Self-optimizing Control Mechanisms and Some Principles for More Advanced Learning Machines

A. M. ANDREW

Value of Self-optimization

Self-optimization may be a valuable feature of a system for automatic control or information processing under either of two sets of conditions. The first is where the situation is too complex to be analysed and, therefore, an efficient system cannot be designed. The second is where the environment is liable to change in such a way that the mode of action of the system must be modified accordingly if it is to operate efficiently.

Machines, actual and proposed, which incorporate self-optimization range from relatively simple control mechanisms to computers programmed to play chess or prove theorems. Some simpler devices optimize their control functions by adjusting a number of parameters, the adjustments being determined by computed statistical quantities such as correlations. It will be shown that the principles used in these mechanisms can be extended to allow aspects of behaviour usually associated with more complex systems. First, it is necessary to review the principles which have been used in self-optimizing devices.

Principles of Self-optimizing Devices ('Learning Machines')

Definition of learning

Thorpe[1] tentatively defines learning as 'that process which manifests itself by adaptive changes in individual behaviour as a result of experience'.

A relatively simple device like the conditional probability computer described by Uttley[2–4], can show behaviour which conforms to Thorpe's definition. In the present paper, however, the term 'Learning Machine' will only be applied to fairly complex self-optimizing systems.

Self-programming computers

The term 'learning' would be readily applied to a machine which could utilize its experience in becoming proficient in a complicated task such as chess-playing. One line of development which may lead to such advanced machines is the attempt to make a digital computer evolve its own programme. Friedberg[5] has programmed a computer to do this for some very simple tasks.

This line of approach may be fruitful in connection with certain applications of learning machines but seems unlikely to contribute to automatic control for a long time.

Logical-type systems

In a number of self-optimizing systems, the signals in the input and output channels are two-valued. These may be termed logical-type systems. The majority operate by continuously computing certain statistical quantities, and modifying their mode of action according to the value reached.

Uttley[2–4] and Russell[6] have discussed devices for the computation of conditional probabilities and have demonstrated a practical computer which can be incorporated in self-optimizing systems.

Many-element systems

Some of the self-optimizing systems proposed consist of large networks of simple elements. MacKay's system[7] is in this category, since he visualizes a large number of statistically adjustable linkages connected together. In other proposals, the property of self-optimization is a property of the assemblage, and not explicable by consideration of a single element. Some steps have been taken towards analysing the behaviour of such nets by Rosenblatt[8] and von Foerster[9].

Parameter-adjusting systems

In the simplest case these adjust quantities which can be held at steady values, such as temperatures, pressures, etc. Draper and Li[10] applied such a controller to adjust the ignition

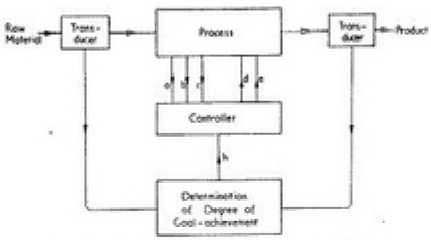

Figure 1. *A process with a self-optimizing controller*

timing and the proportions of gas–air mixture to maximize the power developed by an internal combustion engine. A controller for chemical processes which similarly adjusts the values of two variables to maximize a third has been described[11].

Feldbaum[12] has discussed controllers of the parameter-adjusting type, and Box and Wilson[13,14] have provided statistical theory applicable to them.

In more advanced systems, the parameters are incorporated in mathematical functions. *Figure* 1 shows a process and self-optimizing controller where d and e are controlled variables, and a, b and c are quantities continuously measured in the process. The quantities d and e are functions of a, b and c. In a simple controller a linear relationship might hold:

$$d = K + La + Mb + Nc \tag{1}$$

and the self-optimizing procedure adjusts the values of K, L, M and N.

The previous type of controller can be represented by

$$d = K \tag{2}$$

plus a similar equation for e.

In *Figure* 1, h is a measure of the degree of goal-achievement, termed 'hedony' after Selfridge[15].

Gabor[16] has described a self-optimizing filter which, in its linear form, may be represented by equation 1, where a, b, $c \ldots$ are values $f(t), f(t - \tau), f(t - 2\tau) \ldots$ of the input signal $f(t)$. The parameter values can be adjusted to give a desired filter characteristic. The non-linear form of the filter incorporates numerous product terms containing ab, abc, etc.

Kalman[17] describes a self-optimizing controller in which the parameters are not incorporated in control functions but in equations representing the transfer function of the process.

Correlation

Systems for parameter adjustment may operate by making an experimental change in one or more parameters and waiting to determine whether an increase in hedony results. If it does, the change is maintained, otherwise, it is cancelled.

Where the controlled process is subject to an unpredictable disturbance, the above 'move-and-stick' type of operation is only possible when the effects of the disturbance are small, or slow compared to the time between an experimental change and its effect on hedony. The alternative is to let the parameter values fluctuate repeatedly and to correlate the changes in hedony with these fluctuations. If a correlation coefficient reaches a value which is statistically significant and positive, the corresponding parameter is increased, or if significant and negative, it is decreased.

Need for Interpolation

The application of logical-type self-optimizing systems in automatic control is unlikely to be fruitful unless they can be made to exploit the continuity of their environment. Satisfactory control cannot be effected by a system which treats all measured variables as two-valued.

Logical-type machines could be extended to take account of quantitative information by coding it in binary form and using several two-state channels to convey the information from each transducer.

The complexity needed to satisfy any non-trivial control requirement in this way is formidable. A more serious objection is that the device would take a very long time to carry out any useful amount of self-adjustment. The remedy is to incorporate some form of interpolation. This feature can be introduced in a logical-type device by arranging that statistics are not collected for all discriminable situations but only for a moderate number of points scattered over its field of operation. *Figure* 2(*a*) represents a two-dimensional phase space in which every discriminable situation is represented by a dot; it is clear that a device which evolves its control action separately for each situation will be slow to optimize. *Figure* 2(*b*) shows the same phase space with fewer situations. A control system could operate by relating all of its statistical information to these. Then if a situation arises which is represented by point P, the action taken by the controller is a weighted average of the actions appropriate to the points A, B, C, D at the corners of the square in which P lies, the weightings being determined by the distance of these points from P. When data become available concerning the effect of this control action on hedony, the statistics referring to A, B, C, D would be modified to take account of it. The amounts of modification would depend on the distances of the respective points from P.

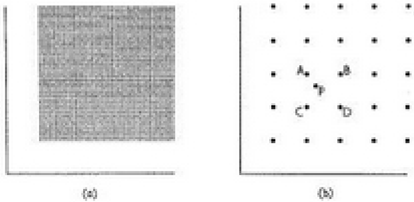

Figure 2. (a) Phase space with all discriminable situations represented; (b) the same with fewer situations represented

An alternative way in which any functional relationship between a, b, c and d, e can be expressed is by an equation of the form

$$d = K + La + Mb + Nc + Pa^2 + Qab + \text{other product terms} \tag{3}$$

and a similar equation for e. The number of terms on the right-hand side depends on the form of the functional relationship and the accuracy of representation required. A self-optimizing system which operates by adjusting the parameters K, L, M, N, P, Q—automatically embodies the feature of interpolation.

Further Possibilities Using Parameter Adjustment

Self-organization

Besides operating to find the best values, a parameter-adjusting system can be made to alter the actual form of the equations. It can incorporate criteria which enable it to decide which additional polynomial terms can profitably be included.

In general the correlation between a parameter and hedony will be computed as a continuous or 'running' value. It is therefore possible to compute a correlation between this and some other variable. Suppose the control equation is simply

$$d = K + La + Mb + Nc$$

Let r_{Lh} be the running value of the correlation between L and h when L fluctuates. Then if a significant correlation exists between r_{Lh} and a the conclusion can be drawn that the control equation could profitably include a term in a^2. Similarly, if there is a significant correlation between r_{Lh} and b the equation could profitably include a term in ab, and so on. With the addition of these terms the equation becomes

$$d = K + La + Mb + Nc + Pa^2 + Qab$$

Criteria may also be devised to show when an existing term in the equation is serving no useful purpose. The circuitry used in computing this term might then be re-allocated to compute new terms which have been selected as described above.

The new terms introduced to the polynomial functions need not be restricted to new combinations of the variables already appearing in them. Signals can be made available to the controller which are not initially known to be relevant to the control task. Let p be such a signal, perhaps representing

the atmospheric pressure under which the process is working. Then if there is a significant correlation between r_{Kb} and p, the control equation could profitably include a term in p.

It would not be possible to compute continuously all the correlations which are possibly of interest in a system capable of this kind of self-organization. The operation must depend on 'wandering correlators' which randomly test for correlations between pairs of variables not known to be correlated.

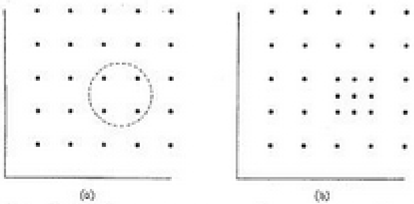

Figure 3. (a) Phase space with unsatisfactory representation in the enclosed region; (b) the same with additional points

Methods of self-organization could also be devised for a controller using the kind of interpolation which was described in terms of phase space. The controller could incorporate some means of detecting regions of its phase space in which its control policy is unsatisfactory because the key points are too widely spaced. Let the region enclosed by the broken line in *Figure 3(a)* be such a region. Then the control can be improved by introducing further key points as in *Figure 3(b)*. This is analogous to the introduction of further terms in a polynomial control function.

Sub-goals and concept formation

Any application of a self-optimizing controller to a complex process would involve the use of sub-goals. Part of the process would be linked to a sub-section of the controller in the general manner represented by *Figure* 1, but the quantity recognized as hedony and maximized by this part of the controller would not be the same as the overall goal of the controller. For a chemical process, for instance, the overall goal would be expressed in terms of the yield and quality of the product, but a sub-goal might be to maintain a particular temperature-distribution in a fractionating column. The sub-goals could be represented by polynomials or other functions of the measured variables, and could be modified by a correlation procedure in the same way as the control functions.

The specifications for sub-goals reached by a self-optimizing process are functions of the input data of the controller which may profitably be computed, and are therefore similar to concepts which a person might form when evolving a method of performing a task[18].

Learning-to-learn

In designing a self-optimizing system, values have to be chosen for such quantities as the time constants of smoothing of the statistical quantities which are computed, and the level of statistical significance the machine will require of its data before making a change. The mathematical determination of the best values for these is virtually impossible, especially since they depend on the nature of the process to be controlled. In fact, the design of the self-optimizing controller is precisely the kind of awkward mathematical problem which it is hoped to by-pass by the use of self-optimization. It seems likely, therefore, that there will eventually be no fixed values incorporated for many of the quantities determining the behaviour of the system. Instead, they will be varied in a trial-and-error fashion to find values which maximize the rate of increase of hedony.

Experimental Approach

Computer simulation

The foregoing discussion indicates that work on parameter adjusting systems can lead to devices showing great flexibility of operation, which can appropriately be described as 'learning machines'. It was therefore decided to make a study of this kind of optimization by simulating a process and a controller on the DEUCE digital computer. It was decided that the parameters to be optimized should not be static values of inputs to the process but should be incorporated in a control equation. It was also decided that the process should be subject to a random disturbance.

The process which has been simulated is one which is used for student demonstrations in the City of Gloucester Technical College. It consists of four vertical tubes containing liquid, connected at their lower ends through constrictions, as shown in *Figure* 4. Liquid flows into tube No. 1 at a rate determined

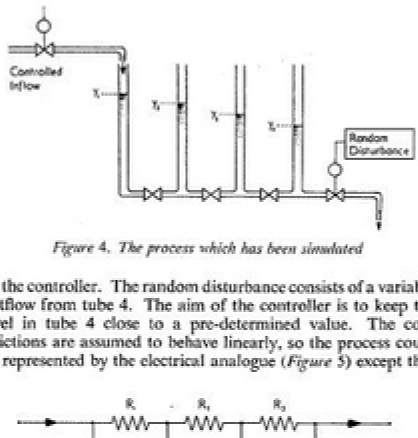

Figure 4. The process which has been simulated

by the controller. The random disturbance consists of a variable outflow from tube 4. The aim of the controller is to keep the level in tube 4 close to a pre-determined value. The constrictions are assumed to behave linearly, so the process could be represented by the electrical analogue (*Figure 5*) except that

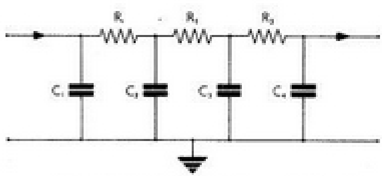

Figure 5. The electrical analogue of Figure 4

the levels in the tubes are not allowed to exceed certain (overflow) levels nor to go below zero.

The controller applies the control function previously considered

$$d = K + La + Mb + Nc$$

in which d represents the inflow to tube 1, and the significance of a, b and c can be altered by making minor adjustments to the programme. Each of them can represent any one of the four liquid levels, or the random disturbance, or values of these delayed in time by anything up to 16 units of simulated time.

The obvious form of control function when experimental fluctuations K_f, L_f, M_f, N_f are added to the parameters is

$$d = K + K_f + (L + L_f)a + (M + M_f)b + (N + N_f)c \quad (4)$$

and the programme can readily be adjusted to simulate a controller incorporating this. In the experimental runs reported here, however, the variables a, b and c were subjected to quasi-differentiation (as by an R–C circuit) to form their fluctuating components a_f, b_f, c_f and the control equation was

$$d = K + La + Mb + Nc + K_f + L_f a_f + M_f b_f + N_f c_f \quad (5)$$

The reason for preferring the mode of operation represented by equation 5 is as follows. In equation 4 an experimental fluctuation such as L_f is multiplied by the mean value $\bar{a}$ of a as well as the fluctuating part a_f, and it is likely that the detection of correlation between K_f and hedony will be rendered more difficult by the contributions $L\bar{a}$, $M\bar{b}$, etc., which are eliminated from equation 5. The superiority of the form of operation represented by equation 5 has not been conclusively demonstrated; it is expected that further work using this computer simulation will provide data for comparison of the two modes.

The process simulation consists of the Runge–Kutta method for numerical solution of simultaneous differential equations as programmed by Gill[19]. It is arranged to take 16 steps per unit of simulated time.

The differential equations are

$$C_1 \frac{dy_1}{dt} = d - \frac{y_1 - y_2}{R_1} \quad (6)$$

where d is given by equations 5 or 4,

$$C_2 \frac{dy_2}{dt} = \frac{y_1 - y_2}{R_1} - \frac{y_2 - y_3}{R_2} \quad (7)$$

$$C_3 \frac{dy_3}{dt} = \frac{y_2 - y_3}{R_2} - \frac{y_3 - y_4}{R_3} \quad (8)$$

and

$$C_4 \frac{dy_4}{dt} = \frac{y_3 - y_4}{R_3} - z \quad (9)$$

where z is the random disturbance.

In order to minimize derivative effects, the waveform of the parameter fluctuations has no discontinuities. Periodic variations are used at different frequencies for the four parameters. A symmetrical triangular waveform proved simpler to generate than a sine wave.

The amplitude slowly increases from zero until a significant correlation is detected. The rate of increase is determined by numbers inserted in the programme; it is a linear change at first, and then exponential up to a limiting value. This amplitude, and also the correlations, are returned to zero under different conditions in the two modes of operation used. These will be termed type A operation in which only the values associated with the parameter successfully correlated are returned, and type B in which *all* are returned as soon as *any* correlation becomes significant.

The numbers y_1–y_4 representing the liquid levels are permitted to vary between 0 and 1. The 'desired value' of y_4 is taken as $\frac{1}{2}$. The quantity taken as hedony is defined as

$$h = -(y_4 - \tfrac{1}{2})^2 \quad (10)$$

The hedony value is smoothed to give $\bar{h}$, and then $h - \bar{h}$ is computed, squared and smoothed with another time constant to give $\overline{(h - \bar{h})^2}$. The normalized hedony value is computed as

$$h_n = \frac{h - \bar{h}}{\{\overline{(h - \bar{h})^2}\}^{\frac{1}{2}}} \quad (11)$$

This value is multiplied by signals representing the fluctuations in the parameters (but of constant amplitude), and the products smoothed with yet another time constant to give measures of correlation. These measures are compared with threshold values in order to determine whether they are statistically significant.

The programme also computes correlations between hedony and the moduli $|K|$, $|L|$, $|M|$, $|N|$. When one of these values becomes significant and negative the corresponding fluctuation-amplitude is made to decrease instead of increase. The aim of this is to avoid large fluctuations when the range of variation straddles an optimal value. All the correlations are actually computed at three different time displacements, so the number of values computed is 24.

When a correlation of the first kind (referring to a fluctuation and not its modulus) becomes significant, the corresponding parameter is increased or decreased according to the sign of the correlation, and the amount of the parameter change is proportional to the amplitude of fluctuation. In the experiments conducted up till now, the constant of proportionality has been unity.

Since the control function is equation 5, rather than equation 4, the experimental fluctuations in L, M and N multiply only the fluctuating parts of a, b and c. The increments, or decrements in the values of L, M and N, multiply the real values of a, b, c, however. To allow for this anomaly and to avoid abrupt changes in d when a parameter other than K is altered, it is arranged that any change in L, M or N is accompanied by a compensating change in K. A change ΔL in L, for instance, is accompanied by a change ΔK where

$$\Delta K = -\bar{a} \,.\, \Delta L \quad (12)$$

where $\bar{a}$ is derived from a, by smoothing with the same time constant as was used in computing a_f. In fact, the quantities $\bar{a}$, $\bar{b}$, $\bar{c}$ are computed first, and then

$$a_f = a - \bar{a} \quad (13)$$

The numbers stored in the computer to represent K, L, M, N are limited to the range -1 to $+1$ in value. The value computed for d is multiplied by 8 before being applied to the process and then limited to the range 0 to 1 (a simple modification allows factors other than 8 to be used). Thus the effective values of K, L, M, N vary in the range -8 to $+8$.

Aims of the experiments

The general aim of the experiments is to use the programme as a 'test bed' in which variations of the self-optimizing controller can be tried. Different values can be inserted for the time constants involved, for the frequencies of the parameter-fluctuations, and the rate of increase and decrease of their amplitudes. Variations can also be made in the statistical properties of the random disturbance z.

Besides quantitative variations, a number of qualitative changes can be made in the method of self-optimization, and the computer simulation should make it possible to choose between these. One of these qualitative changes has already been mentioned; the programme may embody either type A or B operation as described above.

Another variation is to let the controller operate according to equation 4 rather than to equation 5 as described previously. Yet another is to arrange that the amount by which a parameter such as L is altered when it has been correlated depends on a measure such as $\overline{(a-\bar{a})^2}$ of the amount of fluctuation undergone by the corresponding process variable a. This would tend to ensure that the controller did not come to ignore one of its input signals during a period when the signal was inactive.

Experimental results

All of the resistance and capacitance values in the process (*Figure* 5) were set equal to unity. Thus the time constant of any R–C combination equals one unit of simulated time.

The generator of random disturbance used incorporated DEUCE sub-routine No. 283, which generates a sequence of effectively random numbers $r_1, r_2 \ldots$ of 31 binary digits according to the rule

$$r_n = [(2^{15} - 3)r_{n-1} - 1](\text{mod. } 2^{31}) \qquad (14)$$

A number was generated by this at intervals of one unit of simulated time, and the disturbance value moves linearly from

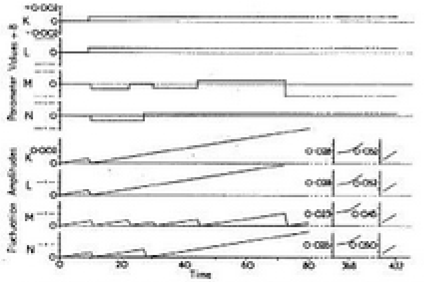

Figure 6. Records from the first computer run, in which fluctuation-amplitudes were individually returned to zero

one of these numbers to the next. The magnitude was scaled to correspond to a variation in the range of 0 to 1/32 of a unit of flow in the simulated process.

The periods of the fluctuations in K, L, M, N were respectively 16, 20, 29 and 48 time units, giving a frequency-ratio of $4\frac{4}{5}$, $3\frac{4}{5}$, $2\frac{2}{3}$, $1\frac{3}{5}$. These values, in conjunction with the time constant of smoothing of the correlations (200 time units), were chosen so that the period of the fastest variation is long compared to the process time constants, and so that the difference-frequency components which appear in the correlations are adequately smoothed out. The full mathematical justification of the choice will not be entered into here.

The time constant used in deriving $\bar{a}$, $\bar{b}$, $\bar{c}$, and thence a_f, b_f, c_f was 16 time units, that for smoothing $\bar{h}$ was 25 and that for $\overline{(h-\bar{h})^2}$ was 50. a, b, c were made to correspond to y_4, y_3 and y_2 (*Figure* 4).

The values of K, L, M, N which maximize h and towards which the programme should ideally converge have not been determined exactly, but K should become positive and the others negative.

Figure 6 represents a run of the programme in which initial values of K, L, M, N were zero, and the threshold value for the correlations corresponded to a coefficient of 0·042. The form of operation was type A as described in the section on computer simulation. The values of K and M became, respectively, positive and negative, but L and N remained positive. After 80 time units, no further significant correlations were detected, and the fluctuation-amplitudes rose to large values. At $t = 368$

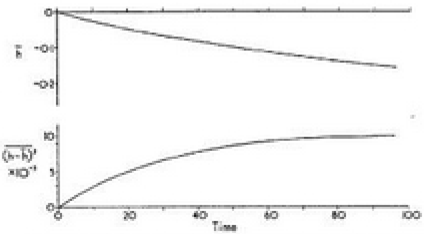

Figure 7. Further records from the first run. (Although the value of h becomes more positive during the run, the value of $\bar{h}$ initially falls. This is because $\bar{h}$ is smoothed as though by an R–C circuit, and starts from zero)

the rate of increase of the fluctuation-amplitudes was increased by a factor of 4 but no significant correlations appeared before the run was terminated at $t = 432$.

Further runs were made with lower threshold values for the correlations, of 0·007 and 0·014 respectively, but in each of these the value of K changed sign repeatedly and it was clear that there was no convergence. A value of 0·021 brought K and L to positive and negative values, respectively, but M and N remained positive.

Subsequently, runs were made to start from the values of $\bar{h}$ and $\overline{(h-\bar{h})^2}$ which applied at $t = 16$ in the first run. This avoided the initial steep rise in these quantities shown in *Figure* 7. The initial values of y_1, y_2, y_3 and y_4 were zero for all the runs.

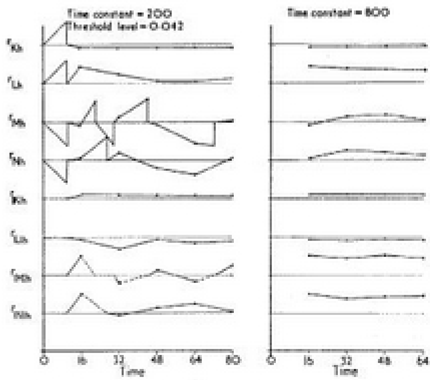

Figure 8. Fluctuations in the measures of correlation, smoothed with two different time constants. Only the values at intervals of 16 time units are plotted, except where it is known that statistical significance was reached and the corresponding measures were returned to zero

In order to see how the fluctuations in correlation measure depend on their time constant of smoothing, a short run was made in which this was 800 instead of 200 units. This started from conditions which applied at $t = 16$ in the first run, and the variations of the eight correlations computed without time displacement are shown in *Figure* 8, with the corresponding values from the first run for comparison. It is possible to derive the relationship between the indicated value of correlation and the probability P of the correlation being accidental from the relationship between the magnitude of these fluctuations and the time constant of smoothing.

Figure 9 is a record of a run in which the programme was successful in bringing K to a positive value while L, M, N became negative. This run also started from the conditions which applied for $t = 16$ in the first, except that all correlation

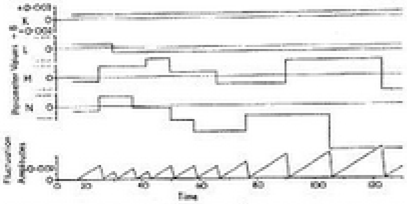

Figure 9. Records from a run which brought the parameters to values having correct signs. The fluctuation-amplitudes were returned to zero together

measures and fluctuation-amplitudes were returned to zero. Type B operation was used here. In this run the threshold level for the correlations was 0·021, and due to the experimenter's impatience the rate of increase of fluctuation-amplitude is 4 times the value previously used. There is reason to believe that a slower rate would have been preferable.

Conclusions

These results provide a comparison of the type A and type B methods of operation discussed in the section on computer simulation. Under the conditions of the test type B was preferable. This conclusion might be modified if some improved way of limiting the growth of fluctuation-amplitude were devised. The provision discussed in the section on computer simulation (for reducing the amplitude in response to a significant correlation with a modulus of a fluctuation) did not come into operation in these tests. Possibly the use of a lower significance threshold for these correlations would be advantageous.

The results were obtained in a few preliminary runs with the programme, and much further information will be available before June 1960. Data cannot be obtained in a short time, for although DEUCE is a high-speed computer, one minute of its running time is required for each unit of simulated time. Work is proceeding on an analogue version of the self-optimizing system suitable for connection to an actual process.

Results obtained using the computer simulation are, of course, only strictly applicable to a controller applied to a process of the kind which has been simulated. It is expected, nevertheless, that the general conclusions will be valuable in providing starting points for diverse applications of self-optimization.

I am indebted to the Director of the National Physical Laboratory for permission to present this paper.

References

1 Thorpe, W. H. *Learning and Instinct in Animals.* 1956. London; Methuen
2 Uttley, A. M. A theory of the mechanism of learning based on the computation of conditional probabilities. *Proc. of 1st Int. Congress of Cybernetics, Namur.* 1956. Paris; Gauthier-Villars, 1958
3 Uttley, A. M. Conditional probability computing in a nervous system. *N.P.L. Symposium** 1958
4 Uttley, A. M. The design of conditional probability computers. *Information and Control* 2 (1959) 1
5 Friedberg, R. M. A learning machine. *I.B.M. J.* 2 (1958) 2
6 Russell, G. (1957). Interim report on learning machines and adaptive control mechanisms. *Memo. no. 1369, Radar Research Establishment, Malvern, Worcs*
7 MacKay, D. M. Towards an information-flow model of human behaviour. *Brit. J. Psychol.* 47 (1956) 30
8 Rosenblatt, F. Two theorems of statistical separability in the perceptron. *N.P.L. Symposium** 1958
9 Von Foerster, M. H. Inaccuracy and reliability in biological computers. *2nd Int. Congress of Cybernetics, Namur.* 1958
10 Draper, C. S. and Li, Y. T. *Principles of Optimalizing Control Systems and an Application to the Internal Combustion Engine.* 1951. A.S.M.E. publication
11 Burt, D. A. and Van Nice, R. I. *Optimising Control Systems for the Process Industries.* 1959. Technical publication, Westinghouse Electric Corporation
12 Feldbaum, A. A. Automatic optimalizer. *Automat. Telemech. Moscow* 19 (1958) 731
13 Box, G. E. P. and Wilson, K. B. On the experimental attainment of optimum conditions. *J. R. statist. Soc. B* 13 (1951) 1
14 Box, G. E. P. *et al. Design and Analysis of Industrial Experiments* ed. O. L. Davies. 1956. Edinburgh; Oliver and Boyd
15 Selfridge, O. G. Pattern recognition and learning. *Information Theory*, ed. E. C. Cherry. 1956. London; Butterworth
16 Gabor, D. Communication theory and cybernetics. *Trans. Inst. Radio Engrs.* CT-1, No. 4, 19 (1954)
17 Kalman, R. E. Design of a self-optimising control system. *Trans. A.S.M.E.* 80 (1958) 468
18 Pask, G. Physical analogues to the growth of a concept. *N.P.L. Symposium** 1958
19 Gill, S. A process for the step-by-step integration of differential equations in an automatic digital computing machine. *Proc. Camb. phil. Soc.* 47 (1951) 96
20 Andrew, A. M. Learning machines. *N.P.L. Symposium** 1958

* *Symposium on The Mechanisation of Thought Processes at the National Physical Laboratory, Nov. 1958.* To be published by Her Majesty's Stationery Office.

Summary

Machines, actual and proposed, which incorporate self-optimization, range from relatively simple control mechanisms to computers programmed to play chess or prove theorems. Some of the simpler mechanisms optimize their control functions by adjusting parameters, the adjustments being determined by computed statistical quantities such as correlations. It is shown that the principles used in these mechanisms can be extended to allow aspects of behaviour usually associated with more complex machines. These aspects include: (*a*) self-organization, which allows the machine to modify the form of its control functions, and its own internal organization for computing

them, as well as adjusting parameters; (*b*) concept-formation, by which the machine decides which functions of its input data may profitably be computed; and (*c*) the facility of learning-to-learn, by which the machine's methods of optimization can themselves be subject to adaptive modification.

Because of these potential developments, and also with more immediate applications in mind, a study has been undertaken of the parameter-adjusting type of self-optimizing controller. A programme for a digital computer is described which simulates such a controller and also a process to which it is coupled. The simulated process is a system of capacities and restrictors and is subject to an unpredictable disturbance in the form of a variable outflow from one point. The controller determines the inflow d at another point according to the equation

$$d = K + La + Mb + Nc$$

where a, b and c can be current or time-displaced values of any variables in the process. The controller adjusts K, L, M, N with the aim of developing a control policy which keeps the level in one of the capacities close to a desired value.

The programme allows for different ways of organizing the controller, and different values can be inserted for such quantities as the time constants of smoothing of the statistical quantities computed. The effectiveness of the different arrangements can be compared. Data from several computer runs is presented, and a great deal more will be available before June 1960.

Sommaire

Les machines existantes ou proposées qui incorporent une auto-optimisation, vont des mécanismes de commande relativement simples, aux calculateurs programmés pour jouer aux échecs ou démontrer des théorèmes. Certains des mécanismes les plus simples optimisent leurs fonctions de commande en ajustant des paramètres, les ajustements étant déterminés par des grandeurs statistiques calculées telles que des corrélations. On montre que les principes utilisés dans ces mécanismes peuvent être étendus à des machines plus complexes. Ces aspects comprennent: (*a*) auto-organisation, qui permet à la machine de modifier la forme de ses fonctions de commande, et sa propre organisation interne; (*b*) formation de concept, par laquelle la machine décide quelles fonctions de ses données d'entrée peuvent avantageusement être calculées; (*c*) facilité d'apprendre à apprendre, par laquelle les méthodes d'optimisation de la machine peuvent être sujettes elles-mêmes à une modification adaptative.

On a entrepris une étude du type à ajustage de paramètres. On décrit un programme pour un calculateur numérique qui simule un tel dispositif de commande, ainsi qu'un processus auquel il est couplé.

Le programme autorise différentes façons d'organiser le dispositif de commande, et on peut insérer différentes valeurs de grandeurs telles que les constantes de temps de lissage des grandeurs statistiques calculées. Les efficacités d'arrangements différents peuvent être comparées. On présente des résultats provenant de plusieurs essais sur calculatrices et un nombre bien plus grand de résultats seront disponibles avant juin 1960.

Zusammenfassung

Ausgeführte und vorgeschlagene Maschinen mit Selbstoptimierung reichen vom relativ einfachen Regelgerät bis zu Rechenmaschinen, die zum Schachspielen oder zum Beweis von Lehrsätzen programmiert sind. Eine Reihe der einfacheren Apparate optimieren ihre Regelfunktionen durch Parametereinstellungen, die aus der Berechnung statistischer Größen, z.B. Korrelationen, vorgenommen werden. Es wird gezeigt, daß sich die in diesen Apparaten verwendeten Prinzipien in Richtung auf ein Verhalten erweitern lassen, welches sonst nur bei aufwendigeren Maschinen zu finden ist. Zu den Einzelheiten dieses Verhaltens gehören: (*a*) die Fähigkeit, sich selbst zu organisieren und Parameter einzustellen; (*b*) die Bildung eines Konzepts, nach dem die Maschine entscheiden kann, welche Funktionen der Eingangsdaten am günstigsten zu errechnen sind; und (*c*) die Möglichkeit, das Lernen selbst zu erlernen, wodurch die Maschine ihr eigenes Optimierungsverfahren einer anpassenden Korrektur unterwerfen kann.

Es werden Untersuchungen selbstoptimierender Regler mit Parameterverstellung durchgeführt und ein Digitalrechnerprogramm zur Nachbildung eines derartigen Reglers und einer damit zu regelnden Strecke beschrieben. Die nachgebildete Strecke besteht aus Speichern und Drosselstellen und wird unvorhersehbaren Störungen in Form eines veränderlichen Ausflusses an einem Punkt unterworfen. Der Regler ermittelt den Zufluß d an einer anderen Stelle nach der Gleichung

$$d = K + La + Mb + Nc$$

Dabei können a, b, und c gleichzeitige oder zeitlich verschiedene Werte von beliebigen Veränderlichen der Strecke sein. Der Regler verstellt die Werte K, L, M, und N mit dem Ziel der Festlegung eines Regelverhaltens, durch das das Niveau in einem der Speicher nahe dem Sollwert gehalten wird.

Eine Reihe von Ergebnissen der Arbeiten auf der Rechenmaschine wird mitgeteilt; weit umfangreichere Ergebnisse werden noch vor juni 1960 zur Verfügung stehen.

DISCUSSION

O. J. M. Smith (*U.S.A.*)

I did not understand what is readjusted in the last figure.

A. M. Andrew, *in reply*. The coefficients which control the ratios of components in the flow of fluids into the system are readjusted. In the paper they are denoted by the symbols K, L and M.

O. A. G. Ivakhnenko (*U.S.S.R.*)

How is self-optimization carried out in the system shown in the right-hand side of *Figure* 2?

A. M. Andrew, *in reply*. I have not considered this type of system in detail, but by the general method we would produce experimental fluctuations of the stored quantities and calculate the correlation between these fluctuations and changes of 'quality'. If the correlation is appreciable and positive, the stored quantity should be increased. If it is appreciable in magnitude and negative, the quantity should be decreased. It should be possible to improve the system by increasing its complexity, storing the probability distribution at each point in place of the numerical quantities.

A. V. Napalkov (*U.S.S.R.*)

It seems to me that the paper by Andrew is very interesting. In the complicated and important problem of designing-learning, self-organizing and concept-processing machines a further important step has been made. Principles have been developed which lead to the establishment of more highly perfected machines. The ideas of the necessity of introducing interpolation, and of the possible appearance of intermediate goals, appear to be very important. At the same time I would like to remark that to solve the problem of designing self-organizing systems a more detailed study of the principles of interaction and circulation of information between the object and the control system is desirable.

We are studying this problem on the basis of an analysis of brain function. We have come to the conclusion that to solve many

problems it is not so important to penetrate into the structure of the brain and into the black box, as it is to clarify the rules and principles defining the processing of external information. In our experiments we have found the definitions of a system of rules (algorithms), permitting the brain to select and store only the necessary information and to carry out only movements useful to the problem. Our work has been presented at the UNESCO Congress in Paris. It seems to us that these results may be found useful for the realization of the very interesting work being carried out by Mr. Andrew.

E. Letski (*U.S.S.R.*)

I would first like to congratulate Mr. Andrew for an interesting paper.

Several words on the use of the concept 'learning'. The definition of learning as adaptation through experience appears very broad. If we introduce into engineering the concept of 'learning' taken from psychology it is useful to understand by learning not only a process of random search but the operation of a certain more complicated system, where automatic search is only one element of behaviour. I think that it would be useful to try to introduce certain systems of biological learning into technical systems. In 1959, at the Cybernetics Laboratory of the Moscow Power Engineering Institute and at the Experimental Laboratory of the Academy of Sciences of the U.S.S.R., under the leadership of Professor Braines we have developed a learning automaton reproducing certain principles of biological learning—specifically the development of a chain of conditional reflexes. Besides the undoubted contribution of this automaton to the modelling of certain aspects of psychological behaviour of animals, we have been able to draw a number of conclusions regarding the direction of further work in the development of learning systems.

Any learning system which pretends to any degree of usefulness should have elements of 'intelligence', permitting the reinforcement of the biological method of learning by trial and error with the possibility of finding the best method of learning, and the possibility of constructing certain hypotheses or of making definite conclusions in the absence of information sufficient for the exact solution of the problem. From this point of view the ideas presented in Andrew's paper are of interest.

It should, of course, be noted that the correlation method connected with the production of a disturbance in the process is used by us only for certain particular cases admitting a fairly simple controlled object. With increase in the number of parameters the complexity of the control system increases substantially. The use of 'wandering correlators' leads to lengthening the search process. The convergence of the process is then problematical. The presence of test fluctuations is sometimes undesirable or completely impossible. In this case the problem of developing systems with maximum probability of determining correct conditions without carrying out test operations arises. It seems to me that work in this direction is very promising.

E. M. Braverman (*U.S.S.R.*)

At the present time there is already a large number of machines which from intuitive considerations we term 'learning'. A striking factor is the great variety of problems solved by these machines and the correspondingly great number of dissimilar machines. We need only mention Shannon's 'mouse', Ashby's homeostat, Gelernter's geometry-theorem proving machine, Selfridge's 'pandemonium' and others. Below we propose a classification of the problems solved by these machines which I hope will facilitate understanding the inter-relationships between various machines.

First Class. Typical formulation of problems: From a given set of elements it is required to find an element with a given property. A typical machine is Ashby's homeostat. In this class two important sub-classes may be defined.

(*a*) To find the coordinates of an extremum of a certain function. We note that to solve such problems we apply, in particular, the method of random search (frequently termed Ashby's method) with slight modifications.

(*b*) In principle a very important sub-class—the problems are formulated as follows. Two areas A and B are given in a certain labyrinth; find the path joining these areas. In precisely this way it is possible to formulate in general terms the problem solved by the theorem-proving machine.

If Ashby's method is used traditionally for this problem, i.e. a point is moved along a random trajectory from area A until it arrives at area B, the probability of this event at each step is, as a rule, very small. However, if simultaneously from area B we also start a point, the required path will be found when these trajectories intersect, while the wandering points need never coincide. The probability of intersection of two random trajectories is substantially larger than the probability of wandering points arriving at a prescribed area. These considerations, in essence, are a generalization of the method of sub-goals.

Second Class. Typical formulation: There exists a machine with input and output. A finite set of situations may arise at the input and output. It is required that after the learning process, leading as a rule to the progressive solution of a problem of the first class, the machine will respond to a given situation at the input by a required situation at the output. Initially this correspondence between situations is not given to the machine. The following limitations exist: The memory volume of the machine permits storing all input situations; and all possible situations appear at the input during the learning period.

Typical machines: Shannon's 'mouse', the Braines–Napalkov machine for modelling conditioned reflexes. We note that in these machines each input situation is stored in the result.

Third Class. Typical formulation: The same as the problems for the above class of input and output machines. It is also required that the machine establishes the required correspondence between input and output situations. However, either the memory volume of the machine is small in comparison with the number of possible input situations, or during the learning time only a part of the input situations can arise. In this case it is no longer possible to store each concrete situation. Other methods of memorizing are necessary. The most interesting machine of this class is Selfridge's 'pandemonium'. This type of machine presents the greatest perspectives since it is capable of imitating the human ability to establish abstractions.

A. M. Andrew, *in reply*. I thank Napalkov and Letski for their very valuable remarks. A combination of their systems with mine would give better results. I have considered a system which reacts only to continuous variation of the parameters. The systems of the preceding speakers are sequential discrete systems.

N. V. Grishko (*U.S.S.R.*)

The paper poses questions concerning the principles of improved learning automata, in which connection I would like to state an algorithm realizing perfected control with great *practical* significance. I shall illustrate it by a very simple example.

Let there exist an object with extremum, containing the object proper, some pre-existing automaton and a computer, calculating the effectiveness y. As a whole, such an object may be described by the equation (*see*, for example, Feldbaum's paper)

$$y = y_m + \Sigma L_{2i} (L_{1i} x_{iu} - x_{im})^2 \qquad i = 1, \ldots, n$$

The position of the extremum x_{im} with respect to the ith control organ x_{iu} depends on $x_{j\,\mathrm{ex}}$—the external signal, the process parameters etc.; i.e. $x_{im} = U(\ldots, x_{j\,\mathrm{ex}}, \ldots)$ in a simple particular case $x_{im} = \Sigma_j L_{ij} x_{j\,\mathrm{ex}} (L_{1i}, L_{2i}, L_{ij}$ are operators reflecting inertial properties). In the ideal case the control signal $x_{iu} = L_{1i}^{-1} x_{im}$. If we knew U we could synthesize equipment F, realizing optimal control of the process, approaching ideal to a certain degree, but we do not know U. In this initial period certain x_{iu} are fixed; some of them vary under the influence of the existing automaton, which in the general case is not optimal.

It is intuitively clear that there exists a continuous process of improvement of the control system. There thus arises the problem consisting of *determining the properties of the optimum dynamic process of improving the control system in connection with the optimal process of accumulating knowledge about the controlled object during the control period.*

The following procedure is a certain approximation to this process. An automatic optimizer C is constructed and *immediately begins* to control the object: $x_{iux} = C(y)$. As is well known, the possibility of purely experimental control is limited in conditions where the object is multidimensional, inertial, and with rapidly shifting extremum position $x_{im}(t)$. During the period of control using C, calculation of the correlation functions $Rx_{iux}x_{j0x}(t)$ is begun. If certain of them are not constants, then among the x_{j0x} and x_{ix} there is placed a device F_{ji} of a certain degree of complexity with arbitrary parameters $x_{n+r;ix}$, $r = 1, \ldots, m$. In the present case it processes one component $x_{ix}F_{ji} = F_{ji}(x_{j0x}; x_{n+r;ix})$. This component is added to the function processed by C. Thus, $x_{ix} = x_{iux} + \sum_j x_{ix}F_{ji}$. The parameters $x_{n+r;ix}$, i.e. the magnitude and sign of the gain factor, the inertial properties F_{ji} in the first approximation, are adjusted on the basis of the correlation functions $Rx_{iux}x_{j0x}$. Improved adjustment of $x_{n+r;ix}$ is carried out by means of C, i.e. the number of action points of C is increased; in $x_{n+r;ix}$ we have the adjustment of C itself.

As a result of this process, the control functions in C are transferred to F, i.e. we pass more and more from control, on the basis of search, to control by algorithm.

We note that if the $x_{j0x}(t)$ are stationary, in the limit $x_{iux} \equiv 0$, while F_{ij} ideal $= L_{1i}/L_{ji}$ (here $x_{n+r;ix} = \text{const.}$). If $x_{j0x}(t)$ are non-stationary, the 'non-stationarity detectors' are necessary, and then the above cycle is repeated, etc., as long as economically expedient (or when some cycle demonstrates that the process on this scale is practically stationary). We note that it is possible to construct $x_{n+r;ix}$ immediately by means of C, but apparently calculation of R accelerates the process.

It is natural to take as the criterion of optimalness of the process

$$I = \overline{\overline{y_m - y}} + \overline{s(t)}$$

where $s(t)$ are the *most exact* estimates of the consumption of energy, equipment and human labour.

We have considered a simple example of a process, omitting certain important particular cases, and certain general conclusions may already be drawn. In solving the problem we must first find the optimum variation of the test signal frequencies of C, the optimum sequence of adjustment of individual x_{iux} and $x_{n+r;ix}$, the optimum utilization of the computation of R, etc.

The realization and solution of this problem has great economic significance, particularly in the period of broad introduction of automation into new, unstudied processes, for example, in the chemical industry.