TOWARDS A GENERATIVE SYSTEM FOR INTELLIGENT DESIGN SUPPORT

JOHN HAMILTON FRAZER, MING XI TANG, and SUN JIAN
School of Design, The Hong Kong Polytechnic University

Abstract. In the development of intelligent computer aided design systems, three important issues need to be considered: how to support the generation of product concepts using evolutionary computation techniques; how to use intelligent databases and constraint management systems for detailed exploration of product embodiment; and how to integrate rapid prototyping facilities for product evaluation. In this paper, we present a brief review of knowledge based design and evolutionary design and discuss ways of integrating both in the development of a generative design system. Based on this review, we present the model and its applications of a generative design system utilizing a number of AI and evolutionary computation techniques. This generative design model is intended to provide a generic computational framework for the development of intelligent design support systems.

1. Knowledge-based design

Existing computer-based design systems provide little support for the evolution of design solution concepts that satisfy all physical constraints. They rely on precise geometric information to specify the model of an artifact. They cannot support early stage design tasks because the geometry of an artifact is not sufficiently definite at the early stage. The design of complex products containing many components require a number of alternative solutions to be explored, each being modified by making changes to key design parameters, until a satisfactory design solution is found. The generation and exploration of design concepts based on incomplete design requirements, and the handling of a multitude of variables and constraints embedded within 3D design objects is a challenging task for designers.

Four main strategies may characterize the current developments in knowledge-based design systems:
• The intelligent CAD approach;
• The building block approach;
• The prototype approach; and
• The constraint-based approach.
The intelligent CAD approach extends the capability of a CAD system by employing heuristic knowledge on top of geometric models of the artifacts. An intelligent CAD system plays an active rather than a passive role in the design process, by incorporating a significant amount of design knowledge into the system. Functional modeling, for example, is an approach that attempts to build intelligent CAD systems through a mapping from a function space to a conceptual solution space using functional instead of geometric components and their interfaces [Chakrabarti and Tang 1996]. This approach uses abstract knowledge to represent the function and behavior of elementary design components, parts and their causal relationships. This abstract knowledge is used to generate initial conceptual solutions, which are subsequently selected, evaluated and specialized further. This approach is based on the assumption that designers use abstract and qualitative knowledge at the early stage of the design process to generate a conceptual model of an artifact. This conceptual model can be generated without detailed geometric information such as dimensions.

The building block approach formulates design process as a sequence of tasks that can be tackled individually by different classes of CAD tools and AI methods [Brown et al 1989]. Chandrasekaran classified these tasks into three categories:
1. Class 1 - design tasks in which the components of the artifact being designed are unknown;
2. Class 2 - design tasks in which the components of the artifact being designed are known, but the design plans, i.e. the methods of how to design, are unavailable in a compiled form; and
3. Class 3 - design tasks where ways of decomposing a design problem are already known and for which compiled design plans are available for each stage of the design process

In this approach, design tasks under class 3 are considered routine designs that can be conveniently supported by automated computer programs. The others are more difficult and require significant creative inputs from human designers.

The prototype design approach divides design process into three different activities: prototype refinement, prototype adaptation and prototype creation based on a library of design prototypes [Gero et al 1989]. Gero classified design as: routine design where the functions, expected behaviors and structural variables of the design are known, and the problem of design is one of instantiating values for structure variables; innovative design, where certain aspects of a defined design space need to be modified or extended because no existing solution within that space meets the design requirements; and creative design, in which an entirely new design problem space is to be defined [Coyne et al 1990]. Prototype refinement involves instantiating the variables in a design prototype and determining their values. Prototype adaptation becomes necessary
when a design prototype is found to be inadequate in some way, and needs to be adapted to a new design problem, or made generally more useful for other design problem solving. Prototype creation is seen as the ultimate design endeavour in this approach. So far research based on this approach has mainly concerned the problems of prototype refinement and prototype adaptation. Little progress has been reported on the issue of prototype creation and it remains a difficult subject.

The constraint-based approach models an engineering system as a constraint network or a hierarchical structure, i.e. collections of constraints that are interconnected by design variables [Smithers et al 1990]. Instantiation of this structure, or part of it, forms a basic structure of a new design problem. Computer-based constraint-based reasoning techniques are used to derive the values of unknown design variables from some initial data provided by the designers to form the solutions to the new design problem. Computational methods are used to detect constraint violation and to suggest alternative values for design variables. The application of design strategies or design methods is reflected in the way in which design variables are created, constrained, and manipulated.

The original aim of AI and knowledge based design was to produce general purpose, domain independent AI tools that would automate design tasks requiring intelligence. The early promise of knowledge based design has not been fulfilled because these approaches:
- did not scale within and across domains,
- unable to learn and adapt to design contexts,
- failed to handle complexity, consistency and unpredictability,
- could not acquire knowledge satisfactorily, and
- could not model and support creativity.

2. Evolutionary design

Recent advancement in evolutionary computing provides new opportunities to re-examine the issue of intelligent design support. Genetic algorithms are widely accepted as powerful generative and adaptive techniques generally applicable to many design activities. Evolutionary design is an approach that utilizes different evolutionary computation techniques in different stages of the design process. Evolutionary design is capable of generating astonishing imaginative and innovative designs [Bentley 1999]. The strength of evolutionary design comes from the factor that controlled evolution can be formulated as a general purposed problem solver with ability similar to human design intelligence but with a magnitude of speed and efficiency. Traditional AI methods such as rule-based reasoning have to model design intelligence explicitly in terms of knowledge both in representation and inference. These
methods have serious drawbacks because the process of how human designers actually use this kind of knowledge is not necessarily fully understood.

Bentley classified evolutionary design into four categories. These are:

- **evolutionary design optimization** that is concerned with optimizing existing designs by evolving the values of suitably constrained design parameters;
- **creative evolutionary design** that generates entirely new designs from little abstract knowledge to satisfy functional requirements;
- **conceptual evolutionary design** that deals with the production of high level conceptual frameworks of preliminary designs; and
- **generative evolutionary design** that directly produces forms of designs contributing to the emergence of implicit design concepts.

These evolutionary design approaches combine several vital aspects of design intelligence in an evolutionary process, including modeling design data and information, concept formation, idea generation, optimization, learning, and evaluation. Once a design problem is properly formulated in an evolutionary process, the computer is able to generate a large number of candidate solutions before reaching an optimum one. The candidate solutions are unpredictable but the process and the final outcome are controllable by the designers.

The evolutionary design approach has an excellent potential for developing more intelligent design support tools. For example, Frazer used genetic algorithms in his evolutionary architectural design to evolve unpredicted forms of architectures and their possible interactions with the environments [Frazer 1995, 1996 and 1997]. Chakrabarti developed a functional synthesis program that generates a large number of abstract design concepts from functional requirements and abstract building blocks of engineering elements [Chakrabarti and Tang 1996]. Thornton utilised genetic algorithms as constraint management tools in the process of embodiment design [Thornton 1994].

However, the development of evolutionary design tools is still at its early stage. So far, many genetic algorithms have been used and tested only in design problems of small scale. A theoretical understanding of evolutionary design and its applications in design process is necessary. A computational model is needed in order to formulate product evolution processes in which genetic algorithms can be used as general purposed problem solvers. In a highly automated and evolutionary computational process, the role of designers in creative decision-making must be strengthened rather than weakened. In this sense, evolutionary design techniques as general purposed problem solvers or design support tools need to be integrated with knowledge-based design techniques in order to reflect designers’ expertise and experience in any automated generative processes.
3. A generative system for intelligent design support

A generative design system is being developed in the School of Design in the Hong Kong Polytechnic University. This system is intended for the study of adding generative capability, design intelligence, and rapid prototyping facilities in computer aided design systems. A particular issue in the development of this generative design system is to investigate how genetic algorithms can be used in knowledge-based design systems as general purposed problem solvers to support conceptual design creatively.

We have identified three tasks in product design process where evolutionary design techniques can be utilized to support the designers:

- generation of abstract design concepts configuring elementary design components that satisfy functional design requirements,
- generation of design embodiments that satisfy geometric and assembly constraints, and
- generation of 3D forms and components for easy manufacturing and assembly.

In our approach, genetic algorithms are implemented as design knowledge sources to be invoked during the design process by the designers to support these three tasks.

The first task is generative in nature. This is to support the generation of a large number of abstract design concepts represented by a set of key functional variables (or key elementary functional components) giving an initial design requirement. This set of key functional variables or functional components defines at an abstract level the structure of the design problem and its conceptual solutions.

The second task deals with the specialization of a particular design concept selected by the designers. An abstract design concept can be specialized by substituting its functional variables or components with 3D geometric components for embodiment and assembly. New design variables can be added to the design concept through a hierarchical knowledge structure representing different levels of complexity in terms of product function, dynamic behavior and assembly etc. To complete the embodiment of an original design concept involves exploring the values of these design variables in order to satisfy embodiment and assembly constraints.

Many alternative designs need to be explored in order to get the optimum solution at the embodiment design stage. However, human designers can hardly cope with them all because many constraints are mathematically, spatially, or logically related to each other. It is difficult to know the consequences of any changes in embodiment design. This is where genetic algorithms can be used to search for those values of the design variables that maximizes the satisfaction of the constraints. However, before genetic algorithms can be invoked, the
constraint network representing the design problem space needs to be partitioned into small areas where certain fitness functions or evaluation criteria can be defined by the designers. This process requires the integration of evolutionary design techniques with knowledge-based design techniques such as symbolic constraint management techniques.

The third task is concerned with the generation of geometric components and parts that are easy to manufacture and assemble. So far the application of evolutionary design techniques in areas such as industrial design has been limited. The difficulty arises from the fact that industrial design requires a variety of knowledge and information at the early creative stage of the design process. This knowledge is difficult to formulate as a knowledge-based or an evolutionary process. This kind of knowledge and information are related to how to determine the function, form, materials, human factors, ergonomics, environmental impact of a product that are difficult to measure in quantitative terms as required in genetic algorithms. In particular, the determination of the form and structure of a product has important implications on manufacturing processes and production cost. Our approach is to develop a generic computer framework in which genetic algorithms are used to support the process of automatically generating geometric forms of products using primitive shells and blobs that are easy to manufacture.

In our generative design system, conceptualizing, learning, designing, evaluating, optimizing and form generating activities are supported by a number of AI methods including machine learning and genetic algorithms in a cooperative and integrated manner. It is argued that this integration is vital for industrial design process in which little is known about the concept, form and structure of the product to be designed. In this system, genetic algorithms are used both as searching tools for design optimization, and as conceptual thinking tools for generating creative design concepts, forms, and structures.

4. Implementation

The computational model of our generative design system is illustrated in Figure 1. In this model of generative design system, three essential design tasks, namely, generate, evaluate, and select, are supported by a generative design toolkit. The toolkit integrates symbolic constraint management and design concept learning techniques with genetic algorithms to support a generative design process within a knowledge-based framework. This generative design process consists of three main stages: design concept generation, design concept clustering and design concept specialization. The process starts at an abstract or general level and becomes more and more specialized.

Abstract functional components and their 3D substitutions are stored in a design knowledge base. Functional components can be randomly selected by the
TOWARDS A GENERATIVE SYSTEM FOR INTELLIGENT DESIGN SUPPORT

system or by the designers to generate design concepts using genetic algorithms or other AI-based generative algorithms. The concepts generated at this stage are not necessarily the optimum ones because only abstract knowledge is used. However, these concepts satisfy the functional requirements specified by the designers. Alternative concepts are also explored and retained.

In order for the designers to evaluate the design meaningfully, the concepts generated at the first stage by the system are clustered into categories or design concept trees with the support of a design concept learning system utilizing inductive concept formation techniques [Tang 1995]. The purpose of design concept learning is to organize the knowledge

Figure 1: A generative design system model
generated so far by the system in a way in which, features and
differences of many similar design concepts are recognized.
This allow designers to focus on interesting conceptual
solutions whilst remain informed of other important alternatives.
At the first stage the process of generating many design
concepts is implicit and automatic. The design concept learning
system compromises this by making the similarity and difference
of all concepts explicit at a selected level of abstraction
determined by the designers.

Any chosen design concept must be further specialized by mapping an
abstract design concept to a product data model that contains actual 3D
components, parts and assembly of the intended design. This mapping from an
abstract level to a detailed level results in new variables and constraints being
introduced. These variables and constraints define a design problem space in
which more alternatives are to be further explored, but at a much more detailed
level involving geometric and spatial relations of the 3D components. Symbolic
constraint management techniques and genetic algorithms are used at this stage
for the effective exploration of this design problem space and optimization of a
final embodiment.

Throughout this generative design process, designers are closed involved in
all the vital decision making processes. For example, designers can select what
functional components are to be used at the concept generation stage. The
genetic algorithms and other AI-based generative programs generate solutions
according to the functional requirements specified by the designers. At the
concept clustering stage, designers can specify clustering heuristics or similarity
measurements for the design concept learning system to generate a design
concept tree in a way to reflect designers’ preferences. That is, designers can
specify which important attributes are to be considered first when clustering the
design concepts. At the design concept specialization stage, designers can define
an optimization goal for a selected part of the design problem space to allow
genetic algorithms being efficiently used. Designers are also responsible for the
selection of a final design solution.

The generative design system resembles the architecture of a knowledge-
based design system in which the generative design toolkit is an integration of
independent and self-contained design knowledge sources. All the knowledge
sources in the toolkit are controlled consistently and can be selected by the
designers at certain stages of the design process. Within this knowledge-based
design system architecture, visualization, simulation, and domain specific design
support systems are be integrated.
5. Applications

Three applications are currently being used as case studies in our development of a generative design system. The first application is in the area of engineering design. In a project called “adding generative capability, database intelligence and rapid prototyping facilities for computer aided design”, genetic algorithms are integrated with a mechanical engineering design system. In this system, genetic algorithms are used to optimize a path for a robot arm to reach certain positions in a 3D space without clashing with any obstacles.

In the second application, genetic algorithms are being used to develop a system that supports design for manufacturability. This system is intended to generate easy assembly components and parts for injection molding by classifying a series of basic structures that can be used as the basis for evolving new shapes and shells.

In the third application, genetic algorithms are being used to support the process of Chinese typeface design. In this application, a number of new processes of Chinese typeface design are being explored. These processes integrate heuristics representing cultural, visual and aesthetic intentions of the designers with generative techniques using genetic algorithms. This allows reversion, recursion, mutation, geometric and schematic transformations to be performed in Chinese typeface design to balance the effects of complexity and simplicity, regularity and randomness, consistency and unpredictability.

6. Conclusions

This paper has reviewed knowledge-based design and evolutionary design in order to combine the two to develop computational models and systems for intelligent design support. A generative model of design integrating AI and evolutionary computing techniques has been presented. A number of research projects being carried out in the School of Design in the Hong Kong Polytechnic University are presented.

Many problems remain to be further studied. These problems are associated with how to formulate creative design tasks at the early stages of the design process so that genetic algorithms and other AI-based generative techniques can be effectively utilized. We have attempted to add generative techniques such as genetic algorithms on a knowledge based design support framework in which evolutionary methods are treated as general problem solvers for essentially two kinds of tasks, conceptual design and design optimization. More research and software development are needed before a generative tool can be added on top of today’s many commercial design systems.
7. Acknowledgements

Our research is supported by a number of grants from Hong Kong Research Grant Committee and the Hong Kong Polytechnic University. We would like to acknowledge David Wilson, Manit Rastogi, Cristiano Ceccato for their contributions to our research on generative design techniques.

References


