Engineering with Computers 2, 21–29 (1987)

Engineering with Computers

Mechanical Design Methodology: Implications on Future Developments of Computer-Aided Design and Knowledge-Based Systems

David G. Ullman
Department of Mechanical Engineering, Oregon State University, Corvallis, OR 97331

Thomas A. Dietterich
Department of Computer Science, Oregon State University, Corvallis, OR 97331

Abstract. The Current Computer-Aided Design (CAD) and Knowledge-Based Systems (KBS, expert systems) tools are changing mechanical engineering design. Future development and integration of these technologies is dependent on an understanding of the methodology of the mechanical design process, an area of little study and one that is poorly understood. This paper reports on the progress of an effort to understand how practicing engineers perform design. The approach is to record engineers' verbalization of their solution of carefully constructed design problems. The recordings are reduced to determine the intellectual tasks and problem-solving methods used. The results will determine what needed capabilities future intelligent CAD systems will need to aid design engineers.

1 Introduction

Current Computer-Aided Design (CAD) and Knowledge-Based Systems (KBS, expert systems) tools are changing the way we do mechanical engineering design. CAD has increased productivity, added never before available visualization, and tied graphic representation to complex analysis tools, making them easier and faster to use. Knowledge-based systems are just beginning to move from the research of artificial intelligence (AI) to becoming useful engineering tools. Some KBS tools are being used to configure computers, select materials, diagnose failures, and assist in project planning.

As new CAD tools are developed, it is anticipated that they will include more aspects of KBS. They will embody more of the logic that enables KBS tools to deal with symbolically structured problems. What capabilities the use of KBS will add to CAD is a matter of conjecture. In the area of mechanical design, development of these computer-based technologies is dependent on an understanding of the methodology of the mechanical design process, an area of little study and one that is poorly understood. Thus, it can be said that the next generation of CAD tools is limited by a lack of basic understanding of how mechanical design engineers do their job.

This paper briefly reviews the status of CAD and KBS systems and the current theories on mechanical design methodology. Some preliminary results from an ongoing study are used to develop a list of research areas in which further work is needed. Progress in these areas will support the development of future intelligent CAD systems.

2 The Current State of CAD Design Tools

For the purposes of this paper the term CAD is defined as the use of interactive computer graphics to help solve design problems. Current CAD tools aid the mechanical design process in four ways: as an advanced drafting tool; through assisting in the visualization of hardware and data; by improving data organization and communication; and through being used as a pre- and postprocessor for computer-based analytical techniques such as finite element analysis, weight and mass properties, kinematic analysis, and so on. As mentioned previously, these capabilities are having a profound effect on mechanical engineering design. As will be developed subsequently, these capabilities serve only a part of the mechanical design process.

3 The Current State of KBS Tools in Design

Knowledge-based systems, or expert systems, are just beginning to make their way into mechanical

---

Reprint requests: D.G. Ullman, Associate Professor of Machine Design, Department of Mechanical Engineering, Oregon State University, Corvallis, OR 97331.

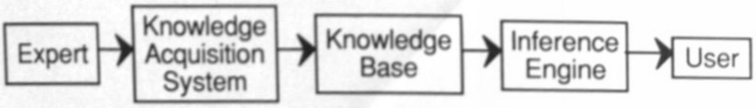

Fig. 1. Simple knowledge-based system structure

engineering design. The application of KBS in mechanical engineering was the focus of the 1985 ASME Computers in Mechanical Engineering Conference (Boston, May 1985). At this conference a number of papers were presented specifically oriented toward mechanical design. To set the stage for the points made in the following sections, we will present the makeup of a KBS.

A knowledge-based system (or expert system) is a computer program that employs explicit knowledge structures (e.g., rules and facts) to solve problems that otherwise could only be solved by human experts. The knowledge structures capture and symbolically represent the expertise of the human expert. This expertise is typically a combination of factual knowledge (e.g., of material properties and manufacturing processes) and heuristic knowledge (e.g., rules of thumb for estimation). To construct such systems conveniently, symbolic programming languages are usually employed. In its simplest form, a KBS is structured as shown in Fig. 1. An expert, or group of experts, impart their knowledge to the knowledge base through the knowledge acquisition system (typically with the assistance of a knowledge engineer). A user can then make use of this knowledge through the inference engine. Thus, the KBS, the link between the expert and the user, is like a book with the expert the author and the user the reader. The chief difference is that, unlike a book, the KBS can interact with the user (by asking questions and providing explanations) to solve particular problems.

The knowledge base is subdivided into two parts: facts and rules. Facts are knowledge descriptions of the variables in the problem. These may be quantitative, such as the size or weight of an item, or more qualitative, how an item is perceived. Rules are the embodiment of expertise. They represent the knowledge of how to put the facts together to generate a solution. In most current knowledge-based systems, these rules are represented as IF-THEN-ELSE statements relating problem features to solution features.

The inference engine (or controller) embodies the problem-solving methods and strategies necessary to combine the rules and form the solution. In most current systems these are fixed, with the method being backward chaining and the strategy being depth first search.

The exact form of the KBS is dependent on the underlying knowledge, the intellectual tasks, and the problem-solving methods employed by the experts. To date, the breadth of the tools available only cover a small part of the intellectual tasks and problem-solving methods used in mechanical design: those that have a very limited choice of solutions, are easily broken down into independent subproblems, have their constraints independent of the solutions, and have reliable data available for the knowledge base. Thus, few design problems are solvable with the current KBS tools. More detail on KBS systems and their limitations is available (1,2).

4 Mechanical Design Methodology

The term ‘‘mechanical design’’ is most often defined as the creative decision-making process for specifying or creating physical devices to fulfill a stated need. This definition describes what mechanical design is but gives no indication as to how it is done. The actual process of mechanical design is not very well understood, even though there are a number of books on the subject (3,4). There has been a recent interest in Europe in the subject (5,6). This European literature has received little attention in the United States. Typically, all these books have some hints as to how to organize the design process, but they are, by necessity, very general and based on the authors' own techniques. They are really texts on what the design engineer *should* do during design but are not based on studies of how designers do design. Only one paper has been found in the mechanical design literature that is based on actual design observations (7), and it appears to be a mixture of analysis and conjecture.

All of these references break down the design process much as it is in a product-based industry. Namely, the design problem usually starts with a problem assignment. The first phase of the design process is to transform the problem into a well-formed set of design specifications (problem solu-

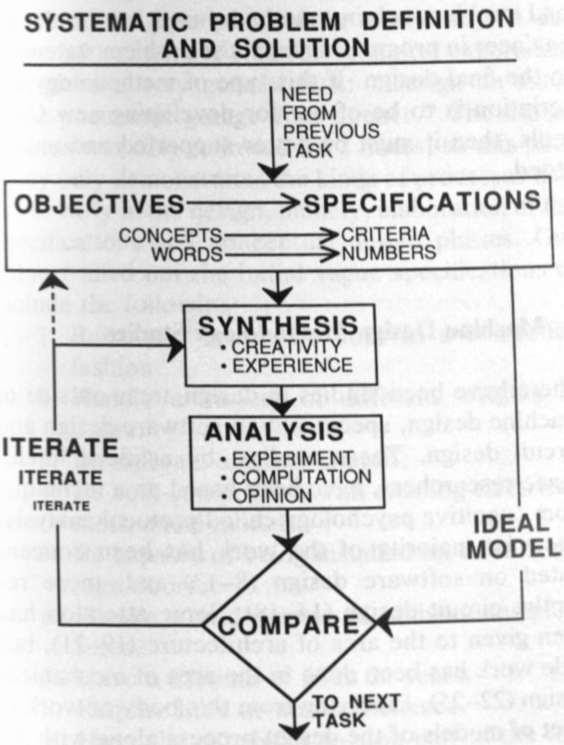

Fig. 2. Task model design

tion constraints). This specification development phase includes establishing the state of the art and attempting to represent all specifications as quantitatively as possible. These specifications form an ideal description or model toward which the overall design is aimed and against which all potential designs are compared.

The second phase is the conceptual or preliminary design phase. Here the product or machine as a whole is the main focus, and the goal is to establish functional structures and concepts to compare to the ideal design. This phase of design requires much creativity, as limiting the quantity of concepts here limits the final solution. The focus in this phase is on the design as a whole with individual assemblies and components essentially modeled as nebulous black boxes. Simulation of the operation of the machine is difficult here because few quantitative analysis tools are available and only qualitative descriptors are used for many of the components and operations.

When a number of preliminary designs are found to meet the design goals, the black boxes can be fleshed out into hardware and a layout can be established. This is termed the layout phase, and its goal is to reduce the acceptable designs to one fully developed, "optimal" design ready for detailing. After a layout is successfully developed, the detail design can take place. This is the interface between the prototype and production, where every component is detailed and topics such as materials, processes, vendors, manufacturability, maintainability, and assemblability are finalized. These four major phases—specification development, conceptualization of feasibility, layout design, and detail design—characterize the design process. They describe the state of the design at various points in the process and are guides as to what is done in the development of the design but give no information on how the design is accomplished. There is not enough detail in this description of the design process to realize the intellectual tasks, problem-solving methods, and the underlying knowledge brought to bear on the problem.

These phases do, however, speak to the necessity of the design process moving from a general overview to an increasingly detailed development of each component. This implies a structure that can be represented as a tree with the problem statement at the root and the problem dividing and subdividing as more detail is added. Each branch of the tree can represent a subtask that is characterized by a set of specifications defining the design goals of the task. A company or a specific engineer breaks up each task into subtasks in a manner that can be dependent on hardware and intellectual boundaries. How this subdivision is made, how the design progresses through these tasks, and how the results from one task effect the specifications of other tasks are not understood at all. It is realized that the ways that a problem is broken down into subproblems and how these subproblems are solved varies from one designer to another. This blocking out of the design process into tasks and subtasks can be further clarified if each task, each branch on the tree, is looked at in more detail. Figure 2 shows a potential detailing of each task. This task model is thought to hold at any level in the design process, be it the overall specification development for the entire problem or the detail design of a specific component.

In this model, the design specifications for the specific task are determined not only from the specifications and design results of more preliminary tasks, but also from those on a similar or even more subordinate and detailed level. This interdependence or communication between specifications is an important part of the design process that is not evident in the references and not addressable by current KBS systems. These specifications must be clear before any ideas for solution are synthesized, since unclear specifications by nature converge to meet the solution (i.e., the solution drives the specifications, not the other way around). Additionally, the specifications must be as precise and numerical as possible. This is necessary for any formal type of comparison or optimization using currently available tools. After the specifications are determined, ideas for the solution can then be synthesized. Documenting the ideas is important because as the design matures and the specifications are updated, previously discarded designs may be applicable.

Designs are synthesized or created in a manner not yet well understood. Part of the creative process is based on experience, either personal or through outside experts, literature, or other data base. Part of the creative process is the ability to put together the experience in a manner that meets the design objectives. Often the terms innovation or invention are used as synonymous with synthesis.

The analysis or simulation might be as simple as an opinion based on gut feeling or as complex as a full finite element analysis coupled to an optimization routine to find the "best" shape to meet some stress specification. Once an acceptable result is obtained (what is acceptable is not clear nor is it constant), the results can then be used to develop specifications for more detailed elements of the design and to update the specifications of other tasks. It is of interest to note that the level of analysis required in a specific task is not the same during the feasibility phase as it is during the detail design. This inconsistency in the depth of analysis required is an area of needed study.

In the design process, synthesis and analysis are wrapped together in an iteration loop. Ideas are analyzed to enough depth to ascertain their validity and are used to develop other ideas, with the depth of analysis increasing as the final choice is approached. This synthesis-analysis loop is fundamental to the design process.

After analysis, each concept is compared to the ideal model which is determined by the specifications. Optimization techniques perform this in a formal way, whereas informally, acceptable solutions are often sufficient to reach a final design. The resulting design is then used to determine specifications for subtasks and may also influence tasks at a higher level in the task tree.

The preceding description is merely a conjectural overview of the design process. It is based on scant evidence and gives little detail as to intellectual problem-solving methods used by the design engineer in progressing from the problem statement to the final design. If this type of methodology description is to be of use for developing new CAD tools, then it must be better supported and understood.

5 Machine Design Methodology Studies

There have been studies in design areas outside of machine design, specifically in software design and circuit design. These studies, by artificial intelligence researchers, have been based on a technique from cognitive psychology called protocol analysis. The vast majority of the work has been concentrated on software design (8–13) and, more recently, circuit design (14–18). Some attention has been given to the area of architecture (19–21), but little work has been done in the area of mechanical design (22–25). Emerging from this body of work is a set of models of the design process along with AI methods for implementing design programs. Mostow (26) gives an excellent overview of the current state of AI models of design.

To address the lack of machine design methodology understanding, an ongoing study of how mechanical designers solve design problems has been undertaken. The approach is to apply methods from information processing psychology (27,28) to gather data on the actual behavior of mechanical designers. The need for such studies has been widely recognized. Mostow (26), for example, states that the development of cognitive models of design is an important research issue for the next decade. The method called "protocol analysis" is being applied to gather data on designer behavior. A designer is given a problem statement for a particular design problem and asked to "think out loud" as he or she solves the problem. The session is videotaped, and these tapes are then analyzed to identify the intellectual tasks, problem-solving methods, and control strategies that the subject has carried out.

Thus far, a pilot study has been carried out to test the methods and help plan for future, more detailed studies. Hence, the results presented in this paper are very preliminary. Nevertheless, several interesting behaviors have been identified, and it is believed that the protocol analysis methodology shows great promise for helping create an artificial intelligence model of design.

Here is a description of the pilot study. A designer with extensive electromechanical experience was given a very general task: "Design an automatic pneumatic garage door opener." The design process was only followed for 2 hours, so this pilot study only demonstrates the kinds of processes that occur early in the design, namely, elaboration of the specifications and conceptual design phases. Our subject filled out the initial vague specifications to include the following:

1. Raise and lower the door in a controlled fashion.
2. Handle a variety of different weights of doors.
3. Incorporate a safety shutoff.
4. Be cost-competitive with existing electrical-mechanical models.
5. Be capable of being installed on existing garage doors.
6. Handle a wide range of ambient temperatures.
7. Have a lifetime of 15 to 20 years.
8. Require little or no maintenance.
9. Have the capability to stop part of the way open or closed.

Of these specifications or constraints, 1 through 4 were discovered almost immediately by considering the design of existing garage door openers. The remaining specifications were introduced gradually as possible designs were considered.

During the design process, the subject focused primarily on specifications 1 and 3. The other specifications influenced the design process mostly by serving to eliminate possible alternatives. Specification 1 was broken into two subgoals: (1a) to raise and lower the door and (1b) to do so in a controlled fashion. Most attention was given to goal 1a initially, but possibly 1b would have been addressed later had the design been allowed to progress. Goals 1b, 3, and 9 were handled by patching the design developed for 1a. Other goals were handled through the choice of subgoals and components. It is likely that goal 4 (cost) was not satisfied.

The final design (after 2 hours of work) must still be considered a preliminary design. It calls for an electric motor to drive an air pump, which in turn drives a sliding seal-style cylinder. A rope or cable that connects to the door is attached to the piston. The piston also contains a mechanical break that locks in the absence of sufficient air pressure. There are controls to detect when the door is fully raised and to control the rate of raising and lowering the door. Gravity is employed to lower the door. Safety is provided by sensing the changes in the pressure inside the piston.

As discussed earlier, the main goal of this pilot study was to identify the intellectual tasks and problem solving methods employed in mechanical design. The analysis of the pilot protocol uncovered a wide variety of tasks and methods. For example, one important problem-solving method employed by our subject was the failure-driven patching strategy. In this approach (29,30) two subproblems are solved independently. Then the global solution is checked for correctness. If a "bug" is found, an attempt is made to analyze the nature of the problem and the design is "patched" to repair it. In this protocol, the initial electric-pneumatic design called for using a simple 8-foot long cylinder to raise the door. Analysis of this idea uncovered two basic problems. First, when fully extended, the cylinder/piston combination would require 16 linear feet, and this was too large to fit in an ordinary garage. Second, a push-type piston would require an extra seal where the rope or cable entered the cylinder. The subject attempted a series of eight different patches to get out of this problem. Here is a section of the protocol where the first two patches were attempted (analysis enclosed in square brackets):

S: "If I'm going to put pressure in the area behind the piston, there's uh, I've got an added seal. That is not, that's not a good idea." [Summary of analysis]

S: "And I can't suck a vacuum on the other side if I need 50 psi because the best I can get is 14 psi atmospheric, so that's out." [Patch #1: Use vacuum]

S: "I was just gonna try to make the piston go in the other direction toward the door somehow feeling I could get the pressure on the side that this piece of cable materials wasn't on, but I can't." [Patch #2: Reverse piston direction]

From this section of the protocol, we see three different intellectual tasks: (1) analysis of a configuration, (2) component patching, and (3) configuration patching. Patch #1 is a component patch, because a different kind of component is being substituted for the original push-type piston. Patch #2 is a configuration patch because it involves altering the way chosen components are configured. For each patch, analysis techniques are being applied immediately to evaluate the proposed patch. The analysis of patch #1 is clear (i.e., atmospheric pressure is 14 psi and this is the most that can be achieved in a suction-type piston), but the protocol does not reveal how S analyzed patch #2.

Detailed analysis of this kind has uncovered a large number of intellectual tasks and their associated problem-solving methods. Some of these seen in the example protocol are listed here. They are divided into those used in the specification development phase and those used in the preliminary design phase. This breakdown is subject to continued development but even in this preliminary form it shows directions of needed research and development.

Also noted for each intellectual task there is a comment about KBS systems that is available to assist in the task. These systems are divided into three groups: tools (generally usable products); research prototypes (still in university or commercial development); and commercial custom systems (tools developed for a specific application, not generalized and usually proprietary). There are no CAD systems listed for each of these intellectual tasks as none exists to assist the engineer at the stages of development addressed. More comments on this follow the listing.

Intellectual tasks associated with establishing and refining the specifications are as follows:

1. Specification development
   Intellectual task: Identify additional constraints that should be added to the specifications.
   Problem-solving method: Apply background knowledge to analyze the environment in which the device will eventually be used. There is a body of expertise involving potential problems to check for (e.g., extreme temperatures, extreme pressures, neglected maintenance, and so forth). The design of existing garage door openers was employed to guide the development of the specifications. This use of analogy to other designs was seen in many of the tasks.
   KBS systems: None are known.
2. Specification assessment
   Intellectual task: Determine if the specifications are complete.
   Problem-solving method: In the protocol, our subject appears to employ the lack of knowledge inference to determine completeness of the specifications. In other words, when no other specifications can be found, the specifications are assumed to be complete.
   KBS systems: There are some commercial custom systems that are proprietary.
3. Specification refinement
   Intellectual task: Determine a set of subgoals or constraints whose combination would satisfy a given goal.
   Problem-solving method: Several methods were employed here. One interesting method applied qualitative simulation to a partial design in order to identify additional goals that needed to be satisfied. Another approach involved applying existing refinement rules (e.g., if motion is required, a power source is needed).
   KBS systems: There are some research prototypes [e.g., MOLGEN (31) and VEXED (15)].
4. Specification patching
   Intellectual task: Modify a goal so that it can be more easily satisfied.
   Problem-solving method: In the pilot study, two kinds of goal changes were observed. First, simulation and further design uncovered additional goals that need to be achieved (e.g., the ability to adjust the rate of raising and lowering the door). Second, design commitments entailed that some constraints (specifically, cost constraints) needed to be relaxed.
   KBS systems: None are known.
5. Innovation search
   Intellectual task: Find a new way to accomplish a task for which designs already exist.
   Problem-solving method: Traverse a given design history considering alternative decisions that might have been made at each choice point. Carbonell (32) terms this method derivational analogy.
   KBS systems: None are known.

Intellectual tasks associated with the preliminary phase of design are as follows:

1. Component refinement
   Intellectual task: Augment the current description of a component (e.g., "cylinder") to include further design choices.
   Problem-solving method: The generate-and-test method was employed here. The designer has knowledge of the possible ways to refine a component. Each of these is generated and then tested to see if it will work in this case.
   KBS systems: There are some research prototypes [e.g., MOLGEN (31) and VEXED (15)].
2. Component patching
   Intellectual task: Modify a component so that it satisfies a given constraint.
   Problem-solving method: Two basic tech-

niques were employed. One approach is to substitute a slightly different component (e.g., substitute a custom cylinder for an ordinary commercially available cylinder). The other approach is to augment the component with brakes, valves, seals, mountings, and so on. Simulation was employed to determine where the augmentations should be positioned.

KBS systems: There are a number of commercial systems [e.g., XCON (2) and VT (33)]. VT has been generalized as a tool called SALT (33).

3. Component configuration

Intellectual task: Determine how to position and interconnect components so they achieve the desired function.

Problem-solving method: Most of the configuration decisions were dictated by the flow of power through the system. Geometrical reasoning (and perhaps "abstract generate and test") was employed to identify possible locations for the cylinder.

KBS systems: There are a number of commercial systems [e.g., XCON (2) and VT (33)].

4. Configuration patching

Intellectual task: Find a perturbation of the existing configuration that satisfies some constraint.

Problem-solving method: Operations such as inversion and rotation were employed to patch configurations. The relative positions of two components could be inverted, or the force relationships between two components could be reversed (e.g., "push" instead of "pull"). A third kind of patch involved changing which components are stationary and which are in motion. A fourth kind of patch involved changing the number of components or their size. Finally, the direction of motion could be reversed or altered to involve rotation rather than translation.

KBS systems: There are a number of commercial systems [e.g., XCON (2) and VT (33)].

5. Design recognition

Intellectual task: Analyze an existing design to determine the specifications and principal constraints underlying it.

Problem-solving method: Apply recognition heuristics to identify stereotypical design configurations (bottom-up recognition). Perform the design over again, consulting the existing device to determine which design decisions to make (top-down recognition). Similar techniques have been employed by Genesereth (34) to perform plan recognition. This was employed in determining how current garage door openers operated.

KBS systems: None are known.

6. Refinement evaluation

Intellectual task: Analyze a proposed refinement for feasibility and desirability.

Problem-solving method: Many methods were employed here including qualitative simulation, standard analytic techniques, knowledge of manufacturability, knowledge of cost, and so on.

KBS systems: There are a number of commercial systems [e.g., XCON (2) and VT (33)].

As can be seen, the variety of tasks performed in just developing the specifications and in the preliminary design phase is substantial. KBS tools are available or being developed to assist in some of the preliminary design tasks, but CAD tools, oriented toward quantitative descriptions of components and configurations cannot be applied until the configuration is more concrete. Current CAD systems only benefit the layout and detail design phases. Thus, it could be of no help in the protocol study described earlier, which never progressed beyond the conceptual design phase.

6 Needed Directions of CAD-KBS Development

Based on the effort to associate the fields of CAD, KBS, and design methodology and on the protocol study discussed previously a number of needed tools or capabilities become evident. This listing makes no differentiation between CAD and KBS as they will need to blend to satisfy many of the items.

6.1 *Assurance of Complete Design Specifications*

In the initial understanding of the problem and in the design of assemblies and components, how can there be assurance that the design specifications are complete? This is especially important when considering the design of assemblies that interact. The changing of one component affects the others and tracking the interrelationships is difficult. In the protocol example the subject used at least five identifiable intellectual tasks. Only in assessing and refining the specification has any work been undertaken.

6.2 *Design Methodology*

Assuming there is a preferred design methodology, current CAD/KBS tools give no guidance on how to proceed through the design. This is a touchy point as design methodologists advocate that there is a preferred method, whereas design engineers, American design engineers in particular (35), are very sure that design method is individualistic. No evidence exists that there is a preferred way or that everyone designs his or her own way. It is not clear from the theory of methodology or the protocol study how the designer determines what to do next. If tools can be developed in this area, their form is not clear. Further studies into the design process are needed to identify the effects of measures such as expertise and familiarity with the problem being solved.

6.3 *Rough Sketching and Modeling*

CAD is a very effective tool for finished drawings and detailed analysis of components and some assemblies. It is, however, of little use in the conceptual and early layout phases where sketching and qualitative simulation are needed. Designers use sketching and simplistic, nonanalytic modeling during these phases. The depth of modeling is limited to what can be done mentally, because there are no tools that can match the depth of modeling required at this stage of the design. In other words, tools are needed that can assist in back of the envelope, cardboard model, and hand-waving design.

6.4 *Design History Recording*

If the history of the design were recorded from specification onward, failure in meeting the specifications at some point in the design process would only require backtracking to a previous point in the design history for a new attempt. Additionally, specification changes from an existing design would only require backing through the design history to the point where the change has effect. This assumes the development of a vocabulary to characterize mechanical design which goes beyond that necessary for solid modeling. It requires the development of a machine design language capable of dealing with the specifications, components, processes, and so on. This design history is basic to many of the intellectual tasks, especially innovation searching and the patching tasks. A history of the existing garage door opener design would have given a basis for the effort discussed earlier.

6.5 *Design Catalog and Handbook Interfacing*

Currently CAD systems have a very limited ability to utilize handbook and catalog information for modeling and analysis. This information would assist in patching and innovation searching. Group theory and data base research is currently dealing with this area.

6.6 *The Integration of KBS and CAD Tools*

The integration of KBS into CAD is not yet available; however, the concept of a CAD system that has the knowledge structures of KBS systems has implications of greatly streamlining the design process. The development of such systems, however, are limited by the tools available. Nonetheless, the capability exists to develop systems in limited, well-defined design areas. A combined system could suggest components or parts of components such as flanges, ribs, and so on during the layout process based on analysis of the current design and the expertise in the knowledge base.

6.7 *KBS Tools Configured for Design*

Most design problems are characterized by large solution spaces, where the potential solutions cannot be itemized prior to the beginning of the problem. Current expert systems require a well-defined solution space and thus are not applicable. The limitations of current KBS systems is seen in the scarcity of mechanical-design-oriented KBS literature cited. Specifically, the developmental needs fall into two broad areas: representation and control.

In order to develop KBS systems that are oriented toward the problem of mechanical design, we need to develop vocabularies and reasoning methods that are sufficient for representing the specifications, the components, and assemblies, and for the design steps themselves. These were alluded to in item 4, the design history. The description of the components or artifacts includes attributes such as geometry, structure, behavior, manufacture, assemblability, and so on.

Control needs are for more flexible schemes and a formalization implementation of the problem-solving methods used in design.

7 **Summary and Concluding Remarks**

Many engineers have enthusiastically embraced the knowledge-based systems technology emerging from artificial intelligence research. This paper has examined the state of KBS and CAD technologies in light of our current knowledge of design methodology. The main conclusion is that although there are important ways in which KBS and CAD systems can be combined, existing tools are only capable of addressing a small portion of the overall process of design. Indeed, our understanding of the design process is so sketchy and vague that it is very difficult to see how the simple KBS and CAD tools available today can be aimed at clearly definable subtasks of the mechanical design process. Further research is needed both to understand the design process and to develop more powerful and more flexible KBS tools to be integrated into CAD systems. An ongoing research project at Oregon State University is addressing both of these research needs.

Acknowledgments

This material is based upon work supported by the National Science Foundation under Grant No. DMC-8514949.

References

1. Hayes-Roth, F.; Waterman, D.A.; Lenat, D.B. (1983) Building Expert Systems. New York: Addison-Wesley
2. Harmon, P.; King, D. (1985) Expert Systems. New York: Wiley
3. Love, S.F. (1980) Planning and Creating Successful Engineered Designs. Advanced Professional Development, Inc., Los Angeles
4. Ostrofsky, B. (1977) Design, Planning, and Development Methodology. Englewood Cliffs, NJ: Prentice-Hall
5. Hubka, V. (1982) Principles of Engineering Design. London: Butterworth
6. Pahl, G.; Beitz, W. (1984) Engineering Design. London: Springer-Verlag
7. Marples, D.L. (1960) The decisions of engineering design. Eng. Design. December, pp. 1–16
8. Adelson, B.; Soloway, E. (1984) A cognitive model of software design. Rep. No. 342, Department of Computer Science, Yale University, Princeton, NJ
9. Balzer, R.; Goldman, N.; Wile, D. (1976) On the transformational implementation approach to programming. In: Proc. 2nd Int. Conf. Softw. Eng., pp. 337–343
10. Fickas, S. (1982) Automating the transformational development of software. Doctoral Dissertation, Computer Science Department, University of California, Irvine
11. Kant, E.; Newell, A. (1983) An automatic algorithm designer: An initial implementation. Proceedings of the National Conference on AI, AAAI–83, 177
12. Swartout, B.; Balzer, R. (1982) On the inevitable intertwining of specification and implementation. Commun. ACM 25(7), 438–440
13. Wile, D.S. (1983) Program developments: Formal explanations of implementations. Commun. ACM 26(11), 902–911
14. Lenat, D.B. (1983) The nature of heuristics, II and III. AI 21(2,3), 31–98
15. Mitchell, T.; Steinberg, L.; Shulman, J. (1984) VEXED: A knowledge-based VLSI design consultant. Rutgers AI/VLSI Project Working Paper No. 17
16. Katz, R.; Subrahmanyam, P.A.; Scacci, W. (1984) Design environments for VLSI and software. Jo. Sys. Softw. 4, 1–13
17. Mostow, J.; Balzer, B. (1984) Application of a transformational software development methodology to VLSI design. Jo. Sys. Softw. 4
18. Subrahmanyam, P.A. (1983) Synthesizing VLSI circuits from behavior specifications: A very high level silicon compiler and its theoretical basis. In: VLSI '83. (ed. F. Anceau) Amsterdam, North-Holland
19. Eastman, C.M. (1968) Explorations of the cognitive processes in design. NTIS AD 671158. Doctoral Dissertation, Carnegie-Mellon University, Pittsburgh, PA
20. Maher, M.L. (1984) HI-RISE, a knowledge-based expert system for the preliminary structural design of high rise buildings. Design Research Center Rep. No. DRC-02-22084. Doctoral Dissertation, Dept. of Civil Engineering, Carnegie-Mellon University, Pittsburgh, PA
21. Srirarm, D.; Maher, M.L.; Fenves, S.J. (1984) Knowledge-based expert systems in structural design. Design Research Center Rep. No. DRC-12-24-84, Carngie-Mellon University, Pittsburgh, PA
22. Brown, D.C.; Chandrasekaran, B. (1983) An approach to expert systems for mechanical design. In: Trends and Applications, '83. IEEE Computer Society, NBS, Gaithersburg, MD, pp. 173–180
23. Brown, D.C.; Chandrasekaran, B. (1984) An approach to expert systems for mechanical design: A progress report. In: Proc. Comput. Eng. Conf., Las Vegas, August
24. Dixon, J.R.; Simmons, M.K. (1983) Computers that design expert systems for mechanical engineers. Comput. Mech. Eng., November, 10–17
25. Dixon, J.R.; Simmons, M.K. (1984) Expert systems for engineering design: Standard V-belt drive design as an example of the design-evaluate-redesign architecture. In: Proc. Comput. Eng. Conf., Las Vegas, August
26. Mostow, J. (1985) Toward better models of the design process. AI Magazine 6(1), Spring, 44–57
27. Newell, A.; Simon, H. (1972) Human Problem Solving, Englewood Cliffs, NJ: Prentice-Hall
28. Newell, A.; Simon, H. (1976) Computer science as empirical inquiry: Symbols and search. Commun. ACM 19(3), 113–126
29. Sussman, G.J. (1973) A computer model of skill acquisition. New York: American Elsevier
30. Sacerdoti, E.D. (1975) A Structure for Plans and Behavior. New York: American Elsevier
31. Stefik, M. (1982) Planning with constraints. AI 16(2), 111–140
32. Carbonell, J.G. (1983) Derivational analogy and its role in problem solving. Proceedings of the National Conference on AI, AAAI–83, 64
33. Marcus, S.; McDermott, J.; Wang, T. (1985) Knowledge acquisition for constructive systems. In: Proc. Int. Jo. Comput. AI 85, Morgan Kaufmann Publishers, Los Altos, CA, pp. 637–639
34. Genesereth, M.R. (1982) The role of plans in intelligent teaching systems. In: Intelligent Tutoring Systems (ed. D. Sleeman; J.S. Brown) London: Academic Press
35. Eder, W.E. (1981) Design engineering methodologies: Comparisons of English- and German-language regions and influences of culture. Int. Conf. Eng. Des., ICED '81, Rome, Italy, March; and WDK 5, Konstruktionsmethoden in Ubersicht (ed. V. Hubka) Zurich: Heurista