# PRESENTATION OF A MAZE-SOLVING MACHINE

CLAUDE SHANNON
*Bell Laboratories, Murray Hill, N. J.*

THIS IS A maze-solving machine that is capable of solving a maze by trial-and-error means, of remembering the solution, and also of forgetting it in case the situation changes and the solution is no longer applicable. I think this machine may be of interest in view of its connection with the problems of trial-and-error learning, forgetting and feedback systems.

As you can see (Figure 8), there is a maze on the top panel of the machine which has a range of 5 x 5 squares. The maze can be changed in any desired manner by rearranging the partitions between the twenty-five squares. In the maze there is a sensing finger, which can feel the partitions of the maze as it comes against them. This finger is moved by two motors, an east-west motor and a north-south motor. The problem facing the machine is to move the finger through the maze to the goal. The goal is mounted on a pin which can be slipped into a jack in any of the twenty-five squares. Thus you can change the problem any way you choose, within the limits of the 5 x 5 maze. I will turn it on so you can see it, in the first place, trying to solve the maze. When the machine was turned off, the relays essentially forgot everything they knew, so that they are now starting afresh, with no knowledge of the maze.

*Savage:* Does than mean they are in a neutral position, neither to the right nor the left?

*Shannon:* They are in a kind of nominal position. It isn't really a neutral position but a meaningless one.

You see the finger now exploring the maze, hunting for the goal. When it reaches the center of a square, the machine makes a new decision as to the next direction to try. If the finger hits a partition, the motors reverse, taking the finger back to the center of the square, where a new direction is chosen. The choices are based on previous knowledge and according to a certain strategy, which is a bit complicated.

*Pitts:* It is a fixed strategy? It is not a randomization?

*Shannon:* There is no random element present. I first considered using a probability element, but decided it was easier to do it with a fixed strategy. The sensing finger in its exploration has now reached

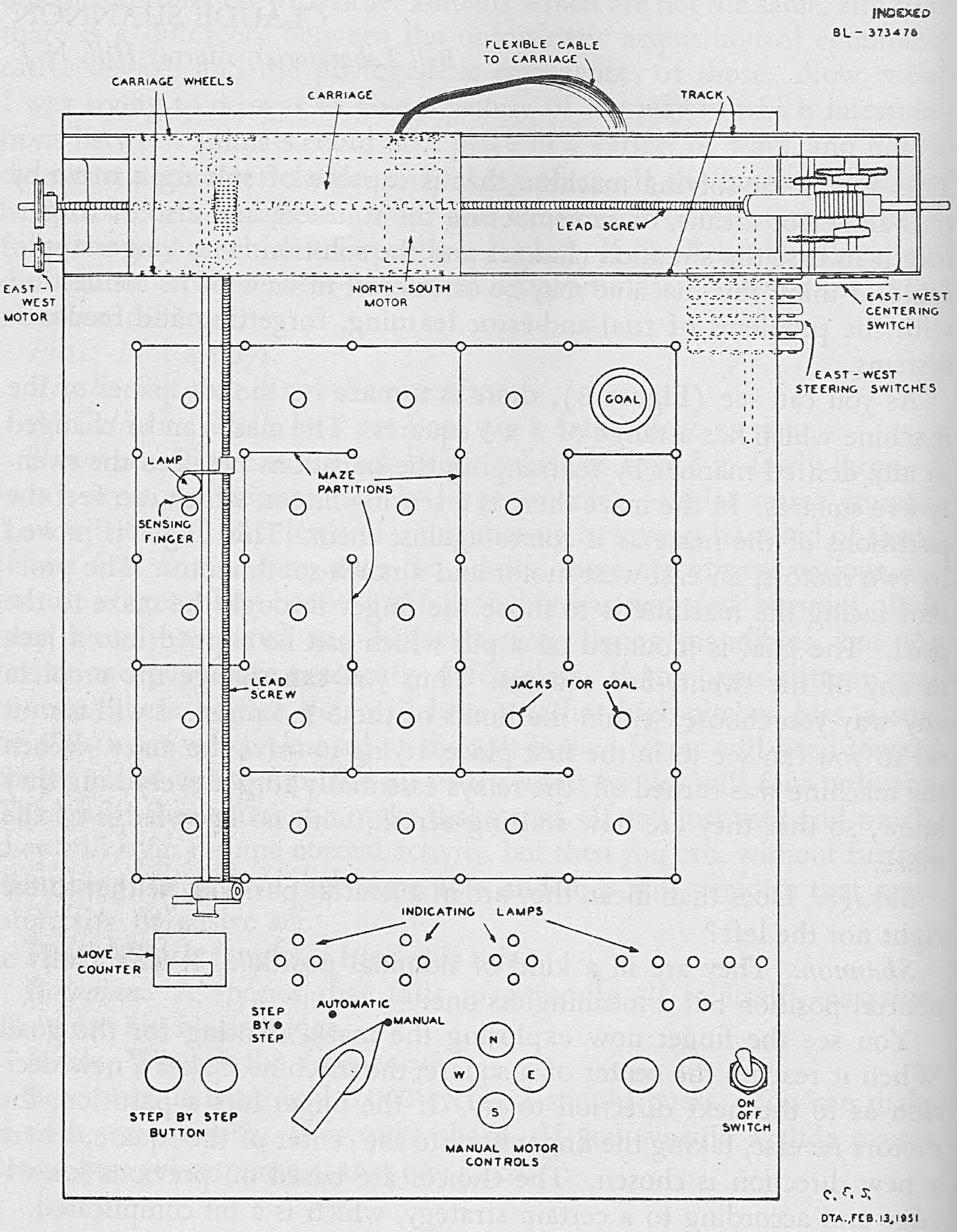
INDEXED
BL – 373478
FLEXIBLE CABLE TO CARRIAGE
CARRIAGE WHEELS
CARRIAGE
TRACK
LEAD SCREW
EAST-WEST MOTOR
NORTH-SOUTH MOTOR
EAST-WEST CENTERING SWITCH
EAST-WEST STEERING SWITCHES
GOAL
LAMP
SENSING FINGER
MAZE PARTITIONS
LEAD SCREW
JACKS FOR GOAL
INDICATING LAMPS
MOVE COUNTER
AUTOMATIC
STEP BY STEP
MANUAL
N
W
E
S
STEP BY STEP BUTTON
MANUAL MOTOR CONTROLS
ON OFF SWITCH
C. E. S.
DTA., FEB. 13, 1951

FIGURE 8

the goal, and this stops the motors, lights a lamp on the finger, and rings a bell. The machine has solved the maze. I will now run the finger, manually, back to the starting point, and you will see that the machine remembers the solution it has found. When I turn it on, it goes directly to the goal without striking the partitions or making side excursions into blind alleys. It is able to go directly to the goal from any part of the maze that it has visited in its exploration. If I now move the finger to a part of the maze that it has not explored, it will fumble around until it reaches a known region. From there it goes directly to the goal.

Now I should like to show you one further feature of the machine. I will change the maze so that the solution the machine found no longer works. By moving the partitions in a suitable way, I can obtain a rather interesting effect. In the previous maze the proper solution starting from Square A led to Square B, then to C, and on to the goal. By changing the partitions I have forced the machine at Square C to go to a new square, Square D, and from there back to the original square, A. When it arrives at A, it remembers that the old solution said to go to B, and so it goes around the circle A, B, C, D, A, B, C, D. . . . It has established a vicious circle, or a singing condition.

*Gerard:* A neurosis.

*Shannon:* Yes.

*Savage:* It can't do that when its mind is blank, but it can do it after it has been conditioned?

*Shannon:* Yes, only after it has been conditioned. However, the machine has an antineurotic circuit built in to prevent just this sort of situation.

*Mead:* After it has done it a number of times?

*Shannon:* After it has gone around the circle about six times, it will break out. The relay circuit includes a counter which stops this behavior at the twenty-fourth count.

*Frank:* How many relays are there in it?

*Shannon:* All told, there are about seventy-five relays.

*Savage:* It doesn't have any way to recognize that it is "psycho;" it just recognizes that it has been going too long?

*Shannon:* Yes. As you see, it has now gone back to the exploring strategy.

*Teuber:* Now, does it have to relearn the entire maze, or can it still utilize some form of it?

*Shannon:* No. As it stands, it can't utilize any information it had before.

*Savage:* But it is trying to utilize it, I suppose. It is moving as it would move.

*Shannon:* As a matter of fact, the old information is doing it harm.

*Bigelow:* I think it's getting to it.

*Shannon:* Yes, it is gradually working over toward the goal. I should like to spend the rest of my time explaining some of the things which are involved in the operation of the machine.

The strategy by which the machine operates can be described as follows: There are two modes of operation, which I call the "exploration strategy" and the "goal strategy." They are both quite simple. The exploration strategy is used when it is first trying to find the goal. For each square in the maze, there is associated a memory, consisting of two relays. These are capable of remembering one of four possible directions: north, east, south, or west. The direction that is remembered for a square is the direction by which the sensing finger left the square the last time it visited that square. Those are the only data the machine remembers about the course of the finger through the maze. There are some other memory functions in the computing part of the circuit, but these remembered directions are the data which allow it to reproduce its path at a later time.

Now, let's call the remembered direction for a particular square, D, considered as a vector. In exploration strategy, the machine takes the vector D and rotates it 90° as the first choice when it comes into a square. For example, suppose it left a square in the easterly direction at the last visit. If it comes to that square again, it will try the northern direction as the first choice. If it hits a barrier and comes back, it again rotates 90°, because it has just put this northern direction into the memory, and, advancing 90°, it tries the westerly direction, and so on. The choices progress around counterclockwise, starting with the direction by which it left the square last time—with one exception: it also remembers the direction by which it came into the square at the current visit, and on the first rotation of the vector D, it skips that direction of entrance. This is to prevent the path repeating too much. Before that feature was installed, there was a tendency to explore up to a new square, go back through the entire maze, and then go one square further, and so on; and it took a very long time to solve the maze. It required about three times as long as it does now, with this skipping feature added.

When it hits the goal, a relay operates and locks in, and the machine then acts according to the goal strategy, which is also based on this vector D.

In the goal strategy, the machine takes as its first choice direction D, which is the direction by which it left the square on its last visit. This is very simple to do, and it has many convenient features for maze solving, because it cancels out all blind alleys and circular paths. Since a blind alley must be left by way of the same square through which it was entered, the direction D retained for that square will necessarily lead to the goal directly rather than by way of the side excursion into the blind alley. In a similar way, if the machine follows a circular or re-entrant path in exploring its way to the goal, the direction retained for the last fork in this path must be that going to the goal rather than around the side loop. As a consequence, the machine follows a fairly direct path to the goal after it has first found its way there.

The final feature of forgetting is obtained as follows: After reaching the goal, suppose we move the sensing finger to a different point in the maze and start it operating. The machine then starts counting the number of moves it takes, and if it does not reach the goal within a certain specified number of moves, which happens to be twenty-four in this case, the machine decides that the maze has been changed or that it is in a circular loop, or something of that sort, and that the previous solution is no longer relevant. The circuit then reverts to the exploration-type strategy which is mathematically guaranteed to solve any finite solvable maze.

There are a few other points about the machine which may be of some interest. The memory is quite undifferentiated in the sense that I can take the group of wires leading from the rest of the circuit into the memory, shift them over either in the north-south or east-west directions, and the machine will still operate correctly, with no significant change, although the data corresponding to a square are then stored in a different part of the memory.

Another point is that there are, of course, a large number of feedback loops in this system. The most prominent is the feedback loop from the sensing finger through the circuit to the driving motors and back to the sensing finger, by mechanical motion of the motors. Normally, if you have a feedback loop and change the sign of the feedback, it completely ruins the operation of the system. There is ordinarily a great difference between positive and negative feedbacks. This maze-solving machine, however, happens to be such that you can change either or both of the signs in the feedback connections, and the machine still operates equally well. What it amounts to within the circuit is that the significance of right and left is interchanged; in other words, the effect on the strategy if one of the feedback loops is changed is that the advance of 90° counterclockwise becomes an advance of 90° clockwise. If both of them are changed, the strategy is not altered.

*Von Foerster:* If there are two different ways to reach the target, certainly the machine is only able to find one. Does the possibility point to its making a choice of the better way?

*Shannon:* No, it does not necessarily choose the best way, although the probabilities are in favor of its choosing the shorter of two paths. Incidentally, the exploration strategy of this machine will solve any maze whether it be simply or multiply connected. Some of the classic solutions of the maze problem are satisfactory only in the case of simply connected mazes. An example is the method of keeping your hand always on the right-hand wall. While this will solve any simply connected maze, it often fails if there are closed loops.

*Savage:* This cyclical feature that you illustrated occurred because the machine was not then in really searching condition?

*Shannon:* No, it was in the goal strategy rather than in the exploratory.

*Savage:* A goal strategy is to go the way you last went, but what are you to do if the attempt to do that is frustrated?

*Shannon:* Then it returns to the center of the square and advances 90° and tries that direction. But it still remains in goal strategy.

*Savage:* I see. When it gets into the next square, it tries to go ahead in the accustomed direction?

*Shannon:* That's right. The purpose of this is that it may have learned most of a maze in its first exploration, but not quite all of it. If we put it into a square it has not visited, it explores around by trial and error until it reaches a familiar square, and from there goes directly to the goal. The previously unknown squares have by this process been added to its previous solution.

*Bigelow:* You can then put new loops on any known path; it will learn those new loops immediately and not get into trouble. Is that right?

*Shannon:* That's right.

*Bigelow:* Because when you come back to the main stream, the search goes in the right direction, if it recognizes that square.

*Shannon:* I am not sure I understand what you mean.

*Bigelow:* It forms a single-directional path. Now, then, if you introduce a new path which brings it out of the known path into strange territory, back into the known path again—

*Shannon:* Such a side path is completely canceled when it has gone into the goal strategy.

*Bigelow:* But once you start it around that circuit, then the procedure is correct after the starting point.

*Shannon:* If it is in goal strategy, yes, but not in exploratory.

*Bigelow:* What would you have to do to minimize running time—in order to make it learn on repeated trials eventually to take the shortest possible path in a more complex maze?

*Shannon:* I think that would require a considerable amount of memory in the form of relays, because of the need to store up a number of different solutions of the maze as well as additional computing relays to compare and evaluate them. It surely could be done, but it would be more difficult; it would mean a much more complicated machine than this.

*Savage:* And it would have to decide when to invest the effort to seek a new path. That is really a very important problem in any kind of real human learning. If you can already peel a potato, why should you take the trouble to find a better way to peel it? Perhaps you are already peeling it correctly. How do you know?

*Von Foerster:* What happens if there is no goal?

*Shannon:* If there is no goal, the machine establishes a periodic path, searching for the goal; that is, it gradually works out a path which goes through every square and tries every barrier, and if it doesn't find the goal, the path is repeated again and again. The machine just continues looking for the goal throughout every square, making sure that it looks at every square.

*Frank:* It is all too human.

*Brosin:* George Orwell, the late author of *1984,* should have seen this.*

*Von Foerster:* And after that? For instance, if you put a goal into the path after the machine has established such a periodic motion, what happens then?

*Shannon:* When it hits the goal, the machine stops and changes into the goal strategy, and from there on it goes to the goal as placed there. Incidentally, it is interesting to think of this—if I can speak mathematically for a moment—in the following way. For each of the twenty-five squares, the memory of the machine retains a vector direction, north, east, south, or west. Thus, as a whole, the memory contains a vector field defined over the 5 x 5 maze. As the sensing finger moves through the maze, it continually revises this remembered vector field in such a way that the vectors point along possible paths of the maze leading to the point currently occupied by the finger.

*Teuber:* If you rotate the field through 180°, would it continue to function?

*McCulloch:* Suppose you reverse the connections and leave the motor, so that you reverse your direction of rotation; can it still find its way?

*Shannon:* Only if I reverse some switches within the machine which tell it what square it is currently occupying. If I reverse the motors, I must change these switches to compensate. Otherwise, it would think it was moving one way and put that in the memory and actually be moving in a different direction.

*Orwell, G.: *1984.* New York, Harcourt, Brace & Co., 1949 and Signet Books, 1950. No. 798

*Gerard:* That would be like cross-suturing the motor nerves of animals and getting flexion when you want extension.

*Bigelow:* Have you considered how difficult it would be to have a circuit which, instead of forgetting everything, goes back to the origin and remembers what it did at the first square but tries something else, say, the opposite search sequence? When that produces no new solution, go back where it was, in the second square, but try the opposite, therefore asking for the possibility of replacing each square in its memory as it goes systematically through. In other words, this would require a very small addition of memory because it need only remember the entire past pattern once, but then, having reached the state where goal behavior is no longer a solution (which it knows by exceeding "N" trials), then, instead of erasing its entire thinking, you have a switching technique where it goes back to the origin, and then tests each hypothesis in turn, and finds the particular one to replace.

*Shannon:* I haven't considered that, but I think it would be rather slow, because there is a great deal of backtracking in that procedure, back to the origin, as it tries out different hypotheses.

*Bigelow:* If it knows how to get from the origin to the target, does it not always know how to get from the target back to the origin, by a very simple reversal of the switches?

*Shannon:* No. You see, this vector field, if you like, is unique in going in the direction of the vectors, but going backward, there are branch points, so it does not know where it came from.

*Savage:* Does this vector field flow into the target from every point?

*Shannon:* Yes, if you follow the vectors you will get to the goal, but, going in reverse, you may come to branch points from which you may go in any of various directions. You can't say where the sensing finger came from by studying the memory.

*Savage:* It is not organized around any particular initial point; and that is one of the features of it, that once it has learned the maze, if you start it anywhere where it has been on its way to the maze, it continues; if you start it where it hasn't been, it finds one of those places where it has been, and then continues.

*McCulloch:* Like a man who knows the town, so he can go from any place to any other place, but doesn't always remember how he went.