present value has therefore been substituted for NPV in this comment. Secondly, there is some ambiguity in the references to a system's capital cost, this is referred to as the system's fixed cost but the annual fixed costs are also called fixed costs. It seems preferable to make a clear distinction and this note distinguishes between the initial capital investment (C) and the annual fixed costs (F). Thirdly, the authors refer to the profit as being represented by the area between the variable cost line and the revenue line. It is more usual in break-even analysis to indicate profit by the vertical distance between these two lines at the relevant level of activity. If profit is to be represented by area we would need to have a break-even chart that deals with marginal costs and marginal revenues.

The authors argue correctly that where V_t varies from one year to the next, it is insufficient to adopt the simple average

$$\left(\bar{V} = \sum_{t=1}^{M} V_t / M \right)$$

in the analysis (where M is the life of the system). They propose that a weighted average should be used to give recognition to 'the time value of money'. They suggest that the unit variable costs should be converted to a present value by using the formula

$$PV = \sum_{t=1}^{M} \frac{V_t}{(1+r)^t}$$

where r is an appropriate interest rate. They then proceed to convert this present value to an annual value (which they refer to as $\bar{V}_M$) by multiplying by the interest rate, r. The correct procedure is rather to multiply by an annuity factor to convert PV into an actuarially equivalent annuity. The correct factor is

$$\frac{1 - (1+r)^{-M}}{r}$$

not r, so that

$$\bar{V}_M = \frac{1 - (1+r)^{-M}}{r} \sum_{t=1}^{M} \frac{V_t}{(1+r)^t}$$
$$= \frac{(1+r)^M}{(1+r)^M - 1} \left[r \sum \frac{V_t}{(1+r)^t} \right]$$

where in the last version the square brackets enclose the formula given by Agmon and Borovits. In fact, although the authors give the formula quoted, they do not use it in their illustrative example, neither do they use the correct approach unless some of the calculations have been rounded.

Another error occurs in the calculation of the system's annual fixed cost (F). Agmon and Borovits obtain a value for F by dividing the capital cost of the system (C) by its anticipated life (M). This 'straight line' approach is commonly used by accountants for external financial reporting purposes but it is recognized as being incorrect for decision making. The correct procedure is to multiply C by the appropriate annuity factor to obtain the equivalent annual capital charge ie

$$F = C \frac{1 - (1+r)^{-M}}{r}.$$

For the illustrative example provided by Agmon and Borovits this gives a fixed cost of £71 557 in place of £30 000. With the interest rate standing at 20% the firm will be indifferent as between paying £300 000 now or paying £71 557 p.a. at the end of each of the next ten years whereas it would clearly prefer to pay £30 000 annually for the next ten years. The present value of an annuity of £30 000 for ten years is only £125 774.

The break-even point is

$$Q^* = \frac{F}{P - V}.$$

For the example given in the article this is $71557/(5 - 3.281) = 41\,627$, whereas Agmon and Borovits obtain a break-even activity level of 17 763. It is clear that such a large error could lead to incorrect decisions.

The authors have sought to combine DCF investment techniques with break-even analysis to provide a tool that will assist with decisions relating to investments in information systems. Comments could be made on other aspects of the article; the general usefulness of the suggested approach, possible difficulties with the analysis when there is a choice between mutually exclusive investment alternatives and the treatment of risk but this note has focused solely on an error in the proposed procedures. Unfortunately the error can lead to a serious understatement of the level of activity that is required to justify the investment and the example used by Agmon and Borovits describes a situation where this does occur.

Yours faithfully
M. J. MEPHAM
Dean of Faculty of Economic and Social Studies
Heriot-Watt University
Edinburgh
UK

Dear Sir,

L. V. Atkinson discussed in his article 'Jumping About and Getting into a State' a couple of program structures for a simple problem posed by Knuth. I wonder, for which reasons he left out the endless loop construction, which is particularly well suited for this case. In Modula-2, for instance, one may express the algorithm conveniently by

```
i := 1;
LOOP
    IF a[i] = x THEN b[i] := b[i] + 1; EXIT
                                         END;
    IF i = m THEN m := m + 1; a[m] := x;
                    b[m] := 0; EXIT END;
    i := i + 1;
END
```

This structure is simple and clear, it expresses exactly what is intended. There do not arise any problems with the conjunction of conditions, with index overflow or with additional boolean variables. The elegant and concise structure is of course due to the facilities of

Modula-2. In Pascal we would get the heavier structure

```
i := 1; looping := TRUE;
WHILE looping DO
    IF a[i] = x
        THEN BEGIN b[i] := b[i] + 1;
                      looping := FALSE END
    ELSE IF i = m
        THEN BEGIN m := m + 1;
    a[m] := x; b[m] := 0; looping := FALSE END
        ELSE i := i + 1
END
```

The suppression of this solution by L. V. Atkinson indicates perhaps that Pascal programmers are accustomed to think only in while loops and repeat loops. Although it is in many respects wise to stick to a few well known patterns, it seems that it sometimes narrows the sight.

Yours faithfully
P. RECHENBERG
Institut für Informatik
Universität Linz
A-4040 Linz

The author replies:

Apart from introducing state transition loops, I discussed no loop constructions other than those presented in the letters which prompted the paper. These letters did not discuss the endless loop construction with a controlled exit or its emulation in Pascal.

Although not directly available in Pascal, this is yet another approach. I agree with Prof. Rechenberg that the structure is simple but would take issue over its clarity. The loop construct itself gives no indication as to how many, if any, reasons there are for exit, what these reasons are or which one caused the exit. For further information, one must delve into the loop, find the EXITs and then examine the corresponding if-tests.

With a state indicator approach in Pascal, all reasons for exit are explicitly named and if, as is often the case, the loop is followed by a case statement using the state variable as a selector, control flow subsequent to loop exit is immediately apparent. For further transparency, the loop termination test can check set membership:

```
state := s0;
REPEAT ... UNTIL state IN [s1, s2, ... ];
CASE state OF
    s1 : ...;
    s2 : ...;
    ...
END {case}
```

Even in Modula-2 with controlled exit from a loop, program transparency is improved by the use of state variables with exit to a case statement.

Dear Sir,

It is often said that the effect of computer jargon on our care for fine language has been

CORRESPONDENCE

analogous to the effect of myxomatosis on rabbits. Indeed many examples tend to confirm this pessimistic view. Recently I had occasion to study a 1933 paper in electronics[1] and found its good English usage refreshing after much of what one reads today. It hardly needs to be said that the paper is not about computers, which in their present form did not then exist, but nevertheless 'data' is throughout correctly treated as a plural, and 'set of data' as singular. One wonders how many modern authors would have got these right, and how many referees, to say nothing of editors, would insist that they be made correct. The paper by M. C. Er[2] suggests however that the influence of computing on language is not all bad. Consider for example his condition C1 in the Towers of Hanoi problem: 'Only one of the topmost discs may be moved at a time'. This, exactly as the problem which he is discussing, may appear trivial because the reader has seen a solution, but attempt to re-compose this sentence and it becomes evident how good it is.

In an accompanying letter,[3] Mr Er uses the construction that someone 'criticises that', and also that an equation 'has to be used costly'. Both these constructions struck me as unexpected, but after thought I believe both to be correct and desirable.

Yours faithfully
PETER FELLGETT

References

1. L. J. Sivian and S. D. White, On minimum audible sound fields, *Journal of the Acoustical Society of America* **4**, 288–321 (April 1933).
2. M. C. Er, The generalised colour towers of hanoi: an iterative algorithm, *The Computer Journal* **27**(3), 278 (August 1984).
3. M. C. Er, The tower of hanoi as a trivial problem—a reply, *The Computer Journal* **27**(3), 285 (August 1984).