

Not only computing – also art

JOHN LANSDOWN

Down under

During the summer I spent a pleasant month as a Senior Research Fellow at the Department of Architectural Science in Sydney University. Among many other things, the Department, under the indefatigable Professor John Gero, is doing pioneering work on the use of Prolog for developing architectural design programs. Prolog is a different type of language to the ones most of us are used to and allows the very easy programming of logical statements and inferences. As Clocksin and Mellish say in the preface to their book, *Programming in Prolog* (Springer-Verlag 1981): '... the task of writing a Prolog program is not like specifying an algorithm ... in a conventional programming language. Instead, the Prolog programmer asks more what formal relationships and objects occur in his problem, and what relationships are "true" about the desired solution. ... The Prolog approach is rather to describe known facts and relationships about a problem, than to prescribe the sequence of steps taken by a computer to solve the problem.' This is equivalent to saying that, in Prolog, we tell the computer what to know rather than what to do. Thus, we might express the concept of connectivity of rooms by a set of statements of the form:

connects (dining-room, kitchen).
connects (guests-lounge, lobby).

and so on. We might, further, express the idea of 'reachability' by the statements:

reach (x, y): connects (x, y).
reach (x, z): connects (x, y).
 reach (y, z).

The first of these statements means: 'You can reach x from y if x connects with y'. The second is a recursive definition and means: 'You can reach x from z if x connects with y and it is possible to reach y from z'.

By using such statements and developments from these a PhD candidate, Tuncer Akiner, has written a program to question the computer about aspects of the design of a small (but real) hotel. The sort of questions that can be asked are: 'What rooms are on the same level as the terrace? Where is the kitchen and how can it be reached from the lobby? What rooms are adjacent to the lounge? Which spaces have direct access to the stairs?' and so on. The main advantage of such an approach to programming is that it allows us to create basic descriptions of complex objects and to infer novel facts from them – an essential feature if we are to write intelligent programs

to assist the designer. Programs such as Tuncer's could assist in such things as the automatic analysis of plans from the point of view of security or means of escape in case of fire; checking the suitability of layouts for disabled persons and minimising pipe and duct runs (remember that, in a modern building the services can account for more than half the total cost of the scheme!)

Capability Coyne

Another doctoral student, Richard Coyne, is using Prolog to experiment

with Expert Systems as an aid to building design. This work is at too early a stage to deal with in any detail but, from what he has done so far, it is clear that there is considerable promise. Richard is both an architect and a landscape architect and his Master's thesis, written when he was at Melbourne University in 1982, concerns the use of computers in landscape design. Landscape architects generally are not particularly well served with computer programs and Richard deals in his thesis with a program, VISIT, which he has written to assist designers in visualising sites. As he says in his introduction, 'VISIT is essentially a drafting tool intended to free the landscape designer from pouring energy into renderings at the early stages of the design process, and to enable the speedy generation of several solutions which can be depicted three-dimensionally'. VISIT is a suite of eight Basic programs designed to run interactively on a Tektronix 4054 desktop graphics system and can be used to produce views of terrain

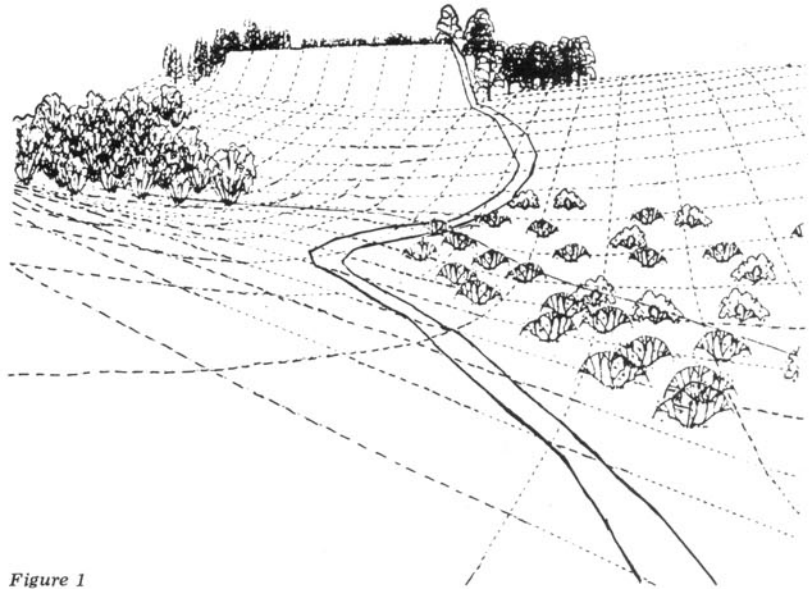


Figure 1



Figure 2

containing vegetation as well as linear features such as roads, hedges and so on. Typical views are illustrated in Figures 1 and 2. I think we will hear a lot more of Richard's work in the coming years.

Points of view

Coincidentally, just before I left for Australia, I had a visit from an antipodean artist, Jeffrey Shaw and his colleague, Fabienne de Quasa Riera, both of whom now work in Holland and use computers in their work. Fabienne uses a video frame grabber and an Apple II to create pixel graphics of the type shown in Figure 3. A series of pictures such as this were included in her recent Amsterdam exhibition called *Dell' Inferno di Dante*. Jeffrey



Figure 3

also had exhibits in Holland this year and displayed a real-time computer piece called *Points of View (1)* which is described in the catalogue as an 'interactive audio-visual theatre work'. However, instead of a real stage and actors, the protagonists of the play are computer generated symbols projected on to a large screen by means of a video projector. The symbols, which derive from Egyptian hieroglyphics and have a meaning, are set on a drawn stage in the manner of Figure 4. The audience's point of view of this graphic representation is changeable because the projected image can move in three-dimensional space allowing the whole to be seen at any angle and from distant view to close-up. The movement from one point of view to another is controlled by a member of the audience who sits in one of the chairs in the room (Figure 5) which has two control devices: the left one moving the image forwards and backwards; the right one controlling the angle of view. In effect,

this member of the audience becomes the director of the work for the time being and, by virtue of the fact that the controllers also affect the sound, is able to choose the way in which the performance progresses.

The sound is just as important as the visual representation and it, too, expresses different points of view in that, depending on the position of the controllers, one or more of 15 different recorded sounds will be heard. These include works by Artaud, Marinetti, Rilke, R. D. Laing, selections from the

Bhagavad Gita, music, and childrens' voices. As the catalogue of one of the performances says: 'As a total work this piece explores contrasting attitudes to reality as a consequence of the activity of taking different positions to look at it'. The fact that Jeffrey only has access to simple hardware makes the graphical output somewhat limited but it is clear that this work exploits the capabilities of the computer in a special way. I look forward to further developments.

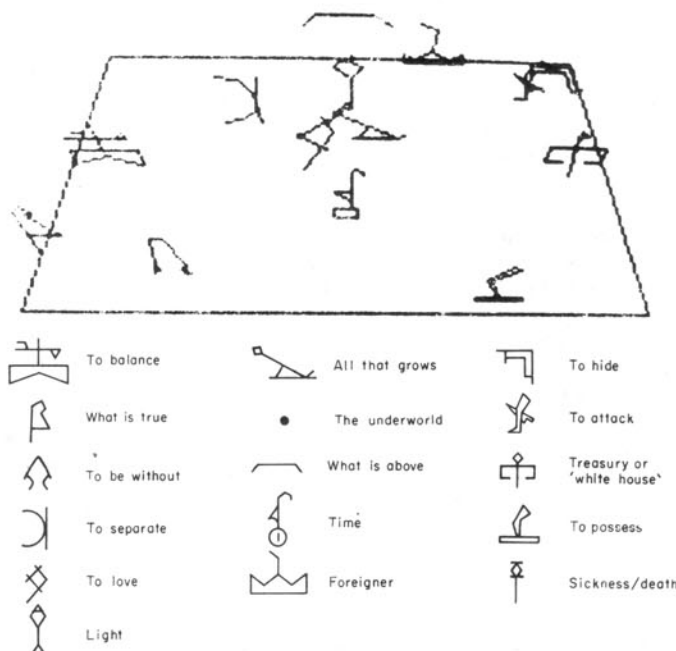


Figure 4 The projected computer graphic image represents a stage with 15 characters. Each character is a sign derived from the Egyptian alphabet; the meanings of these signs are shown above.

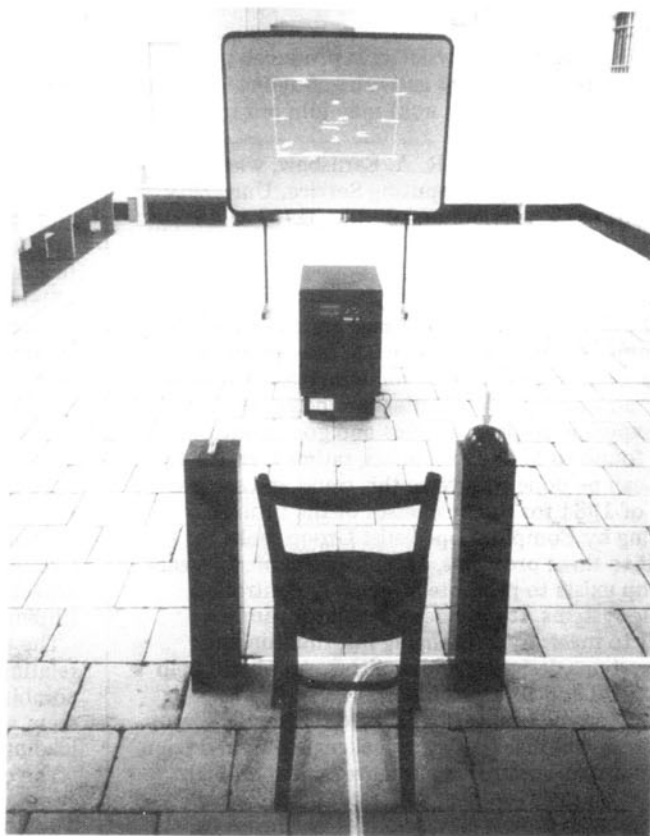


Figure 5