INFORMATION PROCESSING USING A MODEL OF ASSOCIATIVE MEMORY

Kaoru Nakano & Jin-ichi Nagumo
Faculty of Engineering
University of Tokyo
Tokyo, Japan

Summary

Association mechanisms are considered to play Important roles in thinking in the human brain. Such mechanisms should be ana]ysed and utilized In machine intelligence in order to further work in the field. This paper describes a model of a neural network named 'Associatron', and considers its applications for Information processing. The Associatron stores a lot of entities in the same region of its structure, and recalls the whole of any entity from a part of it without any sequential search. From its structure, some properties are derived that are expected to be useful for human-like information processing. Analysis of these properties is briefly described. An Associatron with i80 neurons has been simulated by a computer and has been applied to simple examples of concept formation and game playing. Hardware realization of an Associatron with 25 neurons made for trial purposes, is outlined, too.

Introducti on

The purpose of this paper is to outline an approach) to simulating certain functions of a human brain from the viewpoint of association. The first aspect of this work Is to design an associative memory device that is considered to be reasonable as a model for association mechanisms. The second 1s to distinguish the characteristics of information processing using the device, from that of conventional information processing.

Association in the human brain has been studied mainly in the field of psychology, and some semantic models for association have been presented in the last few years. On the other hand, biological studies are beginning gradually to reveal the structure of the nervous system. But Knowlege about it Is still not enough to construct the structure artificially, although models *of* nerve cells have been presentedl.

In this situation, it will be important to try to construct a machine with homogeneous structure, where the microscopic behavior of Its components combines to form the macroscopic behavior, such as pattern recognition, concept formation, game playing and so on. Perceptron[2], Adaline[3] and other learning machines have this property to some extent, although they do not provide the functions of a lifelike memory or a memory-oriented information processor. Work concerning a rather biological associative memory was presented by Post[4]. His model 1s a distributed memory device which memorizes triplets of entities and recalls one of these entities from two other entitles in the specific triplet. The model should be helpful for realizing lifelike Information processing.

The Assoclatrorn[5,6] proposed here is one of this kind of model, whose structure is more similar to the nervous system in the human brain and more general so that a larger[1] part of the function of the brain will possibly be explained by the use of the model. In this model a lot of entities are stored in the same region of its structure, and any stored entity can be recalled from parts of it without any sequential search. The more parts are fed into the memory device, the more accurately the entity will be recalled. The principle 1s based on the application of auto-correlation functions. If the auto-correlation function of an entity is held in the memory, the entity can be easily reproduced from only a small part of it. Since storing all these auto-correlation functions is too redundant to be practical, they are linearly added in the memorizing process. Therefore, if the stored entitles are numerous, entities cannot always be recalled completely, but they are expected to be reproduced only probabilistically.

This kind of associative memory has the following properties, differing from those of conventional memory;

(1) Reliability is improved in the sense that the failure of some components of the memory device does not cause the loss of the whole of any entity, but cause only inaccuracy in recalling.

(2) Although some uncertainties appear in recalling, they are rather useful for realizing an artificial intelligence with flexibility and applicability. This is because a strictly logical comparison, for example, is apt to discard useful data which resembles certain other data. The uncertainty resulting from a confusion among similar data will make the

device extract automatically, from a number of entities, the essence useful for a certain job.

(3) Since information is accessed associatively without sequential search, the speed of access or information retrieval is very high.

These properties differentiate information processing using the associative memory from that of conventional computers.

Recently, the thinking process has been studied in detail using so-called heuristic programs. For example, the learning program for the game of checkers[7,8] is well known. The model in this paper is related to such heuristics in game playing, general problem solving and possibly to the thinking process itself.

A Model of Associative Memory

A mode] of associative memory realized in the following way is named an 'Assoclatron':

Presume that an entity is represented by a row vector

$$x = (x_1, x_2, \ldots, x_n), \tag{1}$$

where $X_1 = -1, 0,]$. An entity is composed of several patterns and of neutral areas. Each pattern is composed *of* -l's, 0's and l's, while the neutral area Is composed only of 0's. Each pattern has a simple *or* complex meaning, and the entity represents the association of these patterns.

Now the memorizing process is considered. Let x^r denote the transposed vector of x. The inner state of the memory, after k entitles $x^{(1)}, x^{(2)}, \ldots,$ $x^{(k)}$ have been stored, is defined as an n x n matrix

$$M = x^{(1)\tau} x^{(1)} + x^{(2)\tau} x^{(2)} + \ldots + x^{(k)\tau} x^{(k)}. \tag{2}$$

To recall entities, first we define the quantizing function,

$$\phi(t) = \begin{cases} -1 & \text{if } t < 0 \\ 0 & t = 0 \\ 1 & t > 0 \end{cases} \tag{3}$$

Suppose that it also can be applied to a vector u = (uj) and a matrix A = (a_{ij}), as

$$\begin{aligned} \phi(u) &= (\phi(u_j)) \\ \phi(A) &= (\phi(a_{ij})). \end{aligned} \tag{4}$$

From an input $y = (y_1, y_2, \ldots, y_n)$, the same kind of vector as x, the memory device can recall a row vector z as the output, where

$$z = \phi(y\phi(M)). \tag{5}$$

If y is composed of a neutral area and a few patterns which are also the components of a stored entity x, then it is possible that y is nearly equal to x, even when a lot of entities are memorized. This would enable recalling the whole of a stored entity from a part of it. Consequently, a few patterns previously processed can be associated with the rest of the patterns of entity x.

Figure 1 depicts the behavior of the Assoclatron as a neural network. Any pair of neurons xi and xj is connected through a memory unit mij; , which corresponds to the synapse. This model differs from general neural networks in the following points,

(1) All neurons are mutually connected .

(2) Kach neuron has three possible states.

(3) For any pair of neurons, the values of synaptic conductance are assumed to be equal for both directions. The value of the memory unit my Is multivalued and Is modified during a memorizing process by adding the product Xi, Xj to the previous value. However, in propagation of a stimulus, the quantized value 1 s used .

(4) Refractory time is not considered .

From these items, It can be said that the Assoclatron Is at best a considerably simplified model of a neural network. This simplification might cause a situation where separation of patterns is not very good, reversed patterns cannot be separated and the direction of recall is out of control. The first problem will be solved by increasing neurons, and the other problems by modifying the model. But the modification is not always necessary, because the results turn out predictably when the model is applied appropriately.

Properties of the Model of Associative Memory

In this section, properties of memorizing and recalling entitles are discussed. It was previously mentioned that an entity is composed of patterns, each of which has a meaning. Here we consider it in a little more detail. Figure *2* illustrates the relation of an entity and its component patterns; that is Fig. *2* (a) shows the array of components of vector x and (b) shows an example *of* entities. The entity is composed of patterns 1, 2, 3 and a neutral area, where, pattern 1 has the meaning of 'red', pattern *2* the meaning 'spherical', etc.. Now it is reasonable to consider that a pattern or a few patterns establish a concept. In the present example, pattern 1 has the concept of 'red', pattern 2 the concept of 'spherical', the set of patterns 1 and 2 has the complex concept of 'red and spherical', the set of patterns 1, ? and 3 has the concept of 'apple' and so on. In the Associatron, when entities or associations of patterns are stored, not only can patterns be recalled, but also various concepts are gradually formed.

The following discussion considers the properties of memorizing and recalling.

Presume that Q denotes the set of all n-dimentional vectors

$$x = (x_1, x_2, \ldots, x_n) \tag{6}$$

where $x_i = -1, 0, 1$. Now, we consider the recalling function

$$\alpha : Q \to Q.$$

If $X \in Q$, and α is defined as

$$z = \alpha(x) = \phi(x \cdot \phi(M)), \tag{7}$$

then z is also a member of Q. Now let us introduce the index vector

$$v = (v_1, v_2, \ldots, v_n), \tag{8}$$

where V; is equal to 0 or 1 . This vector' represents a certain area of the neural network. Defining elementwise multiplication * as

$$v * x = (v_1 x_1, v_2 x_2, \ldots, v_n x_n), \tag{9}$$

we call v * x the concept of x at the area v. Presume that both x and y are members of Q. If

$$v * x = v * y, \tag{10}$$

then x and y are said to have the same concept at area v. The measure of area v is defined by

$$l(v) = \sum_{i}^{n} v \tag{11}$$

Now It is presumed that the entity x or the association of patterns A and B is stored in the memory device, as in Fig. 3 (a), whose representation is the same as Fig. 2. If the order of the elements of x Is changed to put together the elements constructing the same pattern, the entity x is represented by a row vector as

$$x = (A, B, 0), \tag{12}$$

where A and B are row vectors, and 0 denotes zero-vector. Then the matrix M which stores only x will be

$$M = (A, B, 0)^t (A, B, 0) = \begin{pmatrix} A^tA & A^tB & 0 \\ B^tA & B^tB & 0 \\ 0\,0 & 0 & 0 \end{pmatrix} \tag{13}$$

If the index vectors of A and B are v_A and v_B, respectively, the concept at area v_B of the recalled pattern from $v_A * x$ is

$$\begin{aligned} v_B * \phi((v_A * x)\cdot\phi(M)) &= v_B * \phi(A\cdot\phi(A^tA), A\cdot\phi(A^tB), 0) \\ &= (0, A\,\phi(AB), 0) \\ &= (0, B, 0) \end{aligned} \tag{14}$$

This means that the pattern of concept B is completely recalled from A.

Where associations A-B and C-P are stored independently, as shown in Fig. 3 (b), :n the same way it can be seen that a pa*tern of each pair is completely recalled from another.

In the case of Fig. 3 (c), two associations, A-B and C-B', are stored in such a way that pattern B and B' are over lapped. Stored vectors are

$$\begin{aligned} x &= (A, B, 0, 0), \\ y &= (0, B', C, 0). \end{aligned} \tag{15}$$

The matrix is

$$M = \begin{pmatrix} A^tA & A^tB & 0 & 0 \\ B^tA & B^tB+B'^tB' & B'^tC & 0 \\ 0 & C^tB' & C^tC & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \tag{16}$$

The recalling process is as follows,

$$\begin{aligned} &v_B * \alpha(A, 0, 0, 0) = \phi(0, A\cdot\phi(A^tB), 0, 0) \\ &= (0, B, 0, 0) \\ &v_A * \alpha(0, B, 0, 0) = \phi(B\cdot\phi(B^tA), 0, 0, 0) \\ &= (A, 0, 0, 0) \end{aligned} \tag{17}$$

In the case of Fig. 3 (d), where two associations, A-B and C-F, are stored, A and C, B and F partially overlap, respectively. This is rewitten as Fig. 3 (e), where the overlapping areas are delt with as new patterns. From the previous result, it is clear that F and F are indifferent to the recalling AB -> CD, where AB means the complex concept of A and B etc, so they are eliminated. Therefore, stored vectors can be considered as

$$\begin{aligned} x &= (A, B, C, D, 0), \\ y &= (0, B', 0, D', 0) \end{aligned} \tag{18}$$

The recalling process from AB to CD is as follows,

$$\begin{aligned} &v_{CD} * \alpha(v_{AB} * x) \\ &= (0, 0, \phi(A\phi(A^tC) + B\phi(B^tC)), \phi(A\phi(A^tD) \\ &\quad + B\phi(B^tD + B^tD')), 0) \\ &= (0, 0, C, D, 0) \end{aligned} \tag{19}$$

Thus, CD can be completely recalled from

AB, because only two patterns are overlapped at VB. When this number is arbitrary, the following argument will hold.

Let k pairs of A_j -B_j be stored as shown in Fig. 3 (f). For simplicity of calculation, k is presumed to be an odd number. Besides, it is presumed that $\ell(VA) = s$ is odd and that the entities are random patterns at areas v_A and v_B. These assumptions do not necessarily hold in actual use of the associative memory. But it matters little, because this only causes a small error in estimation of accuracy of recalling. In any recalling process, the probability that a memory unit votes for the right state of a neuron for the specific entity stored in the memory device is

$$(k + 1)/2k. \tag{20}$$

The probability that the state of a neuron decided by majority is right, is represented by the sum of the first (s+1)/2 terms of the binomial expansion of

$$\left(\frac{k + 1}{2k} + \frac{k - 1}{2k}\right)^s \tag{21}$$

That is,

$$P = \left(\frac{1}{2k}\right)^s \sum_{i=0}^{\frac{s-1}{2}} {}_sC_i (k + 1)^{s-i} (k - 1)^i , \tag{22}$$

where C ; is the number of combination. As this summation cannot be written as a simple form, to determine the behavior of this equation, the graphical expression is taken by calculating the values of P for various values of k and s. This is shown in Fig. 4. From the graph, it is found that completely accurate recalling can be done when the number of stored entitles is very small or when the number of elements constructing the input pattern is very large. But it is important that the capability of an associative memory is not judged only by the accuracy of memory. Characteristics of information processing using an associative memory are shown in the following sections.

Computer Simulation and Some Experimental Results

An Associatron composed of 180 neurons has been simulated by a digital computer. Using this simulated model, several experiments were performed. To evaluate the results, the correlation coefficient between a stored pattern and the recalled pattern is used. Presume that both patterns A and B consist of n elements, and that corresponding elements of patterns are a and bi, respectively. The correlation coefficient is defined as

$$r = \frac{1}{n} \sum_{i=1}^{n} a_i b_i . \tag{23}$$

r = 1 means that A and B are the same. When r equals zero, they are indifferent, and when r equals - 1, one is the reversed pattern of the other. Although input patterns for recalling should also be memorized, for convenience of evaluation of the behavior of the memory, they are not memorized in these experiments.

<u>Example 1</u> Three areas A, B and C, each of which consists of 40 neurons are taken on the neural plane. Each of the names of 6 things, 3 colors and 3 shapes is coded into a 40 bit pattern of l's and -l's. A thing, its color and its shape are stored at areas A, B and C, respectively. After 6 triplets are stored, the memory device is made to recall the color and the shape from a thing, and vice versa. The results are shown in Fig. 5. The device recalls the color and the shape completely from the thing, but the accuracy of reverse recalling is about 92 %.

<u>Example 2</u> An experiment concerning simple concept formation was accomplished. In this case, for example, red things and the word 'red' such as Apple-red-spherical-word'red', brick-red-boxlike-word 'red' are shown one by one to the memory device. At first, the memory device might take the word 'red' for 'spherical', but as it is trained, it forms the concept 'red' as shown in Fig. 6, where symbols are used instead of things, shapes colors etc.. Thus, the pattern of 'red' is associated with the pattern of the word 'red', and the memory device recalls one from another. Besides, if entitles apple-red and apple-spherical are stored at other times, the device recalls 'red' and 'spherical' from 'apple' and vice versa. That is, although the triplet apple-red-spherical is not a stored entity, it is formed in the memory device.

<u>Example 3</u> When the associative memory device is applied to games, a set of patterns effective for winning the game is expected to be extracted automatically. The following game is used for a demonstration of the learning process in the associative memory. There are n chips on the board initially. Two players take any number of one to three chips alternately. The player who is forced to take the last chip loses the game. The restriction is that a player must not take the number which his opponent took at his previous move. For simplicity, presume that there are only six chips on the initial board. To play the game using the Associatron, first the components of vector x are assigned as shown in Fig. 7 to four patterns of the board position, the move, the mutual effect of the board

position and the move, and the image of the result. Learning is performed in such a way that, all sets of patterns of board position, strategy, mutual effect and the result, which have appeared in the game, are memorized one by one. The player with the associative memory (Player A), at every move, recalls the image of the result from the board position, a possible strategy and the mutual effect. The strategy, from which the image of 'win' is recalled, is taken. If the images are the same through possible strategies, one strategy is chosen at random. In this simple way, Player A is expected to make progress in developing his skill in the game, utilizing the property of the associative memory. In this experiment, it is assumed that the opponent (Player R) chooses one of possible strategies at random. In training after each game, for the opponent's move, the image of the result Is reversed from 'win' to 'lose' and vice versa, and Is used. Therefore, Player A makes progress using his opponent's strategies, too. Eighteen games have been played. As the result, the winners are H, A, A, A, A, A, A, A, A, R, A, A, A, R, R, A, R, A. Examination of the game tree yields the best player's strategy at every board position. Strategies of Player A are examined after a certain number of games, as shown below. At every possible board position, a check is made as to which image of 'win' or 'lose' Player A recalls from the set of board position, possible strategy at the board and their mutual effect. The result of the examination is shown in Fig. 8. The winning rate of Player A to Player R is shown, too. Figure 9 shows it graphically. From the graph, It is found that the associative memory can learn how to play the game well.

Hardware Realization

The hardware can be constructed as an iterative circuit as shown in Fig. 10 (a). Since the matrix in Eq. (2) is symmetrical, only half of the nxn memory units are required. The memory unit is represented by a simple automaton operating synchronously. Presume that the notation Is assigned as in Fig. 10 (b). The automaton is described as

$$\begin{aligned} S(t+1) &= S(t) + X_1(t)\,X_2(t) \\ Y_1(t+1) &= \phi(S(t))\,X_2(t) + X_3(t) \\ Y_2(t+1) &= \phi(S(t))\,X_1(t) + X_4(t) \\ Y_3(t+1) &= X_1(t) \\ Y_4(t+1) &= X_2(t), \end{aligned} \tag{24}$$

where S sorresponds to the value of the memory unit and t is time. For the value of the unit, few levels are enough to operate the device specifically.

In this way, a memory device with 25 neurons, that is, 325 memory units, has been constructed for trial. Figure 11 shows the appearance of the device. The levels of the memory unit previously mentioned are taken to be seven. Each memory unit is composed of 20 integrated circuit elements. A block diagram of this device is shown in Fig. 12. For convenience, input and output parts are clearly separated In this device, and if the 'Memory Hold' is used, the device wi]l recall entities without changing its inner state. Any entity can be set in the input register with the input switches. The output can be transferred to the input register manually or automatically. If some of the selecting switches are turned on, corresponding states of the input register are obtained from the Input switches instead of the output register. Thus, the recalled pattern can be immediately modified and be used as the next input. Although this device is very small in number of neurons, it is effective for some experiments concerning sequential recalling. The Importance of this kind of experiment is shown in the next section.

Thinking by the Sequence of Associations

If the recalling process is repeated in the method where recalled pattern is used as the next input, a chain of associations may be traced. The process is considered as one of the thinking processes, rather than one of recalling the stored entitles, because the structure of the associations was not directly memorized but has been formed in the memory through experience from a set of associations .

Though the real thinking process may be far more complicated, here we consider the simplest loop A->B->C->A by setting

$$x = (A, B, O), \quad y = (O, B, C), \\ z = (A, O, C) \tag{25}$$

where $\ell(v_A) = \ell(v_B) = \ell(v_C) = 60$. After a number of such loops have been stored, an initial input of u = (A', 0, 0) lo- used to recall patterns sequentially, where A' is an arbitrary pattern. Jn case the stored loop number Is less than six, the loops can be separated completely. That coincides with the result of theoretical evaluation. When the number of loops is larger, various kinds of confusion take place. The behavior of the associative memory in sequential recalling is being studied, but no particularly interesting results have been obtained yet.

Conclusions

Information processing based on as-ssociative memory was studied. The mode], named an Assoclatron, memorizes entities distributively and recalls them associatively. Consequently, it has different properties from conventional memory devices. Those properties were analyzed and found useful for realizing machine intelligence. The ability of an Assoclatron increases as the number of neurons increases. A few examples, using the model as simulated by a digital computer, show that concept formation and game playing with learning are possible. In the Example of game playing, it is found that the application of the method is not restricted to a special game, because it fully utilizes the properties of associative memory and the method itself is very primitive and simple. Hardware of the Assoclatron with 25 neurons, made for trial purposes, uses integrated circuit elements. Although consideration is limited to 'static' properties, or the properties of single recalling, it is suggested that 'dynamic' behavior, or' the sequence of associations, is more important .

Acknowledgement

This work was supported by a grant from the Science and Technology Agency of Japan.

References

1. E. R. Caianiello, "Outline of a Theory of Thought Processes and Thinking Machines," Journal of Theoretical Biology, vol. 1, no. 2, pp. 204-235 , April 1961.

2. F. Rosenblatt, "The Perceptron—A Probabilistic Model for Information Storage and Organization in the Brain," Psychol. Rev., vol. 65, pp. 386-407, 1958.

3. B. Widraw, "Generalization and Information Storage in Networks of Adaline 'Neurons'," Self-organizing Systems, pp. 435-461, Spartan Books, Washington, D. C. , 1962.

4. P. B. Post, "A Lifelike Model for Association Relevance," Proc. Int. Joint Computer Conf. on Artificial Intelligence, pp. 271-280, May 1969.

5. K. Nakano and J. Nagumo, "Studies on Associative Memory Using a Model of Neural Network," Conf. on Medical Electronics of Institute of Electronics and Communication Engineers of Japan, June 1970.

6. K. Nakano, "Learning Process in a Model of Associative Memory," Japan, U. S. Seminar on Learning Process in Control systems, August 18-20, 1970, Nagoya, Japan.

7. A. Samuel, "Some Studies in Machine Learning Using the Game of Checkers," IBM J. Res. Develop, vol. 3, no. 3, pp. 210-229, July 1950.

8. A. Samuel, "Some Studies in Machine Learning Using the Game of Checkers II—Recent Progress,"IBM J. Res. Develop, vol.11, no. 6, p. 601, November 1967.

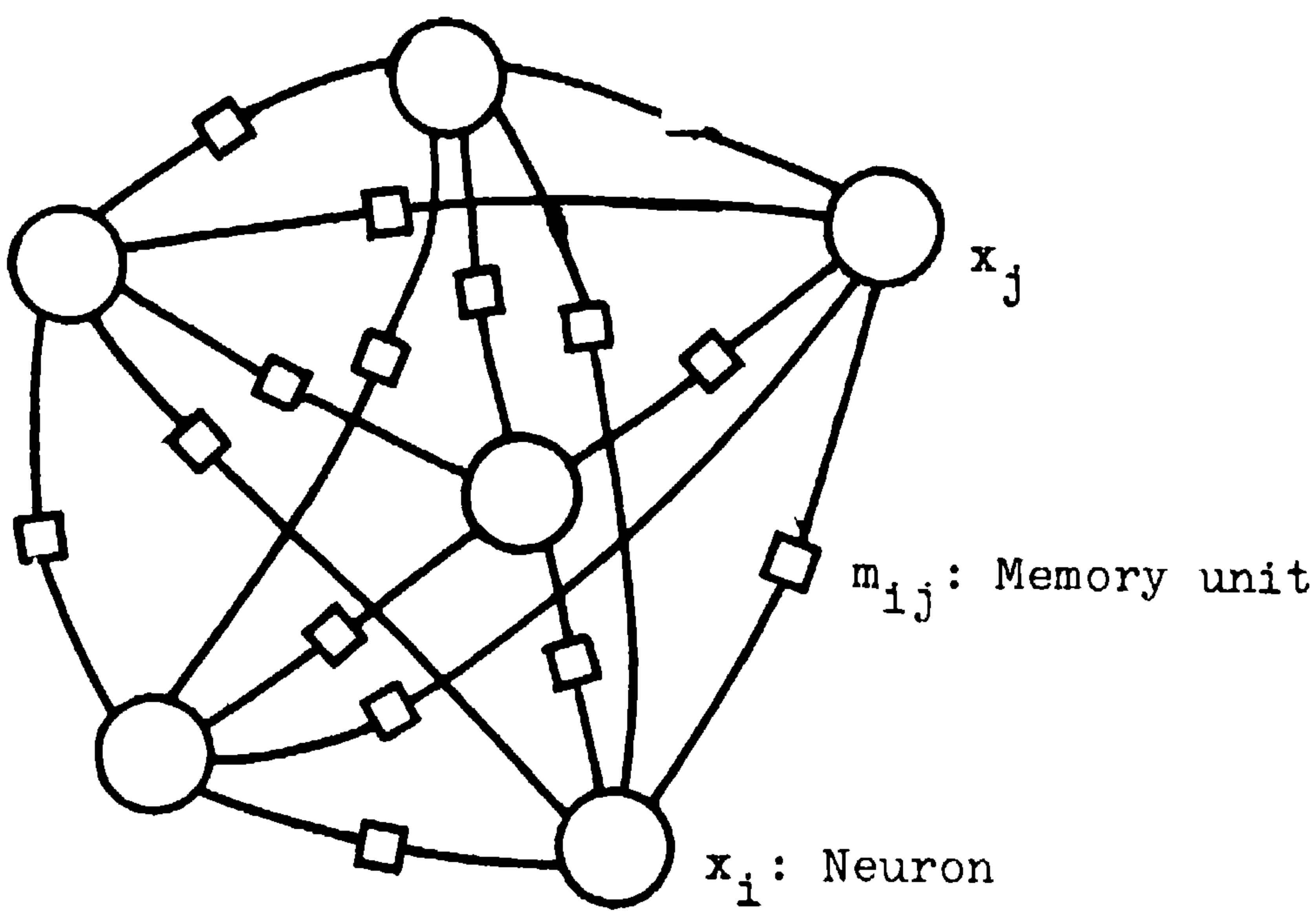

Fig. 1 Associatron as a neural network

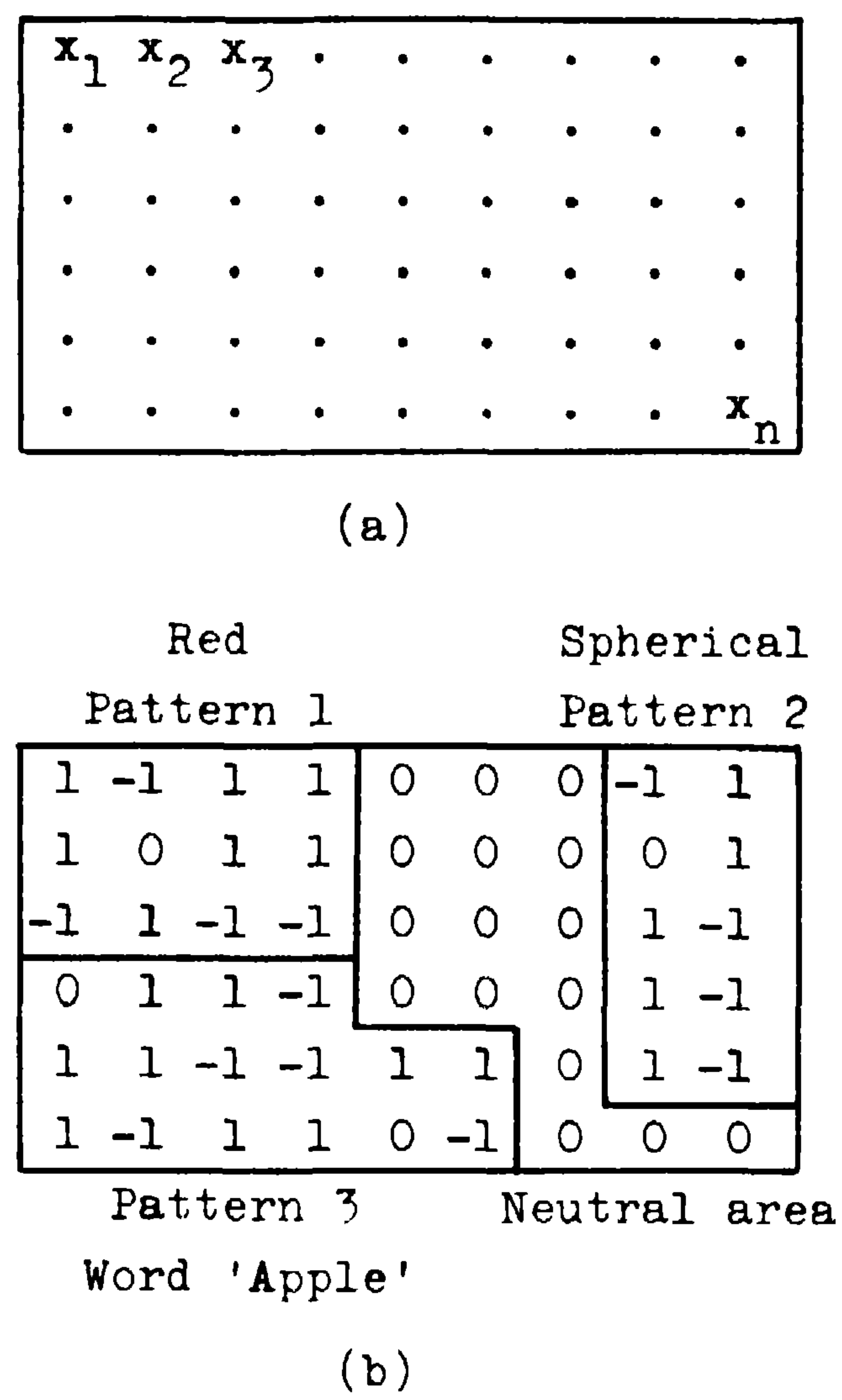

Fig. 2 Configuration of an entity

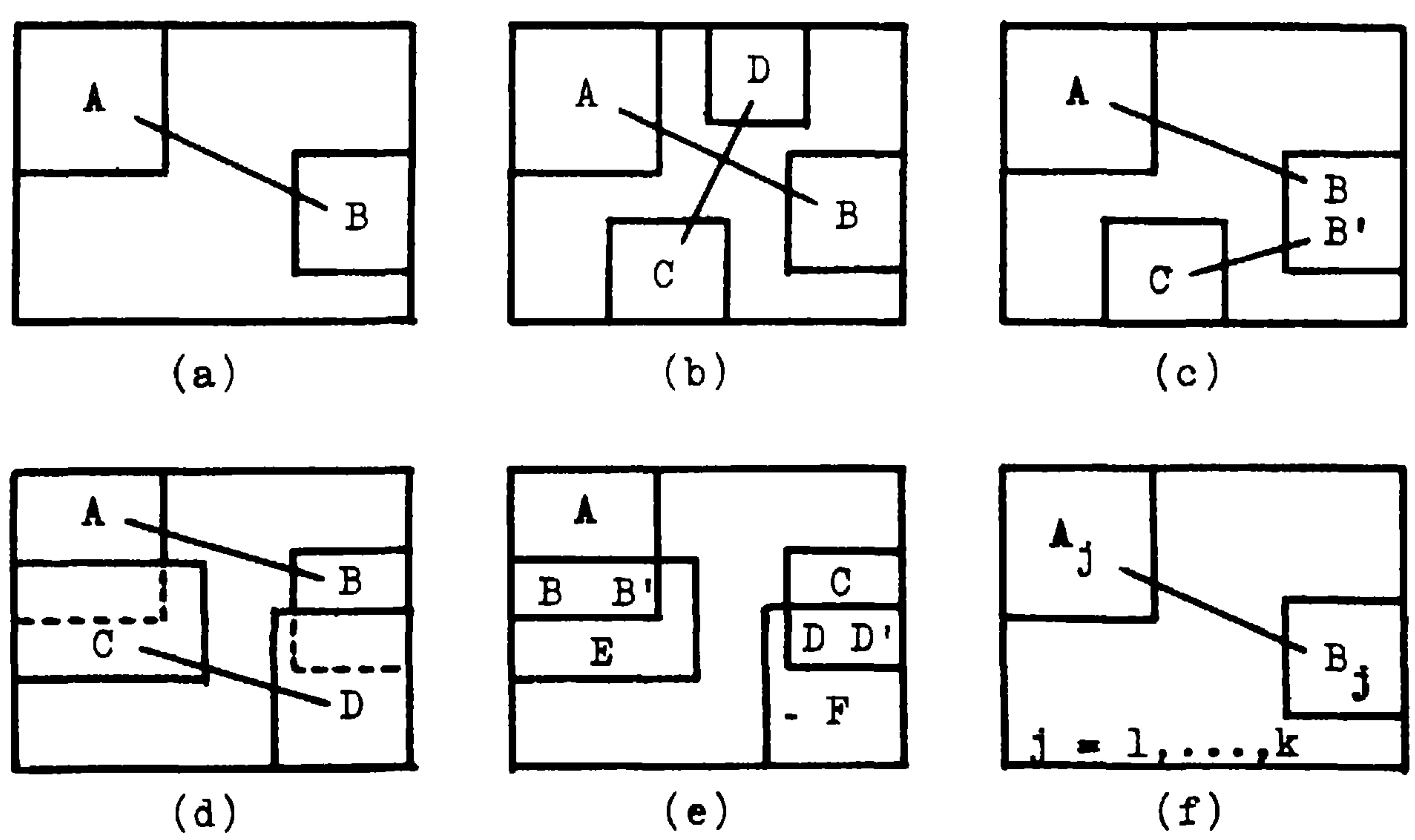

Fig. 3 Various associations

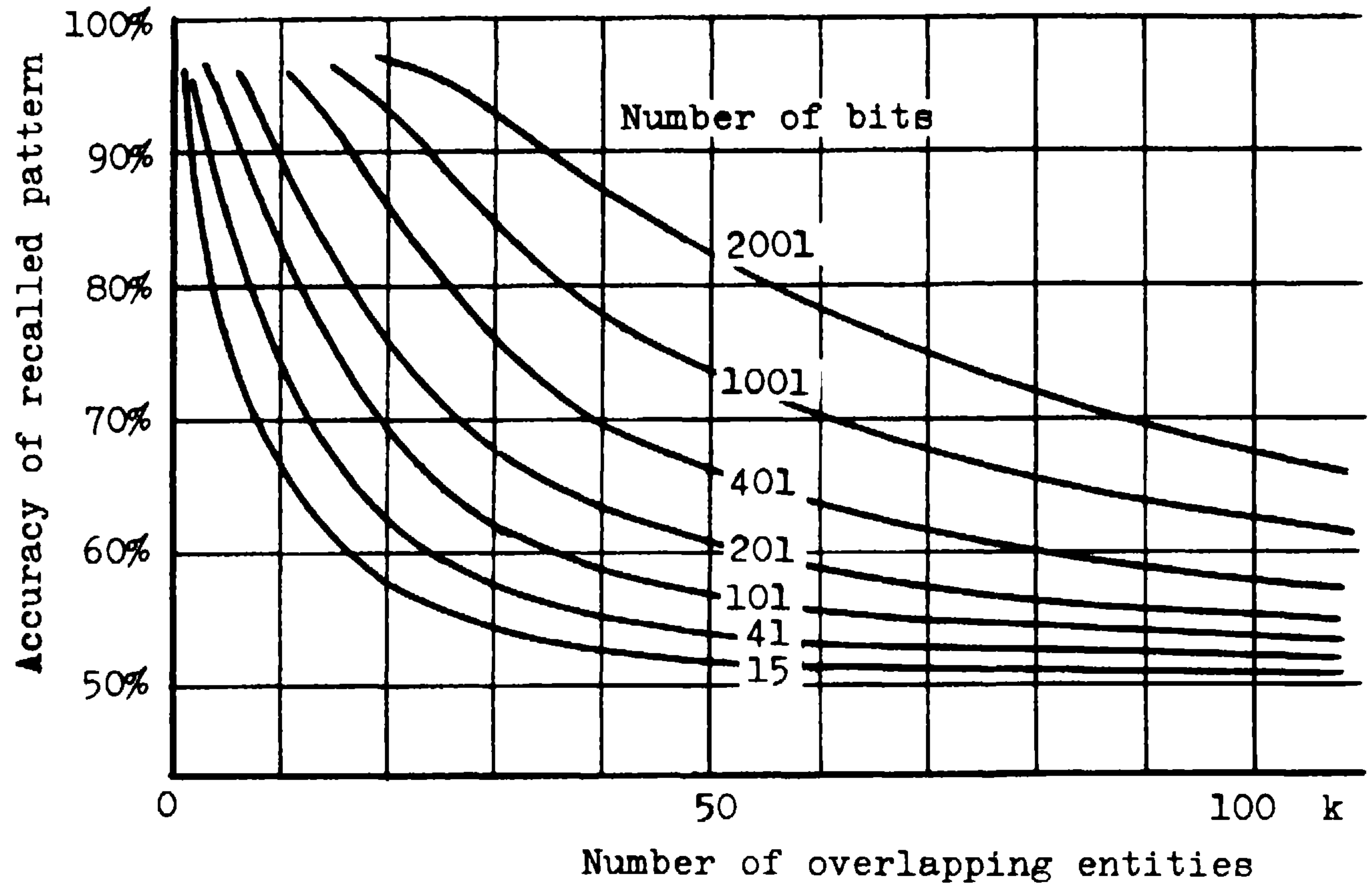

Fig. 4 Accuracy of recalling

Memorize

Stored triplets

1. Apple-Spherical-Red
2. Watermelon-Spherical-Green
3. Banana-Sticklike-Yellow
4. Wooden box-Boxlike-Yellow
5. Cucumber-Sticklike-Green
6. Brick-Boxlike-Red

Recall

	(Accuracy)
Apple → Spherical, Red	(r = 1.00)
Watermelon → Spherical, Green	(r = 1.00)
Banana → Sticklike, Yellow	(r = 1.00)
Wooden box → Boxlike, Yellow	(r = 1.00)
Cucumber → Sticklike, Green	(r = 1.00)
Brick → Boxlike, Red	(r = 1.00)
Spherical, Red → Apple	(r = 0.78)
Spherical, Green → Watermelon	(r = 0.75)
Sticklike, Yellow → Banana	(r = 0.78)
Boxlike, Yellow → Wooden box	(r = 0.88)
Sticklike, Green → Cucumber	(r = 0.78)
Boxlike, Red → Brick	(r = 0.88)

Fig. 5 Concept formation 1

Memorize

Thing-Shape-Color-Word

1. $A_1 - B_1 - C_1 - C_1'$
2. $A_2 - B_2 - C_1 - C_1'$
3. $A_3 - B_3 - C_1 - C_1'$
4. $A_4 - B_2 - C_2 - C_2'$
5. $A_5 - B_1 - C_2 - C_2'$
6. $A_6 - B_3 - C_2 - C_2'$
7. $A_1 - B_1 - C_1 - B_1'$
8. $A_2 - B_2 - C_1 - B_2'$
9. $A_4 - B_2 - C_2 - B_2'$
10. $A_5 - B_1 - C_2 - B_1'$

Word → Concept	Concept → word
$C_1' \to C_1$	$C_1 \to C_1'$
$C_2' \to C_2$	$C_2 \to C_2'$
$B_1' \to B_1$	$B_1 \to B_1'$
$B_2' \to B_2$	$B_2 \to B_2'$

Thing → Shape, Its word, Color, Its word

$A_1 \to B_1,\ B_1',\ C_1,\ C_1'$

$A_2 \to B_2,\ B_2',\ C_1;\ C_1'$

All recallings are completely accurate.

Fig. 6 Concept formation 2

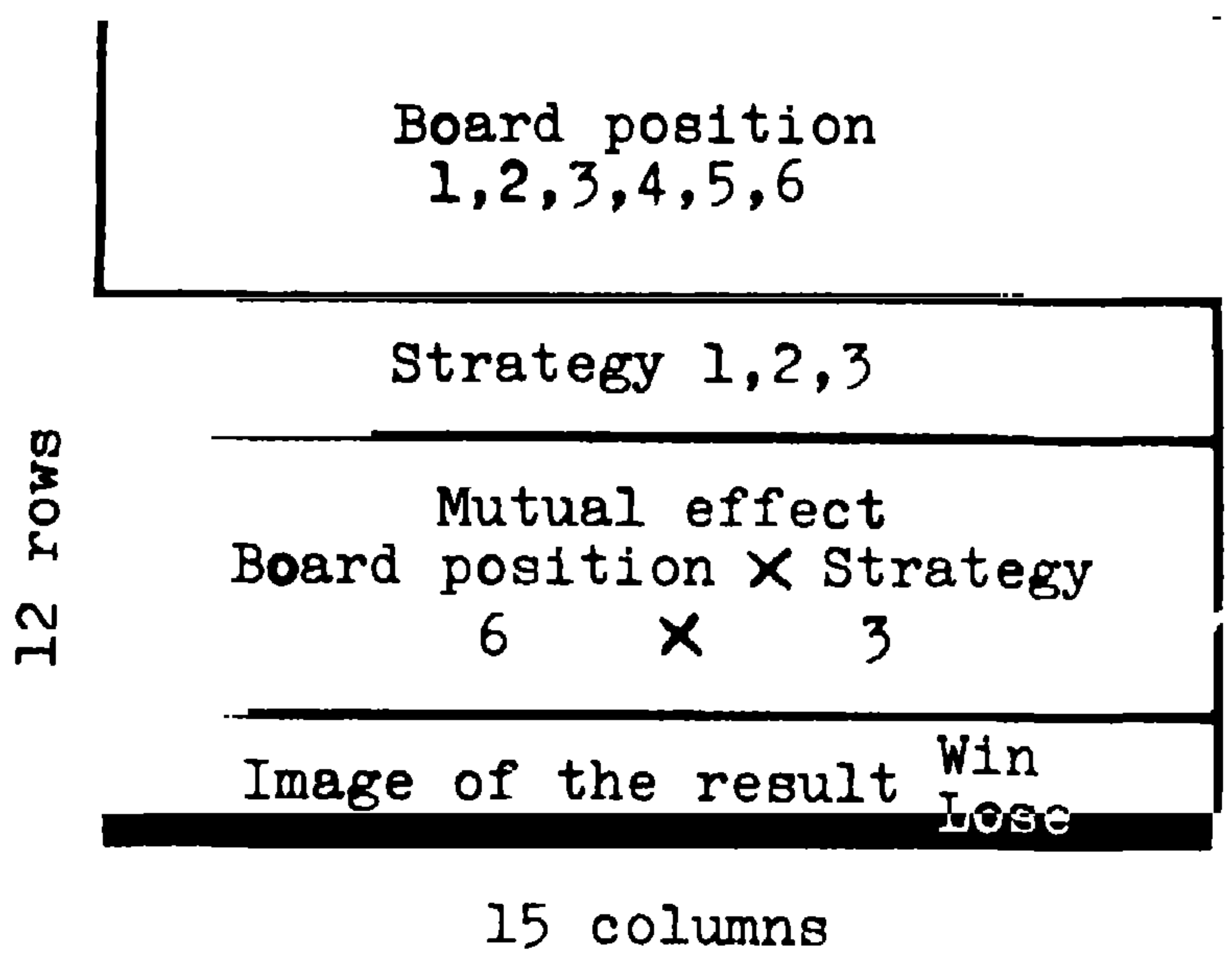

Fig. 7 Allocation of an entity in game playing

Board	Strategy	After n'th game 2	3	5	9	16	Best player
0 - 6 -	1	X	0	0	0	0	0
	2	X	0	X	X	X	X
	3	X	X	X	X	X	X
1 - 5 -	2	X	X	-	-	-	X
	3	X	X	X	X	X	X
2 - 4 -	1	X	0	0	X	X	X
	3	X	X	X	X	X	0
3 - 3 -	1	0	0	0	0	0	0
	2	0	0	X	0	=	0
2 - 3 -	1	0	0	0	0	0	0
	3	=	0	0	0	0	X
1 - 3 -	2	0	0	X	0	=	0
	3	X	0	=	0	0	X
3 - 2 -	1	X	X	X	0	=	0
	2	X	X	X	X	X	X
1 - 2 -	2	X	X	X	X	X	X
	3	X	X	X	X	X	X
Winning rate %		58.3	54.2	62.5	58.3	70.9	91.7

0 : Win X : Lose = : Neither
- : Reverse image of 'Win'

Fig. 8 Learning process in game playing

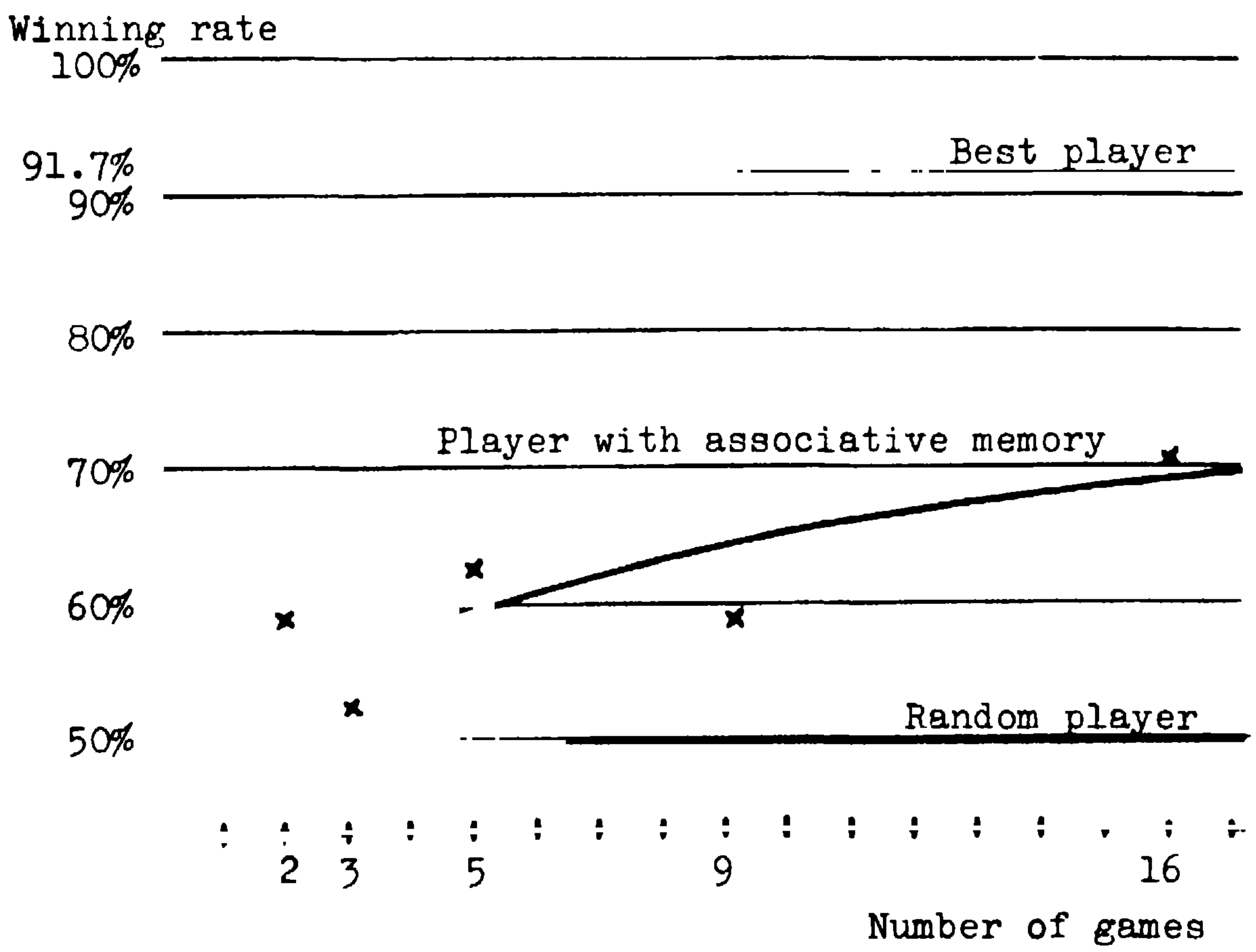

Fig. 9 Graphical expression of learning

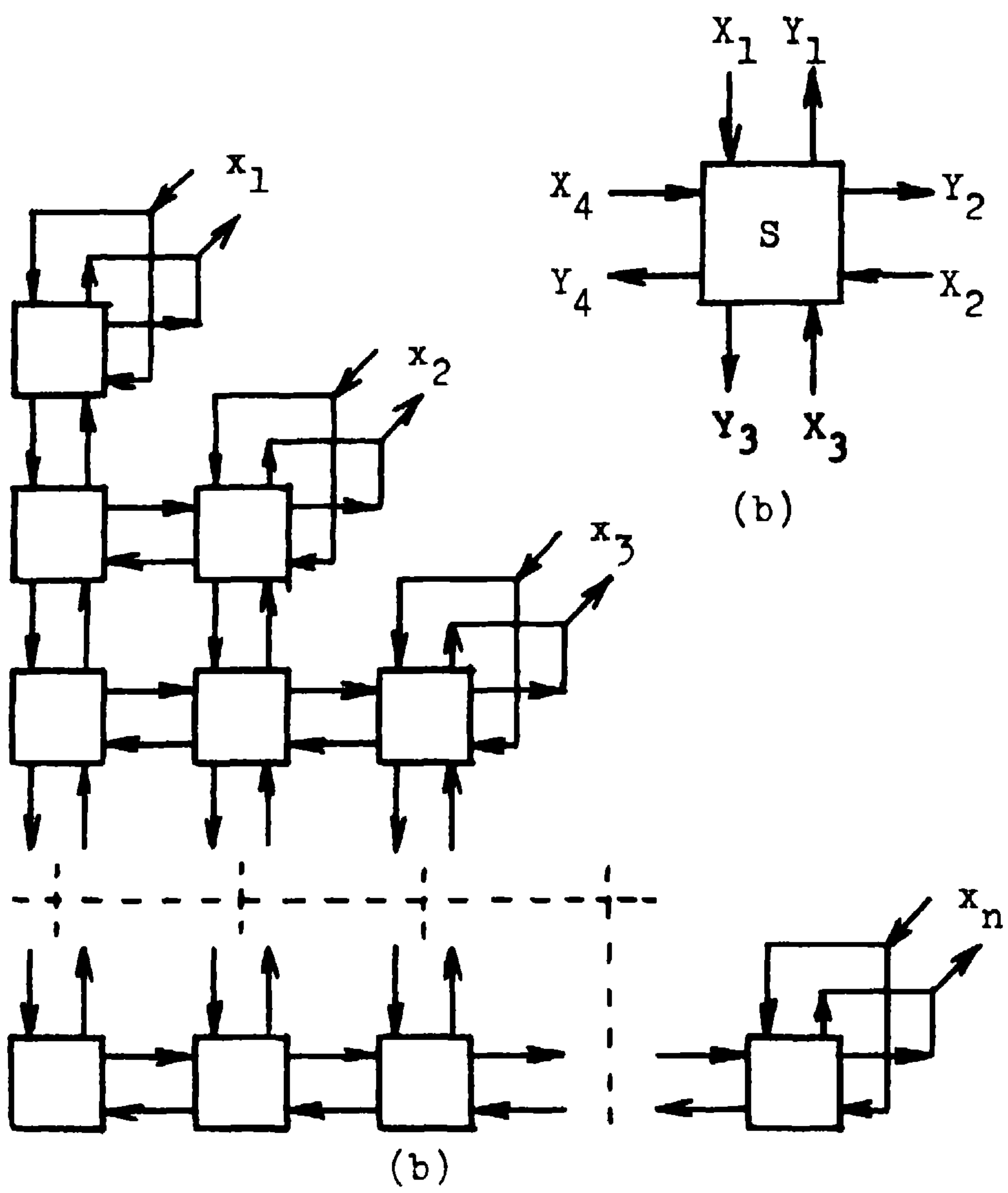

Fig. 10 Hardware realization

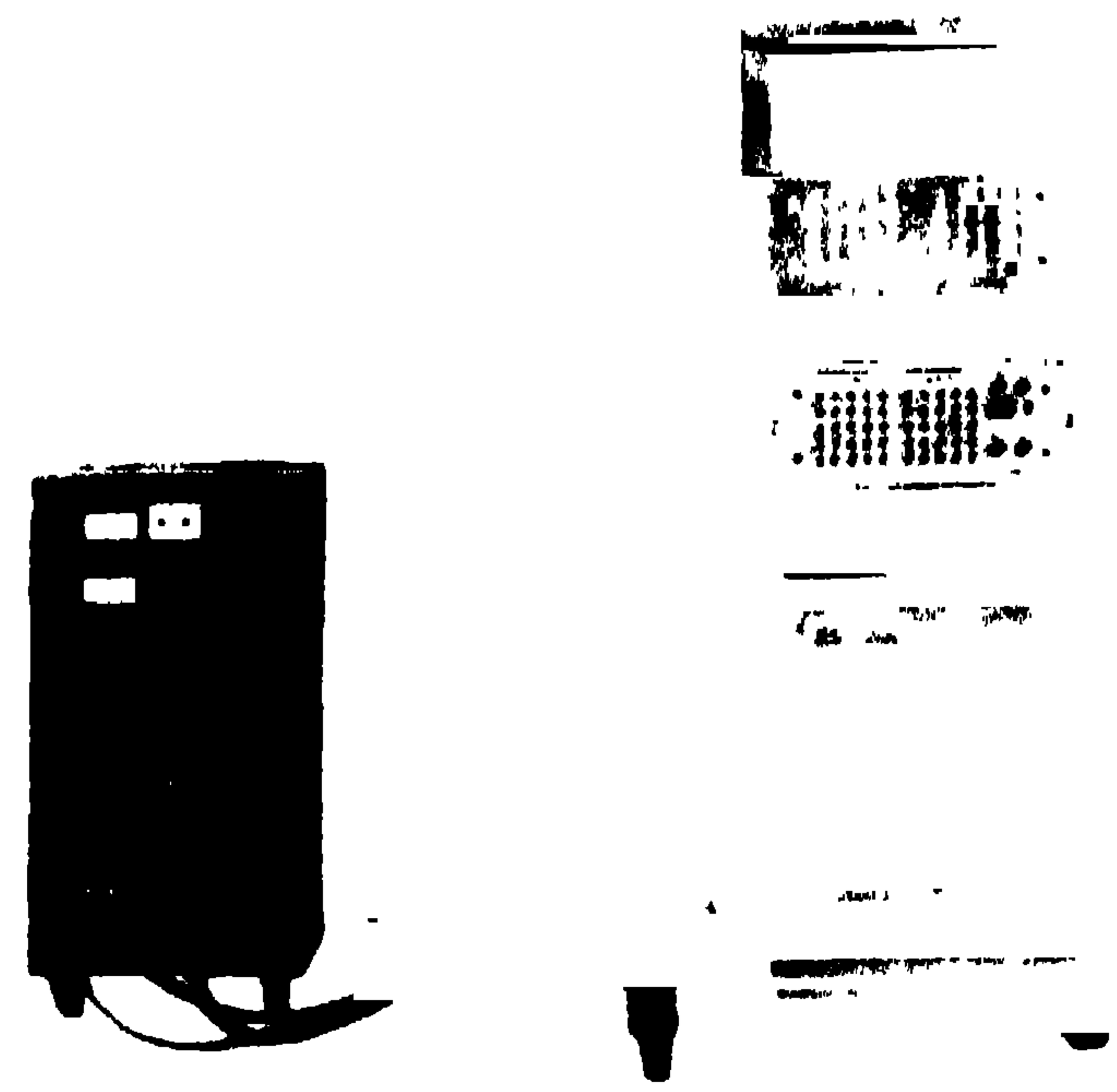

Fig. 11 The Associatron made for trial purposes

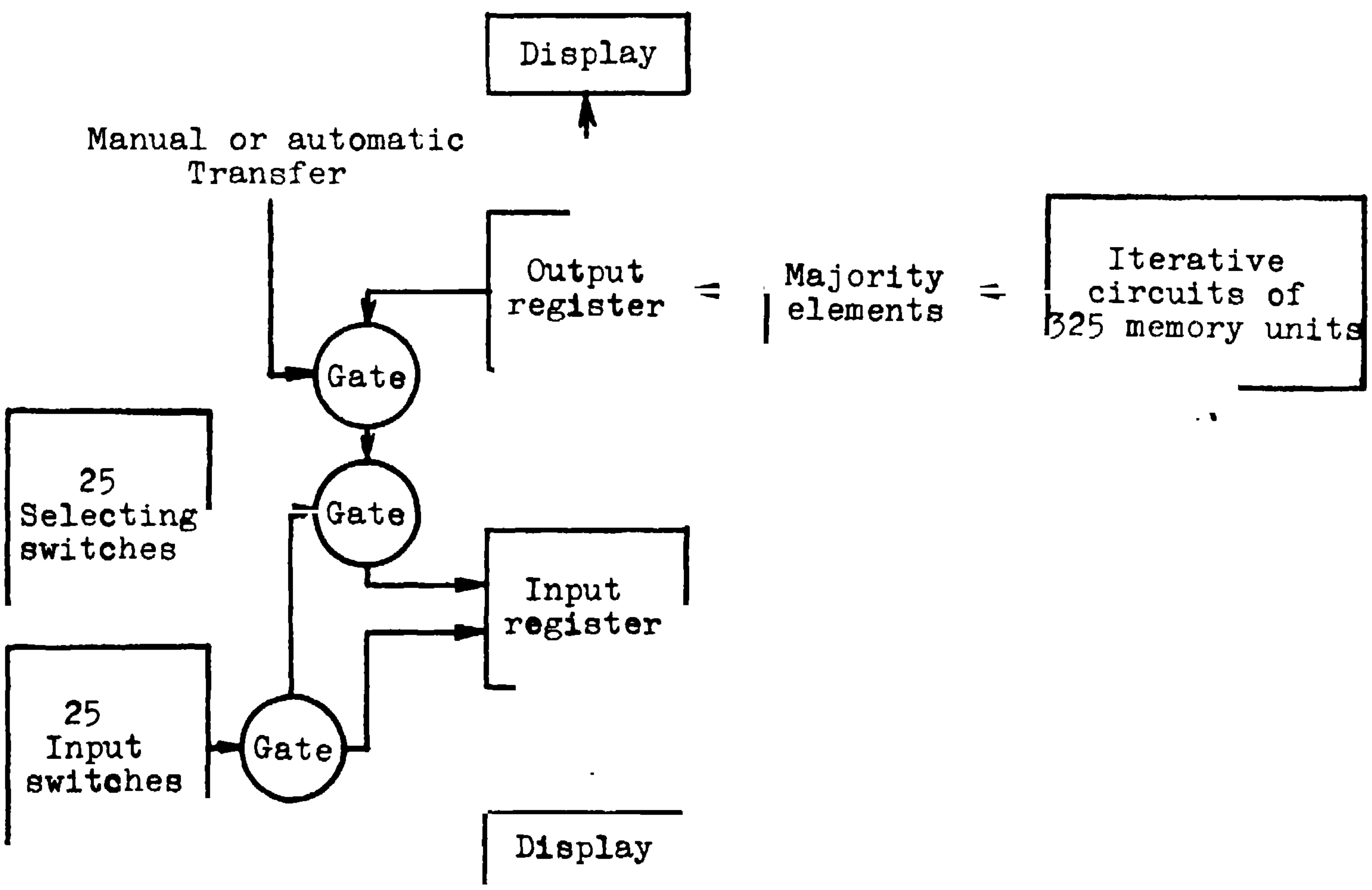

Fig. 12 Block diagram of the Associatron