can be expressed as a norming of a vector which is the resultant of the difference of two other vectors. This norming of a difference as a decision function can be expressed as follows:

$$||\underline{p}_d - \underline{p}_p|| \rightarrow m \quad . \ . \ . \ . \ . \ . \qquad (4.15)$$

Even though the above expression conveys an intended mathematical meaning, it also symbolically expresses a relationship bewteen sets of information. Differences between sets can be the designer's feelings or some mathematical quantity, and norming can be the designer's composite intuitive sense of the design or a weighted sum of distinct quantities.

The concept expressed in formula 4.15 can easily be understood in context of an optimization problem by conversion into a penalty function problem where the norm is a simple sum and values are assigned to the difference in each dimension by some discontinuous function. Likewise, 4.15 complies with the idea of satisficing if all values are binary (good or bad) and Boolean logic is used for norming. Another interpretation of 4.15 using fuzzy sets will be presented in the next chapter.

Many technques are available for decision making, and many value systems influence the designer in this process, but in the end, decisions are highly human functions. Society, economic conditions, or statutory processes may come into play in charting a course of action, but the designer must be able to make a decision for the process to continue through to fruition. It is felt that both expressions 4.8 and 4.15 describe this aspect of the structural design process in a symbolic manner.

4.4.6 The Conceptual Operator (X)

Creativity is a characteristic of any good designer. Every aspect of the design process requires that the designer be an imaginative, creative individual; however, it is in the conceptual operator, X, that creativity plays its most important role, for here the designer actually develops the design solution. The conceptual aspects of the design process, whether it be a coarse or fine metric, are viewed as the identification of a vector, $\underline{d}$, (or a part of that vector) in a design space, D'. The introduction of the term 'identication' is intended to clarify the meaning of a conceptual operator and to help avoid possible confusion in the use of the word creativity. Both Guilford and Jones (see section 1.2.2 and 2.3.1 respectively) have discussed divergent and convergent production relative to such design identification. Divergence is the expansion of the design space, the identification of the space's dimensions and the dimension's possible values. Convergence is the identification of a particular vector within that design space.

At the beginning of a design process there must be a conceptual operator that has both divergent and convergent properties before an initial design kernel can be identified. Alexander's "Pattern Language" is a process with the aforementioned characteristics that is intended for the identification of design kernels. On the other hand, the morphological approach is basically divergent, developing the defined design space but not assisting in the identification of a design kernel. Beyond the initial identification of a design kernel, the conceptual operator can be thought of as generally being convergent.

Although the builder synthesized his designs from empirical laws and intuition, today the engineer still uses much the same identification methodology. Experience is no longer necessarily gained through the seemingly crude trial and error processes associated with the builder, but experience does serve the same basic function for the contemporary engineer. The builder could only gather experience from his own personal observations of actual structures or through the cumulative experience of other builders. Today, when concepts of structural behavior have become more sophisticated, the engineer still gains his experience through feedback on existing structures; however, the engineer is also able to gain experiences through theoretical and experimental studies of structural behavior. The conceptual or synthetic process is still one in which the designer attempts to identify a solution that will behave in an acceptable manner. It is this sense or understanding of behavior, provided by experience, that controls the designer's ability to identify a potential solution.

In a simple manner, the identification of a solution is the inverse of evaluating the same solution. Given a partially delineated design space, D', the problem's context, C, and the performance space, P, the designer can attempt to identify an acceptable solution, $\underline{d}$. To illustrate this relationship between X and H, an example is presented. The problem discussed in Section 2.2.3 will be used with simplification that the only active criterion is that in bending the extreme fiber should not exceed the allowable stress. The evaluative function for extreme fiber stress was g_1 and is given below:

$$f_b = \text{extreme fiber stress} = g_1(b,h) = \frac{3wL^2}{4bh^2}$$

The problem then is to determine the required depth, h, if the width, b, is already known; thus, g_1 can be reconstructed into a new function, g, where:

$$f_b = g(1/h^2) \quad \text{or} \quad g = \frac{3wL^2}{4b} \times \frac{1}{h^2}$$

By changing the inequality constraints placed upon g_1 to equality constraints and setting the extreme fiber stress to the allowable stress, h can be determined in the following manner:

$$1/h^2 = g^{-1}(f_{\text{allowable}}) \quad \text{or}$$

$$h = 1/(g^{-1}(f_{\text{allowable}}))^{1/2} = \left(\frac{3wL^2}{4bf_{\text{allowable}}}\right)^{1/2}$$

In this example, the inversion is mathematically meaningful; however, the principle is the same regardless of the metric of the information.

The modern structural engineer is confronted with numerous collections of information that have a synthetic purpose. Examples of this are found in codes, manufacturer's product data, reports of research institutions and technical journals.

But, at the level of the design kernel, there is still very little substitute for experience and creativity.

The initial design kernel provides the roots for the design's hierarchical development, and it is to this area that the art of designing is often directed. While the inverse nature of identification becomes more apparent as the metric is refined, the initial kernel identification is not as simply understood and can be as well explained by Wallas' incubation as by inversion. Anything that provides a better understanding of a structure's response to various environments provides information not just for evaluation but also for the identification of a solution.

4.5 Summary

This chapter has presented a descriptive model of a structural design process. This model is based upon the acquisition, manipulation and generation of information. From the moment of need recognition, the design process is seen to spiral upward in a hierarchical structure of information that ends with the development of an acceptable solution. This spiraling process is symbolized by the cyclic interaction of sets of information that correspond to the problem solving cycle of synthesis-evaluation-decision.

Because of the strong interaction of these design sets, they must evolve together, particularly at the initial stages of a design process. With the discrimination of information (assignment of membership) in this initial stage of the process, there eventually emerges a design kernel. This design kernel is then carried upward with increasingly finer metrics until it evolves into a detailed and acceptable solution. The upward spiral is not really a direct linking of well defined sets of information, but is, rather, a process full of the solution of sub-problems that, when viewed together, make up fuzzily defined sets of information associated with a structural design process.

The intent and scope of this model are meant to be both flexible enough to provide descriptive mechanisms for an observer of an arbitrary structural design process and rigid enough to suggest the improvement of present design methods and the development of new methods to aid the designer. The next chapter introduces such a new method. The method is introduced not so much for the sake of its acceptance but rather to illustrate how this model can lead to new approaches and deeper understanding of structural design.

CHAPTER FIVE

THE STRUCTURAL DESIGN PROCESS - A METHODOLOGY

5.1 Introduction

In this chapter an interactive methodology is developed that is intended to stimulate and to expand a structural designer's ability to identify design kernels. There is a limited human capacity for information manipulation that is to be considered a critical barrier in problem solving; that is, every person has a finite limit to the amount of information that he can manipulate at any one time [79]. Quoting A. Horman from her paper, "Machine Aided Value Judgements Using Fuzzy-Set Techniques":

> When the complexity of the situation exceeds the capacity of man to cope with it, oversimplification and premature conclusions often result.
>
> [81,pg.3]

The method to be presented is an attempt to increase the amount of information that can be brought to bear in the solution of a problem. This method is one of the possible extensions of the model presented in the preceding chapter and, in that sense, is also meant to support the existence of that model.

The basic concept for this methodology is found in the concept of design as the acquisition, manipulation, and generation of information. This concept leads logically to the implementation of some form of information system, quoting D. Lefkovitz from his book, <u>File Structures for On-Line Systems</u>:

> Information systems can broadly be classified into two levels, the first called storage and retrieval systems and the second, control systems. The former performs a very singular function and is essentially a mechanized extension of the present library concept. It stores and updates a data collection, catalogs (or indexes) it, and enables retrieval of stored data.
>
> The control information systems contain within them a storage and retrieval system and, in addition, provide further processing that imparts to them a semblance of intelligent or heuristic behavior. These systems aid in decision making by automatically retrieving and correlating records and preparing graphs, decision tables, and summaries.
>
> [82,pg.4-5]

The information system that emerges is a primitive control system that combines the additional concepts of fuzzy sets and a morphological approach (see section 2.3.2.2) into a method called MAGID (The Manipulation and Generation of Information for Design).

This chapter is divided into three additional sections. The next section develops the information storage and retrieval system required by MAGID. The information storage and retrieval system is based on the idea of relevance indexing using fuzzy sets. The second section describes a prescribed sequence of requests that the control system uses in aiding the designer in the construction of a design space. The identification of design kernels in the previously constructed design space is the subject of the last section. This identification of design kernels takes place in what is defined as a "fuzzy environment" where a "decision is equal to the confluence of goals and constraints" [83,pg.21].

5.2 An Information Storage and Retrieval System

An information storage and retrieval system is basically one in which documents are stored in a manner which allows for their retrieval in some meaningful fashion. Documents can loosely be defined as an intelligible body of information, which might range from simple data to articles or books. W. Cooper in his paper "The Mathematical Structure of Reference Retrieval Systems" [84] offers the following "conceptual scheme within which virtually all reference retrieval systems, existing or proposed, can be conveniently analyzed."

> A reference retrieval system is a quadruple (R,I,V,T) in which
>
> (i) R is a nonempty set (of allowable "requests");
> (ii) I is a nonempty set (of "index records");
> (iii) V is a nonempty set (of possible "retrieval status values"); and
> (iv) $T:R \rightarrow V^I$ (the "retrieval function" of the system) is a function from R into V^I.
>
> [84,pg.49]

It is now within the context of the above mathematical model that an information storage and retrieval system can be developed for use in aiding a structural design process.

5.2.1 Application in a Structural Design Process

The information storage and retrieval system that is the center of MAGID is a synthesis of the mathematical model of a reference retrieval system just presented and the model of a structural design process presented in the previous chapter. The documents of this system are BIT's of information, and their aggregate is called the library, L. The BIT's stored in L form a two level hierarchical system. The first level consists of primary BIT's (PBIT's) which are referenced by the use of an indexing system. The second level consists of secondary BIT's (SBIT's) which are linked to primary BIT's and, in that sense, referenced by the indexing system through the primary BIT's.

The indexing system for the primary BIT's of L is based upon their relevance to a particular index term or concept; thus, I is a collection of fuzzy sets I(i), where

$$I(i) = [\mathrm{PBIT}, \mu_{I(i)}(\mathrm{PBIT})] \qquad \mathrm{PBIT} \in L$$

and

$$\mu_{I(i)} = \text{a measure of the relevance of PBIT to "i."}$$

In turn, the linkages of secondary BIT's to a primary BIT also constitute a fuzzy set where the membership function is determined by the relevance of a secondary BIT to the primary BIT.

Requests, R, for information can be made of L in the form of logical statements incorporating union (OR), intersection (AND) and negation (NOT or -). The mapping function, T, then becomes the rules of Boolean logic as they may be applied to fuzzy sets. The product of T's mapping is another fuzzy set, V^I, in L, where the membership is determined by the relevance of the primary BIT's to a given request. By ordering such a set, and the possible introduction of a threshold (an α-level set, see Section 3.3.2), the response to a request can be an ordered series of the most relevant primary BIT's to that request. In addition, the secondary BIT's linked to the retrieved primary BIT's may also be displayed, if desired, as part of the response to the request.

5.2.2 The Retrieval Function

The retrieval function is defined as the rules controlling the logical operations of union, intersection and negation of fuzzy sets. The technical aspects of the following discussion have been extracted from several papers by L. Zadeh [68,69,71]. To aid in discussing the rules of logic, three fuzzy sets (A, B, and D) are given below:

$$A = [(1,1.0),(2,0.5),(3,0.2),(4,0.0),(5,0.8)]$$
$$B = [(1,0.3),(2,0.5),(3,0.6),(4,0.8),(5,1.0)]$$
$$D = [(1,0.5),(2,1.0),(3,0.9),(4,0.2),(5,0.4)]$$

"The union of two fuzzy sets A and B with respective membership functions $f_A(x)$ and $f_B(x)$ is a fuzzy set C, written as $C = A \cup B$ whose membership function is related to those of A and B by

$$f_C(x) = \mathrm{Max}[f_A(x), f_B(x)] \qquad x \in X"$$

[68,pg.341]

For example:

$$C = A.OR.B = [(1,1.0),(2,0.5),(3,0.6),(4,0.8),(5,1.0)]$$

The concept of intersection is normally defined as a new fuzzy set, C, written as $C = A \cap B$, whose membership function is related to those of A and B by

$$f_C(x) = \mathrm{Min}[f_A(x), f_B(x)] \qquad x \in X$$

However, this implies a harsh interpretation of intersection. It is possible to develop softer definitions of intersection; this more generalized operation is referred to as the max-star relation and is symbolized by an asterisk, * .

$$f_C(x) = f_A(x) * f_B(x) \qquad x \in X$$

It is necessary for * to be operationally associative and monotone non-decreasing in each of its arguments; an example suggested by Zadeh of such a function is multiplication [0136,180]. In the case where only two arguments are to be involved in logical operations, two other possibilities are an arithmatic average and a geometric average.

$$\text{Arithmatic average} \quad f_C(x) = (f_A(x) + f_B(x))/2$$

$$\text{Geometric average} \quad f_C(x) = (f_A(x) \cdot f_B(x))^{1/2}$$

Examples of all four possible interpretations of intersection are given below:

C = A.AND.B = [(1,0.30),(2,0.50),(3,0.20),(4,0.00),(5,0.80)]
C_{product} = [(1,0.30),(2,0.25),(3,0.12),(4,0.00),(5,0.80)]
C_{average} = [(1,0.65),(2,0.50),(3,0.40),(4,0.40),(5,0.90)]
C_{root} = [(1,0.55),(2,0.50),(3,0.35),(4,0.00),(5,0.89)]

The concept of negation is defined as the complement of a fuzzy set and is expressed either as C = NOT.A or C = -A; thus,

$$f_C(x) = 1.00 - f_A(x) \qquad x \in X$$

With these basic rules as a mapping function, it is now possible to transform a request into a fuzzy set in L which orders the primary BIT's according to their relevance to that request. Let it now be assumed that L only has five primary BIT's in it and that fuzzy sets A, B, and D are indexes related to L. If the harsh (minimum) interpretation of .AND. is then used, the request

(A.OR.B).AND.-D

gives the fuzzy set

[(1,0.5),(2,0.0),(3,0.1),(4,0.8),(5,0.6)]

If this set were ordered and a threshold established with a value of 0.5, then the response to the request would be

$PBIT_4$ - (0.8)
$PBIT_5$ - (0.6)
$PBIT_1$ - (0.5)

On the other hand, if a softer interpretation of .AND. were used (e.g., the geometric average), the response to the previous request would be

$PBIT_4$ - (0.80)
$PBIT_5$ - (0.77)
$PBIT_1$ - (0.70)

If there had been secondary BIT's linked to one of the three primary BIT's in the response to the request and the linkage had a value greater than the threshold value, they too would have been listed in the response. The problem created by the lack of a precise definition of intersection is not satisfactorily resolved. In establishing the main information storage and retrieval program described in the next section, a harsh or minimum definition was used, but in running example problems, this proved to be too harsh. In the control program developed as part of MAGID and discussed later, the root of the product was used for the definition of an intersection and proved more acceptable but was somewhat awkward from a computational viewpoint.

5.2.3 An Information Storage and Retrieval Program

A computer program was developed to implement an information storage and retrieval system as described in the previous section. This program was named MAGID because of its central role in the evolving methodology. MAGID was written in Fortran IV for use on the Control Data Corporation 6400 computer at the University of California, Berkeley. MAGID was developed as a batch mode program because of systems limitations; however, the basic concepts are equally well suited for on-line, real time systems. The program was so structured that it was well adapted for use with teletypes; this included such features as an ability to separate bulky parts of the output for printing in a more efficient fashion than provided by a teletype (e.g., a high speed printer).

The basic capabilities of MAGID were isolated into five basic subroutines that reflected the program's functions:

STORE - This subroutine allowed for the storage of up to 500 primary BIT's and 1000 secondary BIT's where a BIT was defined as from 1 to 128 consecutive lines of 80 alpha-numeric characters. The input of secondary BIT's in STORE could also include their linkages to primary BIT's.

INDEX - This subroutine allowed for the introduction and modification of up to 130 different index sets.

REQUEST - This subroutine allowed the user to make a logical request of the library in the form of a simple string of characters that included index names and logical symbols AND, OR, and - (for negation). For example;

((INDEX1.AND.-INDEX5).OR.INDEX3).OR.-INDEX9)

The logical intersection (AND) was interpreted as a minimum function. A limitation on a request string was that it could not contain more than four levels of nested logic (e.g., the above sample request contains three levels of nested logic). The user could also specify a desired threshold to limit the program's response and indicate what type of output was desired; that is, primary BIT numbers, primary BIT's, or primary and secondary BIT's.

LIST - This subroutine had the capability of listing all files existing in the library or any part of those files.

MODIFY - This subroutine allowed for the modification of existing files by changing lines in BIT's, adding or changing linkage values between primary and secondary BIT's, by changing an index name and retaining the related set, and by deleting a full index set.

The library associated with MAGID was stored of four separate files:

FILE1 - This file was used for storing the operational information required by MAGID, for example, the library name, index labels, or BIT locations.

FILE2 - This file contained the index sets.

FILE3 - This file contained the primary BIT's.

FILE4 - This file contained the secondary BIT's.

The nature of MAGID's retrieval and modification functions required the ability to access parts of the library files in a random manner. To accomplish this random accessing, a system subroutine called TSDISK was employed. This subroutine converted files 2 through 4 from sequential files to random files when MAGID required random accessing.

With this information storage and retrieval system implemented, it becomes possible to develop a control program to help the designer in structuring information to aid in the construction of a design space.

5.3 The Development of a Design Space

In the early stages of a design process, that is, just after need recognition, the processing of information is often subjective. It is the intent of this methodology to create a library meaningful to the designer by converting this subjective information discrimination process into the assignment of membership for related BIT's in index sets. These index sets are derived from the design sets identified in the model of a structural design process so that BIT's are classified on the basis of their functional relevance to the design process. By classifying information in this manner, it is possible to aid the designer through a sequence of requests to the related library.

This sequence of requests is based upon the concept that a conceptual operator can be considered the inverse of an evaluative operator. Thus, by establishing the performance dimensions, p_j, associated with a need, it is possible to work backward through postulated evaluative functions to establish the design dimensions, d_i. These design dimensions form the design space in which a kernel can then be identified. A morphological box is then created with the determination of the possible values to be associated with each design dimension.

The library of BIT's associated with such a methodology would hopefully be quite extensive and in a continual process of evolution. In theory, such a library could exist at the beginning of a problem and only have to be supplementally indexed on the basis of the recognized need. The library would then be expanded each time it was used to aid a designer. Unfortunately, no such library presently exists, and it is well beyond the scope of this research to create one; therefore, in the example problem worked in the next chapter, the BIT's used are ones generated to suggest the possible content of a design library.

Another program was developed in order to implement the control system necessary to aid the designer in the construction of a design space and in the eventual identification of design kernels in that space (discussed in Section 5.4). The program is called REQUEST and is a sequence of ten input/output couplets. Each couplet is called a level (not to be confused with the previous hierarchical useage). These REQUEST levels (the first eight are discussed in this section) consist of storing information (BIT's and membership values) in a library by using MAGID and extracting informtion from that library by using REQUEST. The actual input required of REQUEST is a statement of what level is being worked on, what threshold is to be used, and what form of output is desired. The intent of REQUEST's output is to solicit more input from the designer and to interact with the designer in the construction of the design space. Because of experience with MAGID, it was decided to use a softer interpretation of the logical AND (i.e., a geometric mean) in the development of REQUEST.

The different levels of REQUEST are not intended to be sharply delineated steps but rather to allow the designer the flexibility to move in a dynamic sense within the method, to continually add or modify the library. The idea is to extend the designer's ability to manipulate information in the initial search for a design kernel. Another application of this method would be in helping design teams coordinate their work and aid in making of initial decisions.

To aid in the presentation of the different levels of MAGID a very simple example problem is used. The BIT's related to the problem are kept to a minimum, for the problem is only intended to help the reader understand the technical aspects of the methodology.

5.3.1 Level One

The basic input for level one is the BIT's expressing the client's need. Along with this need, the designer is encouraged to enter any other information deemed relevant. The output of level one is simply a listing of the current library with a request that the designer assign membership values (index) for the BIT's on the basis of their relevance to the design sets: DESIGN, PERFORM, CONTEXT, EVAL, DECISION, and CONCEPTION.

The client's need for the sample problem is given by the following three primary BIT's:

$PBIT_1$ – A beam is needed to span 15 feet.

$PBIT_2$ – The beam must support an estimated uniform load of 0.3 kips/ft.

$PBIT_3$ – The beam should allow for ease of construction.

The indexing of these initial BIT's can be very important, because they often contain the roots required for the subsequent hierarchical development of the problem. The following is a brief comment on the logic of membership assignment for this problem (Table 5.1, found in the discussion of the eighth level, is a summary of all membership assignment for the sample problem). Determining membership values is the responsibility of the designer, who must have an understanding of the design process model. $PBIT_1$ provides a basic identification of the design solution and therefore will have high membership in both DESIGN and CONCEPTION. $PBIT_2$ is basically contextual (membership in CONTEXT), but it also implies a need for a performance criterion and thus will receive membership in PERFORM. $PBIT_3$ is basically a performance criterion (high membership in PERFORM) but also will influence the need for evaluative and decision functions (membership in EVAL and DECISION).

5.3.2 Level Two

The input of every level is the information (BIT's and indexing) resulting from the request of the previous level; therefore, the input of level two would be the membership values assigned in level one. Because of this obvious relationship, the remaining discussion of MAGID is limited to the output of a level and the response of the designer in the sample problem. The output of level two is the current state of the design sets. The designer is requested to enlarge and refine upon the information in these sets.

If an extensive design library were to already exist, the designer would review that library on the basis of its existing indexing and extend the problem oriented library by cross indexing the design library BIT's to the design sets. If such a library is not in existence, the designer must then begin to gather information and enter it into the problem's library. Level two must be repeated until the designer is satisfied that the library contains the BIT's necessary for the hierarchical roots of the problem. In the development of this information base, the designer is free to use MAGID to make additional requests that might help in the development of the library; in the same sense, the designer can introduce any other indexing that may be desirable for the same reason.

At level two the designer may also begin to clarify the PBIT's by the introduction of SBIT's, the beginning of the information hierarchy. Following the sample problem, the designer may ask the client what was meant by "ease of construction." The response may then be recorded as secondary BIT's linked to the appropriate primary BIT's, for example:

$SBIT_1$ - Weight is to be kept to a minimum. Links ($PBIT_3$,1.0)
$SBIT_2$ - Simple techniques would preclude the use of reinforced concrete. Links ($PBIT_1$,0.9),($PBIT_3$,0.9)
$SBIT_3$ - The connection detail should be simple. Links ($PBIT_1$,0.6),($PBIT_3$,0.9)

The addition of SBIT's may well increase the influence of the PBIT's to which they are linked; such changes may require the reindexing of the associated PBIT's . Throughout the method, the designer is continually encouraged to add new BIT's, modify old BIT's and to revise indexing. Modification of the library might require that the designer return to some lower level and again begin the progress upward from level to level.

5.3.3 Level Three

The third level is intended to establish the dimensions of the performance space. The output of this level attempts to aid the designer in the choice of these dimensions. As in all levels this is accomplished by a sequence of logical requests from the problem library. These requests are given below (in a form followed in the rest of this chapter) along with a brief explanation of their intended purpose.

PERFORM	-	Information directly related to desired performance.
CONTEXT	-	Information related to the context, so that designer may assess possible implication for performance dimensions.
EVAL	-	Information related to evaluation functions, so that designer may assess possible implication for performance dimensions.

The designer is asked to respond to this query by introducing new PBIT's, new indices, and additional membership values. The definition of the performance space is accomplished by associating one PBIT with each dimension, such that that PBIT describes that dimension. This PBIT which describes a performance dimension may already exist in the library or may now have to be introduced into the library.

A new index set is introduced in association with each of these performance dimensions. This index is of the form PJ where J is an integer identifying the dimension (e.g., actual indices would be P1 or P13). The PBIT that describes the dimension will receive a membership value of 1.0, and the rest of the library will also receive membership values based upon their relevance to the specified performance dimension. In addition, another index, P, is introduced. This index is used to identify the dimensions of the performance space. Thus, each PBIT that describes a performance dimension receives a membership value of 1.0 in P. P then is a set of information that describes the performance space.

In the same problem, the designer has decided that PBIT_3 can serve as one performance dimension and that one more PBIT must be introduced as another dimension, thus:

PBIT_4 - The beam should be capable of carrying prescribed loads.

This new PBIT is entered by the designer essentially to express his basic concern, that is, the strength of the structure. PBIT_3 is indexed as the first performance dimension, P1, and PBIT_4 as the second performance dimension, P2. Again the associated indexing can be found in Table 5.1.

5.3.4 Level Four

Level four attempts to establish the evaluative methods necessary for the mapping of a design into the performance space. For each performance dimension the designer is asked to establish one or more evaluative techniques that can be used in assessing the related criterion. These evaluative techniques can be analytical, physical, heuristic, subjective or identity mappings. Since more than one evaluative function may be specified relative to each performance dimension, it is desirable to extend the previous notation η_j to η_j^k where j refers to the evaluative function. The following request is repeated for each dimension in P.

PJ.AND.P - This will cause the display of the description of performance dimension J.

PJ.AND.PERFORM - Additional performance information relating to performance dimension J.

PJ.AND.EVAL - Information that may already describe available evaluative functions relating to performance dimension J.

PJ.AND.CONTEXT - The contextual information relating to performance dimension J - the context may easily influence the choice of or need for a particular evaluative function.

The designer is asked to respond by introducing or identifying one PBIT for each evaluative function required. These evaluative BIT's are then to receive a membership value in EVAL of 1.0 and a membership value in the related PJ in accordance with its relevance; in addition, these evaluative BIT's should be indexed to all design sets.

Upon receiving the output from this level, the designer in the sample problem delineates the following evaluative needs:

η_1^1 - determination of weight,
η_1^2 - assessment of the ease of connection,
η_2^1 - determination of weight, and
η_2^2 - determination of strength.

It is noted that both performance dimensions require the determination of the beam's weight; however, this is much more important in assessing constructability than in the determination of strength, and this difference can then be reflected in the assignment of membership values. The designer now introduces three evaluative BIT's:

PBIT_5 - Weight is determined by the product of cross-sectional area x length x density.

PBIT_6 - Ease of connection is to be subjectively evaluated.

PBIT_7 - Strength can be determined using a linear elastic model and calculating the extreme fiber bending stress and the mid-height shear stress.

5.3.5 Level Five

The purpose of the fifth level is to identify and index the dimensions of the design space. This is achieved by determining what design descriptors will be required for each of the evaluative functions resulting from the previous level. The following sequence is intended to aid the designer in this process:

EVAL - By setting a threshold equal to 1.0, this request will return the evaluative BIT's identified in level four.

DESIGN.OR.CONCEPTION - This information is intended to convey the current state of the design.

The designer is asked to respond to this query by introducing PBIT's that describe a dimension in the design space. In addition, new indices are required and the library is to be reviewed in light of these indices. An index, DI, is introduced for each dimension of the design space (where I is an integer identifying that dimension, e.g., D1 or D7). Each of the PBIT's describing a design dimension will receive a membership value of 1.0 in the appropriate design set. The library should then be indexed on the basis of these sets and the new PBIT's should be indexed on the basis of the old design sets. Another index set, D, specifying the design space is also to be introduced. All PBIT's describing a design dimension will then receive a membership value of 1.0 in the index set D.

Of critical importance is the assignment of membership of these descriptor PBIT's in the PJ index sets. These values are to be based on the conditional relationship of the design descriptors to the related evaluative function and the related evaluative function to the appropriate performance dimension. With the possibility of multiple relationships between a design dimension and a performance dimension, this membership value is to be taken as the maximum of the possible values. The formula expressing this relationship between DI and PJ is given below:

$$\mu_{PJ}(DI) = \max_{k}[(\mu_{PJ}(\eta(k)) \cdot \mu_{\eta(k)}(DI))^{1/2}] \qquad (5.1)$$

where $\mu_{PJ}(DI)$ - is the membership of the PBIT describing the I-th design dimension in the index set associated with the J-th performance dimension,

$\mu_{PJ}(\eta(k))$ - is the membership of the PBIT describing the k-th evaluative function in the index set associated with the J-th performance dimension, and

$\mu_{\eta(k)}(DI)$ - is the membership of the PBIT describing the I-th design dimension in an interm index set associated with the k-th evaluative function.

In the sample problem the designer examines each evaluative function and determines what descriptive BIT's are necessary and also assigns the membership function $\mu_{\eta(k)}(DI)$. This procedure is abstracted below:

$\eta(1)$ $PBIT_5$ - Weight is determined by the product of cross-sectional area x length x density.

Required design descriptors	$\mu_{\eta(k)}$
cross-section	0.7
material	1.0

$\eta(2)$ $PBIT_6$ - Ease of connection is to be subjectively evaluated

Required design descriptors	
material	0.8
connection type	1.0

$\eta(3)$ $PBIT_7$ - Strength can be determined using a linear elastic model and calculating the extreme fiber bending stress and mid-height shear stress.

Required design descriptors	
material	1.0
cross-section	0.8
connection type	0.5

From these required design descriptors, the designer identifies three PBIT's that adequately describe the design space:

$PBIT_8$ - D1 - Material Type

$PBIT_9$ - D2 - Cross-sectional Type

$PBIT_{10}$ - D3 - Connection Type

The designer then calculates the membership of these descriptors in the performance dimension index sets; below are two examples of that calculation:

$$\mu_{P1}(D1) = \max[(1.0*1.0)^{1/2},(0.7*0.8)^{1/2},(0.3*1.0)^{1/2}] = 1.0$$

$$\mu_{P2}(D3) = \max[(0.7*0.7)^{1/2},(0.0*0.8)^{1/2},(1.0*0.5)^{1/2}] = 0.71$$

The results of the rest of these calculations are given in Table 5.1.

5.3.6 Level Six

Level six questions the designer about information related to the decision making function (DECISION). The designer is asked to make sure that at least one PBIT exists for each performance dimension describing how acceptability will be determined. This questioning is accomplished by a sequence of requests for each performance dimension, PJ. This sequence is:

PJ.AND.P - This will cause the display of the description of performance dimension J

PJ.AND.DECISION - Information that is related to decision making in the J-th performance dimension.

The desired response is either the necessary indexing to move an existing PBIT into the decision set or the introduction of new PBIT's related to the decision set.

5.3.7 Level Seven

Level seven is intended to establish the potential values to be associated with each of the design dimensions. Thus, for each design dimension the following sequence of requests is made:

DI.AND.D - This will cause the display of the description of the design dimension I

DI.AND.DESIGN - Information that describes the current state of the design in the I-th design dimension

DI.AND.CONCEPT - Information that may help the designer identify alternative values for the I-th design dimension

The designer is asked to respond by entering potential values as SBIT's linked to the PBIT's describing the appropriate design dimension--a hierarchical structure. The linkage value between the SBIT and the PBIT is an initial indication by the designer of the significance of that SBIT in a potential solution. This linkage value may become important if it is needed to reduce the possible number of enumerated designs in avoiding a combinatorial explosion.

In the sample problem, the designer identifies two possible values for each of the design dimensions.

		Linkage Value
D1 - Material Type	$SBIT_4$ - Steel	0.9
	$SBIT_5$ - Wood	1.0
D2 - Cross-sectional Type	$SBIT_6$ - Rectang.	1.0
	$SBIT_7$ - H or T	0.9
D3 - Connection Type	$SBIT_8$ - Bolted	1.0
	$SBIT_9$ - Nailed	1.0

5.3.8 Level Eight

Level eight is a review of all the sets of information as they presently exist in the library. This is simply a request for each set that then allows the designer to examine the content and membership of each BIT related to a particular set. This is the last level before the actual design kernels are to be identified in the last two levels. This is the last time that the designer will be asked to modify the problem's library. From this point on the library will be used to isolate sub-system alternatives for the designer's review and acceptance. The following table is a summary of all the indexing that has been done in relationship to the sample problem.

	PBIT									
Index Terms	1	2	3	4	5	6	7	8	9	10
DESIGN	.90		.30					1.00	1.00	1.00
PERFORM		.60	1.00	1.00						
CONTEXT	.90	1.00	.70	.20						
EVAL	.20	.50	.30	.80	1.00	1.00	1.00			
DECISION		.40	1.00	.50						
CONCEPTION	.60	.50	.70	.50	.50	.70	.30			
P1	1.00	.30	1.00		1.00	.80		1.00	.84	.89
P2	1.00	1.00		1.00	.70		1.00	1.00	.89	.71
P			1.00	1.00						
D1	1.00		1.00	1.00	1.00	.80	1.00	1.00		
D2	1.00		.89	.84	.70		.80		1.00	
D3	1.00	.60	.71	.89		1.00	.50			1.00
D								1.00	1.00	1.00

Table 5.1: Membership Functions for the Index Sets of Sample Problem (only non-zero values are given)

5.4 The Identification of Design Kernels

By the completion of the eighth level, the methodology has facilitated the construction of a design space. In a morphological approach this entire space would then be enumerated (i.e., every possible vector would be generated) and each vector would then be evaluated. Directly enumerating the design space would still leave unresolved the problems of combinatorial explosion and proper evaluative techniques (previously discussed in section 2.3.2.2).

Providing an acceptable solution to these problems, combinatorial explosion and evaluation, is the function of the ninth and tenth levels of MAGID. Combinatorial explosion is slowed down by uncoupling the design space into subspaces, D^j, containing only those design dimensions that are relevant in assessing solutions relative to performance criterion, p_j. Modified design vectors can then be generated in D^j and evaluated using a judgmental technique. Finally, the acceptable modified design vectors from each subspace can be recoupled into a set of acceptable design kernels. This concept of uncoupling is an outgrowth and simplification of work done on "splittable" design techniques and in particular the work of Stewart, previously cited, on "Partitioning and Tearing" (see section 2.3.1). The concepts relating to the evaluative techniques have been suggested by the work of R. E. Bellman and L. A. Zadeh, "Decision-Making in a Fuzzy Environment" [83], and A. Horman, Machine-Aided Value Judgments Using Fuzzy-Set Techniques" [81].

The uncoupling of the design space into subspaces relevant to a particular performance dimension is achieved through the conditional relevance, established at level five, between a design dimension, d_i, and a performance dimension, p_j. This conditional relevance was determined by the relevance of a design dimension to an evaluative function and the relevance of that evaluative function to a performance dimension. By setting some threshold of relevance for the fuzzy index set, PJ, the relevant BIT's containing the design dimensions of the subspace can be uncoupled from the general design space. In this way the subspace is actually an α-level set:

$$D^j = [\mathrm{PBIT} \mid \mu_j(\mathrm{PBIT}) \geq \alpha]$$

where $\mu_j(\mathrm{PBIT}) = \mu_D(\mathrm{PBIT}) \cap \mu_{PJ}(\mathrm{PBIT})$, and

α - is a relevance threshold

In terms of a request to MAGID, the subspace relevant to a performance dimension, P_j, is the response to the request D.AND.PJ. With this subspace identified, it is then possible to generate the alternatives it contains; however, the problem of evaluation still remains.

In the conceptual phase of a design process, the designer is often capricious, for here decisions are generally subjective and seldom analytically or rationally justifiable. This is not necessarily a criticism but a reality that must be accounted for--the designer must retain his ability to use judgment. MAGID encourages the designer to retain this freedom of choice by providing a consistent mechanism for recording judgmental responses. The key to this concept comes from Horman:

> If it is inappropriate to quantify everything and reduce the measures to one single "measure of effectiveness," then change everything into value-oriented judgment.
>
> [81,pg.10]

By introducing a new fuzzy set, A(j), which is a collection of all the vectors contained within a subspace D^j, it is possible to record a designer's judgmental response as the membership function of that set; thus:

$$A(j) = [\underline{d}_j, \mu_{A(j)}(\underline{d}_j)] \qquad \underline{d}_j \in D^j$$

where $\underline{d}_j$ is a vector containing only those design descriptors relevant to p_j,

$A(j)$ is fuzzy set of all design vectors, $\underline{d}_j$, that are acceptable subsolutions for the criterion p_j,

$\mu_{A(j)}(\underline{d}_j)$ is the membership function for $A(j)$ which is a measure of the acceptability of $\underline{d}_j$ to p_j (where a value of 0.0 indicates a totally unacceptable or nonfeasible solution and 1.0 indicates a perfectly acceptable solution).

The evaluation process for a particular performance criteria, p_j, is then achieved by the designer's assignment of a membership value for the design vectors in the related subspace. The designer is then free to carry out any type of evaluative procedure, whether it be a series of preliminary calculations or an expression of intuitive feeling, as long as it is eventually transformed into a membership value for the design vector in question. Because no mechanism has been provided in the present version of MAGID to screen out logically incompatible combinations of values of design dimensions, the designer must carry out this function through the assignment of null membership for these alternatives to the set $A(j)$.

The question then remains as to the recoupling of these modified design spaces. The key here is Bellman's and Zadeh's concept that in a fuzzy environment "a decision [can] be defined as the fuzzy set of alternatives resulting from the intersection of goals and constraints." [83,pg.19] If membership functions exist expressing the degree of acceptability of alternatives in some space for all goals and constraints, then the intersection of these sets is a fuzzy set whose membership is based upon the overall acceptability of the alternatives. However, it must be realized that there "are some situations...in which some goals and perhaps some of the constraints are of greater importance than others." [83,pg.23] To account for this type of imbalance, Bellman and Zadeh suggest the following method for calculating the membership grade of the decision set [83,pg.23-24]:

$$\mu_D(x) = \sum_{i=1}^{n} \alpha_i(x)\mu_{G_i}(x) + \sum_{j=1}^{m} \beta_j(x)\mu_{C_j}(x)$$

and

$$\sum_{i=1}^{n} \alpha_i(x) + \sum_{j=1}^{m} \beta_j(x) = 1$$

where $\mu_D(x)$ membership function of decision set

$\mu_{G_i}(x)$ membership function for goal set i

$\mu_{C_j}(x)$ membership function for constraint set j

n number of goals

m number of constraints

α_i weighting coefficient for goal i

β_j weighting coefficient for constraint j

The recombination of the sets A(j) into a decision set containing desirable design kernels is in many ways a dual of the original enumeration problem. All combinations of the non-zero or feasible vectors in the A(j) sets must be tried. While at first this may appear to be oppressive in scope, many combinations are immediately eliminated because of their incompatibility of values in the same design dimension. Thus, by recombining compatible vectors from the various subspaces and accumulating their weighted membership, the methodology develops a fuzzy set of alternative design kernels where the membership function is a measure of the design kernel's overall acceptability.

The designer expresses the value system active in the problem through the assignment of the weighting functions, α_i and β_j. The distinction between goals and constraints is only one of semantics at this level. In reality the performance criteria serve both as goals and constraints for they actually provide measures of acceptability. In MAGID there is only one weighting function, that which describes the level of influence of each performance criteria. In this sense, a performance criteria that receives a high weighting acts as a value, while a performance criteria that receives a low weighting tends to act only as a constraint. In this manner, if a performance criteria received a weighting of 1.0 and all other criteria received a weighting of 0.0, then the problem will have been transformed back to a classical single valued optimization problem.

The recombination process can still conceivably have a problem with combinatorial explosion. The introduction of a lower bound of acceptability for each set A(j) will facilitate the slowing down of the problem. In the initial evaluation, the designer was asked to assign a membership of 0.0 for unacceptable solutions; however, after this initial evaluation, the level of acceptability can be arbitrarily raised. This raising of the level of acceptability will decrease the number of acceptable solutions and thus diminish the problem of combinatorial explosion in recoupling. This procedure can be questioned, because in a problem with many performance criteria it is possible for an alternative to be marginally acceptable in one criteria and still have a high overall membership as an acceptable design kernel. This is a particularly sensitive problem when the criterion is acting basically as a constraint. The designer can partially mitigate this problem by doing some form of sensitivity study in the tenth level by varying these lower bound thresholds.

This evaluative and decision-making process incorporated in levels nine and ten shows many similarities to the model of a decision function given in expression 4.15. The assignment of the membership function for each subspace relative to a particular performance criteria is very similar to the assessment of the "difference" between a predicted value and a desired value in a performance dimension. The norming procedure is then accomplished by the "intersection" of these sets containing the expression of difference.

5.4.1 Level Nine

Level nine uncouples the design space and presents the relevant alternatives for each subspace along with the information helpful in assessing the acceptability of each alternative. The designer is initially given a listing of all the active performance criteria, the request P. The response to this listing is to be the relative weightings for each performance dimension (MAGID will normalize the weightings).

Once the performance criteria have been listed, information relevant to the evaluation and decision processes for each performance dimension along with the relevant alternatives for that dimension is given. The information preceding the list of subspace alternative vectors is a response to the following sequence of requests:

PJ.AND.P - this will cause the display of the description of the performance dimension J

PJ.AND.PERFORM - additional performance information relating to performance dimension J

PJ.AND.CONTEXT - the contextual information relevant in the evaluation of the performance dimension J

PJ.AND.EVAL - information describing the evaluative techniques to be used in assessing the value of a design in performance dimension J

PJ.AND.DECISION - information regarding the assignment of membership based on acceptability for an evaluated design

PJ.AND.DESIGN - information regardin the current state of the design description as it might relate to performance dimension J

After the above sequence of requests, a special request, D.AND.PJ, is made whose response will be those PBIT's describing the relevant design dimensions to performance dimension, J. The designer has the option of prescribing a special threshold, other than that used for the previous sequence of requests, for this critical uncoupling operation. Once these PBIT's have been isolated, MAGID determines what SBIT's are linked to them with a value greater than the prescribed threshold. These SBIT's can then be used in directly enumerating all possible design vectors in the related subspace. These vectors can then be displayed either symbolically or translated into an actual description.

The designer's response to level nine is then sets of data for each performance dimension and for the weighting of the performance criteria. With the exception of the weighting input, all evaluations need only be input as non-zero values of the acceptable design vectors in the related subspaces. Unlike all previous levels, this input for level ten is entered directly into REQUEST.

In the sample problem, the designer has decided to us an uncoupling threshold, α, of .85 for both performance dimensions. To facilitate the discussion of the example problem, the notation of $\underline{d}_j$ is extended to $\underline{d}_j^i$ which is interpreted as the i-th alternative vector in the j-th subspace. In displaying these vectors, $\underline{d}_j^i$, a zero value indicates that the particular dimension is not relevant to the related performance criterion.

The numbers that appear in the vector refer to the ordered SBIT's of the PBIT referenced by that location. Thus, a design vector, [a,0,b], would mean that SBIT "a" described the design dimension one, that the second design dimension was not relevant, and that SBIT "b" described design dimension three.

The designer has decided that the construction criteria associated with P1 is the real value associated with the problem and has therefore assigned a weight of 80% to that dimension; thus, the second performance dimension related to strength receives a weighting of 20% causing it to act more as a constraint than a value. The following are the relevant vectors to each performance dimension and their evaluated membership.

D1 - Material Type
- 1 - Wood
- 2 - Steel

D2 - Cross-sectional Type
- 1 - Rectangular
- 2 - H or T

D3 - Connection Type
- 1 - Bolted
- 2 - Nailed

Vectors in design subspace D^1

$\underline{d}_1^1$ = [1,0,1] = a bolted wood beam $\mu_{A(1)}(\underline{d}_1^1) = 0.7$

$\underline{d}_1^2$ = [1,0,2] = a nailed wood beam $\mu_{A(1)}(\underline{d}_1^2) = 1.0$

$\underline{d}_1^3$ = [2,0,1] = a bolted steel beam $\mu_{A(1)}(\underline{d}_1^3) = 0.6$

$\underline{d}_1^4$ = [2,0,2] = a nailed steel beam $\mu_{A(1)}(\underline{d}_1^4) = 0.0$

Vectors in design subspace D^2

$\underline{d}_2^1$ = [1,1,0] = a rectangular wood beam $\mu_{A(2)}(\underline{d}_2^1) = 0.8$
$\underline{d}_2^2$ = [1,2,0] = a H or T wood beam $\mu_{A(2)}(\underline{d}_2^2) = 0.5$
$\underline{d}_2^3$ = [2,1,0] = a rectangular steel beam $\mu_{A(2)}(\underline{d}_2^3) = 1.0$
$\underline{d}_2^4$ = [2,2,0] = a H or T steel beam $\mu_{A(2)}(\underline{d}_2^4) = 0.8$

The designer now inputs into level ten the weighting of the performance criteria and the membership functions given above.

5.4.2 Level Ten

Level ten receives as input the weighting of the performance criteria and the membership functions for the acceptable subspace solutions for each of the performance dimensions. In addition, the designer can specify a lower bound of acceptability for each of these membership functions. Level ten then recouples the compatible subspace design vectors and calculates their membership value in the set of acceptable design kernels. The output of level ten is simply a list of the design kernels with the highest level of acceptability.

In the sample problem, the designer has decided to use a lower bound of 0.6 for both performance dimensions. For the first performance dimension there are then only three acceptable subspace vectors, $\underline{d}_1^1$, $\underline{d}_1^2$, and $\underline{d}_1^3$; if their membership values are multiplied by the weighting function, the following respective values result, 0.56, 0.80, and 0.48. For the second performance dimension there are also three acceptable subspace vectors, $\underline{d}_2^1$, $\underline{d}_2^3$, and $\underline{d}_2^4$, whose multiplied values are 0.16, 0.20, and 0.16, respectively. The recoupling process is then summarized below (only compatible combinations are shown):

$\underline{d}_1^1$ and $\underline{d}_2^1$ = [1,1,1] $\mu_D = 0.56 + 0.16 = 0.72$

$\underline{d}_1^2$ and $\underline{d}_2^1$ = [1,1,2] $\mu_D = 0.80 + 0.16 = 0.96$

$\underline{d}_1^3$ and $\underline{d}_2^3$ = [2,1,1] $\mu_D = 0.48 + 0.20 = 0.68$

$\underline{d}_1^3$ and $\underline{d}_2^4$ = [2,2,1] $\mu_D = 0.48 + 0.16 = 0.64$

The results of level ten are then given in the following output:

```
Design Kernel 1      membership 0.96

     a rectangular, nailed, wood beam

Design Kernel 2      membership 0.72

     a rectangular, bolted, wood beam

Design Kernel 3      membership 0.68

     a rectangular, bolted, steel beam

Design Kernel 4      membership 0.64

     a H or T, bolted, steel beam
```

Level ten has then recombined the acceptable subspace solutions and presented the possible design kernels in preferential order. The designer may find it advantageous to explore the design problem by varying such features as the weighting of the performance criteria and the lower bound acceptability threshold.

5.5 Summary

MAGID (The Manipulation and Generation of Information for Design) is an interactive tool intended to aid a structural designer in the identification of design kernels. This is done by developing an information control system that extends the designer's ability to manipulate and generate information. This control system is centered around an information storage and retrieval system based on fuzzy indexing. Using this system, MAGID guides the designer in the construction of a design space. The construction of this design space is achieved by an inverse process of first defining desired attributes (performance dimensions), then determining how these attributes are to be evaluated, and finally identifying what descriptors (design dimensions) are needed for evaluation. Thus, it is the thorough understanding of behavior (the evaluative functions) that is the key to kernel identification.

The design space is then uncoupled on the basis of a design dimension's relevance to a performance dimension. Each resulting subspace, the design dimensions relevant to a particular performance dimension, then has all its vectors enumerated and judgmentally evaluated. These subspace solutions are recoupled into a fuzzy set of design kernels whose membership function expresses the level of acceptability of each kernel to the initial need.

Two example problems have been worked using MAGID. The first problem dealt with the design of a spreader truss for use in a structural fabricating plant. MAGID facilitated a reduction of this first problem's design space of 1620 vectors to the evaluation of 160 subspace design vectors. The second problem, which is the subject of the sixth chapter, was the redesign of the structural system of the Parkmerced Residential Towers. MAGID required the evaluation of approximately 2800 subspace design vectors in this second problem which was a reduction of 23 times the full design space.

The basic ideas used in the development of MAGID appear to be sound and suggest further development; however, the actual methodology is by no means perfected for actual application. The request sequences and scope of index sets available at each level can be further developed. A method should be incorporated in level nine to eliminate the generation of subspace design vectors that contain logically incompatible values for different design dimensions. The possibility that a performance dimension may be related to an individual need suggests the potential use of MAGID in team or multi-disciplinary design situations. Lastly, it should be remembered that MAGID was intended to serve two purposes in this research: one, to point out a potentially fruitful approach to the problem of kernel identification, and two, to help verify the potential usage of a model of the structural design process as was developed in the fourth chapter.

CHAPTER SIX

PARKMERCED - A SAMPLE PROBLEM

6.1 Introduction

To explore the potential of MAGID, a sample problem of substantial size and scope was pursued. This chapter presents that sample problem with the intent of furthering the reader's understanding of MAGID. It is important to note that in the working of this sample problem, the author could, at best, only simulate certain functions of a designer. However, even in light of this qualification, it is felt that the sample problem will help the reader further understand the intent of MAGID and, in that sense, come to a fuller understanding of the previously presented model of a structural design process.

The presentation of the sample problem in its complete detail would be cumbersome in the context of this dissertation. Instead, the problem will be presented in a condensed form that should convey the nature of the information involved and also provide basic insight into the development of the solution. The sample problem is presented in four parts: the statement of the problem, the development of the design library, the construction of the design space, and the identification of the design kernels.

6.2 The Problem

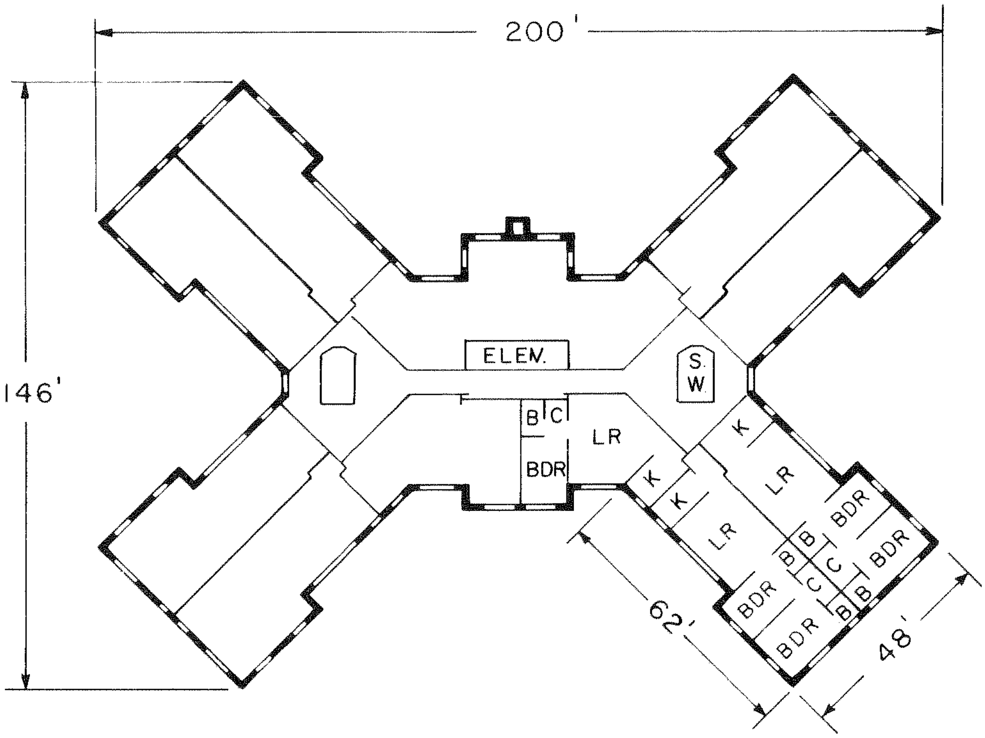
The problem was to develop an alternative structural system for the existing Parkmerced Residential Towers. These towers contain 152 apartments each and are located in the southwestern corner of San Francisco, California. Their basic configuration is given in the following BIT from the design library and is graphically presented in Figure 6.1.

```
- - PRIMARY BIT 11 - -
    THE PLAIN VIEW OF THE EXISTING PARKMERCED APARTMENTS IS AN
    ELONGATED X, 200 FEET ALONG THE MAJOR AXIS AND 147 FEET
    ALONG THE MINOR AXIS.  EACH LEG OF THE X IS APPROXIMATELY
    A RECTANGLE 70 FEET LONG AND 50 FEET WIDE.  THE BUILDINGS
    ARE 13 STORY TOWERS, WITH A STORY HEIGHT OF 9 FEET AND AN
    OVERALL HEIGHT OF 135 FEET.
```

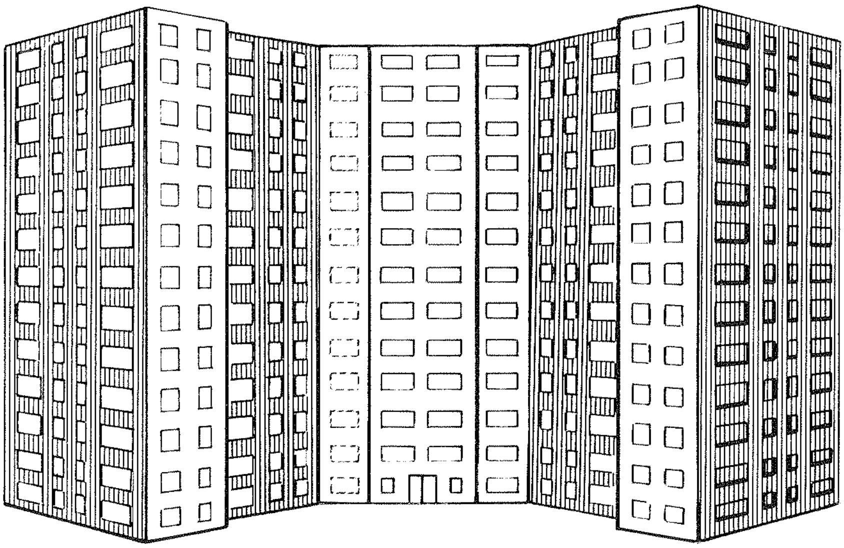
The existing structures were built around 1950 with a basic structural system of interior and exterior bearing walls and cast-in-place flat slab floors (for additional information see J. Gould's article "Multistory Buildings Designed to Resist Earthquakes" [85]).

The imaginary client for this problem stated three basic criteria by which the alternate structural system would be judged. The design should attempt to use steel as a structural material wherever it can be reasonably incorporated. Strong consideration should be given to minimizing on-site labor costs required for the construction of the structural system. And finally, the present architectural plan should be adhered to as closely as possible.

This problem statement was then translated into the



A- PLAN VIEW OF TYPICAL FLOOR



B- ELEVATION

FIGURE 6.1 BASIC CONFIGURATION OF PARKMERCED RESIDENTIAL TOWERS following four BIT's expressing the client's needs:

- - PRIMARY BIT 1 - -
DEVELOP AN ALTERNATE STRUCTURAL SYSTEM FOR THE EXISTING 13 STORY APARTMENT BUILDINGS IN PARKMERCED, SAN FRANCISCO.

- - PRIMARY BIT 2 - -
THE PRIMARY MATERIAL TO BE CONSIDERED IN THE DESIGN IS STEEL.

- - PRIMARY BIT 4 - -
THE BASIC DESIGN GEOMETRY SHOULD REMAIN AS CLOSE AS POSSIBLE TO THAT OF THE EXISTING DESIGN (I.E., THE INTERIOR SPACES SHOULD REMAIN ESSENTIALLY UNCHANGED).

- - PRIMARY BIT 5 - -
IN ATTEMPTING TO FIND AN ECONOMICAL SOLUTION, EMPHASIS SHOULD BE PLACED ON MINIMIZING THE ON-SITE LABOR COST.

The statement expressing the desire of the client for incorporating steel in the new structural system being sought was originally expressed in $PBIT_2$. As information was added to the problem's library, it became apparent that the singular use of steel implied by $PBIT_2$ severely restricted the scope of potential solutions. Of particular importance was the possibility of marginally acceptable behavior in designs incorporating only steel components, especially in reference to floor and wall components. Because of these problems, the client reconsidered the constraints imposed by $PBIT_2$ and accepted the revised statement of $PBIT_{29}$.

- - PRIMARY BIT 29 - -
WHILE ALL MATERIALS MAY BE CONSIDERED IN THIS ALTERNATE DESIGN, EMPHASIS SHOULD BE PLACED ON INCORPORATING STRUCTURAL STEEL ELEMENTS WHENEVER FEASIBLE.

The indexing done for these final four BIT's expressing the client's need is given in Table 6.1.

PBIT	DESIGN	PERFORM	CONTEXT	EVAL	DECISION	CONCEPTION
1	.60	.40	.60	.20	.40	1.00
4	.60	1.00	.00	.00	.80	.95
5	.00	1.00	.00	.40	.90	.40
29	.75	1.00	.00	.00	.50	.90

Table 6.1: Indexing of the Primary BIT's

6.3 The Development of the Design Library

The first, second, sixth, and eighth levels of MAGID are basically intended to expand the problem-oriented library. To explore the nature of the information developed at these levels, several example BIT's are presented. Of particular interest are the origins of these BIT's and the function they have in the developing solution.

6.3.1 Contextual Information

Contextual information can come from many sources. The BIT's discussed come from a statutory representation of a physical phenomenon and the designer's desire to probe more deeply into the context than suggested by the statute.

The first of these BIT's contains the wind loading required by the Uniform Building Code (UBC) [86].

```
- - PRIMARY BIT  8 - -
    THE MINIMUM RECOMMENDED WIND PRESSURE AT ZERO ELEVATION IS
    25 PSF (UBC, FIG 4, PG 131).  VARIATION OF THE MINIMUM
    PRESSURE WITH HEIGHT IS (UBC, TABLE NO 23-E, PG 130):
            0-30 FT     25 PSF             50- 99 FT     40 PSF
           30-49 FT     30 PSF            100-499 FT     45 PSF
```

Here the designer is willing to make use of the required wind pressures without any further inquiry.

On the other hand, the UBC seismic zone requirement, $PBIT_{26}$, was felt to be inadequate in its content.

> \- - PRIMARY BIT 26 - -
> SAN FRANCISCO IS CONSIDERED TO BE A ZONE 3 SEISMIC REGION (AS DEFINED BY FIGURE 1, UBC, PG 122) HAVING A 'Z' FACTOR OF 1.0. THE SEAOC CODE INCLUDES THIS SEISMICITY FACTOR IN ITS BASIC FORMULATION.

The Structural Engineers Association of California (SEAOC) Code mentioned in $PBIT_{26}$ [87] was used for seismic design in this sample problem because of its status as a forerunner to eventual changes in the equivalent sections of the UBC. In an attempt to develop a more explicit idea of the seismicity of the site, the local conditions were ascertained on the first pass basis by examining relevant United States Geological Survey data [88,89,90]. This resulted in the introduction of $PBIT_6$ and $PBIT_7$ with linked secondary BIT's.

> \- - PRIMARY BIT 6 - -
> THE PARKMERCED PROJECT IS LOCATED APPROXIMATELY 2-1/2 MILES NORTHEAST OF THE SEISMICALLY ACTIVE SAN ANDREAS FAULT AND 17 MILES SOUTHWEST OF THE ACTIVE HAYWARD FAULT.

> \- - PRIMARY BIT 7 - -
> THE PARKMERCED PROJECT IS LOCATED ON FROM 150 TO 350 FEET OF ALLUVIAL MATERIAL CLASSIFIED AS THE COLMA FORMATION; IN ADDITION THERE ARE SEVERAL INSTANCES OF ARTIFICIAL FILL. THIS MAIN ALLUVIAL DEPOSIT COVERS THE SEISMICALLY INACTIVE SAN BRUNO FAULT.
>
>> \- - SECONDARY BIT 24 - -
>> EARTHQUAKE STABILITY OF COLMA FORMATION PROBABLY MODERATE TO HIGH.
>
>> \- - SECONDARY BIT 25 - -
>> SHEAR STRENGTH AND FOUNDATION CONDITIONS OF COLMA FORMATION MODERATE TO HIGH SHEARING STRENGTH, APPROXIMATELY 3 TO 6 TIMES GREATER THAN FIRM BAY CLAY. USED FOR PILE AND CASSION SUPPORT. (TABLE 29-B, UBC INDICATES AN ALLOWABLE BEARING PRESSURE OF 8 KSF).

6.3.2 Performance Information

While a great many BIT's can be and are generated from statutory considerations, the designer may often seek a more general statement of his goals for the design. This type of statement is well reflected in the concept of limit states presented in Section 4.4.2. PBIT$_{31}$ was introduced into the problem library to describe the limit states that were the primary concern of the designer.

```
 - - PRIMARY BIT 31 - -
     A LIMIT APPROACH SHOULD BE USED IN THIS DESIGN.  LIMIT STATES
     DESCRIBE STRUCTURAL CONDITIONS FOR WHICH THE STRUCTURE IS NO
     LONGER FIT FOR THE PURPOSE FOR WHICH IT IS INTENDED.
     - - SECONDARY BIT 52 - -
          LIMIT STATE 1
            THE STRUCTURE SHOULD BE ABLE TO UNDERGO A MINOR TO
            MODERATE EARTHQUAKE AND SUSTAIN NO STRUCTURAL DAMAGE
            (MAINTAIN ITS SERVICEABILITY).
      - - SECONDARY BIT 53 - -
          LIMIT STATE 2
            SERVICEABILITY SHOULD BE MAINTAINED UNDER NORMAL LIVE
            LOADING (E.G., NO EXCESSIVE DEFLECTIONS OR DISCOMFORT
            FROM VIBRATION).
      - - SECONDARY BIT 54 - -
          LIMIT STATE 3
            IN THE CASE OF A MAJOR EARTHQUAKE, COLLAPSE SHOULD BE
            PREVENTED.  SUBSEQUENT DAMAGE SHOULD BE LIMITED TO
            THAT WHICH WITH APPROPRIATE REPAIRS WILL ALLOW THE
            RESTORATION OF SERVICEABILITY AND STRUCTURAL INTEGRITY.
```

These limit states then can act as the roots by which the different criteria imposed by statutes and professional concern can be developed. The limitation of MAGID's ability to handle informational hierarchies does not allow directly for this type of development; rather, these criteria of a finer metric must be entered as primary or secondary BIT's. An example of a BIT of this type of finer metric is PBIT$_{14}$ which specifies allowable deflections in the UBC and thus clarifies part of the first limit state.

```
- - PRIMARY BIT 14 - -
    THE DEFLECTION OF ANY STRUCTURAL MEMBER SUPPORTING A
    HORIZONTAL SURFACE SHALL CONFORM TO UBC SECTION 2307
    (PG 115).
      - - SECONDARY BIT  5 - -
          ALLOWABLE DEFLECTIONS  LIVE LOAD ONLY - L/360
                                 LIVE LOAD + K*DEAD LOAD - L/240
          WHERE FOR STEEL K=0
          AND FOR CONCRETE K=0.8, AC=AT; K=1.2, AC=0.5AT;
          K=2.0, AC=0.
```

(Note: AT and AC refer to tension and compression steel, respectively.)

6.3.3 Evaluative Information

The proper evaluation of structures subjected to seismic motion requires a fundamental understanding of both structural dynamics and inelastic structural behavior. The UBC requires the use of a simplified static equivalent load technique given in PBIT_{19} (the secondary BIT given is only an example of many others linked to PBIT_{19}).

```
- - PRIMARY BIT 19 - -
    SEISMIC FORCES ARE TO BE EVALUATED ON THE BASIS OF THE
    RECOMMENDED  LATERAL  FORCE REQUIREMENTS OF THE STRUCTURAL
    ENGINEERS ASSOCIATION OF CALIFORNIA (SEAOC), 1971 EDITION.
      - - SECONDARY BIT 34 - -
          EVERY STRUCTURE SHALL BE DESIGNED AND CONSTRUCTED TO
          WITHSTAND MINIMUM TOTAL LATERAL SEISMIC FORCES ASSUMED
          TO ACT NON-CONCURRENTLY IN THE DIRECTION OF EACH OF
          THE MAIN AXIS OF THE STRUCTURE IN ACCORDANCE WITH THE
          FOLLOWING FORMULA:
            LATERAL FORCE  V  = K*C*W
               WHERE  K  REFLECTS THE GROSS DUCTILE CAPACITY,
               C  IS THE SEISMIC COEFFICIENT AND  W  IS THE
               TOTAL DEAD AND PARTITION LOAD.
```

While providing a reasonable model for serviceability considerations, the concept of seismic behavior reflected in PBIT_{19} provides little insight into the ultimate behavior of a structure. In the conceptual phase of a design process, models descriptive of inelastic behavior may be more helpful in kernel identification than sophisticated analytical modelling techniques. BIT's of this descriptive nature were developed from several sources; for example, B. Bresler's paper, "Behavior of Structural Elements - A Review" [91], which contributed the following BIT's:

- - PRIMARY BIT 27 - -
  SHEAR WALLS ARE TO BE CONSIDERED ELEMENTS DESIGNED TO CARRY PRIMARILY LATERAL LOADS AND AS SUCH ARE CHARACTERIZED AS WALL ELEMENTS (OR BRACING) THAT ACTS PRIMARILY AS A SHEAR ELEMENT IN A FRAME, WHICH RESISTS OVERTURING MOMENTS PRIMARILY BY AXIAL FORCES. THIS IMPLIES THAT SHEAR WALLS CONSTITUTE PANELS WITH FRAME MEMBERS AROUND ALL FOUR EDGES.

    - - SECONDARY BIT 45 - -
    WHEN LATERAL FORCES ARE CARRIED BY SHEAR MECHANISMS, THE RIGIDITY OF THE STRUCTURE IS INCREASED WHICH LIMITS DRIFT BUT REDUCES THE FUNDAMENTAL PERIOD, THIS INCREASES THE LEVEL OF THE LATERAL FORCES TO BE SUSTAINED AND INCREASES THE NUMBER OF LOAD REVERSALS.

    - - SECONDARY BIT 46 - -
    WALLS OR SHEAR MECHANISMS, UNLESS CAREFULLY DESIGNED, MAY FAIL IN A RATHER BRITTLE MODE - GREATLY REDUCING THEIR POTENTIAL ENERGY DISSIPATING CAPACITY.

- - PRIMARY BIT 28 - -
  BEAM WALL SYSTEMS DO NOT HAVE ADEQUATE FRAMING TO RESIST OVERTURING MOMENTS. THIS TYPE OF WALL WILL GENERALLY CARRY BOTH VERTICAL AND LATERAL LOADS.

    - - SECONDARY BIT 44 - -
    A BEAM WALL (FRAMELESS WALL) MUST BE DESIGNED TO RESIST LATERAL SHEAR WITHOUT CRACKING.

- - PRIMARY BIT 30 - -
  TWO CHARACTERISTICS UNIQUE TO EARTHQUAKE INDUCED LOADINGS ARE:

  1. THE DYNAMIC NATURE OF THE LOADING WHICH IS USUALLY CHARACTERIZED BY REPEATED LOAD REVERSALS (LOW CYCLE FATIGUE), AND
  2. THE NEED TO ABSORB AND DISSIPATE LARGE AMOUNTS OF ENERGY.

In addition to these BIT's, it is also possible to obtain information directly from other structural designers or researchers. In this case, the recent research done at the University of California, Berkeley, on the failure of the Olive View Hospital in the San Fernando earthquake contributed a very influential BIT.

> - - PRIMARY BIT 83 - -
> RECENT WORK - RESULTING FROM RESEARCH INTO THE COLLAPSE OF THE OLIVE VIEW HOSPITAL - INDICATES THAT STRUCTURES CLOSE TO ACTIVE FAULTS, SUCH AS PARKMERCED, MAY BE SUBJECTED TO SINUSOIDAL TYPE IMPULSES SUPERIMPOSED UPON NORMAL SEISMIC EXCITATION; THEREFORE, IT IS ADVISEABLE TO HAVE LATERAL LOAD RESISTING SYSTEMS SUFFICIENTLY STIFF (E.G., SHEAR WALLS) TO AVOID DEFLECTIONS LARGE ENOUGH TO INITIATE INSTABILITY.

The interaction of the context and the structure often creates the need to evaluate the micro-context in which the design is actually located. The modification of the fundamental period of an earthquake by soil is an example of this micro-context problem associated with Parkmerced. PBIT_7 indicated both the depth and nature of the soil, while PBIT_{64} attempts to establish the bounds of the soils influence on the potential vibrations to be associated with an earthquake.

> - - PRIMARY BIT 64 - -
> THE RESPONSE SPECTRUM OF AN EARTHQUAKE IS MODIFIED BY THE SOIL LAYER OVER THE BEDROCK FORMATION TO PRIMARILY INCLUDE THE FIRST NATURAL FREQUENCY OF THE LAYER. AN APPROXIMATION (A SHEAR BEAM MODEL) FOR THE FUNDAMENTAL PERIOD OF THE SOIL IS GIVEN BY 4*H/CS WHERE H IS THE DEPTH OF THE SOIL AND CS IS THE SHEAR WAVE VELOCITY IN THE SOIL. ASSUMING A SOIL DEPTH OF BETWEEN 200 AND 300 FEET AND A SHEAR WAVE VELOCITY OF 1000 FPS THE FUNDAMENTAL PERIOD OF THE SOIL IS BETWEEN 0.8 AND 1.2 SECONDS.

There has been a purposeful progression in the BIT's presented in the last three sections. $PBIT_6$ indicated the potential hazard of active faults in the vicinity of the design's location. $PBIT_{31}$ introduced a limit state requiring that collapse of the structure be avoided in a major earthquake. Then the evaluative BIT's given in this section have indicated a strong preference toward a structure that carries seismic loads by form of shear mechanism both to avoid potential instability, $PBIT_{83}$, and stay far removed from the site's potential fundamental period, $PBIT_{64}$. The progression just described indicates the influence that the development of related BIT's can have on the final identification of a design kernel.

6.4 Constructing the Design Space

The design space and the basic relationships used in its construction are presented in this section. The design space is constructed through the development of the performance space (level three), the determination of evaluative techniques (level four), the identification of the design dimensions (level five), and the completion of the design space through the assignment of the potential values for each dimension of the space (level seven).

6.4.1 The Performance Space

In the first attempt to describe the performance space, thirteen desired attributes were listed; however, as the problem progressed, the last six of these attributes were condensed into two attributes resulting in a final performance space with nine dimensions. Of the seven attributes that remained unchanged, three were derived from the original statement of need (BIT's previously given will not be relisted): P1, $PBIT_4$; P2, $PBIT_{29}$; and P6, $PBIT_5$. The discussion then centers around the conversion of the statement of need into performance dimensions, the development of the four dimensions that remained unaltered, and the reasons for and the results of revising the last six performance dimensions.

The client's original statement of need contained what may be loosely interpreted as statements of values and constraints. The values were the premium placed upon the inclusion of steel, $PBIT_{29}$, and the minimization of on-site labor, $PBIT_5$. These BIT's were identified as performance dimensions providing an indication of the amount of steel used in the design and the relative amount of on-site labor required in implementation. In addition, the desire to maintain the present spatial design, $PBIT_4$, serves as a constraint so that there should also be a dimension that contains a description of the design's geometry.

The inclusion of the limit state approach contained in $PBIT_{31}$ suggests the assignment of certain strength criteria. The first of these limit states, $SBIT_{52}$, is a serviceability criterion giving rise to P4, $PBIT_{34}$, while the question of ultimate limit states created the previously mentioned need for revision and will be discussed in the next paragraph. The behavior of the structural system in a fire environment was considered critical both in the maintenance of structural integrity and in the role the structure might play in the prevention of the loss of life. This concern, coupled with related UBC requirements, was reflected in P7, $PBIT_{35}$. A dimension, P3, $PBIT_{33}$, was introduced to supplement P6 by specifying a need for a structural design consistent with available construction techniques and integral with the total concept of the design. It was also felt that the structure might play a role in creating a favorable acoustical environment for this residential building; this gave rise to P5, $PBIT_{15}$.

The question of ultimate limit states was the main source of trouble in developing the performance space. Originally it was felt that six dimensions, reflecting both contextual and descriptive details of the situation, would be useful. Thus, one dimension was introduced for each of four possible earthquake situations, P8 to P11, (i.e., earthquakes inducing forces parallel to the axes of the structure, perpendicular to a leg of the structure, and torsionally in the structure) and for each of two possible wind situations, P12 to P13, (i.e., wind parallel to the axes of the structure). The reason for these six dimensions was the desire to create smaller subspaces by uncoupling possible lateral load carrying mechanisms. Unlike many structures, the geometry of the Parkmerced towers would not permit an idealization of the structure into orthogonal framing systems. Because of this originally incorrect interpretation, the six performance dimensions dealing with ultimate limit states were reduced to two dimensions, P8 and P9, reflecting the limit states expressed in $SBIT_{53}$ and $SBIT_{54}$, respectively.

The final performance space then contained the nine dimensions listed below:

P1 - PRIMARY BIT 4 - -

P2 - PRIMARY BIT 29 - -

P3 - PRIMARY BIT 33 - -
THE STRUCTURAL DESIGN SHOULD BE CONSISTENT WITH AVAILABLE CONSTRUCTION METHODS AND AN INTEGRAL PART OF THE OVERALL DESIGN.

P4 - PRIMARY BIT 34 - -
THE STRUCTURE SHOULD REMAIN SERVICEABLE (I.E., NO DAMAGE OR DISCOMFORT) UNDER NORMAL LOADING CONDITIONS. THESE NORMAL CONDITIONS ARE TO INCLUDE LIVE LOAD AND DEAD LOAD, MINOR TO MODERATE EARTHQUAKES AND MODERATE WINDS.

P5 - PRIMARY BIT 15 - -
CONSIDERATION SHOULD BE GIVEN TO MINIMIZING SOUND TRANSMISSION THROUGH ALL SPATIAL BARRIERS; PARTICULAR ATTENTION SHOULD BE PAID TO THE WALLS AND FLOORS THAT DELINEATE THE BOUNDARIES OF ANY GIVEN LIVING UNIT.

P6 - PRIMARY BIT 5 - -

P7 - PRIMARY BIT 35 - -
THE SAFETY OF OCCUPANTS AND LOSS OF PROPERTY MUST BE CONSIDERED IN CASE OF A FIRE. THIS MUST INCLUDE THE CONTAINMENT OF THE FIRE, MINIMIZING SMOKE EMISSION, AND ASSURING THE CONTINUED INTEGRITY OF THE STRUCTURE IN THE FIRE SITUATION.

P8 - PRIMARY BIT 37 - -
DURING EXTREME LOAD SITUATIONS COLLAPSE OF THE STRUCTURE MUST BE PREVENTED; THIS INCLUDES THE ASSURANCE OF CONTINUED LATERAL STABILITY. THESE LOAD SITUATIONS ARE TO INCLUDE SEVERE EARTHQUAKE AND HIGH WINDS.

P9 - PRIMARY BIT 38 - -
THE INTEGRITY OF THE STRUCTURE MUST BE RETAINED DURING EXTREME LOAD SITUATIONS, SUCH THAT THE REPAIRING OF DAMAGE IS FEASIBLE.

6.4.2 The Dimensions of the Design Space

The key to identifying the dimensions of the design space is found in the statement of the evaluative functions required by the performance dimensions. The evaluative techniques directly applicable to a given design description are generally constrainted by the metric of that description. In this sense, the hierarchical development of that design description is often paralleled by an equivalent transition in the available evaluative techniques. This implies that when the design is described in a general format, i.e., a coarser metric, as in this sample problem, few analytical techniques are directly applicable. Therefore, evaluation tends to be of a more subjective nature. The use of such subjective evaluations, when considering a design described in a coarse metric, is not the only alternative, nor is it necessarily the most desireable alternative.

When considering a particular alternative design, it is possible for a designer to postulate a refinement in the design's metric which can then allow for a more explicit and objective evaluation. This type of evaluation can be observed in what are often called rough or sketch calculations; however, it is also feasible to use more sophisticated analytical techniques when the current description can be extended into a reasonably simplified and more refined description amenable to those techniques. In this way, the designer can use a more objective evaluation to supplement the judgmental assignment of membership for subspace vectors in a set of acceptable design alternatives. This discussion of the relationship between subjective and objective evaluative techniques will be continued with examples in Section 6.5.1.

A total of twelve evaluative functions were developed, of which four provided information for more than one performance dimension. The first two performance dimensions, P1 and P2, required only identity mappings, PBIT_{43} and PBIT_{44}. Evaluating the construction oriented dimensions, P3 and P6, was to be accomplished both subjectively, PBIT_{45} and PBIT_{50}, and with the aid of available cost data, PBIT_{49}. Fire, P7, and acoustic, P5, considerations were to be evaluated subjectively, PBIT_{52}, and in conjunction with stated experimental and empirical rules, PBIT_{51} and PBIT_{48}. The assessment of the structure's physical behavior was to be accomplished using either elastic analysis or approximate techniques, PBIT_{46}, PBIT_{47}, and PBIT_{53}, and inelastic behavior was assessed through the application of empirical rules, PBIT_{54}. Below is a listing of the evaluative BIT's that served as the basis for the conditional relationship between the Performance Space and the Design Space (note the inclusion of the index values for the different performance dimensions):

```
- - PRIMARY BIT 43 - -
    GEOMETRIC CONSIDERATIONS ARE TO BE EVALUATED AS AN IDENTITY
    MAPPING FROM THE DESIGN SET TO THE PERFORMANCE SET. (P1,100)
- - PRIMARY BIT 44 - -
    MATERIAL USEAGE IS EVALUATED AS AN IDENTITY MAPPING FROM
    THE DESIGN SET TO THE PERFORMANCE SET. (P2,100)
- - PRIMARY BIT 45 - -
    CONSTRUCTION CONSIDERATIONS ARE TO BE EVALUATED SUBJECTIVELY.
    (P3,100;P6,85)
- - PRIMARY BIT 46 - -
    THE STRUCTURES INTERNAL FORCES ARE TO BE DETERMINED USING
    EITHER AN ELASTIC ANALYSIS OR AN ACCEPTABLE APPROXIMATE
    METHOD.  (P4,90;P8,100)
```

- - PRIMARY BIT 47 - -
DEFLECTIONS ARE TO BE DETERMINED BY THE USUAL METHODS OR FORMULAS OF ELASTIC THEORY. TIME DEPENDENT CHARACTERISTICS MAY BE ACCOUNTED FOR BY AN APPROPRIATE MULTIPLICATION FACTOR. (P4,100)

- - PRIMARY BIT 48 - -
THE EVALUATION OF SOUND TRANSMISSION IS TO BE QUALITATIVE, FOLLOWING TWO BASIC RULES:
1. BASIC DESIGN CONCEPTS AND DETAILS SHOULD ELIMINATE POSSIBLE SOUND PATHS, AND
2. MASS WILL REDUCE SOUND TRANSMISSION.
(P5,100)

- - PRIMARY BIT 49 - -
RELATIVE COSTS FOR DIFFERENT STRUCTURAL SYSTEMS MAY BE DETERMINED USING DATA AVAILABLE FOR PRELIMINARY ESTIMATES. (P6,100)

- - PRIMARY BIT 50 - -
THE BENEFIT DERIVED FROM POSSIBLE REDUCTIONS IN ON-SITE LABOR IS TO BE SUBJECTIVELY EVALUATED. (P6,90)

- - PRIMARY BIT 51 - -
THE FIRE RESISTIVITY OF DIFFERENT MATERIALS AND COMPONENTS CAN BE ESTIMATED USING DATA PROVIDED IN CODES, BUILDING STANDARDS AND TEST RESULTS. (P7,100)

- - PRIMARY BIT 52 - -
THE OVERALL BEHAVIOR OF THE STRUCTURE DURING A FIRE IS BEST ASSESSED SUBJECTIVELY. (P7,95)

- - PRIMARY BIT 53 - -
THE LATERAL LOADS TO BE USED IN ANALYSIS ARE STATIC EQUIVALENTS THAT APPROXIMATE THE DYNAMIC BEHAVIOR OF EXTREME LOADS.

- - PRIMARY BIT 54 - -
EMPIRICAL RULES WILL BE USED IN ASSESSING DYNAMIC INELASTIC BEHAVIOR OF LOCAL AREAS AND COMPONENTS OF THE STRUCTURE IN EXTREME LOAD SITUATIONS. (P8,75;P9,100)

The design descriptors necessary for carrying out an initial evaluation based on the above BIT's were then determined. These design descriptors were condensed together for the identification of the relevant design dimensions. This process was quite extended, and only a brief comment is given along with the related results. The main problem encountered was again due to the desire to provide the smallest possible subspaces to aid in the eventual task of evaluation. This led to the development of design dimensions with two different metrics. The design dimensions with the coarser metric were intended to describe the basic structural system; these were:

D1 - Primary Structural System (Type)
D2 - Primary Structural System (Elements)
D3 - Primary Structural System (Geometry)
D5 - Structural Walls (Useage)

The major difficulty encountered with this basic definition of the structure was the separation of the structural type into two dimensions, D1 and D5. The intent was that the knowledge of the presence of the structural walls or of a framing type regardless of the existence of the structural walls, would be adequate for certain evaluations. This decision was also based upon the possible existence of more than one type of framing system; however, this possibility has been limited with recent revisions of the related codes [87]. Thus, in hindsight, D1 and D5 could have been effectively combined into one dimension.

At the finer metric, four additional dimensions were defined that described the nature of the walls, the floors and the connections that could be incorporated into the structural system; these were:

- D4 - Structural Walls (Type)
- D6 - Floor Systems
- D7 - Primary Structural System (Connections)
- D8 - Planar, Intersurface and Surface to Structural System, Connections

As the design dimensions were being identified, membership values were being assigned for associating each evaluative BIT with the new index sets, DI, related to the design dimensions. This membership assignment, $\mu_{DI}(\eta(k))$, is identical to the membership values, $\mu_{\eta(k)}(DI)$, required in formula 5.1. The index values associated with the evaluative BIT's and required in the determination of the conditional relationship between the Perfromance Space and The Design Space are given in Table 6.2. Table 6.3 contains the resulting membership values, $\mu_{PJ}(DI)$, derived from formula 5.1.

6.4.3 The Completed Design Space

Once the dimensions of the design space have been identified, the space is completed by the introduction of secondary BIT's containing descriptions of potential values to be associates with the dimension described in the linked primary BIT. These SBIT's are ideally derived from information already contained in the library; however, within the context of the sample problem, some of the values came directly from the designer.

PBIT_{67} is an example of how information in the library may suggest a potential value for a design dimension. This BIT was entered into the library as a result of research

PBIT	P1	P2	P3	P4	P5	P6	P7	P8	P9	D1	D2	D3	D4	D5	D6	D7	D8
43	1.00									.30	.20	1.00	.20	.70	.60		
44		1.00									1.00		1.00		.90	.40	.40
45			1.00			.85				.80	.90	.50	.80	.85	.85	.85	.80
46				.90				1.00		1.00	.50	.85	.20	.90	.40	.10	.10
47				1.00						.70	.80	.50	.10	.30	.90	.10	.10
48					1.00					.60	.65	.30	.95	.85	.95	.20	.50
49						1.00				.80	1.00	.50	.85	.40	.85	.70	.70
50						.90				.80	.90		.85		.85	.80	.90
51							1.00			.50	.85		.95	.65	.95		
52							.95			.75	.75	.50	.85	.85	.85	.50	.50
53				.75				1.00		1.00	.85	.50	.70	1.00	.80	.30	.30
54								.75	1.00	.75	.95	.40	.90	.40	.95	.95	.95

Table 6.2: Index Membership Values for Evalautive BITs
(only non-zero values recorded)

	P1	P2	P3	P4	P5	P6	P7	P8	P9
D1	.55		.89	.95	.77	.89	.84	1.00	.87
D2	.45	1.00	.95	.89	.81	1.00	.92	.92	.97
D3	1.00		.71	.87	.55	.71	.68	.92	.63
D4	.45	1.00	.89	.74	.97	.92	.97	.83	.95
D5	.83		.92	.92	.93	.86	.90	1.00	.63
D6	.55	.95	.92	.95	.97	.92	.97	.92	.87
D7		.63	.92	.46	.45	.86	.68	.84	.97
D8		.63	.89	.46	.71	.90	.68	.84	.97

Table 6.3: Conditional Relationship Between the Performance Space and the Design Space (non-zero values not recorded)

P	α	Relevant Design Dimensions	Vectors in Subspace	Feasible Vectors in Subspace	Sample Vectors	
					Vector	$\mu_{A(J)}$
P1	.90	D3	3	3	[0,0,1,0,0,0,0,0]	1.00
P2	.90	D2,D4,D6	168	168	[0,3,0,6,0,1,0,0]	.30
P3	.90	D2,D5,D6,D7	224	87	[0,3,0,0,1,1,4,0]	1.00
P4	.88	D1,D2,D5,D6	224	58	[2,3,0,0,1,1,0,0]	.80
P5	.90	D4,D5,D6	84	36	[0,0,0,6,1,1,0,0]	.80
P6	.90	D2,D4,D6,D8	672	143	[0,3,0,6,0,1,0,1]	.83
P7	.90	D2,D4,D5,D6	336	144	[0,3,0,6,1,1,0,0]	.90
P8	.90	D1,D2,D3,D5,D6	672	82	[2,3,1,0,1,1,0,0]	.70
P9	.90	D2,D4,D7,D8	384	84	[0,3,0,6,0,0,4,1]	.45
Totals			2767	805	[2,3,1,6,1,1,4,1]	.65

Table 6.4: Summary of the Uncoupling of the Design Space

done in reference to structural behavior in a fire environment, P5.

> - - PRIMARY BIT 67 - -
> CONCRETE ENCASEMENT OF STEEL BEAMS AND COLUMNS, PROVIDED ESSENTIALLY FOR FIRE PROTECTION, SERVES TWO STRUCTURAL FUNCTIONS. FIRST, IT STIFFENS THE MEMBER, REDUCING THE DEFLECTION UNDER LOAD AND INCREASING THE LOAD THAT CAN BE CARRIED BEFORE LATERAL OR TORSIONAL BUCKLING TAKES PLACE. SECONDLY, THE CONCRETE CAN SUPPORT LOAD.

PBIT_{67} suggests the introduction of a value, primary elements of steel encased in concrete, for D2; thus SBIT_{98} containing the above BIT of information was entered and linked to PBIT_{57} describing D2. Another example of this type of influence of evaluative information comes from SBIT_{47} that is linked to PBIT_{27} already given in Section 6.3.3.

> - - SECONDARY BIT 47 - -
> THE OPTIMUM SHEAR WALL DESIGN IS TO PRODUCE A COMBINATION OF A DUCTILE FRAME WITH A DUCTILE WALL PANEL. THIS EFFECT HAS BEEN ACHIEVED IN SOME CASES BY SUCH DEVICES AS A SLIT SHEAR WALL.

This SBIT of information suggested the linkage of SBIT_{108}, with a value of cast in-place modified reinforced concrete walls (e.g., slitted), with PBIT_{59} describing D4. A broad influence is also exerted by PBIT_{5} with the introduction of both prefabricated wall and floor elements.

A problem encountered in assigning values to design dimensions is the handling of mixed systems. For example, if the primary elements of the system are some combination of both steel and reinforced concrete, there must be a SBIT that allows for mixed type connections (e.g., welding and monolithically cast). If given full rein, this type of enumeration would rapidly lead to combinatorial explosion, and yet the handling of it in the context of the problem was not really satisfactory. This appears to be a potential weakness in MAGID which partially clouded the example.

The previous commentary was intended to help explain the origins of some of the SBIT's that contained the potential values of the design dimensions. Below is listed the entire design space; each PBIT describing the design space is given and then followed by all the linked SBIT's containing the dimension's potential values:

D1 - PRIMARY STRUCTURAL SYSTEMS (TYPE)
- SPACE FRAME - DUCTILE MOMENT RESISTING (1D ELEMENTS)
- PLANAR SYSTEMS - BEAM WALLS (2D ELEMENTS)
- MIXED SYSTEMS (1 and 2D ELEMENTS)
- SPACE FRAME - MOMENT RESISTING (1D ELEMENTS)

D2 - PRIMARY STRUCTURAL SYSTEM (ELEMENTS)
- PRIMARY ELEMENTS OF STEEL
- PRIMARY ELEMENTS OF STEEL ENCASED IN CONCRETE
- PRIMARY ELEMENTS OF REINFORCED CONCRETE
- MIXED SYSTEMS - REINFORCED CONCRETE AND STEEL
- PRIMARY ELEMENTS OF PRESTRESSED CONCRETE

D3 - PRIMARY STRUCTURAL SYSTEM (GEOMETRY)
- PRESENT CONFIGURATION - POSSIBILITY OF CHANGE IN FENESTRATION
- PRESENT CONFIGURATION WITH THE FOLLOWING CHANGES:
  1. POSSIBILITY OF CHANGE IN FENESTRATION
  2. GENERAL OR LOCAL THICKENING OF EXTERIOR AND/OR INTERIOR WALLS
- PRESENT CONFIGURATION WITH THE FOLLOWING CHANGES:
  1. POSSIBILITY OF CHANGE IN FENESTRATION
  2. GENERAL AND LOCAL THICKENING OF EXTERIOR AND/OR INTERIOR WALLS
  3. CANTILEVERING OF BEDROOM PROTRUSIONS FROM THE MAIN STRUCTURE

D4 - STRUCTURAL WALLS (TYPE)
  - HOLLOW WALLS ENCLOSING STEEL BRACING
  - HOLLOW WALLS ENCLOSING STEEL DIAPHRAGM
  - CAST IN-PLACE REINFORCED CONCRETE WALLS
  - CAST IN-PLACE MODIFIED REINFORCED CONCRETE WALLS (E.G., SLITTED)
  - PRECAST REINFORCED CONCRETE PANELS
  - NO STRUCTURAL WALLS WILL BE USED

D5 - STRUCTURAL WALLS (USEAGE)
  - NO STRUCTURAL WALLS WILL BE USED
  - STRUCTURAL WALLS (BASICALLY SHEAR WALLS) WILL BE USED

D6 - FLOOR SYSTEM
  - REINFORCED CONCRETE FLAT SLABS - CAST IN-PLACE
  - CONCRETE SLAB POURED ON TOP OF STEEL DECK
  - REINFORCED CONCRETE WAFFLE SLABS - CAST IN-PLACE
  - PREFABRICATED FLOOR ELEMENTS - REINFORCED CONCRETE
  - PREFABRICATED FLOOR ELEMENTS - STEEL
  - COMPOSITE FLOOR - STEEL JOISTS AND CONCRETE SLAB
  - PRESTRESSED CONCRETE FLOOR SLABS

D7 - PRIMARY STRUCTURAL SYSTEM (CONNECTIONS)
  - WELDED PRIMARY STRUCTURAL CONNECTIONS
  - BOLTED PRIMARY STRUCTURAL CONNECTIONS
  - COMBINATION FOR PRIMARY STRUCTURAL CONNECTIONS BOLTED OR WELDED AND CAST
  - CAST PRIMARY STRUCTURAL CONNECTIONS

D8 - PLANAR, INTERSURFACE AND SURFACE TO STRUCTURAL SYSTEM CONNECTIONS
  - CAST PLANAR SURFACE CONNECTIONS
  - MECHANICAL CONNECTION OF PLANAR SURFACES (E.G., BOLTING)
  - WELDED CONNECTION OF PLANAR SURFACES
  - COMBINATION FOR CONNECTION OF PLANAR SURFACES (MECHANICAL OR WELDED AND CAST)

6.5 The Identification of Acceptable Design Kernels

The identification of acceptable design kernels is achieved in three additional steps after the construction of the design space. First, the design space is uncoupled into subspaces and the resulting subspace vectors are then evaluated (level nine). The evaluated subspaces are then recoupled, creating a fuzzy set of acceptable design kernels (level ten). This section attempts to convey how these processes were carried out in working the sample problem.

6.5.1 Uncoupling and Evaluating the Design Subspaces

To uncouple the design space, it is necessary to prescribe a threshold, α, for each of the performance dimensions. Level nine can be used in a preliminary manner that allows the designer to explore the subspaces generated by different values of α without actually enumerating all of the subspace vectors. The major influence controlling the choice of this uncoupling threshold, α, is the desire for the subspace to contain a manageable number of vectors and yet provide an adequate description of the design to allow for evaluation. Data reflecting this uncoupling operation is presented in Table 6.4; this includes the thresholds used in the sample problem, which design vectors were isolated into the subspaces and the resulting number of vectors in each subspace.

Once the designer receives the output of level nine, his first task should be to review the displayed information intended to aid in the evaluation of the subspace vectors. An initial consideration should be to eliminate all of the obviously infeasible, unacceptable, vectors. These unacceptable vectors will include solutions with logically incompatible values such as a moment resisting frame without structural walls or bolted, cast in place planar surfaces (i.e., walls and floors). In addition, the designer may choose to eliminate highly improbable solutions such as welded reinforced concrete primary members or the use of steel decking in a reinforced concrete frame. As a result of this type of elimination process, the original total of 2767 subspace vectors was reduced to 805 feasible subspace vectors. These reductions are given for each individual subspace in Table 6.4. It is felt that a brief discussion and example evaluations for four of the subspaces will adequately convey the process involved in working the sample problem. As was implied earlier, it should be understood that some of the membership assignment in the sets of acceptable subspace vectors was done in the spirit of the sample problem, although an attempt was made to keep these evaluations as realistic as possible. In addition to the following discussion, Table 6.4 contains a sequence of compatible subspace vectors and their assigned membership values in the set of acceptable subspace solutions.

6.5.1.1 Subspace Two - D^2

Subspace two contains those dimension of the design space that are relevant in the determination of the amount of steel incorporated in the design, $PBIT_{29}$. The mapping of these subspace vectors into P2 is achieved by identity mapping, $PBIT_{44}$. In addition, $PBIT_{69}$, given below, was introduced in level six to guide in the assignment of membership values.

```
- - PRIMARY BIT 69 - -
    THE BASIC VALUES TO BE ASSOCIATED WITH THE USE OF STEEL
    ARE AS FOLLOWS:
                  0.5  USE OF STEEL IN PRIMARY SYSTEM
                  0.2  USE OF STEEL IN FLOOR SYSTEM
                  0.2  USE OF STEEL IN STRUCTURAL WALL IF PRESENT
                  0.1-0.3  VALUE OF STRUCTURE REGARDLESS OF
                           MATERIAL
```

It was necessary to extend $PBIT_{69}$ to include steel encased in concrete as having the same value as a steel system.

The subspace vector, $\underline{d}_2^{30}$ [0,1,0,5,0,2,0,0], which describes a steel primary system, a precast reinforced concrete wall, and a concrete floor slab poured on top of a steel deck, received a value of 0.80. Similarly, the subspace vector, $\underline{d}_2^{116}$ [0,2,0,5,0,4,0,0], which describes a steel encased in concrete primary system and precast reinforced concrete wall and floor slabs, received a value of 0.60. The subspace vector, $\underline{d}_2^{120}$, given in Table 6.4 for P2, describes a reinforced concrete primary system, without structural walls and with a cast-in-place reinforced concrete flat slab floor, received a minimum value, for a design without structural walls, of 0.30.

6.5.1.2 Subspace Four - D^4

Subspace four contains those design dimensions relevant to the assessment of a design's serviceability requirements. Serviceability is defined in terms of acceptable performance under normal loading situations, where acceptable performance is interpreted in terms of deformational and vibrational characteristics. In the assessment of serviceability criteria, it is normally assumed that a structure behaves in a linear elastic fashion. The problem is usually uncoupled into the examination of vertical deflections associated with live loads and the lateral drift associated with wind or moderate seismic loads.

The initial consideration of floor deflections is a subjective consideration originating from a general understanding of elastic analysis. The stiffness of a member controls its deflection and is proportional to the term EI (modulus of elasticity x moment of inertia); thus, an increase in either E, a stiffer material, or I, a deeper cross-section, should lead to a floor system with smaller deflections. In this manner, stiffer floor systems were given preference; for example, a reinforced concrete waffle slab was considered better than a reinforced concrete flat slab. PBIT_{14} contains the limiting deflections for floor systems and also contains a simple technique for accounting for time dependent behavior of concrete. The different weighting factors in PBIT_{14} suggest a subjective preference for the use of steel in improving serviceability characteristics.

These initial subjective considerations used in evaluating the problem can be extended by the designer to more objective procedures. To accomplish this, the designer must refine the metric of the description through an approximate sizing of the components for different floor configurations. With this refined description, approximate calculations can be carried out to provide a lower bound for potential deflections. Analyses of this type might include the use of simple beam theory or a generalized plate theory; in addition, available tables may well contain such deformational information. If this simplified analysis does not provide enough information, it might be possible to further extend the evaluation to include a computer analysis that would take into account the floor system and its support members. Such detailed analysis could also be done to provide information relative to creep and cracking of relevant floor systems. While such sophisticated models are normally associated with the latter stages of a design process, there is no reason why they cannot be used by a designer in the earlier stages of the process to help verify an initial subjective evaluation.

Horizontal drift, deflection, resulting from wind or minor to moderate seismic loads is greatly reduced by either the inclusion of shear walls within frames or by using a planar primary structural system. $PBIT_{19}$, static equivalent seismic loads, indicates that seismic forces in a ductile moment resisting frame with shear walls will be considerably less than for box (planar) systems. This indicates that structures using ductile moment resisting frames with shear walls will generally receive a high membership value in the set of acceptable subspace vectors. This type of broad assessment of relative merit comes directly from the information contained in $PBIT_{19}$ relative to the 'K' coefficient.

The geometric configuration of the Parkmerced Towers does not allow for a simple uncoupling of the framing systems as was previously discussed. However, it is still possible, using information from $PBIT_{19}$, to estimate the forces that the structure might be subjected to in a moderate earthquake. An example of these rough calculations is given later in the section. It is also possible for the designer to refine the design description such that it would be possible to use an appropriate dynamic analysis computer program to provide more detailed information in assessing the system's seismic behavior.

The serviceability requirements tend to find preferential solutions in stiffer structural systems. Unfortunately, stiffer structures incur larger seismic forces, SBIT_{45}, and are better transmitters of acoustical and mechanical vibrations. In residential structures, like Parkmerced, these later considerations are outweighed by the deformational considerations giving rise to a decision BIT, PBIT_{74}.

- - PRIMARY BIT 74 - -
THE QUESTIONS RELATING TO THE SERVICEABILITY LIMIT STATE IN RESIDENTIAL DESIGN (E.G., NO HEAVY VIBRATIONS) HAVE A PREFERENTIAL ANSWER IN STIFFER STRUCTURES.

Two examples of typical evaluations are given to help illustrate the previous discussion.

The first illustration deals with subspace vector $\underline{d}_4^9$, [1,1,0,0,1,2,0,0], describing a steel ductile moment resisting frame with structural walls and a concrete floor slab poured on top of steel decking. Along with a majority of other frames incorporating shear walls, the major concern in this evaluation is how to include the shear walls into the existing design geometry. Figure 6.2 shows one possible configuration for the location of these structural walls. Using approximate methods, the base shear of the wall at the end of a leg was estimated to be 150 kps for both the UBC wind loading, $PBIT_8$, and earthquake loading $PBIT_{19}$. Figure 6.3 shows a possible steel x-bracing configuration capable of carrying this base shear at the end of a leg. A 12 x 6 rectangular steel section weighing 31.24 pounds per foot would be required, while if a concrete shear wall with 2% reinforcing were used, it would require an approximate thickness of 10 inches. While the lateral force carrying system would appear to give acceptable behavior, the floor system of concrete on top of steel decking was felt to have a potential deflection problem over longer spans; thus, the membership assigned was 0.85. The same basic design with reinforced concrete primary elements, $\underline{d}_4^{37}$, was considered unacceptable because of the incompatibility of the steel decking with primary reinforced concrete elements. However, a reinforced concrete ductile moment resisting frame with structural walls and a

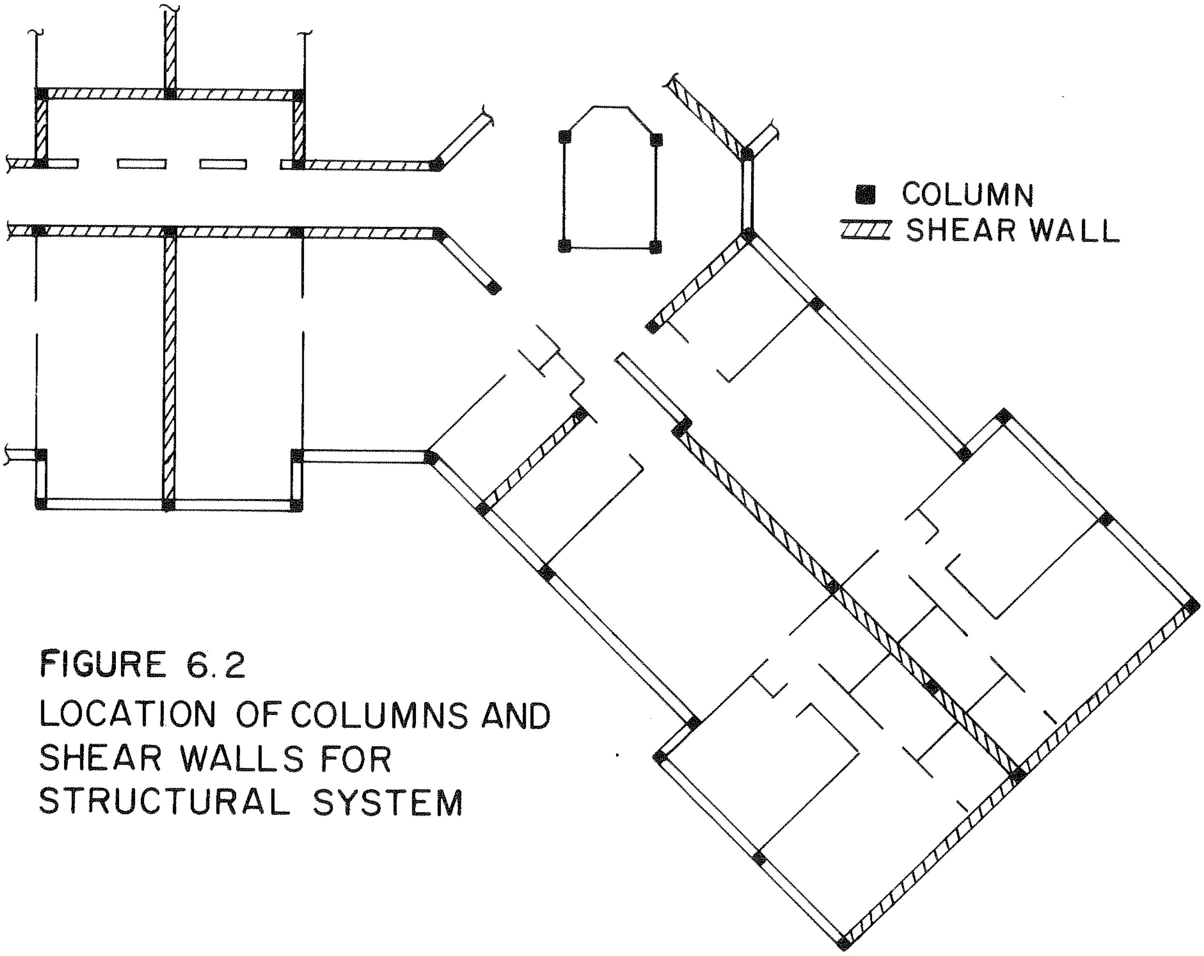


FIGURE 6.2
LOCATION OF COLUMNS AND SHEAR WALLS FOR STRUCTURAL SYSTEM

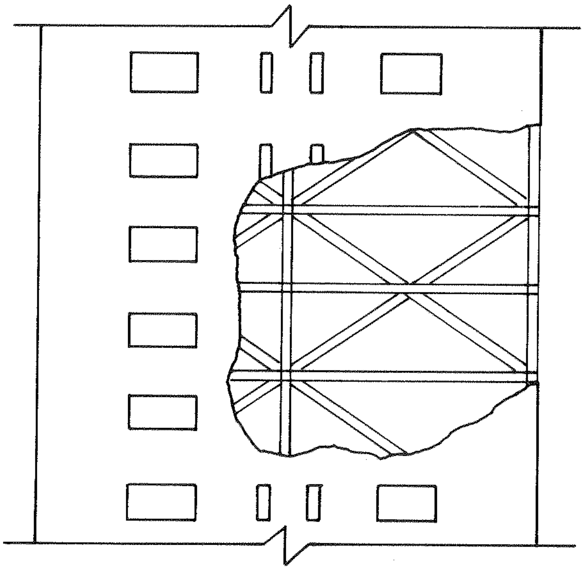


FIGURE 6.3
POSSIBLE CONFIGURATION FOR X BRACING AT END OF BUILDING LEG

cast-in-place, flat slab floor system, $\underline{d}_4^{36}$, received a membership value of 0.80, reflecting the potentially undesireable long and short term deflection characteristics of the floor slab. Another variation on $\underline{d}_4^{9}$ is the elimination of the structural walls, $\underline{d}_4^{2}$, which received a membership value of 0.35 because of the possible trouble with lateral drift.

The second illustration deals with subspace vector $\underline{d}_4^{25}$, [1,2,0,0,2,4,0,0], a steel encased in concrete ductile moment resisting frame with structural walls and prefabricated concrete floor elements. The problem of layout was similar to that previously expressed, and the solution offered in Figure 6.2 is applicable. The problems inherent with this system developed more from the tradeoff between the system's increased member stiffness and the additional mass from the encasement of the concrete. In addition, the prefabricated floor elements were questionable both from their potential flexibility and potential problems relating to system continuity. Figure 6.4 is intended to give the reader an idea of how this system might appear if, in addition to the prefabricated floor elements, the wall elements were also prefabricated. This design received a membership value of 0.75 because of the questionable floor system and the additional mass.

This type of subjective evaluation is not necessarily indicative of preferable behavior of one system over another if both are well designed, but rather, that one system is better suited to be designed for the desired performance. As is seen

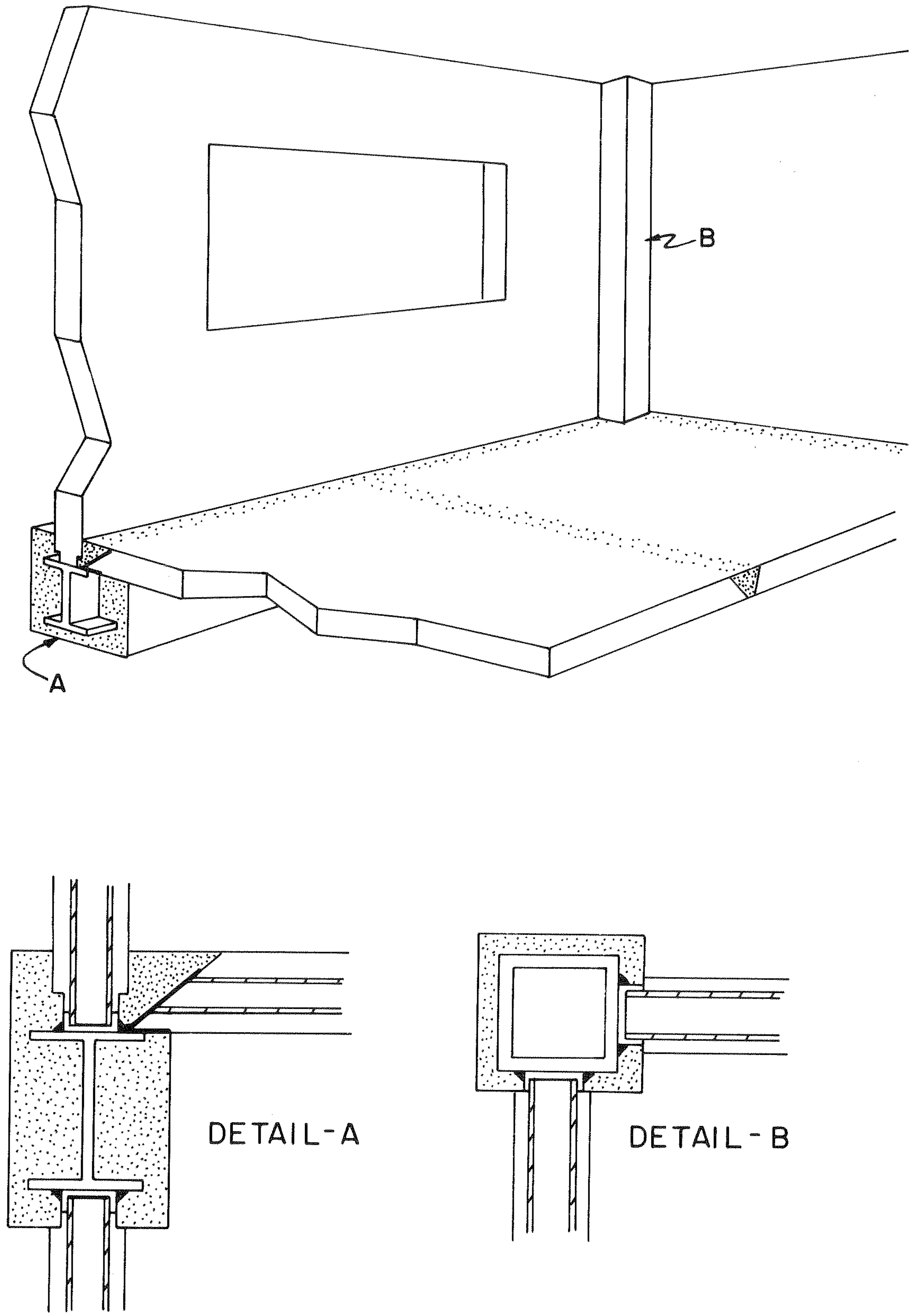


FIGURE 6.4 POSSIBLE DETAILS FOR STRUCTURAL SYSTEM OF STEEL ENCASED IN CONCRETE, WITH PREFABRICATED CONCRETE FLOOR AND WALL ELEMENTS

by the elimination of $\underline{d}_4^{37}$ for constructional reasons, it is difficult for a designer to entirely ignore the influence of criteria other than the one relevant to the subspace being worked on. In some ways, this reality is actually desirable, for in this manner a designer can continually eliminate incompatible values and reduce the required number of evaluations. The evaluation of the subspace is not intended to be an exacting process, but rather, a process in which the designer exercises his judgement relative to a particular conceptual alternative and records the response as the membership function of a set of acceptable design alternatives.

6.5.1.3 Subspace Seven - D^7

Subspace seven contains those design dimensions relevant in assessing the behavior of a design in a fire environment. All of the materials specified in the design space are permissible within the related statutory requirements, $PBIT_{10}$.

```
- - PRIMARY BIT 10 - -
    THE BUILDINGS WILL BE OF CONSTRUCTION TYPE I AND THEY ARE
    ASSUMED TO BE IN A FIRE ZONE 1 AREA.
      - - SECONDARY BIT 15 - -
          THE STRUCTURAL ELEMENTS IN TYPE I BUILDINGS SHALL BE
          OF STEEL, IRON, CONCRETE, OR MASONARY.  WALLS AND
          PERMANENT PARTITIONS SHALL BE OF NONCOMBUSTIBLE FIRE
          RESISTIVE CONSTRUCTION EXCEPT THAT PERMANENT NONBEAR-
          ING PARTITIONS OF ONE-HOUR FIRE-RESISTIVE CONSTRUCTION
          MAY USE FIRE RETARDANT TREATED WOOD (SEE SECTION 407)
          WITHIN THE ASSEMBLY (UBC, SEC 1801, PG 101).
```

The evaluation and assignment of value then was based upon the element's inherent fire endurance, $PBIT_{80}$; that is, its acceptability without additional fire protection and the integration of the elements in relation to fire containment.

> - - PRIMARY BIT 80 - -
> PREFERENCE SHOULD BE GIVEN TO STRUCTURAL COMPONENTS THAT CAN IN THEMSELVES MEET FIRE REQUIREMENTS.

Using this consideration of inherent endurance, the primary structural elements were ranked in the following order: reinforced concrete, steel encased in concrete, mixed systems, and steel (post tensioned elements were eliminated because of an initially weak linkage to D2). The existence of structural walls was generally considered a preferable situation, especially when they also had inherent fire endurance. In this sense, all of the concrete walls were preferred over the hollow walls enclosing steel bracing.

Floor systems provided the greatest variation in assessing fire behavior. It is considered very important to isolate a fire on its floor of origin and to prevent the collapse of the floor immediately above into the fire. The critical factor in rating all the floors was the rate with which the steel present would be heated and therefore lose strength. A key to this evaluation was provided by PBIT_{66}.

> - - PRIMARY BIT 66 - -
> A THINNER WALL OR COLUMN MORE QUICKLY ATTAINS HIGH TEMPERATURES THROUGHOUT ITS THICKNESS AND MIGHT THEREFORE HAVE ONLY A SHORT LIFE AS A FIRE BARRIER. THE TIME FOR WHICH A STRUCTURAL ELEMENT CONTINUES TO FUNCTION SATISFACTORILY IN FIRE IS KNOWN AS ITS FIRE RESISTANCE PERIOD OR, SIMPLY, ITS FIRE RESISTANCE.

Thus, the thinner sections of the waffle slab and the higher strength of the prestressing steel would cause them to be less preferable than a cast-in-place or precast slab. In the same sense, a concrete slab on steel decking is preferable to one on steel joists, because the concrete acts as a heat sink keeping the steel deck cooler than comparable joists. Table 6.5 provides several examples of the membership of subspace vectors in the set of acceptable solutions. These examples provide a reasonable summary of the application of the rankings previously discussed.

The evaluation of fire performance used in the sample problem was highly subjective; however, the evaluation of a structural system in the context of a fire can be placed upon a more objective basis. An example of a more comprehensive evaluation performed on subsystems of approximately the same metric can be found in the work of Nelson and Daley [92]. By further refining the metric of the design description, it is possible to assess the behavior of individual elements in a fire environment through the coupling of a non-linear heat flow analysis capability to an appropriate structural analysis program as suggested by Bizri [93] or Lie and Allen [94]. With this additional metric refinement, it should also be possible to develop a probabilistic model, as suggested by Magnusson [95], that could provide a measure of integrity for the subassemblies of complex structures containing fires. Finally, the work of Pettersson [96] and Kawagoe [97] suggests the consideration of fire as another limit state that encompasses

Subspace Vector	D2 Primary Elements	D4 Walls (Type)	D5 Walls (Useage)	D6 Floor System	$\mu_{A(7)}$
$\underline{d}_7^{8}$	Steel	Hollow Walls Enclosing Steel Bracing	To be Used	Reinforced Concrete Flat Slab	0.50
$\underline{d}_7^{13}$	Same	Same	Same	Steel Joists with Concrete Slab	0.30
$\underline{d}_7^{67}$	Same	Precast Reinforced Concrete Panels	Same	Reinforced Concrete Flat Slab	0.85
$\underline{d}_7^{71}$	Same	No Walls	No Walls	Same	0.75
$\underline{d}_7^{122}$	Steel Encased in Concrete	Cast-in-place Reinforced Concrete	To be Used	Reinforced Concrete Waffle Slab	0.90
$\underline{d}_7^{148}$	Same	Precast Reinforced Concrete Panels	Same	Reinforced Concrete Flat Slab	0.95
$\underline{d}_7^{204}$	Reinforced Concrete	Cast-in-place Reinforced Concrete	Same	Same	1.00
$\underline{d}_7^{224}$	Same	Cast-in-place Modified Reinforced Concrete	Same	Prestressed Concrete Slab	0.95
$\underline{d}_7^{239}$	Same	No Walls	No Walls	Reinforced Concrete Flat Slab	0.90

Table 6.5: Typical Vectors in the Fire Subspace and Their Evaluated Membership in A(7)

all the related problems, such as structural integrity, smoke emission and fire spread. These objective techniques can provide the designer with rational basis for assessing the effect of fire on structures that, in turn, can be effectively used in both initial decision making, as in MAGID, or in the later stages of a design process.

6.5.1.4 Subspace Eight - D^8

Subspace eight contains those design dimensions relevant in assessing a design's ultimate behavior during extreme load situations, particularly during a major earthquake. The primary emphasis of the evaluation and decision process is to control the forces induced in the structure by the earthquake, $PBIT_{39}$ and $PBIT_{40}$, to establish a structure's ability to sustain repeated inelastic reversals, $PBIT_{82}$, to assure the structure's continuing stability, $PBIT_{83}$, and to ascertain the floor's ability to act as a diaphragm capable of transferring shear forces, $PBIT_{81}$.

```
 - - PRIMARY BIT 39 - -
     IN A SEVERE EARTHQUAKE STRUCTURES OFTEN UNDERGO LARGE
     DEFORMATIONS LEADING TO STIFFNESS DETERIORATION AND
     THEREFORE AN INCREASE IN THE STRUCTURES FUNDAMENTAL PERIOD.
 - - PRIMARY BIT 40 - -
     DYNAMIC DESIGN CONSIDERATIONS DICTATE THAT THE FUNDAMENTAL
     PERIOD OF A STRUCTURE SHOULD BE AS FAR REMOVED AS POSSIBLE
     FROM THE FUNDAMENTAL PERIOD OF THE SITE.
 - - PRIMARY BIT 82 - -
     IN AN EFFORT TO GUARD AGAINST ULTIMATE STRUCTURAL FAILURE
     IN AN EARTHQUAKE, A STRUCTURE SHOULD BE ABLE TO CONTINUE
     TO ABSORB ENERGY THROUGH MULTIPLE REVERSALS AND AS HIGHER
     ORDER INELASTIC MECHANISMS COME INTO EXISTANCE.
 - - PRIMARY BIT 81 - -
     THE FLOOR SHOULD BE DESIGNED TO PROVIDE ADEQUATE DIAPHRAGM
     ACTION IN ORDER TO DISTRIBUTE HORIZONTAL LOADS.
```

In evaluating the ultimate behavior of a structure subjected to seismic motion, it is no longer sufficient to use a static equivalent force concept and linear elastic analysis. While the force levels induced in the structure are still critical, the extent of the structure's ability to sustain inelastic deformation is of equal importance. In essence, there is a trade-off between the level of the forces induced by an earthquake and the amount of damage associated with large inelastic deformations (P9). The determination of this inelastic behavior is dependent upon both the behavioral characteristics of the structure and the actual earthquake to which the structure will be subjected; thus, in assessing inelastic behavior, a thorough study of the area's seismicity and the structure's behavioral characteristics is warranted.

Empirical rules, like $PBIT_{40}$, can have a significant impact on the decision for a particular structural configuration. An initial assessment of the site's fundamental period, $PBIT_{64}$, indicated a possible range of from 0.8 to 1.2 seconds. Using methods suggested by $PBIT_{19}$, it is possible to estimate a structure's fundamental period in the elastic range of behavior; from 0.5 to 0.6 seconds for structures using shear mechanisms and from 1.2 to 1.4 seconds for flexible structures like a ductile moment resisting frame. These estimates of fundamental periods coupled with $PBIT_{40}$ indicate a strong preference for stiffer buildings; that is, buildings that carry lateral forces through shear mechanisms. $PBIT_{39}$ suggests the possibility that during an earthquake the fundamental period of a structure might increase in proportion to the amount of damage being done; thus, the choice between structural types cannot be completely answered by the previous rough calculations.

The question of overall structural stability expressed in PBIT_{83} is a major concern of this design because of its proximity to active faults, PBIT_{6}. This problem of stability becomes one of the major controlling influences in the decision to preferentially treat designs incorporating shear mechanisms to resist seismic forces.

While empirical rules can help the designer make initial approximations of a structure's ultimate behavior, a more refined evaluative technique would be required to really answer the questions raised. An initial improvement might be achieved through the estimation of potential collapse mechanisms of the structure keeping in mind the problems created by seismic reversals, PBIT_{82}. To actually carry out a detailed analysis on a computer, it is necessary to decide what form an earthquake might take in the immediate area of the structure. This can be done by studying local seismic conditions and modeling the effect that soil might have in modifying bedrock motion [98,99]. These ground motions could then be used as input in an appropriate computer program for inelastic analysis [100,101] and the structure's potential behavior predicted. In addition, it might be possible to model the interaction between the structure and the soil [102]. While all these advanced analytical techniques are available, their use would require a considerable refinement in the present metric of the design description. With the exception of determining the site seismicity, it would be difficult to use these advanced techniques. Because of this, inelastic behavior is handled principally in the basis of empirical rules.

In addition to examining a structure's overall behavior, it is essential to consider the potential for inelastic behavior for the components that make up the structure. It is in this inelastic behavior that a structure is able to absorb and dissipate energy, $PBIT_{30}$. The question then is how much inelastic deformation can a member undergo and still retain some structural integrity. Again, there are basic empirical rules that allow for judgmental decisions; for example, $SBIT_{46}$ suggests the potential for brittle failure in poorly designed shear mechanisms which gives ductile shear walls a preferential status over the potentially more brittle beam walls. Certain elements should be designed to function only in the elastic range in order to maintain any real structural integrity; for example, a buckled X-brace will lead to the yielding of the other brace,and because of load reversal, a range of no lateral support can develop. Some of the problems dealing with this question of ductile behavior can be answered with preliminary calculations that will not require any significant metric refinement; however, detailed evaluation of a member's potential ductility may require significant metric refinement in order to do computer analyses or possible experimental studies.

The synthesis of all of this information relative to an entire concept would seem to be highly subjective. Several examples should convey the results of the author's attempt at this evaluation. The subspace vector given in Table 6.4, [2,3,1,0,1,1,0,0], essentially describes the existing design. This subspace vector has received a membership value of 0.70 particularly reflecting the questionable use of a planar system.

Another example is $\underline{d}_8^{23}$, [1,1,2,0,2,2,0,0], a steel ductile moment resisting space frame, with shear walls and a concrete slab poured on a steel deck. This subspace vector received a value of 0.90. The only detracting feature of this design was the possibility of trouble with the floor system acting as a diaphragm. Design vector $\underline{d}_8^{70}$, [1,2,20,2,7,0,0], describes a steel encased in concrete ductile moment resisting space frame, with shear walls and a prestressed concrete floor. The added mass of the encasing concrete is considered to be offset by the potentially better inelastic properties of this type of element. The prestressed floor is considered to be the most likely floor system to have trouble under both high shear forces and axial forces; thus, the design received a value of 0.80. The same design without the inclusion of shear walls, $\underline{d}_8^{105}$, received a value of 0.50, showing the definite prejudice against more flexible designs. Design vector $\underline{d}_8^{610}$, [4,3,2,0,2,1,0,0] a reinforced concrete moment resisting frame, with shear walls and a cast-in-place flat slab, received a value of 0.80. This reflected two problems with this design: first, the frame was not ductile, and second, the possibility of higher forces due to the greater mass of a concrete structure. On the other hand, the same design but with a ductile moment resisting frame, $\underline{d}_8^{106}$, received a value of 1.00.

The assessment of membership to subspace vectors is a difficult task when dealing with a problem as complex as a structure's ultimate behavior in a major earthquake. Regardless of the number of empirical rules and experimental facts that the designer has available, the final decision is highly subjective. It is particularly difficult to place a value on the composite design when information is only available for the components of that design. In addition, the best of designs conceptually are only as good as their detailed development and implementation.

6.5.2 Recoupling the Subspaces

Once the evaluation has been completed, the designer assigns weights and lower bound thresholds to each subspace before level ten can recouple them into design kernels. The determination of weights for each subspace is the final statement of values in working the problem. The values of the client expressed in P2 and P6 are each given a weighting of 25. The values associated with the strength of the structure, normally the concern of the structural designer, were reflected in P4, P8, and P9 and received weights of 12, 12, and 10, respectively. Of additional interest to the designer were the performance characteristics of the design associated with fire, P7, and sound transmission, P5, which both received weights of 6. The relative effect of weights of 25 and 6 indicates that P5, for example, will act basically as a constraint relative to P2 or P6. The remaining two performance dimensions, P1 and P3, are relegated to serving almost exclusively as constraints with weights of 2 each.

The original assignment of lower bound thresholds was reasonably arbitrary and was lowered until computation time gave indications of becoming a limiting factor (the progression was from 0.75, for all dimensions, effective cost $1.30, to 0.65, $1.58, and then finally to 0.50). It was noted in these earlier runs of level 10 that the application of a threshold to subspace P2 was very limiting because of the high value associated with the use of steel; thus, it was necessary to lower the threshold for this subspace to 0.01, effective cost $5.30. The value system suggested by the client was altered in an attempt to examine the design's sensitivity to the prescribed values. This alteration was first in the form of decreasing the emphasis placed on the incorporation of steel and second in the form of also decreasing the emphasis placed upon minimizing on-site labor. Table 6.6 gives a summary of the lower bound thresholds used and the three different value systems used.

Subspace	Lower Bound Threshold	Weightings		
		Case 1 Client's Values	Case 2 De-emphasize Steel	Case 3 De-emphasize Steel and On-Site Labor
1	0.50	0.02	0.03	0.03
2	0.01	0.25	0.06	0.08
3	0.50	0.02	0.03	0.03
4	0.50	0.12	0.15	0.20
5	0.50	0.06	0.08	0.10
6	0.50	0.25	0.31	0.08
7	0.50	0.06	0.08	0.10
8	0.50	0.12	0.15	0.20
9	0.50	0.10	0.13	0.17

Table 6.6: Recoupling Thresholds and Weightings

The output of level ten showed that the membership of design kernels in a set of acceptable design kernels was often sensitive to only certain design dimensions. This type of sensitivity meant that the ordered results were often grouped within the context of the more sensitive design dimensions. For example, there was little differentiation between welded and bolted steel connections; therefore, every time steel appeared as a primary material, there would be two closely related designs, one with bolted connections and one with welded connections. The range of membership of the top fifty solutions was only 0.85 to 0.76 for case 1, 0.88 to 0.79 for case 2, and 0.87 to 0.80 for case 3. Because of the subjective nature of the original assignment of the membership values within each subspace and previously mentioned observations, it is more practical to discuss the results of level ten within the concept of groupings of design kernels around the more sensitive design dimensions.

Analysis of all three cases recoupled in level ten indicated certain consistent characteristics within the resulting sets of the fifty most acceptable design kernels. The design kernels seemed to be sensitive to only three of the design dimensions: D1 - the primary structural elements, D4 - the type of structural wall used, and D6 - the floor system. The remaining five design dimensions seemed to either remain basically unchanged or to continually vary, therefore having no real affect. The type of primary structural system, D1 and D5, consistently came out to be either a ductile moment resisting frame or a moment resisting frame where both included the use of structural walls. The differentiation between these two types of framing systems seems reasonably inconsequential, particularly in light of recent code revisions previously mentioned. The geometry of the structure, D3, allowed for possible changes in fenestration and a general or local thickening of the walls. The dimensions describing connections D7 and D8 had continually varying values. It would appear that while connections are a critical aspect of any structural design, their inclusion at this metric is highly questionable. While it may be necessary to conceptualize a connection in the assessment of some criterion, the relevant design dimension cannot reasonably allow for the recording of all such detailed descriptions. Therefore, the generalized connection descriptions used proved to be of little consequence except in the promotion of consistent design descriptions. Because of the insensitivity of the design kernels to the five design dimensions just discussed, it is now possible to examine the results of the sample problem in terms of the three critical design dimensions, D1, D4, and D6.

Given below is the design kernel with the highest membership value in the set of acceptable solutions for case 1:

```
DESIGN NUMBER  1 WITH A VALUE OF  .85
D1 - SPACE FRAME - DUCTILE MOMENT RESISTING (1 D ELEMENTS)
D2 - PRIMARY ELEMENTS OF STEEL
D3 - PRESENT CONFIGURATION WITH THE FOLLOWING CHANGES:
      1.  POSSIBILITY OF CHANGE IN FENESTRATION
      2.  GENERAL OR LOCAL THICKENING OF EXTERIOR AND/OR
          INTERIOR WALLS
D4 - PRECAST REINFORCED CONCRETE PANELS
D5 - STRUCTURAL WALLS (BASICALLY SHEAR WALLS) WILL BE USED
D6 - CONCRETE SLAB POURED ON TOP OF STEEL DECK
D7 - BOLTED PRIMARY STRUCTURAL CONNECTIONS
D8 - COMBINATION FOR CONNECTION OF PLANAR SURFACES (MECHANICAL
     OR WELDED AND CAST)
```

While steel was used in this solution, steel encased in concrete was also well represented in the top fifty design kernels. There were no solutions with primary elements of reinforced concrete. Walls were generally precast reinforced concrete panels and the floor was either a concrete slab poured on top of steel decking or precast reinforced concrete elements. It was not until the 17-th (the ranked order) design kernel with a value of 0.79 that the first cast-in-place floor slab appeared. Steel X-bracing was not particularly common and first appeared in the 16-th design kernel that had a value of 0.79, while a cast-in-place modified reinforced concrete wall did not appear until the 27-th kernel having a value of 0.78. The generalized design kernel suggested by the first case would appear to have primary elements made of either steel or steel encased in concrete, precast reinforced concrete walls and a floor of either a concrete slab poured on top of steel decking or precast reinforced concrete elements.

The second case had a decreased emphasis on the use of steel. The design kernel with the highest membership in the resulting set of acceptable design kernels is given below:

```
DESIGN NUMBER  1 WITH A VALUE OF  .88
D1 - SPACE FRAME - DUCTILE MOMENT RESISTING (1 D ELEMENTS)
D2 - PRIMARY ELEMENTS OF STEEL ENCASED IN CONCRETE
D3 - PRESENT CONFIGURATION WITH THE FOLLOWING CHANGES:
     1.  POSSIBILITY OF CHANGE IN FENESTRATION
     2.  GENERAL OR LOCAL THICKENING OF EXTERIOR AND/OR
         INTERIOR WALLS
D4 - PRECAST REINFORCED CONCRETE PANELS
D5 - STRUCTURAL WALLS (BASICALLY SHEAR WALLS) WILL BE USED
D6 - PREFABRICATED FLOOR ELEMENTS - REINFORCED CONCRETE
D7 - BOLTED PRIMARY STRUCTURAL CONNECTIONS
D8 - CAST PLANAR SURFACE CONNECTIONS
```

Becaue of the decreased emphasis on the use of steel, there was in a sense an increased emphasis of the minimization of on-site labor. This led to a dominant inclusion of prefabricated reinforced concrete floor and wall elements. While steel and steel encased in concrete were still the most common primary elements, reinforced concrete did appear by the 13-th design kernel having a value of 0.84. The first cast-in-place floor appeared in the 12-th design kernel with a value of 0.84, while the first non-precast wall did not appear until the 39-th design kernel with a value of 0.80. In general, the de-emphasis on the use of steel allowed for the introduction of primary elements of of reinforced concrete and strongly promoted the use of precast reinforced concrete floor and wall elements.

The third case had a decreased emphasis in both the ues of steel and the minimization of on-site labor. The design kernel with the highest membership in the set of acceptable design kernels in case 3 had a value of 0.87 and was identical to the highest kernel of case 2 with the exception of the use of a cast-in-place reinforced concrete waffle slab. The essential changes in this case were the increased presence of primary elements of reinforced concrete and a broadened acceptance of various floor systems. These broadly accepted floor systems were: cast-in-place reinforced concrete flat slabs and waffle slabs, precast reinforced concrete elements, and a concrete slab poured on top of steel decking.

The value system presented by the client does seem to promote a trend toward greater use of primary elements of steel; however, there appears to be a conflicting interest in the choice of both wall and floor elements, created by the desire to promote the use of steel and at the same time to minimize on-site labor. If this were an actual design situation, MAGID would have provided the designer a preliminary estimate of the most acceptable design kernels to be further developed in a structural design process.

CHAPTER SEVEN

SUMMARY, SUGGESTIONS AND CONCLUSIONS

7.1 Summary

Structural engineering, while tracing its origins into antiquity, emerges as an identifiable activity with the development and application of tractable analytical models of structural behavior. The comprehension of structural behavior and the ability to predict it have steadily increased, and in the last fifteen years, the introduction of the digital computer has led to a rapid expansion of these predictive abilities. This rapid expansion of predictive abilities coupled to the continually increasing complexity of problems and their innumerable potential solutions are, today, challenging the contemporary processes of the structural designer.

In response to such challenges, this dissertation has attempted to explore structural design within the perspective of human problem solving processes. Structural design has been depicted as a spiraling process in which a need is transformed into a solution -- the design. A section of this spiral reveals definite functions of synthesis, evaluation and decision surrounded by a continual cognitive process. This problem solving spiral is embedded in a medium of information leading to the possibility of interpreting structural design as the acquisition, manipulation and generation of information.

A concept of information has been developed based on BIT's (Basic Information Terms) of information, a primary unit that allows the designer to discriminate. BIT's have been defined as answers to questions and are either descriptive, operational or relational in their content. These BIT's can be collected together in fuzzy sets whose membership is a measure of the BIT's relevance to that particular set. The assignment of these membership values is seen to be a primary function of the designer's continual cognitive processes.

In the review of contemporary approaches to design, it was observed that methods for design often serve their most important function by providing a basis for the externalization of design processes and philosophies. Thus, using the above concepts of problem solving and information, a descriptive model of a structural design process has been developed. From the moment of need recognition, the model envisions a design process that spirals upward in a hierarchical structure of information that ends with the development of an acceptable solution described in sufficent detail necessary for implementation. This spiral process is symbolized by the cyclic interaction of sets of information that correspond to the problem solving cycle of synthesis-evaluation-decision. Because of their strong interaction, these design sets evolve together through a continual cognitive process. With the initial discrimination of information during the early stages of the design process, there eventually emerges a design kernel. This design kernel is then carried upward with increasingly finer metrics until it evolves into a detailed and acceptable solution. The upward spiral is not really a direct linking of well defined sets of information but is, rather, a process full of the solution of sub-problems that, when viewed together, make up fuzzily defined sets of information associated with a structural design process.

A design method called MAGID (The Manipulation and Generation of Information for Design) was developed to show the direct potential of models such as the one just discussed. MAGID is an interactive tool intended to aid a structural designer in the identification of design kernels. This is done by developing an information control system that extends the designer's ability to manipulate and generate information. This control system is centered around an information storage and retrieval system based on fuzzy indexing. Using this system, MAGID guides the designer in the construction of a design space. The construction of this design space is achieved by an inverse process of first defining desired attributes, then determining how these attributes are to be evaluated, and finally, identifying what descriptors are needed for evaluation. The design space is then uncoupled on the basis of a design dimension's relevance to a particular attribute. Each resulting subspace, then, has all of its vectors enumerated and judgmentally evaluated. These subspaces are then recoupled into a fuzzy set of design kernels whose membership function expresses the level of acceptability of each kernel to the initial need.

7.2 MAGID (The Manipulation and Generation of Information for Design

The development of a design philosophy may appear to many as a mere academic exercise; however, this is far from the truth, for every designer either consciously or unconciously operates under the influence of some design philosophy. The real question to be answered is what can be gained from the explicit expression of a design philosophy. The method and sample problem presented in the fifth and sixth chapters are intended to act as an example of what can be directly derived from such a philosophy. In this sense, MAGID is more important as a demonstration of the direct implication of a philosophy than it is as an immediately applicable design method.

MAGID is a direct extension of the model of a structural design process developed in the fourth chapter. The concept of information as a medium to be acquired, manipulated and generated in a design process directly implies the development of an information system to interact with the designer. The relationship between conceptual and evaluative operators was used in the formulation of information structures that would aid the designer in the construction of a design space. The uncoupling and recoupling of that design space in order to identify acceptable design kernels resulted from the concept of isolating relevant information in α-level-sets. Finally, the concept of decision making in a fuzzy environment was related to both the previous use of fuzziness in defining sets of information and the concept of decision making expressed in the decision operator. In these ways, the existence of the previously developed design philosophy aided directly in the development of a method that may well be capable of stimulating and expanding a designer's ability to identify design kernels.

The presentation of the Parkmerced problem has illustrated some of the strengths and weaknesses of MAGID. What is important about this method is the interaction with the designer in a manner that stimulates his ability to identify design kernels; thus, MAGID would be considered equally successful whether an acceptable kernel were identified at the tenth level or even before the tenth level was reached. MAGID's major implications for the designer are the expansion of a problem oriented information base, the externalization of the information affecting the identification of design kernels, the construction of a design space using the conditional relationship between the performance and design spaces, and the availability of a tool that allows for decision making in a fuzzy environment (the uncoupling and recoupling of the design space).

Because MAGID is in its infancy, problems like Parkmerced help suggest directions for further development. The difficulty encountered in establishing well defined and meaningful dimensions in both the performance and design spaces indicates certain basic problems with MAGID. A possible solution to these problems would be to redevelop the method into a more iterative process, such that different dimensions might easily be experimented with and their relative importance established within the process. The actual nature of the sequence of requests used in the method and the related indices are only developmental. What remains important is that the indexing be done on a functional basis with the possible supplementation of subject oriented indices. The inclusion at level nine of an ability to eliminate logically unacceptable subspace vectors is a critical step in turning MAGID into a truly functional method.

MAGID was developed in the context of a single user, and the sample problems were also run in this manner; however, the concept of relevant subspaces strongly suggests a potential use of this type of method in team design problems. The ability to uncouple a problem into subspaces for evaluation relative to diverse criteria and then recoupling these subspaces for developing a solution could be adapted as a format for group interaction.

As was previously mentioned, the development of MAGID was limited to batch mode processing using a large scientifically oriented computer. The method is essentially a data handling process that may well be carried out within considerably smaller and less complex systems. The extension of MAGID into a real time system would allow for a more dynamic interaction between the designer and the method. It would seem that in this type of situation a method like MAGID could really become a true partner for the structural designer.

7.3 The Model of a Structural Design Process

MAGID is a product of the model of a structural design process developed in this dissertation and in that sense is an expression of that model. Likewise, the model is in itself an expression of a philosophy of what is meant by structural design. Philosophy is a study of the truths and principles underlying knowledge and reality. There are many ways to interpret design and therefore to develop related philosophies. The approach chosen in this research has been to develop a descriptive model of the structural design process based on the postulate that design could be interpreted as the acquisition, manipulation and generation of information.

This model of a structural design process is an initial attempt to develop a philosophy of structural design. It is not intended to be a final answer to the question of what is design but is, rather, only a start in that direction. No human process is as simple and unencumbered as models seem to indicate, but it is within the context of such models that one can begin to comprehend those processes being modeled. The use of a fuzzy concept of information was intended as a partial response to the critism that no process as complex as design can be dissected and discretized without some severe loss of reality. For a model of the design process to act as a philosophy, it must provide not only a means of interpreting past and present observations but also a means of predicting future developments.

The model of a structural design process is rooted in the psychological and historical perspective of the first chapter and, in that sense, is consistant with the concepts expressed in that chapter. Such diverse ideas as structural optimization and the conceptual pattern language of Alexander presented in the second chapter can be interpreted as information processes occurring at different metrics within the design process of the model. The concepts of engineering design associated with Asimow are similar to the model, especially in the differentiation of a vertical and horizontal structure, even though there are discrepancies in such ideas as the analysis of information and implementation. The model also closely parallels the descriptive model of engineering design developed by Ramström and Rhenman. In this way the model presented in Chapter Four is consistant with the past and present; however, it is strongly suggested that any continuation of this type of research be accompanied by the development of comprehensive case studies to provide a basis for a more realistic appraisal of contemporary design processes. The potential extension of such design models has been illustrated through the development of MAGID in the fifth and sixth chapters. It is through such related methods that a generalized model of structural design can have its most direct influence on the future evolution of related design processes.

The development of this model of a structural design process emphasizes the importance of the designer's thorough comprehension of structural behavior, for understanding behavior is not merely reflected in the process of evaluation but is, also the essence of synthesis. The conceptual operator is to be interpreted as the mapping by the designer of a vector of desired attributes into a physically descriptive design space. The conceptual operator can then be viewed as the inverse of the evaluative function that maps the design vector into a performance space descriptive of desired attributes. A possible implication of this is that a designer cannot conceive of a structure for which he does not have a fundamental sense of its potential behavior. This fundamental sense of structural behavior not only includes a theoretical understanding of physical phenomena but also a sense of physical reality. Thus, the acquistion of this sense of reality, experience, plays an important role in the final maturation of a structural designer.

7.4 Directions for Further Research and Development

As was mentioned previously, it is necessary for the future development of models such as the one developed in this dissertation, to make use of actual case studies of design processes. The need to develop such studies, then, is a logical extension of the research presented in this dissertation. These studies are not intended to be just a compilation of facts about a given structure, but an in-depth examination of the processes followed by a reasonable cross-section of practicing designers. It will be necessary not just to observe designers at work but also probe the designers as they work to determine why they have carried out their tasks in the manner observed. Design is not to be simply interpreted as the initial identification of a design kernel but as all processes leading through to the last details of a design; therefore, the case study of a design process must be followed from the client to the implementor and through to useage. The assessment of the actual structure as finally implemented, the observation of the real context, and the recording of the actual behavior of the structure can help in the exploration of the influence, both absolute and relative, of different design processes and the related design decisions. Case studies of this type will require a great deal of cooperation between the participating designers or design firms and researchers, but it is from such studies that a truly thorough comprehension of structural design may finally begin to emerge.

Another possible approach to developing a more fundamental understanding of structural design can be found in the simulation studies associated with artifical intelligence, information processing. The purpose of this type of study is not to develop an automated process but, rather, to simulate an existing process. The basic format for such a study is to pose a simple problem or isolate a small aspect of a larger problem that can then be solved by experimental subjects who are structural designers. The designers are to be observed and queried as to the reasons for the observed actions. While this aspect of a simulation study is similar to the gathering of case studies previously discussed, the use of the studies is different. The idea here is to develop a series of protocols for the design processes for different designers under a controlled situation and in the solution of a problem that is simple enough to be reasonably modeled. Using these protocols as a basis, the rules presented in Section 1.2.2 can be applied in the development of computer programs that use basic information processes in the simulation of the previously observed behavior in solving structural problems. The simulation program should then be capable of duplicating the observed behavior and, in addition, allow the researcher to change the input to study the effect of certain parameters in the design situation.

MAGID was intended only as one possible method that can be derived from the model of a structural design process previously developed. Given other simplifications of the model, it is possible for different information processes to be developed that will be capable also of aiding the structural designer. General information storage and retrieval systems could be developed based upon retrieval rules directly related to those aspects of the process that are involved. Of greater interest, though, is the concept of information control systems, methods like MAGID, that would be far more powerful if the designer could be aided in the discrimination of information through the incorporation of heuristic rules. These heuristic rules might well be one of the potential products of the previously mentioned simulation studies.

Through simple examination of the literature in such areas as design methodology or information processing, it is possible to find the seeds for many new design tools capable of aiding the structural designer. The majority of such direct approaches for aiding the designer attempt to make use of the vast potential of the digital computer. These tools are not just more sophisticated evaluative techniques, but concepts that are operative in all aspects of problem solving processes. These are design tools that can help the designer make a more optimal use of the unique human quality of judgment. The full utilization of the tools available to the structural engineer, for example the computer, may well depend upon how well the design process is understood.

REFERENCES

1. Dewey, J., How We Think, D. C. Heath & Co., Publishers, Boston, 1910.
2. Rossman, J., The Psychology of the Inventor, Inventors Publishing Co., Washington, 1931.
3. Wallas, G., The Art of Thought, Jonathan Cape, London, 1926.
4. Johnson, D. M., The Psychology of Thought and Judgment, Harper & Brothers, New York, 1955.
5. Vinacke, W. E., The Psychology of Thinking, McGraw-Hill Book Company, New York, 1952.
6. Duncker, K., "On Problem-Solving," L. S. Lees, Trans., Psychological Monographs, Vol. 58, No. 5, originally published in Berlin, 1935.
7. Wertheimer, M., Productive Thinking, Harper & Brothers Publishers, New York, 1945.
8. Humphrey, G., Thinking - An Introduction to Its Experimental Psychology, John Wiley & Sons, New York, 1951.
9. Wiener, N., Cybernetics - or Control and Communication in the Animal and the Machine, John Wiley & Sons, Inc., New York, .
10. Feigenbaum, E. A. and Feldman, J., eds., Computers and Thought, McGraw-Hill Book Co., New York, 1963.
11. Newell, A., Shaw, J. G., and Simon, H. A., "Elements of a Theory of Human Problem Solving," Psychological Review, Vol. 65, No. 3, 1958, pp. 151-166.
12. Newell, A., and Simon, H. A., "GPS, A Program that Simulates Human Thought," Computers and Thought, E. A. Feigenbaum, and J. Feldman, eds., McGraw-Hill Book Co., New York, 1963, pp. 279-293.
13. Guilford, J. P., The Nature of Human Intelligence, McGraw-Hill Book Co., New York, 1967.
14. Wong, A. K. C., and Bugliarello, G., "Artificial Intelligence in Continuum Mechanics," Journal of the Engineering Mechanics Division, ASCE, Vol. 98, No. EM6, Dec. 1970, pp. 1239-1265.
15. Simon, H., Administrative Behavior - A Study of Decision-Making Processes in Administrative Organization, The Free Press, New York, 1945.

16. McDonald, H., "Origin of the Word 'Engineer'," ASCE Transactions (Vol. LXXVII, p. 1737), from speech given June 2, 1914, reprinted, The Civil Engineer: His Origins, ASCE Historical Publication No. 1, 1970

17. Hopkins, H. J., A Span of Bridges, David & Charles: Newton Abbot, 1970.

18. Crombie, A. C., Medieval and Early Modern Science Vol. II - Science in the Later Middle Ages and Early Modern Times: XIII-XVII Centuries, Doubleday & Co., Inc., Garden City, New York., 1959.

19. Hollister, S. C., "Three Centuries of Structural Analysis," Civil Engineering, Vol. 8, No. 12, Dec. 1938, pp. 822-825.

20. Timoshenko, Stephen P., History of Strength of Materials, McGraw-Hill, New York, 1953.

21. Straub, H., A History of Civil Engineering, The M.I.T. Press, Cambridge, Mass., 1964.

22. Condit, C. W., American Building, The University of Chicago Press, Chicago, 1968.

23. Esherick, J., "Problems of the Design of a Design System," Conference on Design Methods, J. C. Jones and D. G. Thornley, eds., The MacMillan Company, New York, 1963.

24. Churchman, C. W., Ackoff, R. L., and Arnoff, E. L., Introduction to Operations Research, John Wiley & Sons, Inc., New York, 1957.

25. Ackoff, R. L., "The Development of Operations Research as a Science," Operations Research, 4, 265f, 1956.

26. Ackoff, R. L., Scientific Method, Optimizing Applied Research Decisions, John Wiley & Sons, New York, 1962.

27. Dantiz, G. B., Linear Programming and Extensions, Princeton University Press, Princeton, New Jersey, 1963.

28. Hadley, G., Nonlinear and Dynamic Programming, Addison-Wesley Publishing Co., Inc., Reading Mass., 1964.

29. DeNeufville, R., and Stafford, J. H., Systems Analysis for Engineers and Managers, McGraw-Hill Book Co., 1971.

30. Kaufman, A., The Science of Decision-Making, McGraw-Hill Book Co., New York, 1969.

31. Bross, I. D. J., Design For Decision, The Free Press, New York, 1965.

32. Jones, J. C. and Thornley, D. G., eds., Conference on Design Methods, The MacMillan Co., New York, 1963.

33. Gregory, S. A., ed., The Design Method, Plenum Press, New York, 1966.

34. Broadbent, G., and Ward, A., eds., Design Methods in Architecture, Lund Humphries, London, 1969.

35. Moore, G. T., ed., Emerging Methods in Environmental Design and Planning, The M.I.T. Press, Cambridge, Mass., 1968.

36. Jones, J. C., Design Methods - Seeds of Human Futures, Wiley-Interscience, London, 1970.

37. Eastman, C. W., "On the Analysis of Intuitive Design Processes," Emerging Methods In Environmental Design and Planning, G. T. Moore, ed., The M.I.T. Press, Cambridge, Mass., 1968.

38. Jones, J. C., "The State of the Art in Design Methods," Emerging Methods in Environmental Design and Planning, G. T. Moore, ed., The M.I.T. Press, Cambridge, Mass., 1968.

39. Alexander, C., Notes on the Synthesis of Form, Haravrd University Press, Cambridge, Mass., 1964.

40. Stewart, D. V., "Organizing the Design of Systems by Partitioning and Tearing," Nuclear Energy Division, General Electric Co., San Jose, California, 1969-70.

41. Stewart, D. V., "Partitioning and Tearing of Equations," J. SIAM Numerical Analysis, Ser. B., Vol. 9, No. 2, 1965.

42. Archer, L. B., "The Structure of the Design Process," Design Methods in Architecture, G. Broadbent and A. Ward, eds., Lund Humphries, London, 1969.

43. Eder, W. E., "Definitions and Methodologies," The Design Method, S. A. Gregory, ed., Plenum Press, New York, 1966.

44. Best, G., "Method and Intention in Architectural Design," Design Methods in Architecture, G. Broadbent and A. Ward, eds., Lund Humphries, London, 1969.

45. Archer, L. B., "An Overview of the Structure of the Design Process," Emerging Methods in Environmental Design and Planning, G. T. Moore, ed., The M.I.T. Press, Cambridge, Mass., 1968.

46. Zwicky, F., and Wilson, A. G., eds., New Methods of Thought and Procedure, Springer-Verlag, New York, 1967.

47. Norris, K. W., "The Morphological Approach to Engineering Design," Conference on Design Methods, J. G. Jones and D. G. Thornley, eds., The MacMillan Co., New York, 1963.

48. Alexander, C., Ishikawa, S., and Silverstein, M., A Pattern Language Which Generates Multi-Service Centers, Center for Environmental Structure, Berkeley, 1968.

49. Alexander, C., Hirshen, S., Ishikawa, S., Coffin, C., and Angel, S., Houses Generated by Patterns, Center for Environmental Structure, Berkeley, 1969.

50. Alexander, C., and Poyner, B., "The Atoms of Environmental Structure," Emerging Methods in Environmental Design and Planning, G. T. Moore, ed., The M.I.T. Press, Cambridge, Mass., 1968.

51. Archer, L. B., "Systematic Method for Designers," Collection of 7 articles from Design, 1963-64, booklet printed 1965, England.

52. Alexander, C., Interviewed in, "The State of the Art in Design Methodology," DMG Newsletter - Design Methods Group, Vol. 5, No. 3, pp. 3-7, March 1971.

53. Proceedings Second Conference on Engineering Design Education, University of California, Los Angeles, September, 5, 6, 7, 1962.

54. Buhl, H. R., Creative Engineering Design, The Iowa State University Press, Ames, Iowa, 1960

55. Asimow, M., Introduction to Design, Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 1962.

56. Alger, J. R. M., and Hays, C. V., Creative Synthesis in Design, Prentice-Hall, Inc., New York, 1964.

57. Woodson, Thomas T., Introduction to Engineering Design, McGraw-Hill, New York, 1966.

58. Ramström, D., and Rhenman, "A Method of Describing the Development of an Engineering Project," IEEE Transactions on Engineering Management, Vol. EM-12, No. 3, September 1965, pp. 79-86.

59. Teague, L. C., and Hershdorfer, A. M., "Build - An Integrated System for Building Design," Journal of the Structural Division, ASCE, Vol. 94, No. ST4, Proc. Paper 5901, April 1968, pp. 983-994.

60. McCormick, J. K., and Weidlinger, P., "The STORE Project (The Structures Oriented Exchange)," Office of Naval Research International Symposium on Numerical and Computer Methods in Structural Mechanics, Urbana, Illinois, September 8-10, 1971.

61. Wright, R. N., Boyer, L. T., and Melin, J. W., "Constraint Processing in Design," Fifth Conference on Electronic Computation, American Society of Civil Engineers, New York, 1970, pp. 801-827.

62. Goel, S. K., and Fenves, S. J., Computer-Aided Processing of Structural Design Specifications, Civil Engineering Studies Structural Research Series No. 348, University of Illinois, Urbana, Nov. 1969.

63. Fenves, S. J., Gaylord, E. H., and Goel, S. K., Decision Table Formulation of the 1969 AISC Specifications, Civil Engng. Studies Structural Research Series No. 347, University of Illinois, Urbana, Aug. 1969.

64. Spillers, W. R., and Friedland, L., "On Adaptive Strucutral Design," Journal of the Structural Division, ASCE, Vol. 98, No. ST10, Proc. Paper 9245, Oct. 1972, pp. 2155-2163.

65. Shannon, C. E., "A Mathematical Theory of Communication," Bell System Technical Journal, Vol. 27, July and October, 1948.

66. Manheim, M. L., Hierarchical Structure: A Model of Design and Planning Processes, The M.I.T. Press, Cambridge, Mass., 1966.

67. Stevens, S. S., "Measurement, Psychophysics and Utility," Measurement: Definitions and Theories, C. W. Churchman and P. Ratcosh, eds., John Wiley & Sons, Inc., New York, 1959, pp. 18-63.

68. Zadeh, L. A., "Fuzzy Sets," Information and Control, Vol. 8, No. 3, June 1965.

69. Zadeh, L. A., Toward A Theory of Fuzzy Systems, Report No. ERL-69-2, Electronics Research Laboratory, College of Engineering, University of California, Berkeley, 1969.

70. Ashby, W. R., An Introduction to Cybernetics, University Paperbacks, London, 1964.

71. Zadeh, L. A., "Similarity Relations and Fuzzy Ordering," Information Sciences, 3, pp. 177-200, 1971.

72. Fenves, S. J., "Design Philosophy of Large Interactive Systems," Office of Naval Research International Symposium on Numerical and Computer Methods in Structural Mechanics, Urbana, Illinois, September 8-10, 1971

73. Churchman, C. W., The Systems Approach, Dell Publishing Co., New York, 1968.

74. CIB - Masters Lists for Structuring Documents Relating to Buildings, Building Elements, Components, Materials and Services, International Council for Building Research, Rotterdam, 1972.

75. Manstone, R. J., Bianco, L. G., and Harrison, H. W., "Performance Parameters and Performance Specifications in Architectural Design," Building Science, Vol. 3, pp. 125-133, 1969.

76. Specification for the Design, Fabrication and Erection of Structural Steel Buildings, American Institute for Steel Construction, New York, February, 12, 1969.

77. Allen, D. E., "Limit State Design," American Society of Civil Engineers National Structural Engineers Meeting, Portland, Oregon, April 6, 1970, Preprint 1215.

78. Principles of Modern Building, Ministry of Technology, Building Research Station, Her Majesty's Stationary Office, London, 1959.

79. Simon, H. A., The Sciences of the Artificial, The M.I.T. Press, Cambridge, Mass., 1969.

80. Turkstra, C. J., Theory of Structural Design Decisions, University of Waterloo, 1970.

81. Horman, A. M., "Machine-Aided Value Judgments Using Fuzzy-Set Techniques," SP-3590 System Development Corporation, Santa Monica, California, 1971.

82. Lefkovitz, D., File Structures for On-Line Systems, Spartan Books, New York, 1969.

83. Bellman, R. E., and Zadeh, L. A., Decision-Making in a Fuzzy Environment, Report No. ERL-69-8, Nov. 1969.

84. Cooper, W. S., "The Mathematical Structure of Reference Retrieval Systems," Notes for Librarianship 241, Theoretical Problems of Information Transfer and Retrieval, University of California, Berkeley.

85. Gould, J. J., "Multistory Buildings Designed To Resist Earthquakes," Journal of the American Concrete Institute, Vol. 23, No. 1, September, 1951.

86. Uniform Building Code, International Conference of Building Officals, 1970 Edition, Vol. I, Pasadena, California, 1970.

87. Recommended Lateral Force Requirements and Commentary, Seismology Committee, Structural Engineers Association of California, 1968 edition and subsequent Appendices, San Francisco.

88. Bonilla, M. G., Bedrock - Surface Map of San Francisco South Quadrangle, California, United States Geological Survey, 1964.

89. Bonilla, M. G., Preliminary Geologic Map of the San Francisco South Quadrangle and Part of Hunters Point Quadrangle, California, MF-311, United States Geological Survey, 1971.

90. Bonilla, M. G., Schlocker, J. and Radbruch, D. H., Geology of the San Francisco North Quadrangle, California, Map I-272, United States Geologic Survey, 1958.

91. Bresler, B., "Behavior of Structural Element, A Review," Building Practices For Disaster Mitigation, Building Science Series 46, National Bureau of Standards, U. S. Government Printing Office, Washington, D. C., 1973.

92. Nelson, H. E. and Daley, A., "Building Integrity," Proceedings Reconvened International Conference of Fire Safety in High-Rise Buildings, General Services Administration, Public Buildings Service, October 5, 1971.

93. Bizri, H., Structural Capacity of Reinforced Concrete Columns Subjected to Fire Induced Thermal Gradients, SESM Report 73-1, Dept. of Civil Engineering, University of California, Berkeley, 1973.

94. Lie, T. T. and Allen, D. E., Calculation of the Fire Resistance of Reinforced Concrete Columns, Technical Paper No. 378, Division of Building Research, National Research Council of Canada, Ottawa, 1972.

95. Magnusson, S., "Probabilistic Analysis of Fire Safety," paper presented at the 1973 National Structural Engineers Meeting, American Society of Civil Engineers, San Francisco, April 9-13, 1973.

96. Pettersson, O., "Fire Engineering Design of Tall Buildings," TC-8, Summary, International Conference on Planning and Design of Tall Buildings, Leigh University, Bethlehem, Penn., August 21-26, 1972.

97. Kawagoe, K., "Systems Design for Fire Safety," TC-8, International Conference on Planning and Design of Tall Buildings, Leigh University, Bethlehem, Penn., August 21-26, 1972.

98. Ruiz, P. and Penzien, J., Probabilistic Study of the Behavior of Structures During Earthquakes, EERC Report 69-3, College of Engineering, University of California, Berkeley, 1969.

99. Seed, H. B., "Dynamic Characteristics of Soil-Structure Systems," TC-6, International Conference on Planning and Design of Tall Buildings, Leigh University, Bethlehem, Penn., August 21-26, 1972.

100. Clough, R. W., Benuska, K. L. and Wilson, E. L., "Inelastic Earthquake Response of Tall Buildings," Proceedings, Third Conference on Earthquake Engineering, Wellington, New Zealand, 1965.

101. Anderson, J. C. and Bertero, V. V., Seismic Behavior of Multistory Frames Designed By Different Philosophies, EERC Report 69-11, College of Engineering, University of California, Berkeley, 1969.

102. Whitman, R. V., "Dynamic Soil-Structure Interaction," TC-6, D7, International Conference on Planning and Design of Tall Buildings, Leigh University, Bethlehem, Penn., August 21-26, 1972.