MATHEMATICAL GAMES

How to build a game-learning machine and then teach it to play and to win

by Martin Gardner

I knew little of chess, but as only a few pieces were on the board, it was obvious that the game was near its close. . . . [Moxon's] face was ghastly white, and his eyes glittered like diamonds. Of his antagonist I had only a back view, but that was sufficient; I should not have cared to see his face.

The quotation is from Ambrose Bierce's classic robot story, "Moxon's Master" (reprinted in Groff Conklin's excellent science fiction anthology, *Thinking Machines*). The inventor Moxon has constructed a chess-playing robot. Moxon wins a game. The robot strangles him.

Bierce's story reflects a growing fear. Will computers someday get out of hand and develop a will of their own? Let it not be thought that this question is asked today only by those who do not understand computers. In recent years Norbert Wiener has been viewing with increasing apprehension the day when complex government decisions may be turned over to sophisticated game-theory machines. Before we know it, Wiener warns, the machines may shove us over the brink into a suicidal war.

The greatest threat of unpredictable behavior comes from the learning machines: computers that improve with experience. Such machines do not do what they have been told to do but what they have *learned* to do. They quickly reach a point at which the programmer no longer knows what sort of circuit his machine contains. Inside most of these computers are randomizing devices. If the device is based on the random decay of atoms in a sample radioactive material, the machine's behavior is not (most physicists believe) predictable even in principle.

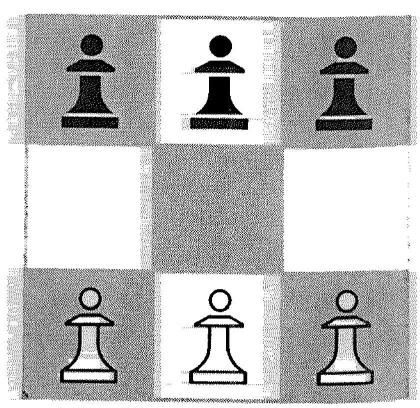
Much of the current research on learning machines has to do with computers that steadily improve their ability to play games. Some of the work is secret—war is a game. The first significant machine of this type was an IBM 704 computer programed by Arthur L. Samuel of the IBM research department at Poughkeepsie, N.Y. In 1959 Samuel set up the computer so that it not only played a fair game of checkers but also was capable of looking over its past games and modifying its strategy in the light of this experience. At first Samuel found it easy to beat his machine. Instead of strangling him, the machine improved rapidly, soon reaching the point at which it could clobber its inventor in every game. So far as I know no similar program has yet been designed for chess, although there have been several ingenious programs for nonlearning chess machines [see "Computer v. Chess-Player," by Alex Bernstein and Michael de V. Roberts; SCIENTIFIC AMERICAN, June, 1958].

Recently the Russian chess grand master Mikhail Botvinnik was quoted as saying that the day would come when a computer would play master chess. "This is of course nonsense," writes the American chess expert Edward Lasker in an article on chess machines in last fall's issue of a new magazine called *The American Chess Quarterly*. But it is Lasker who is talking nonsense. A chess computer has three enormous advantages over a human opponent: (1) It never makes a careless mistake; (2) it can analyze moves ahead at a speed much faster than a human player can; (3) it can improve its skill without limit. There is every reason to expect that a chess-learning machine, after playing thousands of games with experts, will someday develop the skill of a master. It is even possible to program a chess machine to play continuously and furiously against itself. Its speed would enable it to acquire in a short time an experience far beyond that of any human player.

It is not necessary for the reader who would like to experiment with game-learning machines to buy an IBM 704. It is only necessary to obtain a supply of empty matchboxes and colored beads. This method of building a simple learning machine is the happy invention of Donald Michie, a biologist at the University of Edinburgh. Writing on "Trial and Error" in *Penguin Science Survey 1961*, Vol. 2, Michie describes a ticktacktoe learning machine called MENACE (Matchbox Educable Naughts And Crosses Engine) that he constructed with 300 matchboxes.

MENACE is delightfully simple in operation. On each box is pasted a drawing of a possible ticktacktoe position. The machine always makes the first move, so only patterns that confront the machine on odd moves are required. Inside each box are small glass beads of various colors, each color indicating a possible machine play. A V-shaped cardboard fence is glued to the bottom of each box, so that when one shakes the box and tilts it, the beads roll into the V. Chance determines the color of the bead that rolls into the V's corner. First-move boxes contain four beads of each color, third-move boxes contain three beads of each color, fifth-move boxes have two beads of each color, seventh-move boxes have single beads of each color.

The robot's move is determined by shaking and tilting a box, opening the drawer and noting the color of the "apical" bead (the bead in the V's apex). Boxes involved in a game are left open until the game ends. If the machine wins, it is rewarded by adding three beads of the apical color to each open box. If the game is a draw, the reward is one bead per box. If the machine loses, it is punished by extracting the apical bead from each open box. This system of reward and punishment closely parallels the way in which animals and even humans are taught and disciplined. It is obvious that the more games MENACE plays, the more it will tend to adopt winning lines of play and shun losing lines. This makes it a legitimate learning machine, although of an extremely simple sort. It does not make (as does Samuel's checker machine) any self-analysis of past plays



The game of hexapawn

that causes it to devise new strategies.

Michie's first tournament with MENACE consisted of 220 games over a two-day period. At first the machine was easily trounced. After 17 moves the machine had abandoned all openings except the corner opening. After the 20th game it was drawing consistently, so Michie began trying unsound variations in the hope of trapping it in a defeat. This paid off until the machine learned to cope with all such variations. When Michie withdrew from the contest after losing eight out of ten games, MENACE had become a master player.

Since few readers are likely to attempt building a learning machine that requires 300 matchboxes, I have designed hexapawn, a much simpler game that requires only 24 boxes. The game is easily analyzed—indeed, it is trivial—but the reader is urged *not* to analyze it. It is much more fun to build the machine, then learn to play the game while the machine is also learning.

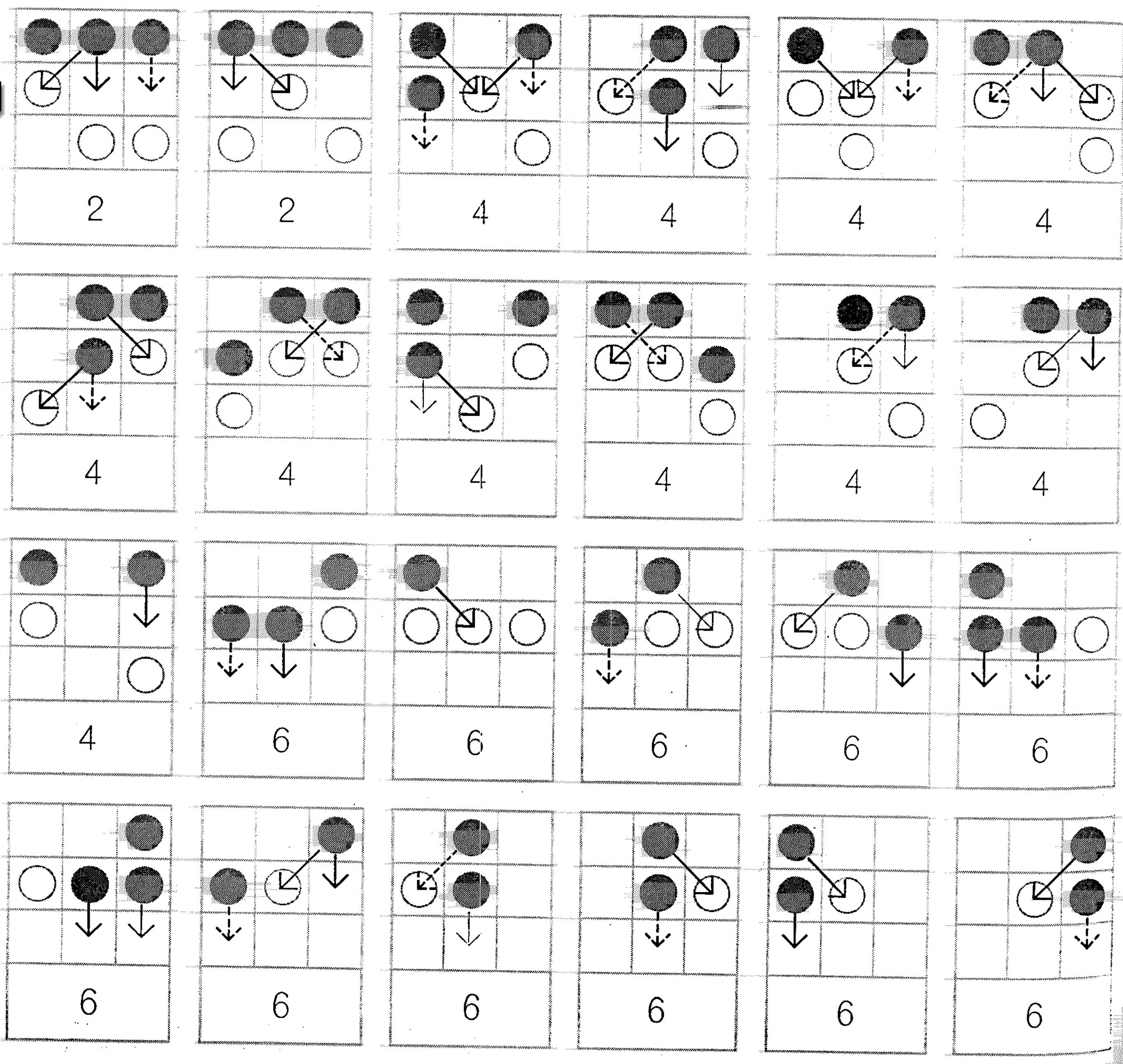
Hexapawn is played on a 3 × 3 board, with three chess pawns on each side as shown in the illustration on page 138. Dimes and pennies can be used instead of actual chess pieces. Only two types of move are allowed: (1) A pawn may advance straight forward one square to an empty square; (2) a pawn may capture an enemy pawn by moving one square diagonally, left or right, to a square occupied by the enemy. The captured piece is removed from the board. These are the same as pawn moves in chess, except that no double move, *en passant* capture or promotion of pawns is permitted.

The game is won in any one of three ways:

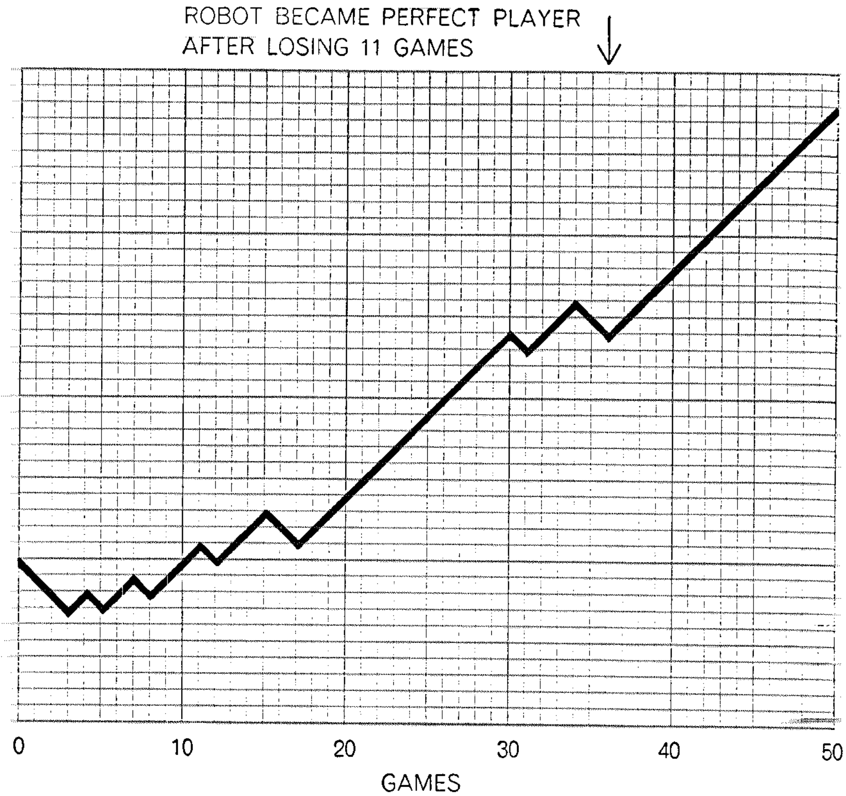
1. By advancing a pawn to the third row.
2. By capturing all enemy pieces.
3. By achieving a position in which the enemy cannot move.

Players alternate moves, moving one piece at a time. A draw clearly is impossible, but it is not immediately apparent whether the first or second player has the advantage.

To construct HER (Hexapawn Educable Robot) you need 24 empty matchboxes and a supply of colored beads. Small candies that come in different



Labels for HER matchboxes

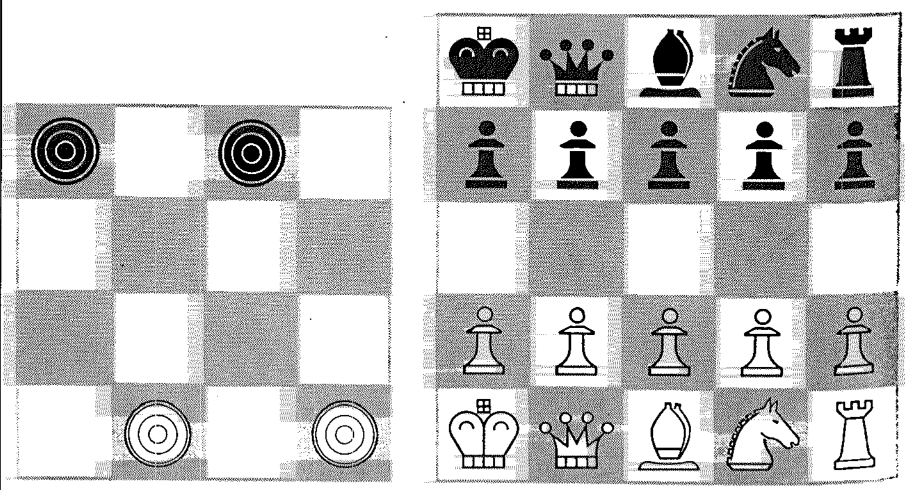


Learning curve for HER's first 50 games (downslant shows loss, upslant a win)

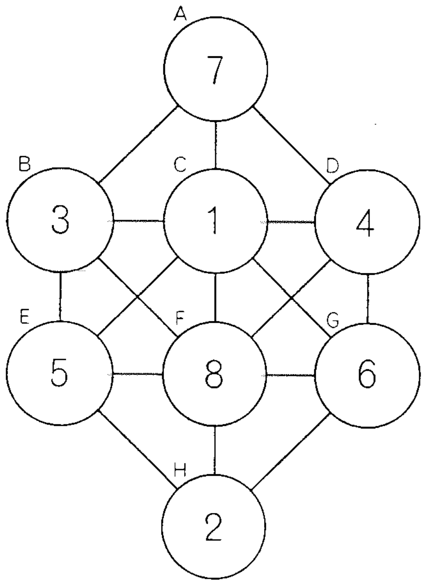
colors work nicely. (I used five five-cent boxes of jujubes.) Each matchbox bears one of the diagrams shown in the illustration on page 140. The robot always makes the second move. Patterns marked "2" represent the two positions open to HER on the second move. You have a choice between a center or an end opening, but only the left end is considered because an opening on the right would obviously lead to identical (although mirror-reflected) lines of play. Patterns marked "4" show the 11 positions that can confront HER on the fourth (its second) move. Patterns marked "6" are the 11 positions that can face HER on the sixth (its last) move.

Inside each box place a single bead to match the color of each arrow on the pattern. The robot is now ready for play. Every legal move is represented by an arrow; the robot can therefore make all possible moves and only legal moves. The robot has no strategy. In fact, it is an idiot.

The teaching procedure is as follows. Make your first move. Pick up the matchbox that shows the position on the board.



Matchbox machine can be built for simplified checkers (left) *but not for chess* (right)



Solution to last month's Problem 1

Shake the matchbox, close your eyes, open the drawer, remove one bead. Close the drawer, put down the box, place the bead on top of the box. Open your eyes, note the color of the bead, find the matching arrow and move accordingly. Now it is your turn to move again. Continue this procedure until the game ends. If the robot wins, replace all the beads and play again. If it loses, punish it by confiscating only the bead that represents its *last* move. Replace the other beads and play again. If you should find an empty box (this rarely happens), it means the machine has no move that is not fatal and it resigns. In this case confiscate the bead of the preceding move.

Keep a record of wins and losses so you can chart the first 50 games. The top illustration on page 142 shows the results of a typical 50-game tournament. After 36 games (including 11 defeats for the robot) it has learned to play a perfect game. The system of punishment is designed to minimize the time required to learn a perfect game, but the time varies with the skill of the machine's opponent. The better the opponent, the faster the machine learns.

The robot can be designed in other ways. For example, if the intent is to maximize the number of games that the machine wins in a tournament of, say, 25 games, it may be best to reward (as well as punish) by adding a bead of the proper color to each box when the machine wins. Bad moves would not be eliminated so rapidly, but it would be less inclined to make the bad moves. An interesting project would be to construct a second robot, HIM (Hexapawn Instructable Matchboxes), designed with a different system of reward and punishment but equally incompetent at the the start of a tournament. Both machines would have to be enlarged so they could make either first or second moves. A tournament could then be played between HIM and HER, alternating the first move, to see which machine would win the most games out of 50.

Similar robots are easily built for other games. Stuart C. Hight, director of research studies at the Bell Telephone Laboratories in Whippany, N.J., recently built a matchbox learning machine called NIMBLE (Nim Box Logic Engine) for playing nim with three piles of three counters each. The robot plays either first or second and is rewarded or punished after each game. NIMBLE required only 18 matchboxes and played almost perfectly after about 30 games.

By reducing the size of the board the complexity of many familiar games can be minimized until they are within the scope of a matchbox robot. The game of go, for example, can be played on the intersections of a 2 × 2 checkerboard. The smallest nontrivial board for checkers is shown at the left in the bottom illustration on page 142. It should not be difficult to build a matchbox machine that would learn to play it. Readers disinclined to do this may enjoy analyzing the game. Does either side have a sure win or will two perfect players draw? (The answer will be given next month.)

When chess is reduced to the smallest board on which all legal moves are still possible, as shown at the right in the bottom illustration on page 142, the complexity is still far beyond the capacity of a matchbox machine. In fact, I have found it impossible to determine which player, if either, has the advantage. The game is recommended for computer experts who wish to program a simplified chess-learning machine and for all chess players who like to sneak in a quick game during a coffee break.

The answers to last month's collection of short problems are given below:

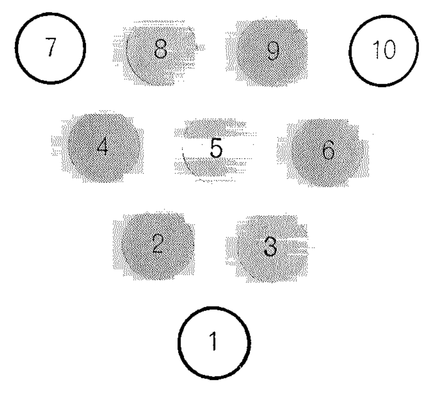
1.

If the numbers from 1 to 8 are placed in the circles as shown in the illustration at the top of this page, no number will be connected by a line to a number immediately above or below it in serial order. The solution (including its upside-down and mirror-image forms) is unique.

L. Vosburgh Lyons, who gave me this puzzle, solved it as follows. In the series 1, 2, 3, 4, 5, 6, 7, 8 each digit has two neighboring numbers except 1 and 8. In the diagram, circle C is connected to every circle except H. Therefore if C contains any number in the set 2, 3, 4, 5, 6, 7, only circle H will remain to accommodate *both* neighbors of whatever number goes in C. This is impossible, so C must contain 1 or 8. The same argument ap-

GAMES	1	2	3	4	5	6	7	8	9
SERVER	R	M	R	M	R	M	R	M	R
WINNER	R	M	R	M	R	M	M	M	M
GAMES WITH SERVICE	X	X	X	X	X	X		X	
GAMES V. SERVICE							X		X

Chart for Problem 3



Impossibility proof for Problem 4