94-22

INFERENTIAL DESIGN THEORY: A CONCEPTUAL OUTLINE

T. ARCISZEWSKI
Department of Systems Engineering Department
School of Information Technology and Engineering
George Mason University
Fairfax VA 22030 USA

AND

R. S. MICHALSKI
Center for Artificial Intelligence Research
School of Information Technology and Engineering
George Mason University
Fairfax VA 22030 USA

Abstract. This paper presents initial ideas toward a new design theory based on the Inferential Theory of Learning, recently developed in artificial intelligence. The theory views engineering design as a process of transforming the initial design specification and design background knowledge into the desired design. This process is performed using certain knowledge operators called 'knowledge transmutations.' Nine basic tenets of the theory are provided, and a system of 22 design knowledge transmutations is proposed. Individual transmutations are defined and explained using examples from the area of conceptual design of wind bracings in steel skeleton structures of tall buildings. The paper also contains initial conclusions and a discussion of future research.

1. Introduction

At present, research on design is concentrated on the development of various monostrategy conceptual design methods, i.e., methods based on a single problem solving algorithm (strategy) implemented in a single computational mechanism. Such methods include, for example, a method of solving conceptual design problems through the elimination of contradictions between various features of the system being designed, as proposed by Altschuller (1988), and Fey and Vertkin (1993), a method based on the random generation of concepts from a given design space, as proposed by Arciszewski (1987), or a constraint search through the design space, as proposed by Cahn and Paulson (1987), Hajdo and Arciszewski (1990), Nadel (1990), and Flemming et al. (1992). Shupe et al. (1987) proposed decision-based design

J.S. Gero and F. Sudweeks (eds.), Artificial Intelligence in Design '94, 295-308.

in which decisions are made based on the theory of living systems (Miller 1965).

In design research, a great deal of effort is devoted to applying the principles of artificial intelligence to design in order to develop new conceptual design methods. For example, Maher (1985, 1986) and Sriram (1986) investigated a knowledge-based approach to conceptual design in which several abstract levels of the design problem are distinguished; a subsolution is produced on each level which satisfies all constraints (decision rules) associated with a given level. Maher (1987) proposed conceptual design based on the retrieval of past experience in the form of incomplete descriptions of past designs which were produced in similar design situations. These solutions are then modified and augmented to produce new design concepts for a given situation. This method was later developed into a method for using analogical reasoning in conceptual design (Zhao and Maher, 1988), and expanded by Gero, Maher and Zhang (1988) into a design method based on the use of prototypes understood as conceptual schemata, or concepts, i.e., generalizations of groupings of elements in a design domain. Arciszewski and Ziarko (1986) proposed a conceptual design method based on the use of a learning system both for knowledge acquisition and for the evaluation of randomly generated concepts. Dyer et al. (1986) investigated the use of machine learning in conceptual design; in his design system, called Edison, concepts are produced from design examples using generalization, analogy, and mutation.

Since human problem solving, including design, is clearly multistrategy, the development of a general conceptual and computational framework for multistrategy conceptual design, i.e., a design theory, is important for at least two reasons: (i) It will improve our understanding of conceptual design, and (ii) it will stimulate the development of a class of multistrategy conceptual design methods for which computational mechanisms could be easily identified and implemented in various design support computer tools. Therefore, in this paper we proposed an Inferential Design Theory (IDT). This theory provides a framework for the integration of a large class of available conceptual design methods in a problem-dependent way. By a problem-dependent way we mean an integration in which a design strategy, or a combination of strategies, is automatically adapted to different design situations. At present, there is no generally accepted scientific design theory available, although the need for such a theory was identified some time ago (Dixon 1988) and several attempts to develop have been taken, for example by Asimov (1965), Miller (1974), Hubka (1982), Pahl and Beitz (1984), Nadler (1985), by Tomiyama and Yoshikawa (1987), and by Suh (1990).

Inferential Design Theory (IDT) is based on the Inferential Theory of Learning, recently proposed by Michalski (1993). IDT consists of a system of tenets and a system of design knowledge transmutations which provide a conceptual, methodological, and computational framework for conceptual design. In the paper, the tenets of IDT are given, the proposed individual design knowledge transmutations are discussed with examples, and future research directions are proposed.

2. Basic Tenets of the Inferential Design Theory

The proposed theory is based on the following tenets dealing with major aspects of engineering design:

2.1. ENGINEERING DESIGN PROCESS

Engineering design is a multistage process which starts when a need for a new engineering system is realized, and ends when an engineering design is produced.

2.2. ENGINEERING DESIGN

An engineering design is a complete description of a future engineering system. This description has two components: (i) an abstract description and (ii) a detailed description. The abstract description, usually called "a concept of an engineering system," is produced using nominal attributes, while the detailed description is produced using numerical attributes.

2.3. ENGINEERING DESIGN PROCESS STAGES

Two major stages in the engineering design process are distinguished: (i) conceptual design, and (ii) detailed design. The objective of conceptual design is to analyze needs and background knowledge and to produce a concept, or a class of concepts, of a future engineering system. The objective of the detailed design is to produce a detailed description, or descriptions, for the concept or concepts produced in the conceptual design stage.

2.4. CONCEPTUAL DESIGN STAGES

Six conceptual design stages have been distinguished, as shown in Figure 1:

1. Analysis of needs
2. Design task formulation
    - 2.1. Identification of design goals
    - 2.2. Building a representation space
    - 2.3. Formalization of design background knowledge

3. Concept generation
4. Concept evaluation
5. Concept selection
6. Knowledge acquisition

2.5. APPLICABILITY OF INFERENTIAL DESIGN THEORY

The proposed theory is formulated for the conceptual design stage No. 3, Concept Generation.

2.6. CONCEPTUAL DESIGN PARADIGM

Concept generation in conceptual design is a goal-oriented process in which both input and background knowledge are used and transformed in order to meet design goals. Input knowledge (P) is any information provided with the design goal, and it may be in the form of facts, procedures, or design examples. Background knowledge (BK) is general engineering/design knowledge which is relevant to a given design problem and which is available to the designing system, living or artificial. Relevant knowledge is knowledge which is useful at any stage of the design process.

Thus concept generation is a search through a design space as defined by the knowledge representation space. It can employ any type of deduction, analogy, or induction. Consequently, the designing systems, both living and artificial, can be viewed as systems which transform a given design task into a concept or class of concepts, and into new design knowledge through a transformation process based on various forms or combinations of inference, as shown in Figure 2. This new design knowledge can be interpreted as design experience, in the form of both factual and procedural (methodological) knowledge gained in a given concept generation process.

The designing system must be able to perform inferences and have a memory that stores the design task and "useful" results of inferences in the form of new knowledge. Thus the conceptual design process can be described by a "conceptual design equation":

conceptual design = inferencing + memorizing

Inferencing is understood here as any type of reasoning or knowledge transformation. *Memorizing* means storing knowledge produced as the result of inferencing.

2.7. THE NATURE OF CONCEPT GENERATION

Concept generation is a multistrategy process in which a combination of various strategies is used.

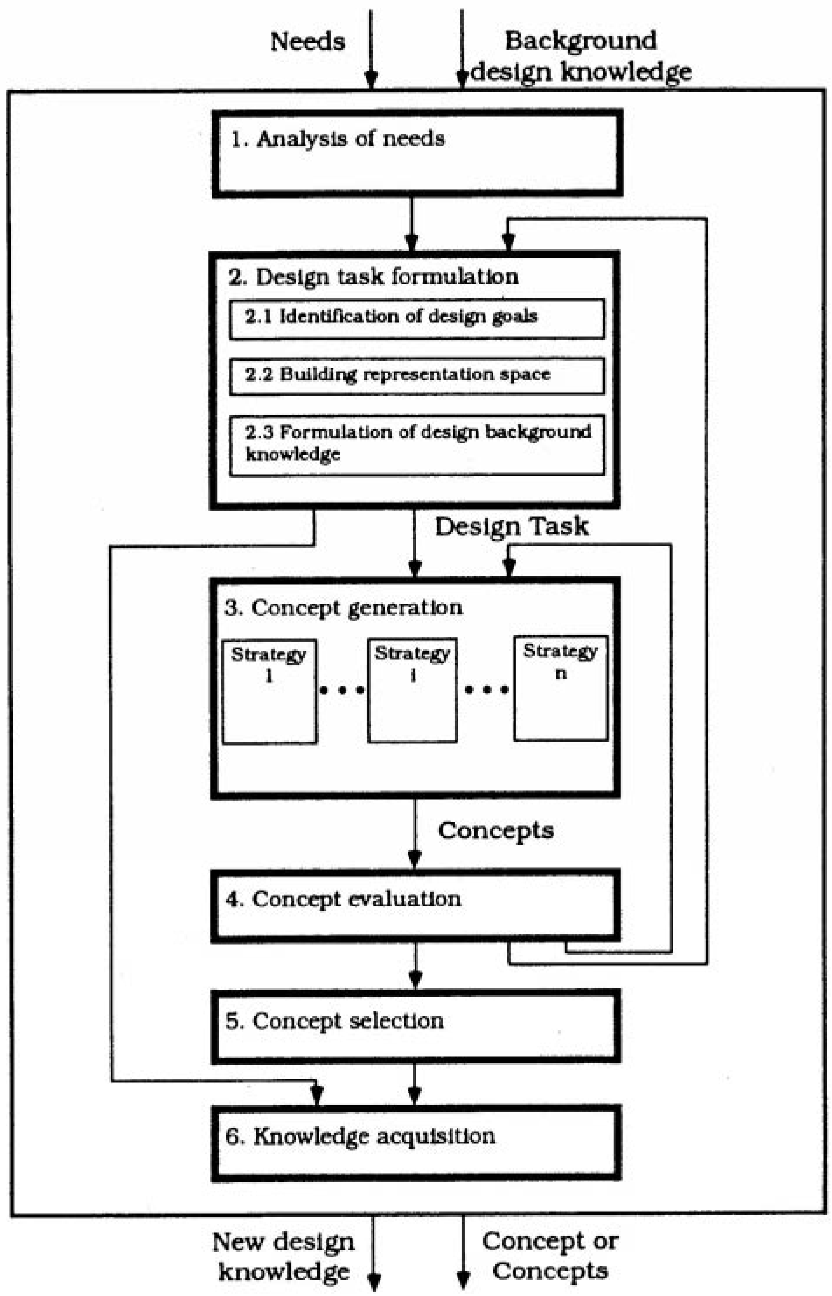

Figure 1. Conceptual design process.

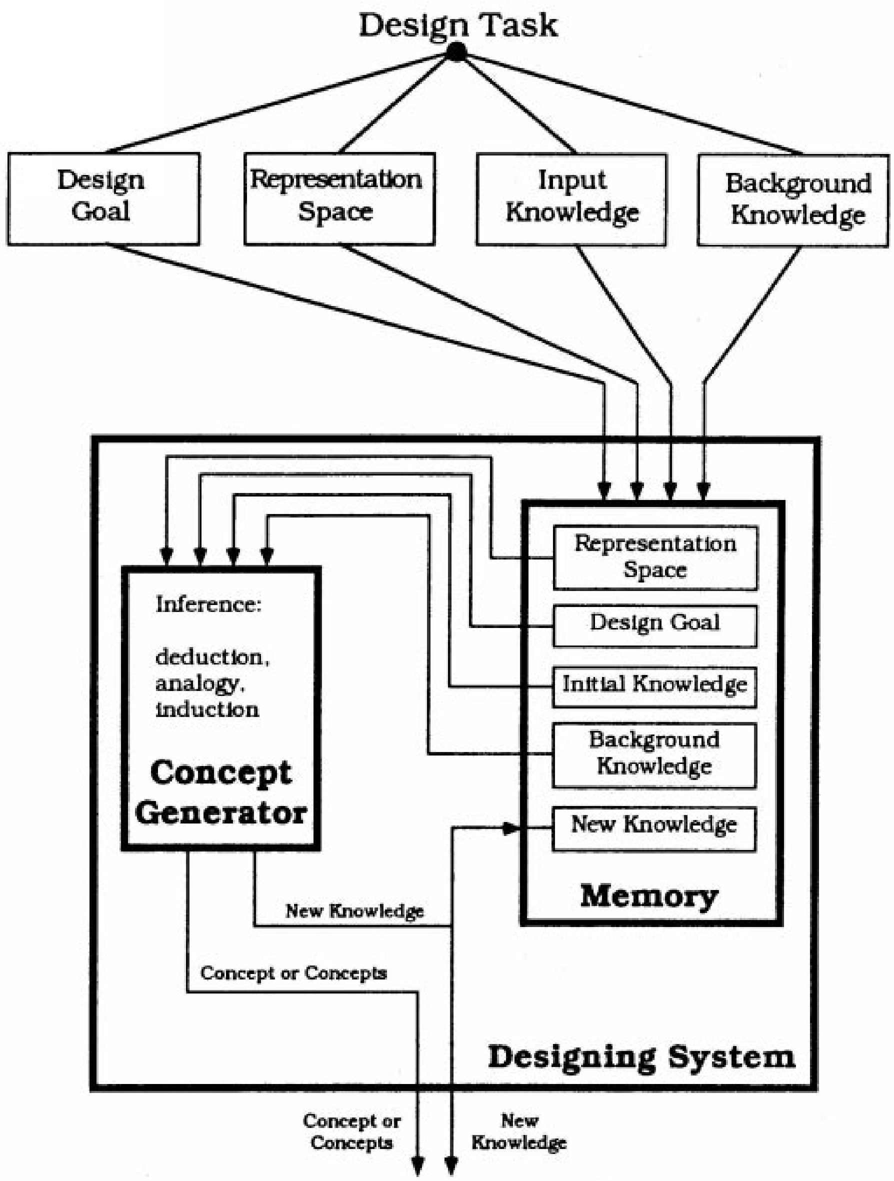

Figure 2. Designing system.

2.8. DESIGN KNOWLEDGE TRANSMUTATIONS

Design knowledge transmutation is a conceptually simple (homogeneous) elementary high-level knowledge transformation that occurs in concept generation. In this transformation a piece of knowledge is derived from a given input and from background design knowledge. In computational terms, a design transmutation is an operator on design knowledge, a search operator that can employ any type of inference. A class of design knowledge transmutations can be identified, and individual types of transmutations can be used for concept generation purposes in a goal-oriented way.

2.9. DESIGN KNOWLEDGE TRANSMUTATION AS INFERENCE

All design knowledge transmutations are based on inference, reasoning, and can be classified according to the validity of the knowledge produced by them. Therefore these transmutations can be described by the fundamental equation for inference (Michalski 1993), which is an entailment:

$$P \cup BK = C$$

where:
- P — a set of statements, input knowledge, called the premise
- BK — a set of statements which are the background knowledge
- C — a set of statements called the consequent

In the case of deductive inference, the consequent C is derived, given P and BK, and the inference is truth-preserving. An inductive inference is hypothesizing premise P, given C and BK, and it is falsity-preserving. Therefore all design knowledge transmutations can be divided into deductive (truth-preserving) and inductive (falsity-preserving) inferences.

Another division of transmutations is proposed according to the nature of the entailment =. This can be '**strong**' when C is a deterministic, or valid, consequence of P and BK, and '**weak**' when C is only a probabilistic or plausible consequence of P and BK. Therefore, when the entailment is "**strong**" the inference is conclusive, while in the case of a '**weak**' entailment, the inference is contingent. When both the truth preservation and the nature of the entailment are considered, four major types of inference can be distinguished, as shown in Table 1.

TABLE 1. Basic types of inference.

		TRUTH PRESERVATION	FALSITY PRESERVATION
ENTAILMENT STRENGTH	STRONG	Conclusive deduction	Conclusive induction
	WEAK	Contingent deduction	Contingent induction

TABLE 2. Attributes and their feasible values.

ATTRIBUTES	FEASIBLE ATTRIBUTE VALUES				
	1	2	3	4	5
1. Number of Stories	6	12	18	24	30
2. Bay Length	20	30			
3. Wind Intensity Factor	1.07	1.11			
4. Joints	Rigid	Hinged	Mixed		
5. Number of Braced Bays	1	2	3		
6. Number of Vertical Trusses	0	1	2	3	
7. Number of Horizontal Trusses	0	1	2	3	
8. Unit Steel Weight	Low	Medium	High	Infeasible	

3. Design Knowledge Transmutations: Initial Examples

Eleven pairs of design knowledge transmutations are proposed. In each pair, two opposite transmutations are given. The transmutations are explained using examples from the area of conceptual design of wind bracings in steel skeleton structures of tall buildings. Concepts of wind bracings are considered from three classes of wind bracings, including rigid frames, braced frames, and trusses. For these bracings and for a three-bay tall building, the representation space is given in Table 2 in accordance with Arciszewski et al. (in print). The first three attributes, Number of Stories, Bay

Length, and Wind Intensity Factor, describe the design case considered. Attributes No. 4 through No. 7 describe the structural system of wind bracing itself, while the last attribute, No. 8, Unit Steel Weight, identifies the nominal value of the relative unit weight of the steel structural system described by attributes 1 through 7. This relative unit steel weight is determined from all normalized unit weights of various types of wind bracings of the same height designed under identical conditions. In this way, Unit Steel Weight represents the goodness of a given wind bracing type from the point of view of weight.

3.1. GENERALIZATION/SPECIALIZATION

The generalization transmutation extends the reference set of the input, i.e., it produces a description that characterizes a larger reference set than the input. This is usually done by removing one or more attributes which describe the initial reference set. For example, the input is in the form of two concepts which represent two types of rigid frames: a one-bay rigid frame and a three-bay rigid frame. Attribute No. 5, Number of Braced Bays, is removed and the input "two types of rigid frames" is generalized into "a class of rigid frames."

The specialization transmutation narrows the reference set of the input. This is usually done by adding one or more attributes. For example, when the input is "a class of truss bracings" the addition of attribute No. 7, Number of Horizontal Trusses equal to 0 specializes the input into "a class of truss bracings without horizontal trusses."

3.2. ABSTRACTION/CONCRETION

The abstraction transmutation reduces the amount of detail in a description of the given input. This is usually done by converting attributes from numerical to nominal, or by changing their values from numerical to linguistic. For example, when attribute No. 2, Bay Length, is considered, its numerical values, 'twenty' and 'thirty feet' respectively, can be converted into linguistic values 'short' and 'long'.

The opposite transmutation is concretion. In this case, additional details are added to the input through the conversion of attributes from nominal into numerical. For example, attribute No. 8, Unit Steel Weight, which is nominal, would be converted to a numerical one, and instead of its nominal values 'low,' 'medium,' 'high,' or 'infeasible,' the actual unit weight values would be used, for instance '22.'

3.3. SIMILIZATION/DISSIMILIZATION

The similization transmutation produces new knowledge about a given reference set using available knowledge about a similar reference set. For example, when the class of rigid frames is considered, the relative effects of using horizontal trusses are estimated using knowledge about the class of truss bracings with horizontal trusses.

In the case of dissimilization, the lack of similarity between two reference sets is used to produce knowledge about one of them, using available knowledge about the other. For example, wind bracings in the form of reinforced concrete shear walls are not similar to truss bracings. Therefore, if we know that the unit weight of the structural system is relatively small in the case of truss bracings, we can infer that the unit weight for reinforced concrete walls will not be small.

3.4. ASSOCIATION/DISASSOCIATION

The association transmutation determines a dependency between entities based on input and/or background knowledge. For example, the taxonomy of wind bracings, which is part of the background knowledge, is used to classify a given type of wind bracing and to associate it, for instance, with the class of truss bracings.

Disassociation identifies the lack of dependency between various entities, also using input and/or background knowledge. Using the same example, the same wind bracing would be classified as not a frame bracing, and it would be disassociated from the class of truss bracings.

3.5. SELECTION/GENERATION

The selection transmutation selects entities from a large class of known entities which satisfy certain selection criteria, or a single criterion. For example, from a large class of wind bracings only truss bracings are selected, and the attribute No. 4, Joints with the value 'hinged,' is used as a selection criterion.

Generation produces examples of entities which satisfy imposed criteria. In our case, a class of truss bracings is produced, and all new wind bracings are described by attribute No. 4, Joints with the value 'hinged.'

3.6. AGGLOMERATION/DECOMPOSITION

The agglomeration transmutation clusters entities, both known and unknown, into groups using the same selection criterion or criteria. For example, all wind bracings with hinged joints cluster into a class of truss bracings.

In the case of the decomposition transmutation, all entities which are clustered together are further divided into subclasses. In our example of truss bracings, these bracings could be decomposed into one-bay, two-bay, or three-bay subclass. Obviously, many decompositions are possible using various criteria.

3.7. CHARACTERIZATION/DISCRIMINATION

The characterization transmutation identifies a characteristic description of a given class of entities. For example, in the case of a class of bracings in the form of rigid frames, this characteristic is attribute No. 4, Joints with the value 'rigid.'

Discrimination determines the description that can be used to distinguish between two classes of entities. For example, when two classes of truss and frame bracings are considered, the discrimination transmutation determines that attribute No. 4, Joints, is sufficient to distinguish between these two classes of wind bracings: for truss bracings this attribute has the value 'hinged" while for rigid frames its value is 'rigid.'

3.8. DERIVATIONS: REFORMULATION/RANDOMIZATION

The reformulation transmutation reformulates input knowledge using background knowledge in such a way that a segment of input knowledge is substituted by a logically equivalent segment of background knowledge. For example, in given knowledge about truss bracings and their diagonals, knowledge provided about x-diagonals is substituted by knowledge about k-diagonals, which is taken from background knowledge.

In the case of randomization, a given segment of input knowledge is transformed into another by making random changes. For example, random generation of wind bracing types based on morphological analysis (Arciszewski 1987) can be considered as a randomization transmutation. In this case, new types of wind bracings are produced through the analysis of various randomly produced combinations of structural wind bracing components.

3.9. INSERTION/DELETION

The insertion transmutation inserts a segment of knowledge from background knowledge into input knowledge. For example, when knowledge about a wind bracing in the form of a rigid frame is the input, a segment of knowledge about horizontal trusses can be taken from background knowledge and added to the input knowledge. In this way, knowledge about rigid frames with horizontal trusses is produced and can be used to develop a class of rigid frames braced by horizontal trusses.

Deletion can be illustrated by considering wind bracings in the form of belt truss systems. In this case, the input knowledge about bracings may contain the knowledge about vertical trusses, belt truss systems, horizontal trusses. The deletion of knowledge about horizontal trusses and their systems transforms the input knowledge into knowledge only about truss bracings and the deleted knowledge is stored for later use as part of the background knowledge.

3.10. REPLICATION/DESTRUCTION

The replication transmutation replicates a segment or segments of the provided knowledge, taken from the input knowledge or background knowledge. For example, when the input knowledge is about a wind bracing in the form of a one-bay rigid frame, this knowledge may be replicated to describe two independent one-bay rigid frames used in a three-bay skeleton structure as a wind bracing.

The destruction transmutation is the opposite process: the unnecessary parts of the input knowledge are deleted first, as in the case of conducting the deletion transmutation, and then destroyed.

3.11. SORTING/UNSORTING

The sorting transmutation changes the organization of the input knowledge according to some criterion. For example, when knowledge about a class of wind bracings is provided as input knowledge, these bracings may be sorted according to their complexity measured by number of structural members. As the result of sorting, bracings are listed from the simplest, with the smallest number of structural members, to the most complex, with the largest number of structural members.

The unsorting transmutation reverses the results of sorting, and in this case the organization of input knowledge is destroyed.

4. Conclusions

The paper is intended to provide only an initial, conceptual outline of the proposed Inferential Design Theory, and to stimulate discussion of the potentials of Inferential Theory of Learning in engineering design. The proposed framework of IDT is conceptually complete, but more design knowledge transmutations may be proposed in the future as research on both the Inferential Design Theory and the Inferential Theory of Learning progresses. The proposed theory can be used as a basis for the development of a class of formal mathematical models for individual design knowledge transmutations, and such work is already planned. These models will be used next to develop a class of experimental design support tools which will be tested in various engineering design domains to determine the feasibility of conceptual design based on the proposed Inferential Design Theory.

Inferential Design Theory provides a new understanding of conceptual design which is consistent with present research on machine learning. Therefore it can be expected that its introduction will lead to the integration of design and machine learning research. This integration may be a challenge, but a challenge to be met for progress in design science.

5. References

Altschuller, G.: 1986, *Finding Ideas*, Science, Novosibirsk (in Russian).

Arciszewski, T. and Ziarko, W.: 1986, Learning expert system for engineering design, *Proceedings of the International Workshop on Knowledge Based Systems*, Avignon, France.

Arciszewski, T.: 1988, Stochastic form optimization, *Journal of Engineering Optimization*, **13**(1).

Arciszewski, T., Bloedorn, E., Michalski, R. S., Mustafa, M. and Wnek, J.: (in print), Learning design rules for wind bracings in tall buildings, ASCE *Journal of Computing in Civil Engineering*.

Asimov, M.: 1965, *Introduction to Design*, Prentice Hall, New York.

Chan, W. T. and Paulson, B. C.: 1987, Exploratory design using constraints, *Journal AI EDAM*, **1**(1).

Dixon, J. R.: 1988, On research methodology towards a scientific theory of engineering design, *Journal AI EDAM*, **1**(3).

Dyer, M. G., Flowers, M. and Hodges, J.: 1986, Edison: An engineering design invention system operating naively, *Proceedings of the 1st International Conference on Applications of Artificial Intelligence to Engineering Problems*, Southhampton, UK.

Fey, V. and Vertkin, I.: 1993, The theory of inventive problem solving for systems engineering and logistics support, *Proceedings of the International Symposium 'Advances in Logistics,'* Exter University, Exeter.

Flemming, U., Baykan, C. A., Coyne, R. F. and Fox, M. S.: 1992, Hierarchical generate-and-test vs. constraint-directed search, *Report*, Engineering Design Research Center, Carnegie Mellon University.

Hubka, V.: 1982, *Principles of Engineering Design*, Butterworth Science.

Maher, M. L., Sriram, D. and Fenves, S.: 1984, Tools and techniques for knowledge-based expert systems for engineering design, *Advances in Engineering Software*.

Maher, M. L.: 1985, *Expert System for the Preliminary Structural Design of High Rise Buildings*, PhD.Dissertation, Department of Civil Engineering, Carnegie Mellon University.

Maher, M. L.: 1987, Using experience to plan the synthesis of new designs, *in* Gero, J. S. (ed.), *Expert Systems in CAD*, North Holland, Amsterdam.

Maher, M. L.: 1987, A knowledge-based approach to preliminary design synthesis, *Report*, Engineering Design Research Center, Carnegie Mellon University.

Michalski, R. S.: 1993, Inferential theory of learning as a conceptual basis for multistrategy learning, *Machine Learning*, **11**, 3-43.

Michalski, R. S.: 1994, Inferential theory of learning: Developing foundations for multistrategy learning, *in* Michalski, R. S. and Tecuci, G. (eds), *Machine Learning: A Multistrategy Approach*, Morgan Kaufmann.

Miller, J. G.: 1965, Living systems: Structure and progress, *Behavioral Science*, **10**.

Miller, J.: 1976, *Systematische Heuristics*, East German Academy of Sciences (in German).

Nadel, B. A. and Lin, J.: 1990, Automobile transmission design as a constraint satisfaction problem, *Report*, Computer Science Department, Wayne State University.

Nadler, G.: 1985, Systems methodology and design, *IEEE Trans. Systems, Man and Cybernetics*, **SMC-15**(6).

Pahl, G. and Beitz, W.: 1984. *Engineering Design*, Springer Verlag, New York.

Shupe, J. A., Allen, J. K., Muster, D. and Mistree, F.: 1987. Decision-based design: Some concepts and research issues, *in* Kusiak, A. (ed.), *Expert Systems: Design and Management of Manufacturing Systems*, Society of Manufacturing Engineers.

Sriram, D.: 1986, *Knowledge-Based Approaches for Structural Design*, PhD Thesis, Department of Civil Engineering, Carnegie Mellon University.

Suh, N. P.: 1990, *Principles of Design*, Oxford University Press, England.

Tomiyama, T. and Yoshikawa, H.: 1987, Extended general design theory, *Design Systems for CAD*, North Holland.

Zhao, F. and Maher, M. L.: 1988, Using analogical reasoning to design buildings, *Report*, Engineering Design Research Center, Carnegie Mellon University.