Where To From Here?:
Some Recent Computer-Aided Design Research

John S Gero[1)]

Key Centre of Design Computing, University of Sydney, NSW 2006, Australia

ABSTRACT

The current focus on the internet and the world wide web masks some other underlying changes, based on current research, which are likely to occur in our use of computers as design aids. This paper draws on current research to present two possible directions for computer-based design aids. Models of design processes based on analogies with natural evolution are providing a fruitful area for ideas. Concepts from genetic engineering are producing results which indicate that there are useful alternate design processes to those modeled on human design processes. Computer-based design aids have assumed that designing occurs independently of the context in which it occurs. Research into "situated" learning is showing that it is possible to develop design aids that learn. The consequence of this is that the tools become more useful over time. This paper examines some associated representation issues.

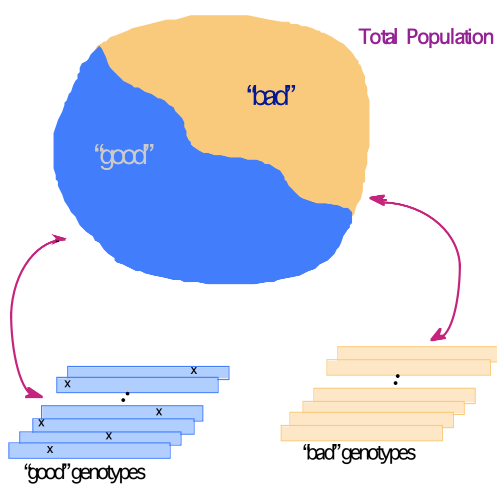
INTRODUCTION

Current computer-aided design systems are built on developments which are founded on research carried out over the last twenty-five years. This has resulted in highly sophisticated computer-aided drafting tools and more recently in computer-aided three-dimensional modeling. Paralleling this work related to graphics has been the development of increasingly powerful analytic approaches based on such techniques as finite element methods and computational fluid dynamics. The most recent development has been the internet and the world wide web as a technology for changing our conception of the availability and transferability of information. This widespread focus on the internet and the world wide web masks some other underlying changes, based on current research, which have the potential to affect our future use of computers as design aids. The remainder of this paper introduces two research directions for the development of computer-based design aids. The first is based on the genetic engineering extension to natural evolution, whilst the second is concerned with the notion of "situation" and its role in the development of design aids.

GENETIC ENGINEERING EXTENSIONS TO DESIGN MODELS BASED ON GENETIC ALGORITHMS

There is already a rich area of research into evolutionary approaches which forms the basis of a design approach which uses the fields of genetic algorithms and genetic programming (Goldberg, 1988, Holland, 1992, Koza, 1992). Genetic algorithms are mathematical models loosely based on Darwinian evolution concepts. They have proven to be very successful in a range of engineering design problems. Genetic engineering in natural systems is the human intervention in natural evolution. Current research, based on genetic engineering concepts, is showing promise in design applications.

Here, an evolutionary model of design is adopted and the assumption is that certain advantageous behaviours of design solutions are caused by unique structures in the genotypes of those designs. Figure 1 demonstrates this concept. The total population of designs is divided into highly performing or "good" designs and into lowly performing or "bad" designs (in practice three groups are used: "good", "neutral" and "bad", with the "good" designs being defined as the top 10% of the population of designs in any given generation). The genotypes of the "good" designs are searched for genetic structures which they all have in common and which do not exist in the genotypes of the "bad" designs. These gene structures are evolved into new genes.



[1)] Professor of Design Science and Co-Director

Fig. 1. Genetic engineering is concerned with locating groups of genes, marked as X in the "good" genotypes, which are causally connected to specified behaviours or performances.

As an example consider Figure 2 which shows 8 genes in the form of shape transition rules. A design is synthesised through an ordered sequence of these rules. Thus, a genotype is made up of a sequence of rule numbers. A design is produced when the rules are executed in the sequence listed in the genotype. An evolutionary system uses crossover and mutation of the genes in the genotype to permutate the sequence and uses evolutionary pressure to move it in the direction of sequences which increasingly perform better and better in terms of some objectives.

Figure 3 shows a set of 10 designs produced using the genes shown in Figure 2. Each of these designs has been put into one of three categories: good, neutral or bad. The genotypes of all the good designs are examined to see if they share some common characteristic. Three of the four good designs in this population of 10 designs exhibit the characteristic in their genotypes that the gene sequence {2, 8, 5} exists, Further, this sequence does not exist in the neutral and bad designs. It is concluded that that gene sequence is a causal contributor to the high performance of those designs and as a consequence the sequence is replaced by an evolved gene whose function is the same as the sequence it replaces.

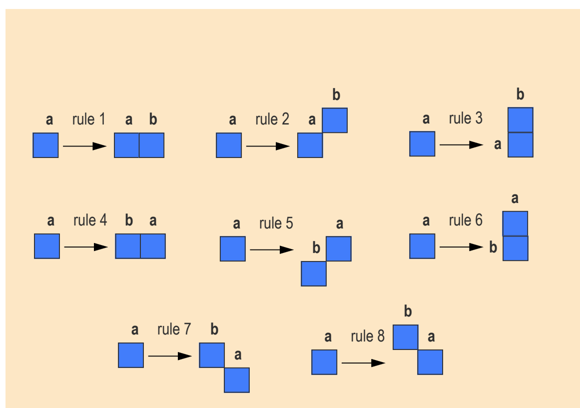


Fig. 2. A set of 8 genes in the form of shape transition rules.

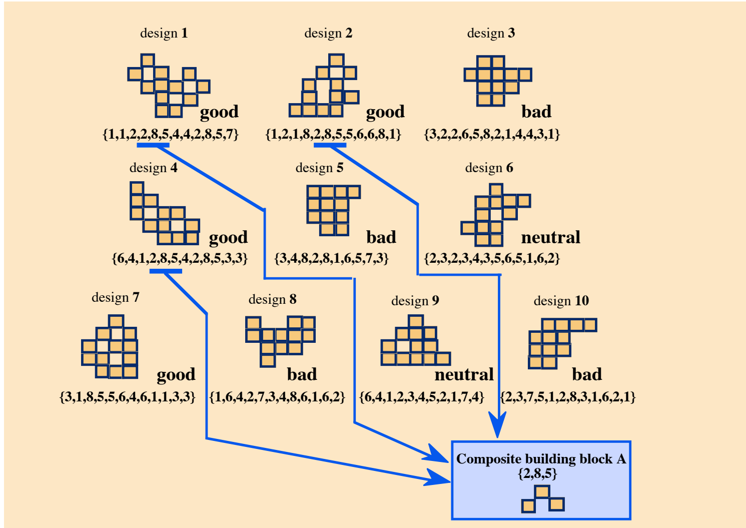


Fig. 3. A set of 10 designs produced with the genes in Figure 2 and evaluated according to their performance in one behaviour. Genetic engineering techniques emerge the gene group {2, 8, 5} as being likely to be the cause of the good performance in that behaviour.

In a similar manner, the gene sequences which are the likely causes of "bad" performance can be found and isolated. The genetic engineer in natural systems has many strategies for manipulating these evolved genes:

radiation therapy which reduces the number of bad genes in natural systems can be modeled in an evolutionary system through the use of high mutation rates;
gene surgery involves the direct removal of bad genes;
gene therapy can be modeled by replacing bad genes with good genes; and
cloning which increases the percentage of good genes in the population of designs.

LAYOUT PLANNING EXAMPLE

A common synthesis problem is the spatial layout planning problem. This is an NP-complete problem which is intrinsically difficult to solve, although it has applications in may domains ranging from VLSI layout, though, production layouts to architectural layouts planning. Typically a set of activities need to be allocated to a set of defined spaces. The activities have a variety of constraints associated with them. These constraints often include area requirements and interaction costs. A good layout is one which allocates all the activities to the available spaces in such a manner as to minimize the cost of activity interaction multiplied by the distance between their locations.

Consider a design with the requirement that the activities be allocated to the four storey building shown in Figure 4. Full details of this problem can be found in Gero and Kazakov (1997a).

Fig. 4. Set of available spaces spread over a four storey building.

If we use a standard evolutionary systems approach we can produce a high quality spatial layout with a good performance. However, if we use the genetic engineering extension to genetic algorithms approach, we not only produce the same result but also evolve sets of genes which contribute to the high quality of the solution. Suppose that after we have proposed a suitable layout the requirements change to allow consideration of an alternate building which has a similar structure but is spread out over only two storeys, Figure 5. Normally this would present itself as a new design problem irrespective of the solution technique employed. It would require a reformulation and would take the same amount of effort to solve as the original problem. However, if the original problem were solved using the genetic engineering extension to genetic algorithms, in addition to the solution a set of evolved genes would have been generated. These evolved genes contain the knowledge involved in the production of the solution. In this case the evolved genes contain sequences of sublayouts which materially contribute the overall highly performing solutions.

Fig. 5. Revised arrangement of available spaces spread over a two storey building.

We are now in a position to compare the two approaches: genetic algorithms with and without genetic engineering extensions. Figure 6 shows a plot of the number of generations that each approach requires to produce the result. The standard genetic algorithm takes, on average, 50 generations to arrive at the same solution that the modified genetic algorithm only takes about 10 generations to produce. This is a saving in computation of 80%, whilst the computational cost of the genetic engineering extension is less than an additional 10% (Gero and Kazakov, 1997b).

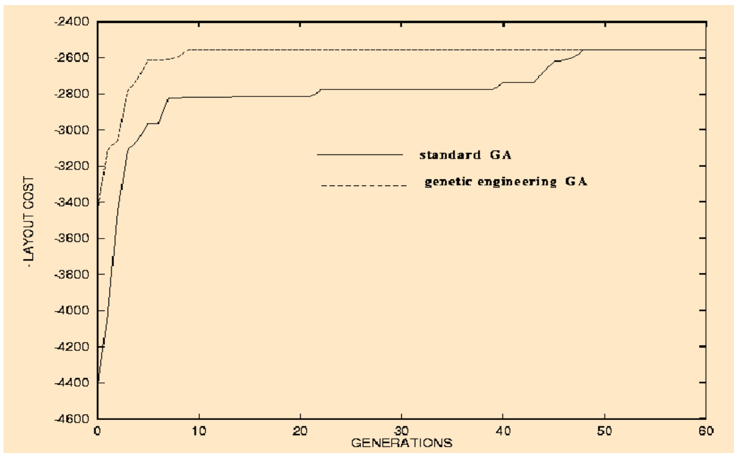


Fig. 6. The effect of using the genetic engineering extension to the standard genetic algorithm in terms of the number of generations required to arrive at the solution.

SITUATED LEARNING AND REPRESENTATIONS

Artificial intelligence has added the concept of learning to our armoury of approaches to computational support for designing. Machine learning for design has largely been concerned with acquiring design knowledge from examples. An implicit assumption behind such work has been that the knowledge is always applicable. However, the situation within which the knowledge is learned plays a role in its future applicability. Thus, in addition to learning design knowledge it is important also to learn the situation so that a basis exists for the reuse of that knowledge. One way to achieve this is to represent situations and contexts explicitly. However, current machine learning design tools do not represent situations and contexts and therefore are situation and context free. This work proposes a situated learning paradigm applicable in designing. In this approach, we borrow and adapt a theory from educational instruction called “situated learning” which states that situations which one experiences as a novice form the developmental conditions of an expert (Ingold, 1995, Brown et al., 1989).

Many authors who refer to situations in design computing as well as in artificial intelligence implicitly use it as a synonym for a state as in AI and problem solving or equivalently as a snapshot of the world. Some researchers like Oki and Lloyd-Smith (1991) and Muller and Pasman (1996) state the importance of situations in terms of applicability of knowledge. The view that we take is that a situation is a partial state composed of a set of a facts that is relative to an agent (Gero and Nath, 1997). Thus, in an agent-oriented world two agents may at the same time be sensing the same world but the situations sensed will be different. What is the situation and what is knowledge relative to the situation is determined by the focus of the agent. This focus of the agent analogically maps onto the figure-ground hypothesis in Gestalt psychology, Figure 7, which states that focus forms a foreground while the rest is the background. This focus also results in interchangeability of the figure and the ground with the restriction that both cannot be focused at the same time. Situations, analogically, are also that part of the relevance field of the agent that is not in focus (background). A situation can also be conceived of as a pattern that is a mechanism for indexing other knowledge or defining the applicability condition of the foreground.

Fig. 7. Is this a white vase on a black background or two black human heads in profile on a white background?

A situated learning environment can be set up using agents as follows:

agents operate in a "world" which can be sensed

each agent's sensors allow it represent that part of the world which is in its "field of relevance"

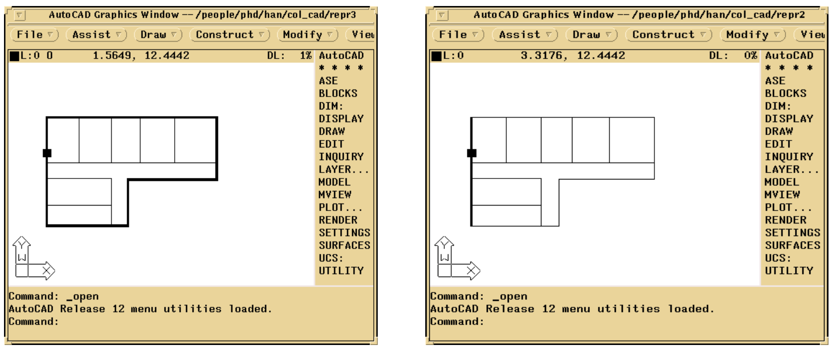
each agent has multiple sensors which provide the opportunity for multiple representations

through its sensors each agent produces a "sensed world"

through its knowledge each agent produces one or more "perceived worlds" which are the sensed world structured

through its knowledge each agent produces one or more "cognated worlds" which are the perceived worlds bifurcated into the situation and focus or knowledge to be learned.

There are two aspects of interest in this approach to situated learning. The first is that the learned knowledge carries with it the situation within which it was learned and the situation is linked to the knowledge which was applicable to it. Thus, the applicability of the knowledge is defined by the situation rather than being assumed to be universally applicable. The second aspect of interest is that what can be learned is not specified a priori as it depends not on what is 'seen' by a human but how it is represented in the computer. Consider the two drawings shown in Figures 8(a) and (b). They both appear to be the same, however, when the designer selects a point on the left hand wall they demonstrate that they are different. Thus, it is likely that what can be learned and what is the situation are richer concepts than first appears if it possible to have multiple representations of objects in computer-aided design systems.



(a) (b)

Fig. 8. Different representations of the same drawing in a computer-aided design system. Selecting the same point does not result in the same set of lines being selected. The resulting selected lines in (a) and (b) are a function of the representation used and not of the image as it appears. The selected line are highlighted by thicker lines (Jun and Gero, 1997).

Consider the image of a square in the upper part of Figure 9. There are multiple representations possible, each of which produces the dame image of the square but which have quite different meanings. For example, the square could be represented as a triangle which has been reflected around a diagonal. With this representation what is learned may be concerned with reflectional symmetry. The significance of this lies in the exploratory nature of conceptual designing where the designer is still looking for ideas and does not want to be restricted by the implicit view of the computer program's implementer. The computer program implementer's view is often only determined by the users as they try to use the system and empirically discover what the restrictions implied by the implementer's approach are.

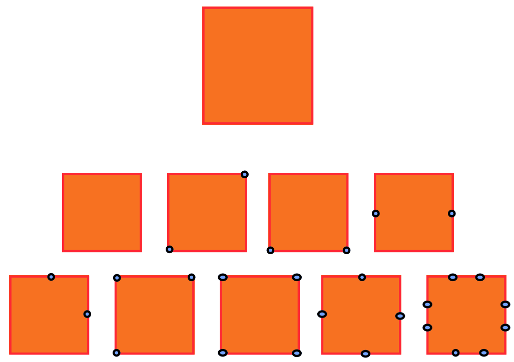


Fig. 9. Multiple representations of the image of a square.

Thus, what can be learned includes both explicit and implicit or emergent knowledge and situations, which adds to the richness of the capacity of the system. The effect of this is that the range of applicability of the knowledge useful at the conceptual stage of designing is extended beyond the direct machine learning approaches. In addition to emergent concepts derived from multiple representations, at the conceptual design stage a designer is unlikely to have finalised all aspects of even the outline of a design. There is likely still to be considerable ambiguity about what is being drawn. Current CAD systems force precision if any form of processing is to be carried out on the drawings since most computational processes assume such precision in the representation. However, this demand for precision is inimical to exploration as occurs at this early stage of designing. What is needed is ambiguous representations which can be made increasingly precise. The effect of such an ambiguous representation may be to provide the opportunity for designers with their design support tools to move in directions not easily found without such ambiguity.

Consider the triangle in Figure 10 which has been represented through sets of competing edgelets in a neural network rather than as three edge vectors. The competing edgelet representation allows for ambiguity in the sense that multiple interpretations of the representation are possible. Thus, instead of the drawing being represented uniquely and only one kind of knowledge and situation being learned the drawing can be viewed as a class of drawings from which multiple knowledge could be learned.

Fig. 10. Ambiguous representation of a triangle (Tomlinson and Gero, 1997).

DISCUSSION

The history of the use of computers in civil and building engineering shows how difficult it is to predict which application areas will develop swiftly. Often the distance between the time the research is done and its application can be large. For example the foundational research for the current programs on 3-D modeling was done 25 or more years ago. Similarly, the research on which finite element programs is founded was done 20 years prior to them becoming popular. More recently, though, this time lag has dramatically shortened. The widespread use of spreadsheets in engineering occurred less than a decade after the foundational research and more recently the changes that the availability of the world wide web are introducing today are beginning to occur only five since the introduction of the world wide web. Although it appears that for applications which are specific to the use of computers in civil and building engineering the time lag has not shortened as dramatically as for general purpose, widely used applications.

Current computer-aided design research provides potential directions for the development of design support tools. Although much of today's activity is focussed on how we can best make use of the world wide web as an information access and delivery vehicle, there are considerable long term benefits to be gained if we can improve the performance of designers and the quality of design in civil and building engineering. Although many of our computer-based support tools are founded on models of how we think human designers work the vast majority of analysis tools are not built using this idea. Recently, tools to aid designers in aspects of their synthesis tasks have been developed which also do not rely on models of how human designers carry out these tasks.

The research related to evolutionary systems does not attempt to model any human activity, whereas the research on situated learning and multiple representations gains its impetus from examining how humans design. It is likely that results from such forms of research will form the basis of future design support tools in civil and building engineering. These future tools are likely to be more "active" than our current "passive" tools.

ACKNOWLEDGMENTS

The work described in this paper has been supported by grants from the Australian Research Council, from Australian Postgraduate Awards, and Overseas Postgraduate Research Awards. Computing resources have been provided by the Key Centre of Design Computing. This work has been carried out in conjunction with Vladimir Kazakov, Gourab Nath and Phillip Tomlinson.

REFERENCES

Brown, J., Collins, A. and Duguid, S. (1989), "Situated cognition and the culture of learning", *Educational Researcher*, **18**(1), 32-42.

Gero, J. S. and Kazakov, V. (1997a), "Evolving design genes in space layout problems", *Artificial Intelligence in Engineering* (to appear).

Gero, J. S. and Kazakov, V. (1997b), "Learning and reusing information in space layout planning problems using genetic engineering", *Artificial Intelligence in Engineering* (to appear).

Gero, J. S. and Nath, G. (1997), "Formalising situtated learning in agent-based design", *CAAD Futures'97,* Munich, Germany (to appear)

Holland, J. (1992), *Adaptation in Natural and Artificial Systems,* MIT Press, Cambridge, Massachusetts.

Ingold, T. (1995), "Lecture at workshop on situated learning within post secondary education", http://www.dur.ac.uk/~dps8zz2/Lave/TimIngold.html.

Jun, H. and Gero, J. S. (1997), "Representation, re-representation and emergence in collaborative computer-aided design", *in* Maher, M. L., Gero, J. S. and Sudweeks, F. (eds), *Preprints Formal Aspects of Collaborative Computer-Aided Design,* Key Centre of Design Computing, University of Sydney, Sydney, pp. 303-319.

Koza, J. (1992), *Genetic Programming,* MIT Press, Cambridge, Massachusetts.

Tomlinson, P. and Gero, J. S. (1997), "Emergent shape generation in design using the boundary contour method", *CAAD Futures'97,* Munich, Germany (to appear).

This paper is a copy of: Gero, J. S. (1997). Where to from here?: some recent computer-aided design research, *in* C-K. Choi, C-B. Yun and H-K. Kwak (eds), *Proceedings Seventh International Conference on Computing in Civil and Building Engineering,* Techno-Press, Taejon, Korea, pp.29-37.