Experiments on the mechanization of game-learning
Part I. Characterization of the model and its parameters

By Donald Michie

This paper describes a trial-and-error device which learns to play the game of Noughts and Crosses. It was initially constructed from matchboxes and coloured beads and subsequently simulated in essentials by a program for a Pegasus 2 computer. The parameters governing the adaptive behaviour of this automaton are described and preliminary observations on its performance are briefly reported.

A reason for being interested in games is that they provide a microcosm of intellectual activity in general. Those thought processes which we regard as being specifically human accomplishments—learning from experience, inductive reasoning, argument by analogy, the formation and testing of new hypotheses, and so on —are brought into play even by simple games of mental skill. The problem of artificial intelligence consists in the reduction of these processes to the elementary operations of arithmetic and logic.

The present work is concerned with one particular mental activity, that of trial-and-error learning, and the mental task used for studying it is the game of Noughts and Crosses, sometimes known as Tic-tac-toe.

From the point of view of one of the players, any game, such as Tic-tac-toe, represents a sequential decision process. Sooner or later the sequence of choices terminates in an outcome, to which a value is attached, according to whether the game has been won, drawn or lost. If the player is able to learn from experience, the choices which have led up to a given outcome receive *reinforcements* in the light of the outcome value. In general, positive outcomes are fed back in the form of positive reinforcement, that is to say, the choices belonging to the successful sequence become more probable on later recurrence of the same situations. Similarly, negative outcomes are fed back as negative reinforcements. The process is illustrated in Fig. 1.

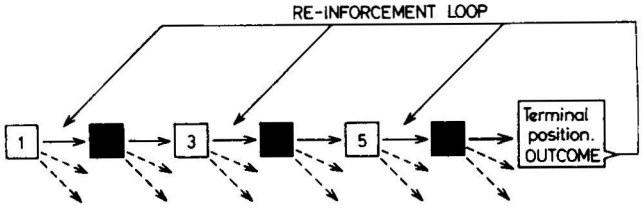

Fig. 1.—Schematic picture of the reinforcement process during trial-and-error learning of a game. The numbered boxes represent the players' successive choice-points, and the black boxes those of the opponent. Arrows drawn with broken lines indicate possible alternative choices open at the given stage

This picture of trial-and-error learning uses the concepts and terminology of the experimental psychologist. Observations on animals agree with common sense in suggesting that the strength of reinforcement becomes less as we proceed backwards along the loop from the terminus towards the origin. The more recent the choice in the sequence, the greater its probable share of responsibility for the outcome. This provides an adequate conceptual basis for a trial-and-error learning device, provided that the total number of choice-points which can be encountered is small enough for them to be individually listed.

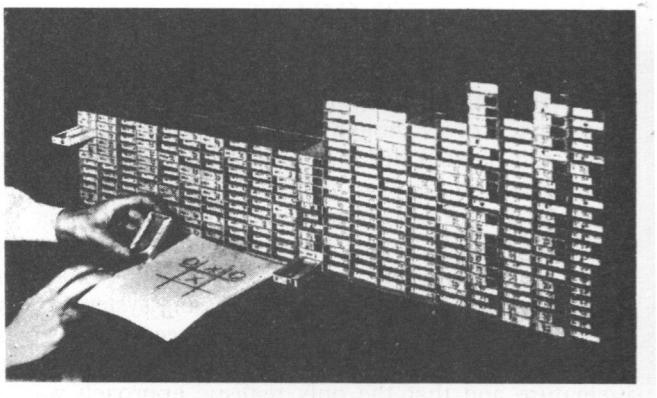

Fig. 2.—The matchbox machine—MENACE

The matchbox machine

Fig. 2 shows such a device, known as MENACE, standing for *M*atchbox *E*ducable *N*oughts *A*nd *C*rosses *E*ngine. The machine shown is equipped to function as the opening player. The principles by which it operates are extremely simple and have been described elsewhere (Michie, 1961). However, a brief recapitulation will here be given.

Every one of the 287 *essentially distinct* positions which the opening player can encounter in the course

of play is represented by a separate box, the face of which bears a drawing of the position and a code-number for indexing. The words "essentially distinct" are emphasized because such variants as those listed in Fig. 3 are treated as one and the same position. Each box contains an assortment of variously coloured beads. The different colours correspond to the different unoccupied squares to which moves could be made, according to the code shown in Table 1. Consider the box corresponding to the position of Fig. 3. A simple convention determines which of the four orientations is to be regarded as standard—in this case the first one listed. At first sight there are seven possible moves available. Considerations of symmetry, however, reduce these to four, namely moves to squares 1, 8, 7 and 6. Hence the box is equipped with white, black, amber and red beads.

Imagine that we wish to play against the machine. In order to ascertain its first move, we remove the box corresponding to the opening position, shake it and tilt it forwards. The beads—in this case white, lilac and gold—run to the front, where a V-shaped partition selects the first to arrive. The colour of this bead defines the machine's opening move. The human opponent, replies, thus generating a fresh position, which might, for the sake of illustration, be the one shown in Fig. 3. The box corresponding to this position is located, shaken and tilted, thus selecting the machine's next move—and so on to the end of the play.

At this stage reinforcements are applied. If the machine has done badly, it is "punished" by confiscation of the selected bead from each of the three or four boxes which have been used during the play, so that it becomes less probable, when any of these positions recur in future play, that the unsuccessful move will be repeated. If the machine has done well, it is "rewarded" by adding to each of the open boxes an extra bead of the same colour as the selected one. The moves in the successful sequence thus become more likely to be repeated if and when any of these positions recur in future.

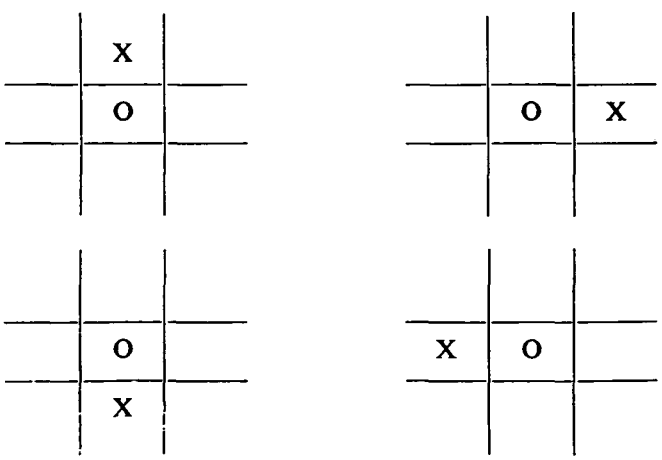

Fig. 3.—Four positions which are in reality variants of a single position

Table 1

The colour code used in the matchbox machine. The system of numbering the squares is that adopted for the subsequent computer simulation program

1 WHITE	2 LILAC	3 SILVER
8 BLACK	0 GOLD	4 GREEN
7 AMBER	6 RED	5 PINK

As stated earlier, it is desirable that the strength of reinforcement should be related to the stage of the game, being maximal for terminal moves and decreasing towards the beginning. This general pattern was ensured by making the number of times each colour in a box was replicated a decreasing function of the stage of play, as shown in Table 2. It can be seen that the system of

Table 2

Variation of the number of colour-replicates of a move according to the stage of play (see text)

STAGE OF PLAY	NUMBER OF TIMES EACH COLOUR IS REPLICATED
1	4
3	3
5	2
7	1

unit bonuses and forfeits will cause more rapid change of probabilities in late boxes than in early boxes.

For MENACE's maiden tournament against a human opponent a draw was reckoned a good result, and received a unit bonus. A win was regarded as an exceptionally good result and was rewarded by three extra beads to each open box. A defeat was punished

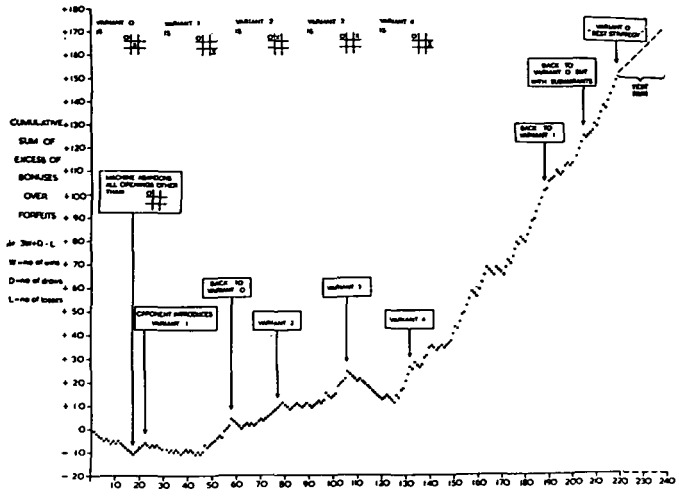

Fig. 4.—The progress of MENACE's maiden tournament against a human opponent. (Reproduced from *Penguin Science Survey*, 2 (1961), p. 139.) The line of dots drops one level for a defeat, rises one level for a draw and rises three levels for a victory. The variants listed along the top indicate the different replies to the machine's opening move which its opponent resorted to

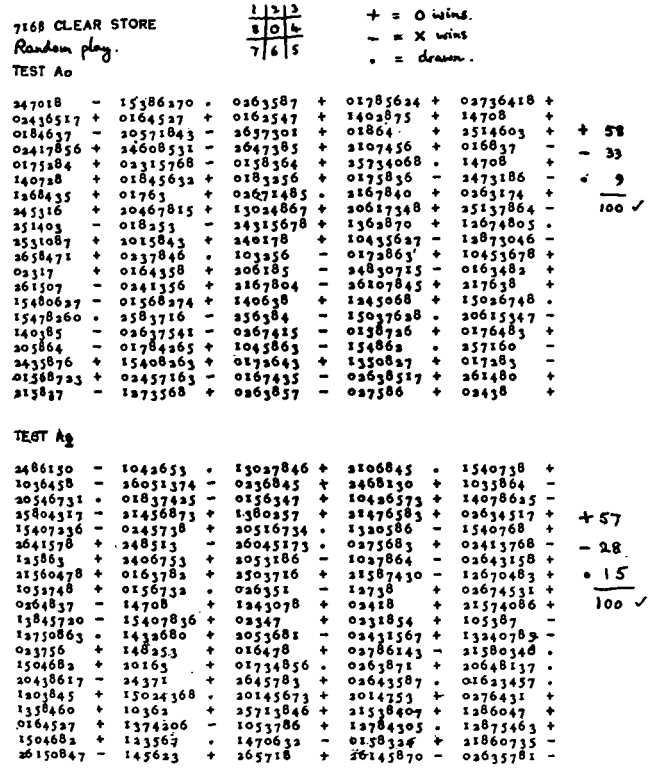

Fig. 5.—Random play at Noughts and Crosses as simulated by the computer program. The numerical coding of moves is as shown in Table 1

by a unit forfeit. Fig. 4 shows the progress of the tournament. The slope of the line of dots measures the prowess of the machine at any stage.

Computer simulation program

With the aid of Mr. D. J. M. Martin, of Ferranti Ltd. a Pegasus 2 computer has been programmed to simulate the matchbox machine. The computer program steps into the shoes of both players, Nought and Cross, and plays them against each other at the rate of about a game a second. Either side can be made to operate as a learner, or as a non-learner, at any desired standard of play from random up to expert. Fig. 5 shows part of a print-out when both sides were playing at random. There is evidently an inherent bias in the structure of the game in favour of the opening player, Nought, to the extent of about 2 : 1. Random games have an extremely idiotic character, as can readily be verified by playing through one or two examples.

The reinforcement system differs from that of the matchbox machine in two main ways. First, the stage of play to which a move belongs is reckoned backwards from the end of the play. Thus, the opening move of a long play might stand as much as eight moves from the end, and hence receive relatively mild reinforcement, since the strength of reinforcement decays for moves successively further from the end. In a short play, Nought's opening move might be the fifth from the end, and be relatively strongly reinforced. This is not unreasonable, since the weight of evidence provided against an opening move by a defeat in five moves is obviously greater than that provided by a defeat in eight moves, and likewise for victories. Similar considerations apply to moves other than the opening move.

The second difference from the MENACE reinforcement system concerns the manner in which the move-probabilities are modified. The computer program handles these in the form of odds, where odds $= \frac{p}{1-p}$, p being the probability with which a given move is selected. The reinforcements are stored as multipliers. Consider a position from which two alternative moves can be made, and suppose that at some stage in the proceedings the probabilities attached to them are $\frac{2}{5}$ and $\frac{3}{5}$ respectively. Suppose that the first of these happens to be selected, and leads to a win after n moves. If the multiplier M_n were, say, 2, the odds on selection of this move in future would be converted from 2 : 3 to 4 : 3, and the corresponding probabilities of the two moves adjusted to $\frac{4}{7}$ and $\frac{3}{7}$. The multipliers for losing outcomes are the reciprocals of those for winning outcomes. Fig. 6 shows the values for the trial run, and the function which was used to generate these values.

Fig. 7 shows the progress of Nought, learning against a random Cross. It is an enormously more difficult and time-consuming task to learn against random play than against play which is expert, or stereotyped in some other fashion. In the latter case only a restricted subtree of the whole game-tree has to be explored. For this

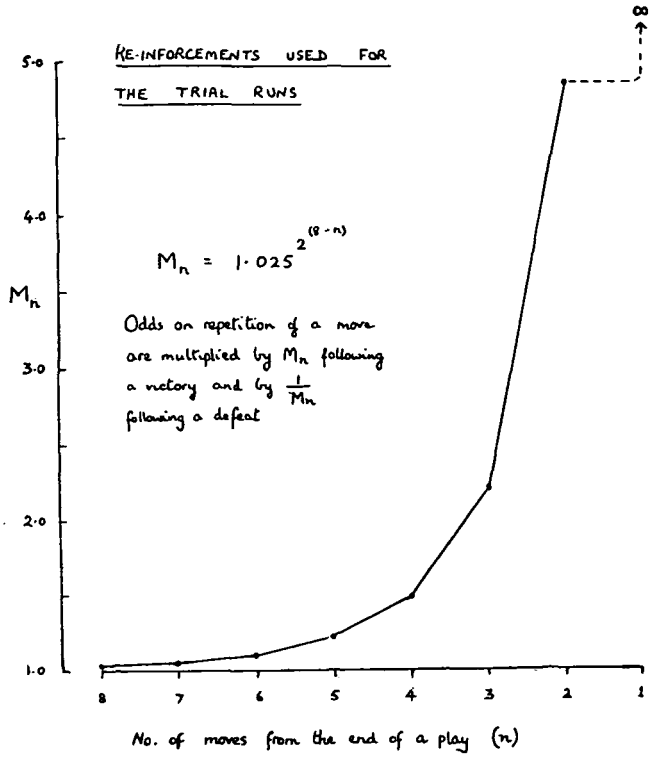

Fig. 6.—The multipliers used for reinforcement in the trial runs of Figs. 6–8. A special case is shown of the general form $M_n = A^{B^{(8-n)}}$

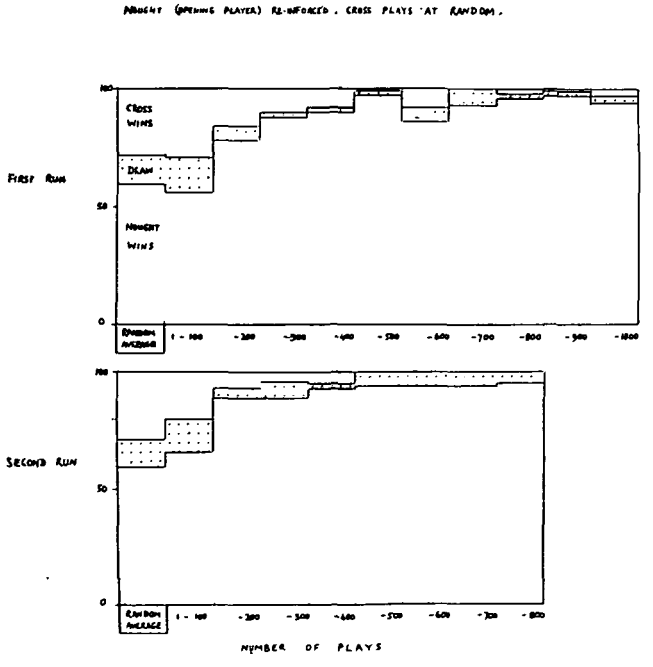

Fig. 7.—Trial runs with the computer program. Nought (the opening player) is learning, while Cross is playing at random throughout. The left-hand block shows the average results when both sides play at random

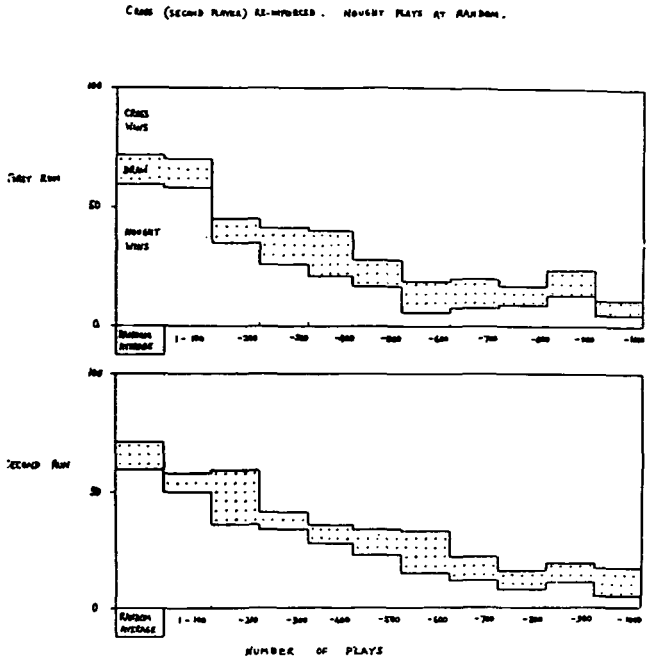

Fig. 8.—Cross is learning, Nought playing at random

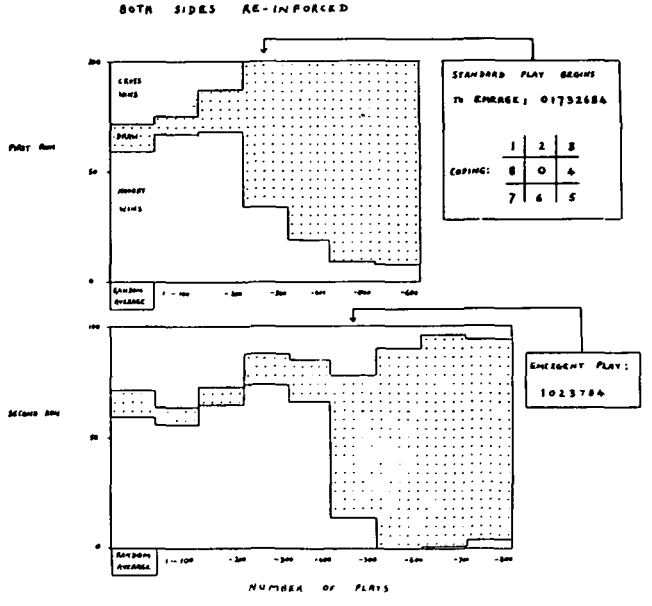

Fig. 9.—Both sides are learning

reason the speed of learning shown cannot be compared with the performance of MENACE in the matchbox machine's maiden tournament. It will be seen that neither of the duplicate runs converged to an infallible standard of play. This is almost certainly because the reinforcements were too strong, so that the machine jumped to premature conclusions from which it never entirely rid itself. Fig. 8 shows Cross learning against random Nought, and presents essentially similar features. Fig. 9 shows what happened when both players were

Table 3

Adjustment of multipliers to a sliding origin

After the *j*th play μ is calculated as

$$\frac{1-D}{D-D^{(j+2)}}\sum_{i=0}^{j} D^{(j\ \ i+\ )}V_i,$$

where V_i is the outcome value of the *i*th play and V_0 is set equal to 0 (value of a win is +1, of a draw is 0, and of a defeat is −1). D is the decay factor and M_n is the unadjusted multiplier for the *n*th stage of the game (see text).

OUTCOME	REINFORCEMENT
Won	$R_n = M_n^{-\mu+1}$
Drawn	$R_n = M_n^{-\mu}$
Lost	$R_n = M_n^{-\mu-1}$

allowed to learn. After a few hundred games the two sides were both producing near-expert play.

Improvements to the program

These results are only preliminary. The program has now been modified so that the value of an outcome is assessed against the average outcome of past plays, instead of remaining fixed. It seems obvious that a draw, for example, should be rewarded when the usual outcome has been defeat, and punished when the usual outcome has been victory. Similar considerations apply to the values of winning and losing outcomes. The method which has been adopted is the following.

The value of a win is rated at +1, that of a draw at 0 and that of a defeat at −1, and a weighted average, μ, of past outcome values is formed using as weight a decay factor D ($0 < D \leqslant 1$). Thus the weight of the last outcome is D, that of the penultimate outcome is D^2, that of the antepenultimate outcome is D^3, and so on. The smaller the value chosen for D, the more weight is given to recent experience; as D approaches unity, increasing weight is given to the experience of the more remote past. In theory, a running calculation is made to evaluate μ after each play, and this is used to adjust the multipliers as shown in Table 3. The implementation in the current version of the program does not actually attain this ideal, but makes an approximation. The decay factor is only applied to the average of each set of one hundred plays.

Our model of trial-and-error learning is thus based on three adjustable parameters, A, B and D (see Fig. 6 and Table 3). The next paper of this series will describe the effects upon learning performance which result from the systematic variation of these parameters.

Acknowledgements

The work described was supported by a Grant-in-Aid from the Royal Society. My thanks are also due to Messrs. Bruce Peebles and Co., Edinburgh, and to Mr. W. A. Sharpley personally, for making computing facilities available to me.

Reference

MICHIE, D. (1961). "Trial and Error," *Science Survey*, 1961, Harmondsworth: Penguin, Part 2, pp. 129–145.

Correspondence

To the Editor,
The Computer Journal.

Dear Sir,

"*Direct coding of English language names*", *The Computer Journal, Vol.* 6, *No.* 2 (*July*), *p.* 113

Surely the duplication in book titles tends to occur at the *beginning*. Could a solution be found for a short unambiguous code in referring to the *last* word, say the first and third, or better still the ultimate and antepenultimate?

e.g.	Selections from Borrow		SLWR
	Selections from Byron		SLNR
	Short History . . . etc. . . .	Augurelius	SOSI
	Short History . . . etc. . . .	Augustus	SOST

Yours faithfully,

E. R. KERMODE.