WHAT ARE WE LEARNING FROM DESIGNERS AND ITS ROLE IN FUTURE CAAD TOOLS

JOHN S GERO
Key Centre of Design Computing
University of Sydney

Recent research into the activity and behaviour of human designers as they design has provide an impetus to carry out research which underpins the development of new CAAD support tools. However, there are computational processes of interest in designing which are not modeled on human design processes. This paper outlines some of the design processes which are being researched based on our understanding of human designers and provides examples from some early implementations.

1. Introduction

Given the large body of design research it is surprising how little we know about designing: the activity carried out by designers. There has been an upsurge in interest in studying human designers formally. Much of this interest has been driven by the efforts of the artificial intelligence in design research community, which has developed a range of computer-implementable designing processes (Gero and Sudweeks, 1994; 1996). However, more recently, studying human designers has been founded on the development and later formalisation of experimental methods based on protocol analysis (Ericsson and Simon, 1993; van Someren et al., 1994; Cross et al., 1996).

Protocol analysis provides a rich source of information on designing as a time-based activity. This then allows design researchers to develop richer models of designing based on the behaviour of human designers. These models in turn provide the basis for a better understanding of designing. Such an understanding then feeds into the development of computer-based support tools for computer-aided architectural design.

Section 2 briefly introduces some results from studying human designers. Section 3 proceeds to describe a computer implementation of a model of designing based on developing cross-domain analogies drawn from the putative behaviour of human designers. Section 4 introduces the notion ambiguity and emergence in images and describes a computer implementation of a model of designing based on utilising such notions.

2. Results from Protocol and Related Studies

Protocol studies are a means of obtaining data from verbal utterances. Designers are asked to "think aloud" while they are designing. While they are designing they are video and audio taped. The designer's verbal utterances are transcribed. The transcription is then used, along with design theory, to develop a coding scheme. The transcription in then coded and finally analysed. The step are listed below:

- taping
- transcription
- code development
- coding
- analysis

Protocol studies are providing detailed evidence of how designers spend their time as they are designing. At a gross level a designer's time can be spent either on postulating solutions, called structure, or in reasoning about the function and behaviour of possible or postulated designs. Figure 1 shows a typical distribution of the time spent between these two large classes of activities by experienced designers. It is interesting to note that it is almost twenty minutes into the session, for this design, before any structure is proposed.

Such behaviour is in contrast to that of inexperienced designers who appear to need to "put pencil to paper" very early in order to have something to work with. This is exemplified in Figure 2.

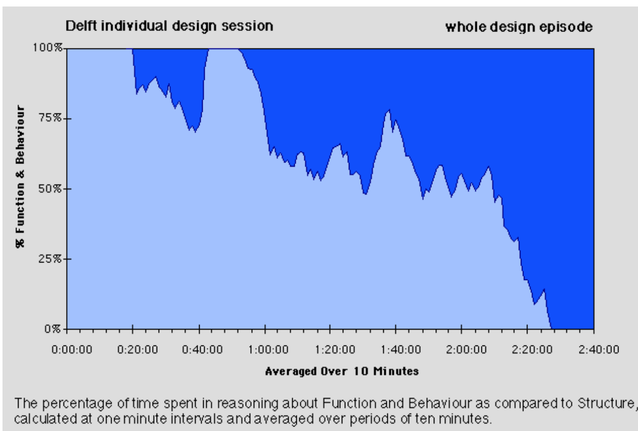


Figure 1. Typical plot of distribution of time spent on function and behaviour (light), as against structure (dark), for experienced designer (Gero and McNeill, 1997).

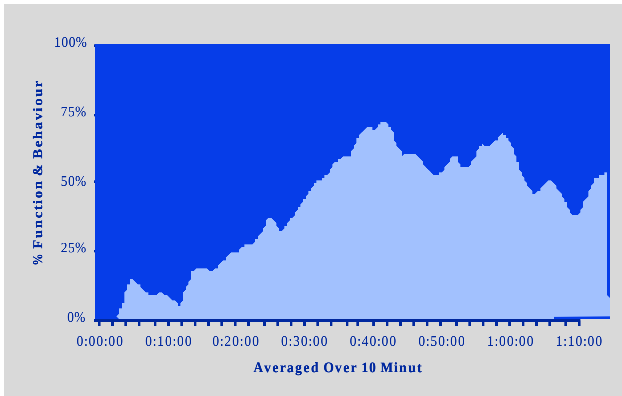


Figure 2. Typical plot of distribution of time spent on function and behaviour (light), as against structure (dark), for inexperienced designers (Gero and McNeill, 1997).

Such studies of designers support previous cognitive studies of human problem solving in areas such as analogy, fixation, emergence and visual ambiguity. They provide detailed information on how designers use such concepts in the development of their design proposals.

3. Using Cross-Domain Analogies

Analogy has long been recognised as an approach in the elicitation of design ideas. In architecture, the most obvious analogies are visual analogies. Here, the surface similarities of the structure are used and transferred to a new design, as shown in Figure 3. However, deep rather than surface similarities require a different approach to their recognition. Similarities associated with function and behaviour provide the locus for deep similarities. While function specifies what a design does, behaviour specifies how it achieves it functions. We may extend the definition of analogous designs to:

> “Two designs are analogous if they have a similar function or behaviour; they may or may not have a similar structure” (Qian and Gero, 1996).

Thus, the aim of analogy-based design is to obtain new ideas of possible structures from an existing design with similar functions and/or behaviours. The use of deep similarities has the potential to allow analogies to be drawn not just from designs in the same domain but from designs in quite different and apparently unrelated domains. It appears that humans may reason about function and behaviour in order to draw cross-domain analogies if we subscribe to the function-behaviour-structure (FBS) characterisation of designing (Gero, 1990).

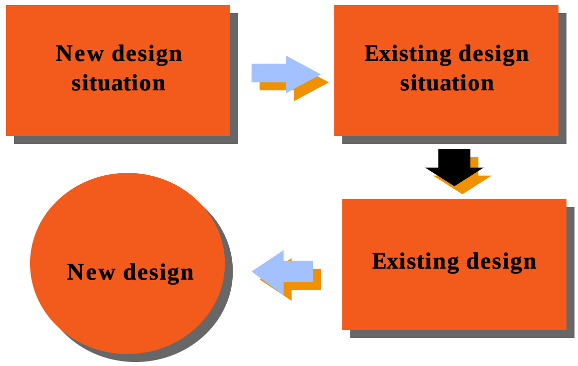


Figure 3. Model of design by analogy based on concepts derived from human designers.

There are at least four modes of reasoning about function (Qian and Gero, 1996):

FBS Type I: achieved by static behaviour, Figure 4
FBS Type II: achieved by a state of dynamic behaviour, Figure 4
FBS Type III: achieved by a set of behaviours occurring contemporaneously, Figure 5
FBS Type IV: achieved by a set of behaviours occurring sequentially, Figure 5.

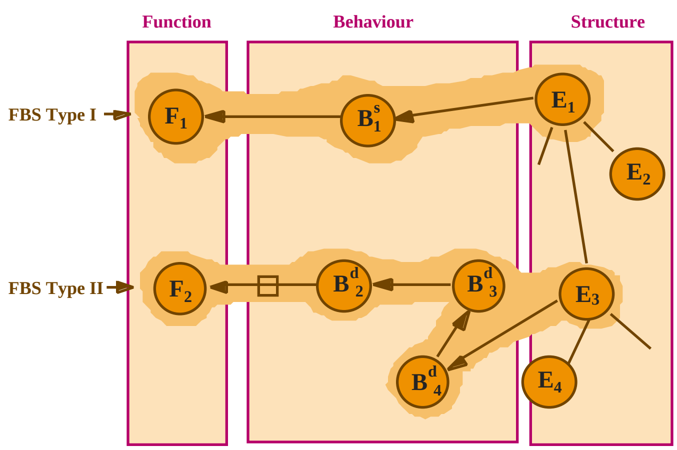


Figure 4. Achieving functions by behaviour paths Types I and II.

The structure is composed of elements, E, from which behaviours, B, are derived. In turn, the functions, F, are derived through the behaviours. The role of the different modes of reasoning about function is to develop a basis for looking for similarities amongst designs which are dissimilar on the surface or structure level.

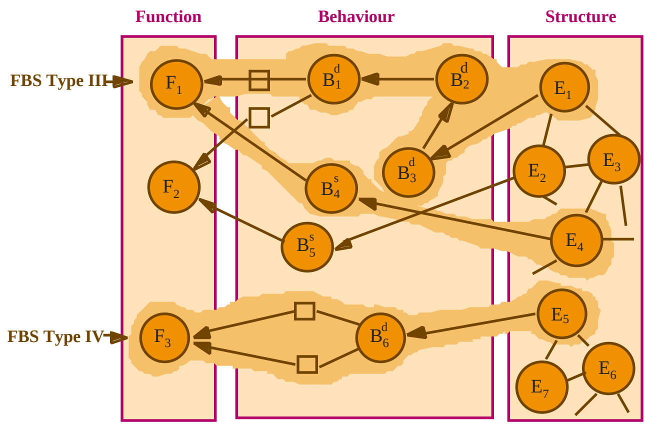


Figure 5. Achieving functions by behaviour paths Types III and IV.

A computer-aided design system built using this paradigm demonstrates in Figures 6 and 7 that it produces cross-domain analogy conjectures. Figure 6 shows that in redesigning doors the system has made a conjectured analogy with the design of an existing domestic water tap. From this figure the proposal is unclear, however, in Figure 7 the proposed analogy is clarified at the structure level. The conjecture is that the hinges of the door could be replaced by the screw mechanism of the water tap.

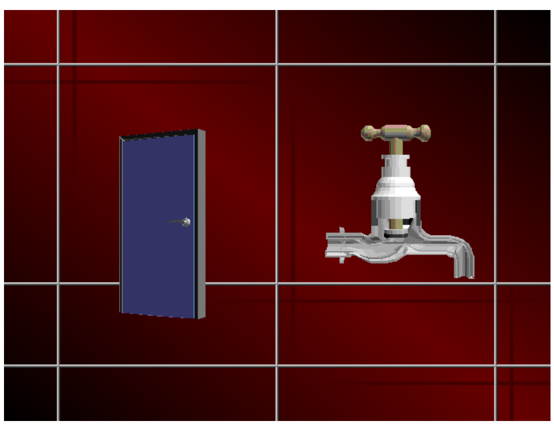


Figure 6. Suggestion by computer system when designing a door that a design analogy could be drawn with a water tap.

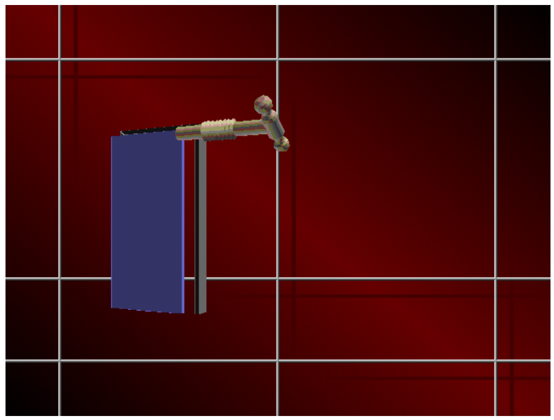


Figure 7. Proposal by computer system to replace door hinges by water tap screw mechanism to produce a new door design.

Figure 8 shows in graphical form the system's reasoning behind the conjecture. This graphical form is derived from a reasoning process concerning the ways both designs achieve their functions and then examining their respective behaviour paths. From this isomorphism the structure mapping is generated. It is only the structure mapping that is shown in Figure 8, ie the door's door frame maps onto the tap's pipe as fixed elements; the movable door leaf maps onto the movable valve; and the hinges that produce rotation which controls the amount of door opening map onto the screw that produces linear motion which controls the amount of tap opening.

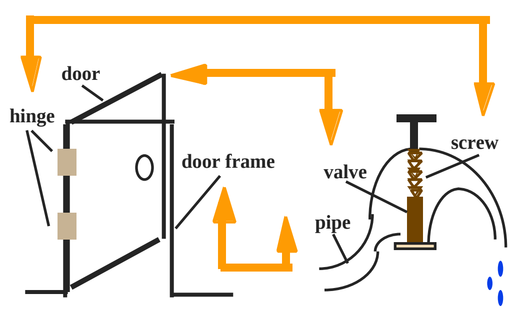


Figure 8. Explanation by computer system, in graphical form, of how the door hinges can be replaced by the water tap screw (after Qian and Gero, 1995).

4. Using Ambiguities and Emergence

Architects when they are designing use shapes to outline the spaces of interest. At the conceptual stage of designing whilst the designer is still exploring possibilities, it is important that the shapes not be fixed and unique but rather that they be open to a variety of interpretations (Schön and Wiggins, 1992). Some artists have used the concept of multiple interpretations as the basis of their work. One of the most effective artists of this genre is M. C. Escher, Figure 9. There are many interpretations possible of this image: white angels on a black background; black 'devils' on a white background; rotationally symmetric shapes; scaled shapes; and so on. Each of these interpretations may be viewed as *emergent.* The interpretations are a result of ambiguities in the representation.

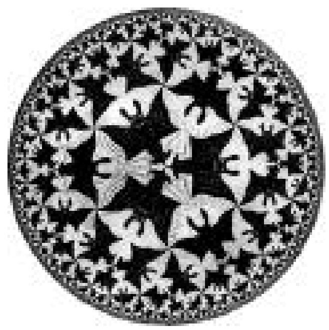
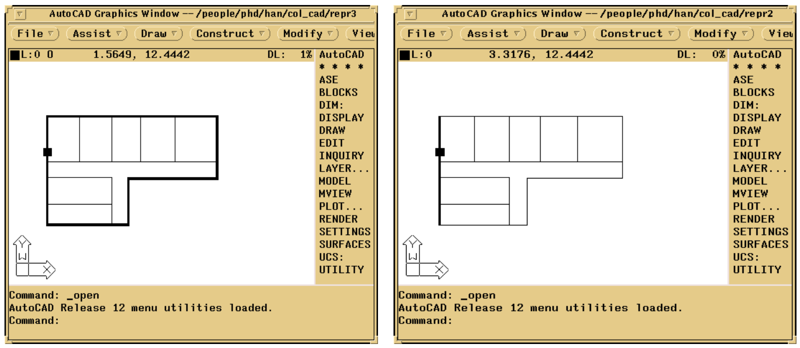


Figure 9. M. C. Escher's Circle Limit IV.

Figure 10 demonstrates this issue. Both the drawings in Figures 10(a) and (b), produced using AutoCAD, appear to be the same, however, selecting the same point in each drawing does not result in the same lines be selected. The actual lines that are consequentially selected depend entirely on the underlying representation and not on the image of the drawing. In paper and pen 'representations' of drawings this issue does not arise since human designers construct in their heads the representation necessary to support the activity they wish to carry out.

The need to support a variety of interpretations creates new requirements for CAAD systems. These new requirements change the expected behaviour of any computational system which is used to support designing. Such computational systems must have the capacity to provide multiple representations of the objects of interest and have the capacity to allow for ambiguous representations. The difference between multiple and ambiguous representations is that with multiple representations there is a set of unique representations for the image, each representation may be used directly. With ambiguous representations there is an interpretation phase required before the representation can be used. Figure 11 shows an example of multiple representations of a square. Each of the representations can be used differently.



(a) (b)

Figure 10. 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(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).

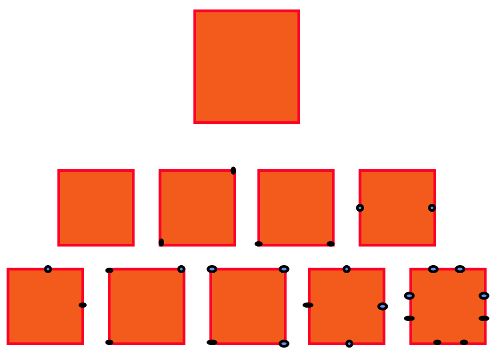


Figure 11. Multiple representations of the image of a square.

Figure 12 shows an example of an ambiguous representation of a triangle, where the triangle is represented by numerous competing ‘edgelets’. With such a representation there is a class of possible triangles represented rather than just the single instance which was drawn initially. The effect of representing classes instead of instances that multiple instances can be derived from the class. This plays an important role at the early stages of designing when the designer does not want to be specific about the details and wants to work at the general level. Class based representations provide an opportunity for this to occur. Previously, it was considered that ambiguous representations should be avoided but it appears that they have a useful role to play in designing.

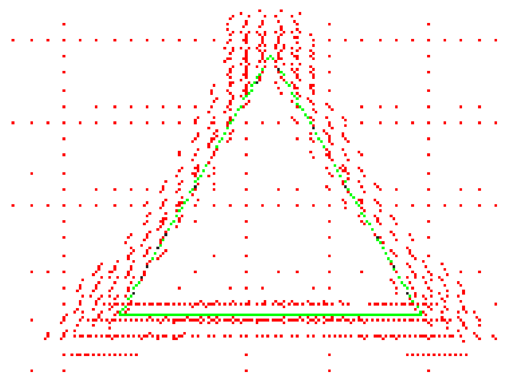


Figure 12. Ambiguous representation (Tomlinson and Gero, 1997).

There are a number of class-based representations being developed. The most common is based on the object-oriented paradigm, although this has some difficulties. Other approaches are based on qualitative representations.

5. Discussion

Recent research into how humans design is providing a rich source of both well-supported as well as anecdotal evidence on which to base computational design paradigms. The effect of utilising such as an approach is not to model human designing processes but rather to use such processes as the bases of analogies for the development of computational processes. Humans are not computers so we should not expect that computational design processes would be models of human designing processes. As we attempt to develop such computational tools we are also developing a better understanding of designing as an activity. We are not limited, however, to basing our tools on analogies with human designing processes. We can construct computationally-based tools founded on conjectures with processes unrelated to human designing: evolutionary systems represent one such approach. We can look to other concepts unrelated to such conjectures and postulate processes founded of axioms and derivations from those axioms.

Acknowledgments

The research described here has been supported by grants from the Australian Research Council, the Australian Postgraduate Awards, the Overseas Postgraduate Awards and the University Research Grant. Computing resources have been provided by the Key Centre of Design Computing.

References

Cross, N., Christiaans, H. and Dorst, K. (eds) (1996) Analysing Design Activity, Wiley, Chichester.

Ericsson, K. and Simon, H. A. (1993) Protocol Analysis: Verbal Reports as Data, MIT Press, Cambridge, Massachusetts.

Gero, J. S. (1990) Design prototypes: a knowledge representation schema for design, AI Magazine 11(4): 26-36.

Gero, J. S. and McNeill, T. (1997) An approach to the analysis of design protocols, Design Studies (to appear).

Gero, J. S. and Sudweeks, F. (eds) (1994) Artificial Intelligence in Design'94, Kluwer, Dordrecht.

Gero, J. S. and Sudweeks, F. (eds) (1996) Artificial Intelligence in Design'96, Kluwer, Dordrecht.

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.

Qian, L. and Gero J. S. (1995) An approach to design exploration using analogy, in J. S. Gero, M. L. Maher and F. Sudweeks (eds), Preprints Computational Models of Creative Design, Key Centre of Design Computing, Department of Architectural and Design Science, University of Sydney, Australia, pp. 3-36.

Qian, L. and Gero, J. S. (1996) Function-behavior-structure paths and their role in analogy-based design, AIEDAM 10: 289-312.

Schön, D. and Wiggins, G. (1992) Kinds of seeing and their functions in designing, Design Studies 13(2): 135-156.

Tomlinson, P. and Gero, J. S. (1997) Emergent shape generation in design using the boundary contour method, CAAD Futures'97 (this proceedings).

van Someren, M., Barnard, Y. and Sandberth, J. (1994) The Think Aloud Method: A Practical Guide to Modelling Cognitive Processes, Academic Press, London.

This paper is a copy of: Gero, J. S. (1997) What are we learning from designers and its role in future CAAD tools, *in* R. Junge (ed.), *CAADFutures 1997*, Kluwer, Dordrecht. pp. 61-70.