Not quite computing—almost art

JOHN LANSDOWN

Setting your house in order

In *Computer Bulletin* for June 1975 I outlined an intriguing arrangement problem which has been puzzling one of the painter-members of the Computer Arts Society for some time. Essentially the problem is to lay out a single-storey building of 15 rooms, each of which has a different configuration of between one and four doors. Whilst no-one has yet come forward with an algorithm which efficiently and exhaustively enumerates the possible solutions to the problem, Ted Allwood, an MSc student in the School of Mathematics, Computing and Statistics of Leicester Polytechnic has sent me a description of an interesting program he has written to help find solutions interactively. Called $HOUSE, the program is in Fortran and runs on a Honeywell DDP-516 machine with a storage screen display and joystick input. There are two modes of running: manual, in which the user builds up the solution on the screen just as he would with pencil and paper but with the added advantage that the computer is able to keep track of the solutions and list out the rooms still to be used whilst performing elementary evaluations on the solutions thus far; and automatic, in which the program attempts to derive a solution algorithmically.

In the automatic mode, a generative procedure is used in which rooms are added one at a time to the plan. At each stage the unplaced room with the most doors is selected and placed in a position such that the total number of doors on the plan is minimised. This procedure continues until either a solution is found or some impasse arises. In this latter case, the program backtracks by random deletion of rooms from the edges of the plan and the procedure re-starts. I think this interesting procedure deserves further study but that other problem-solving strategies should also be incorporated. In addition, it would be useful if the manual mode could incorporate a learning procedure to see if improved algorithms can be derived from the way in which people tackle what is, surprisingly, quite a difficult problem.

I hear you calling me

The role of the computer in human voice synthesis has been mentioned before in these columns and some of the techniques devised by workers have now found their way into digital calculator design. A new hand-held calculator available in the United States and manufactured for the American Foundation for the Blind, New York, has a speaking vocabulary of twenty-four digitally synthesised words which are used to confirm the key-stroke entries and to read out the answers.

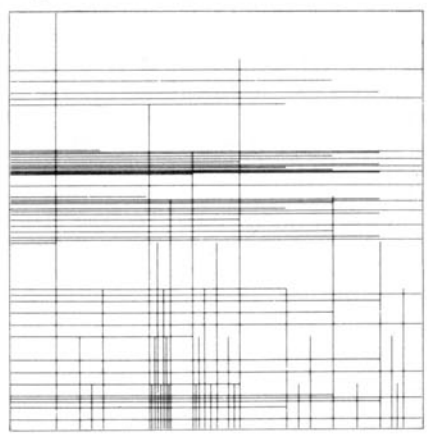
The voice synthesis is carried out by a software algorithm incorporated into an integrated circuit and the simulation of speech sounds derives from analogue voltages converted from digital signals. I have not seen or heard the device, but, if it performs as advertised, it represents a considerable breakthrough both in voice synthesis technology and aids for the disabled. One can conceive of a large number of other applications for such technology not only for the disabled but also more generally and I look forward with interest to further developments.

I know the voice but not the face

Associated with investigations into digital voice synthesis, Frederic Parke, Assistant Professor at the Case Western Reserve University, Ohio, has developed a computer program which models the human head including lip and eye movements and outputs a realistic, full tone picture onto a VDU (Figure 2). The model assumes that the face is made up of several sections some of which are affected by speech or facial expression and, in order to simplify computation, that it is symmetrical so that one side is the mirror image of the other. (This is not the case in real life as you can readily check if you take a photograph of your face and view it with a mirror on its centre lines so that you can see just half the photograph and its mirror image.)

The polygonal skin of one half of the face is assumed to be made up of 250 facets defined by 287 vertices and each eyeball has 50 facets defined by 54 vertices. The manipulation of the face is confined to the eyes, eyelids, eyebrows, lips and jaw so that it is possible to move the eyes, dilate the pupils, arch the eyebrows, open and close the eyelids.

Figure 1



Conventional film animators discovered in the early days of talkies that only four basic lip movements are necessary for successful simulation of speech movement. These are

open lips for the sounds of a, e and i;
close lips for the sounds of b, m and p;
oval mouth for u, o and w; and
lower lip under front teeth for f and v

By using a photogrammatic technique and a data tablet, Parke is able to input details of faces into his program quite readily and it is clear from the photographs I've seen that his animation technique could give a fillip to the creative use of computers.

I is for informatique

A beautifully produced magazine which deserves to be more widely known in Britain is *IBM-Informatique*, published, in French, by Compagnie IBM France. A recent issue, No 13, is wholly devoted to computer graphics and shows, in colour and black and white, a considerable cross-section of recent work in this prolific field. Almost all the well known names are represented, American, European, Japanese and Canadian. The work of British computer artists is less fully covered—for two reasons: firstly, because over here we tend not to specialise in graphics; indeed, except for computer animation, this is an area almost totally neglected by British workers and secondly, because those of us who were asked to write contributions on our specialties failed to do so! As one of the guilty parties I can only conclude that some sort of death-wish is on the British worker.

Alan Sutcliffe's *Skip and Divide* is shown (Figure 1). The two-dimensional result of a clever procedure to subdivide a line, this is an interesting example of procedural art where the means of arriving at a design are just as important as the design itself. Informally, Alan's procedure for subdividing a line A–B is

1 Start with the whole line segment at A;
2 Divide the segment to the right into two, mark the point of division and move to the rightmost end of the segment;
3 If you have finished, stop; otherwise
4 If you are at B, move to A;
5 'Skip' to next point on the line;
6 If you are at B, move to A then go to 2
7 Otherwise go to 2

A PRIDE OF PIONEERS

The photograph immediately below was taken at the opening of the Science Museum's Mathematics and Computing Gallery in December 1975.
Back row (l to r): D. Davies, T. H. Flowers, Grace Hopper, J. H. Wilkinson, T. Kilburn, T. R. Thompson (since decd), M. V. Wilkes, C. P. Marks, A. W. Coombs
Front row (l to r): Mrs E. Hartree, F. C. Williams, M. H. A. Newman, D. J. Wheeler, K. Zuse.

If you try this procedure by hand (or program it) you will see that your line will be subdivided into sections of varying lengths exhibiting the 'bunching' characteristics of many random phenomena procedures but with an overall ordered appearance. Such algorithms are of great interest to computer artists in that they seem to result in patterns which are similar in many ways to those arrived at intuitively. A surprising feature of this procedure is that, quite coincidentally, it turns out on close analysis to be similar to a procedure used by the Dutch computer artist, Lambert Meertens to produce music. In this case, of course, musical intervals rather than line segments are subdivided. A string quartet composed using this procedure won one of the prizes in the Computer Composed Music Competition at Edinburgh IFIP in 1968.

Happy endings

The interest that many computer artists have in devising art works which are somehow responsive to their environments has led to such fascinating devices as *Green Music* and the *Senster*, both of which have been described previously in these columns. A work which, until a day or two ago, was unknown to me however, is a play called *The End*, written by David Edgar and given its first performance by the Bradford University Drama Group in 1972 under the direction of Chris Parr.

On the subject of the dangers of nuclear war, the play was conventionally written except that, from time to time, the action froze while members of the audience were asked to decide a problem relevant to the action posed by one of the actors. The resulting decisions were then fed, *via* an on-stage teletype, to a remote time-sharing computer programmed to determine ultimately which of two optional endings the actors should then perform. Whilst having only two possible endings fails, I think, to exploit the full potential of this promising idea, the concept of the audience participating in the creative process in an oblique way is one which is particularly appealing to me and, with the coming of more readily accessible mini and microcomputers, is likely to be a source of important dramatic developments in the future. The computer was programmed by the brothers Geoff and Brian Wyvill, who have developed a new graphic language for artists, about which more next time.

Figure 2