Journal of Cybernetics, 6:39-64, 1976

FUZZY ROBOT CONTROLS

MASATO URAGAMI, MASAHARU MIZUMOTO, KOKICHI TANAKA

Osaka University
Toyonaka, Osaka, Japan

A fuzzy program is defined as an ordered sequence of fuzzy instructions. In the execution of a fuzzy program, fuzzy instructions are translated into machine instructions by the use of MAX-METHOD and back-tracking. To show how a fuzzy program is executed, a simulation system is exploited which controls a simple inchworm robot. A few examples of computer simulations are presented which deal with the behavior of a stranger (or a robot) searching for his destination in a modeled town under simple fuzzy instructions.

INTRODUCTION

In our daily lives there are many things that we can do without any difficulty. But when we examine their mechanisms, we often find it very difficult to understand them. For instance, consider the descriptions: "If the weather is *fine*, let's go on a hike," "walk *for a while*, then *turn to the right*," and "go *about 100 meters* until you see a *black* building." All of the above descriptions have vague or imprecise conceptions.

Imprecise concepts sometimes play important roles in human life. So it is thought to be very important and interesting to deal with these kinds of matters through the use of computers. This has relations with the fields of artificial intelligence and soft sciences.

In this paper, these vague concepts are analyzed and processed through the applications of the fuzzy theory (Zadeh, 1968, 1973). We consider the case of a "wandering person." The process of wandering is described as

Masato Uragami is currently with the Graduate School of Business, Columbia University, New York, until August, 1978.

Copyright © 1977 by Hemisphere Publishing Corporation

that of interpreting and executing a fuzzy program (Chang, 1972; Tanaka and Mizumoto, 1975).

Backtracking (Filmore and Williamson, 1974; Golumb and Baumert, 1965) is presented as an important factor when we interpret and execute fuzzy programs. Its functions and effects are also explained here. With understanding of these matters, a few examples of computer simulations are given. The system of the computer simulations is exploited, which runs a fuzzy robot (a small inchworm robot that is controlled by a fuzzy program) on a modeled town.

OUTLINE OF THE SYSTEM

First we make an outline of the purpose which the system is designed to fit. Also described is the space where the system is constructed.

Purpose

The system is exploited so as to be able to represent the following situation.

> A stranger calls on a person at a town, on the condition that he was given something like an imprecise and simple map of the town or instructions which bear the rough sketch of the way to his destination. He, with the given information, sometimes wanders and roams. Then he finally arrives at his destination.

We make a model which represents this situation in order to analyze this situation (see Figure 1).

Town

The town where the process of a wandering person (or a robot) is considered consists of two parts. One is the plane space where the robot moves about. The other is the data structure MAP which gives him information of the town. From now on we shall call the former the *real space*, and the latter the *virtual space.*

Real Space

The real space is a space where the robot can actually move about. On the real space, there are no houses, buildings, and even roads which an ordinary

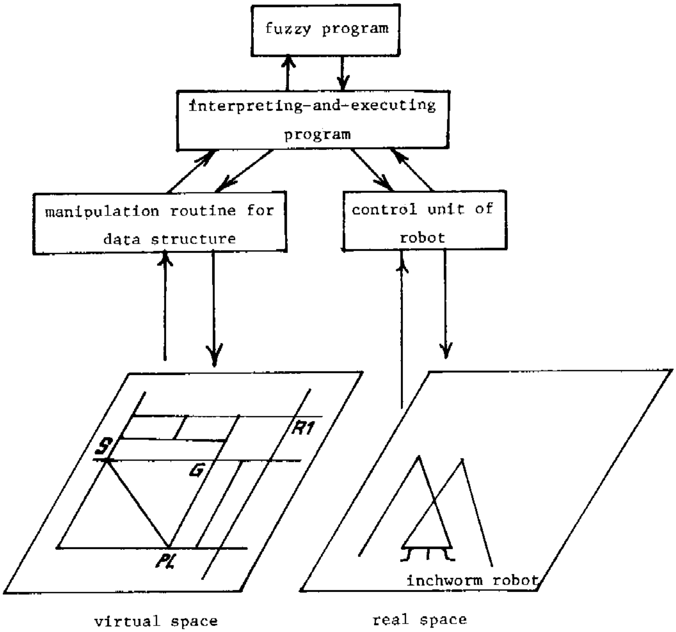

FIGURE 1. System.

town would consist of. There the inchworm robot can perform such movements as "walk" or "turn." The completion of a movement is informed to the interpreting-and-executing program through the interface.

Virtual Space

The virtual space is the data structure MAP which has the structure of the town. In the virtual space, houses, roads, and buildings are recognized as if the town were a real town. The robot is represented as a pointer which indicates a point in the virtual space. The interpreting-and-executing program changes the pointer of the robot in the virtual space when the robot on the real space moves.

Perception of the Town

Information of the circumstances around the robot (for example, he is in front of the building A, at the intersection 2, and so on) is perceived in the following way. Data items (see Figure 2), which relate to the data items representing the present position of the robot in the virtual space, are able

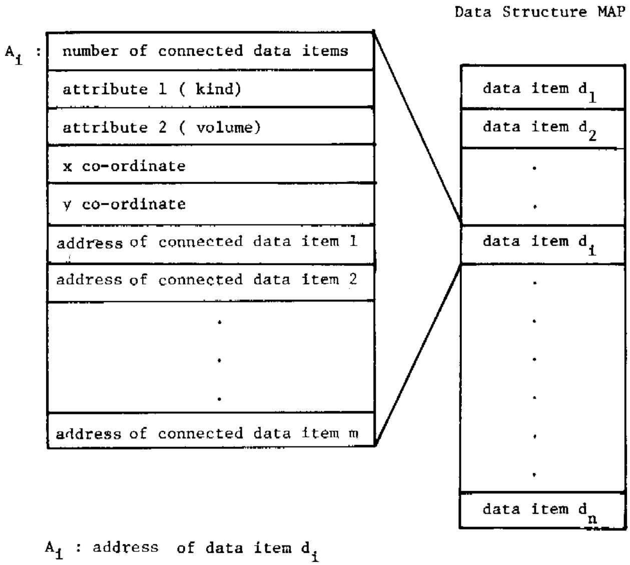

FIGURE 2. Data structure MAP.

to be perceived. But other data items cannot be recognized by the interpreting-and-executing program (each data item represents an objective point—a building, an intersection, etc.). The interpreting process of a fuzzy program is performed through comparison and judgment of the following points—the point which the robot is going to, and the ones which it has passed through. Other points at the town are usually not taken into consideration when the robot decides its way (on backtracking, some of them are used when the robot decides its way). All of the objective points at the town (as we consider the case of a "wandering person," the objective points mean houses, buildings, or intersections where he has alternative ways to take) are stored in the virtual space as data items. On the points other than those represented by data items of the virtual space (for example on a road), the inchworm robot never stops nor changes its way to go.

Data Structure MAP

The data structure MAP which constitutes the virtual space has the structure shown above (Figure 2).

1. The data structure MAP consists of data items d_i's (d_i, $i = 1, \ldots, n$; n is the number of data items, d_i is the ith data item).
2. A data item d_i consists of $5 + m$ fields ($m \in \{1, \ldots, 5\}$).

Each data item represents an object in the virtual space. The fields of a data item d_i have the following contents in this order from the top to the bottom.

1. The number of the data items which connect to the data item d_i.
2. Attribute 1 (the kind of the object, such as a police station, a church, an intersection, or a school, etc.).
3. Attribute 2 (the size of the object; it is presented by its volume).
4. x coordinate.
5. y coordinate.
6. The addresses of m data items which connect to the data item d_i.

The data structure MAP is accessed through the following two routines.

1. Routine FIND: The input argument is the attribute of the data item sought. This routine gets the address of the data item which has the attribute.
2. Routine SEARCH: The input argument is the address of a data item. This routine gets addresses, attributes, and coordinates of data items which connect to the data item with the input address.

Through the data structure MAP and its routines, the interpreting-and-executing program can get information around the robot in concurrence with its movement in the modeled town.

INTERPRETATION AND EXECUTION OF FUZZY PROGRAMS

In this section, we briefly review fuzzy sets and fuzzy instructions which are the fundamental concepts dealing with fuzzy programs. Then we show how the process of interpreting and executing a fuzzy program is represented and processed in the system.

Fuzzy Sets

A fuzzy set A in a set X is defined as

$$\mu_A : X \rightarrow [0,1] \tag{1}$$

where μ_A is called a *membership function*, and the value $\mu_A(x)$ represents *grade*, that is, the grade with which an element $x \in X$ belongs to a fuzzy set A. A fuzzy set A in X can be represented by the following:

$$A = \sum_{i=1}^{|X|} \mu_A(x_i)/x_i \tag{2}$$

Union $\cup$ and intersection $\cap$ of two fuzzy sets A_1 and A_2 are defined as follows.

$$A_1 = \sum_{i=1}^{|X|} \mu_{A_1}(x_i)/x_i \tag{3}$$

$$A_2 = \sum_{i=1}^{|X|} \mu_{A_2}(x_i)/x_i \tag{4}$$

$$A_1 \cup A_2 = \sum_{i=1}^{|X|} \mu_{A_1}(x_i) \vee \mu_{A_2}(x_i)/x_i \tag{5}$$

$$A_1 \cap A_2 = \sum_{i=1}^{|X|} \mu_{A_1}(x_i) \wedge \mu_{A_2}(x_i)/x_i \tag{6}$$

where $\mu_1 \vee \mu_2$ denotes $\max(\mu_1, \mu_2)$ and $\mu_1 \wedge \mu_2$ denotes $\min(\mu_1, \mu_2)$.

Example 1: A fuzzy set which represents "about 50" may be defined by the membership function shown in Figure 3.

Fuzzy Instructions

A fuzzy instruction is an instruction which includes fuzzy concepts. A machine instruction is an instruction which does not contain any fuzzy concepts.

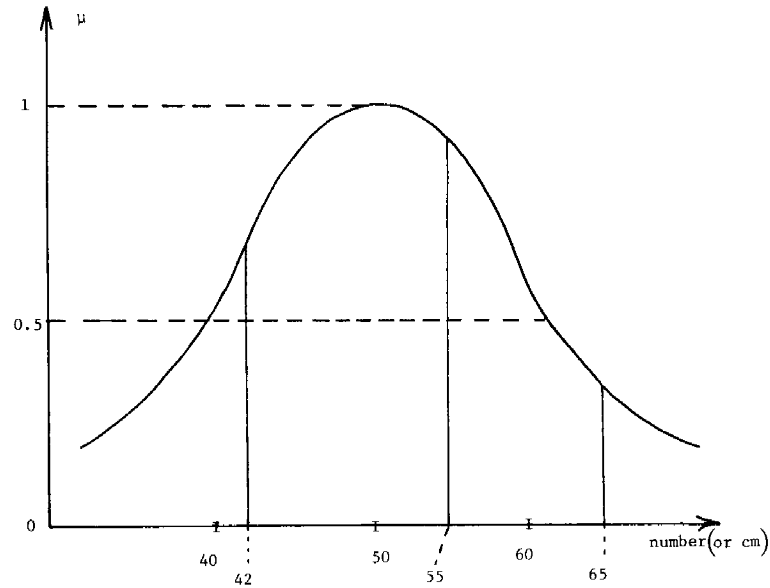

FIGURE 3. Fuzzy set which represents "about 50," or "a stick of about 50 cm."

Example 2: "Walk *about 100 meters*" and "Run *fast*" are fuzzy instructions, since they contain such fuzzy concepts as those in italics.

Example 3: "Walk 100 meters" and "Run at a speed of 30 km/h" are machine instructions.

A fuzzy instruction F_i is said to have an execution E_j by a person H_j under circumstances C_k (Dimitrov and Wechler, 1975), iff

$$F_i \cdot E_j \cdot C_k = \sum_{z=1}^{s'} \mu_z / m_z, \quad s' \leqq s, \quad \mu_z > 0, \tag{7}$$

where $M = \{m_1, \ldots, m_s\}$ is the set of machine instructions, and μ_z is the the grade of a machine instruction m_z on the interpretation of a fuzzy instruction F_i by a person H_j under circumstances C_k.

A fuzzy program is defined as a sequence of fuzzy instructions. We say a fuzzy program is executable when each fuzzy instruction is executable.

Let P be a fuzzy program, S a set of fuzzy instructions, T a set of circumstances, M a set of machine instructions, and U a set of executions. Then,

$$P = F_{i_1} \cdot F_{i_2} \ldots F_{i_t} \tag{8}$$

P is executable iff

$$P \cdot U \cdot T = (F_{i_1} \cdot E_{j_1} \cdot C_{k_1}) \cdot \ldots \cdot (F_{i_t} \cdot E_{j_t} \cdot C_{k_t}) \tag{9}$$

where $S = \{F_1, F_2, \ldots, F_k\}$, $T = \{C_1, C_2, \ldots, C_z\}$, $U = \{E_1, E_2, \ldots, E_m\}$, $M = \{m_1, m_2, \ldots, m_n\}$.

An input fuzzy program is sequencially interpreted and executed through the use of MAX-METHOD denoted below. When the result of the interpretation of a fuzzy instruction is "impossible to execute," the system performs backtracking. If the interpretation of the first fuzzy instruction is "impossible to execute," the interpretation of the entire fuzzy program ends in a failure.

MAX-METHOD

On the interpretation of a fuzzy instruction, the machine instruction with the highest grade is selected first. MAX-METHOD is defined by the max-select function $f_{\max}$,

$$f_{\max} : F_i \cdot E_j \cdot C_k \to m_{k'} \tag{10}$$

where $m_{k'} = \max(\mu_1, \mu_2, \ldots, \mu_{s'})$

The machine instruction $m_{k'}$ selected is called the interpretation I_{F_i} of the fuzzy instruction F_i.

Example 4: Consider the interpretation of a fuzzy instruction "Use a stick of about 50 cm" through the use of MAX-METHOD. Three sticks as shown in Figure 3 are given. In this case, the machine instruction with the highest grade, "Use the stick of 55 cm," is selected first as the interpretation.

Backtracking

On the interpretation of a fuzzy instruction, if the execution of the fuzzy instruction is impossible at all, backtracking is performed. The present state is replaced by the one step before. Then reinterpretation is done. The machine instruction which should be selected on the backtracking process is the one with the highest grade among those that have never been selected.

Example 5: In Example 4, assume that the machine instruction selected before, "Use the stick of 55 cm," is not adequate for some reason and backtracking is performed. Then the fuzzy instruction is reinterpreted. In this case, the machine instruction adopted as the interpretation is "Use the stick of 42 cm." If backtracking is performed again, the machine instruction "Use the stick of 65 cm" will be selected as the interpretation of this fuzzy instruction.

THRESHOLD METHOD

On the execution of a fuzzy instruction, the machine instructions whose grades are lower than a certain value (the threshold method) are not selected as the interpretation of the fuzzy instruction.

In the system exploited, both THRESHOLD METHOD and MAX-METHOD are introduced in order to interpret a fuzzy instruction. From now on this compound method is called MAX-METHOD without any reference.

Example 6: In Example 5, let the threshold value be 0.5, then on the second backtracking, the machine instruction "Use the stick of 65 cm" is abandoned because its grade is lower than the threshold value 0.5. In this case the result of interpretation is "impossible to execute."

We have explained the process of the interpretation of fuzzy programs and backtracking. Here we show two flowcharts: each presents the process of the interpretation of fuzzy programs in Figure 4, and backtracking in Figure 5.

In the process of a wandering person, data items which correspond to the path of the stranger and those points where he may go to next are taken into consideration usually. However, some function is allowed on the backtracking process. That is, on the interpretation of a fuzzy instruction on backtracking, the robot can know other data items than those in ordinary interpretation process. The data items of some points forward on the expected course of the robot are considered to be eligible to see. This is thought to correspond to what we do when we have lost our bearings—asking others the right way, examining the path we took, or scrutinizing

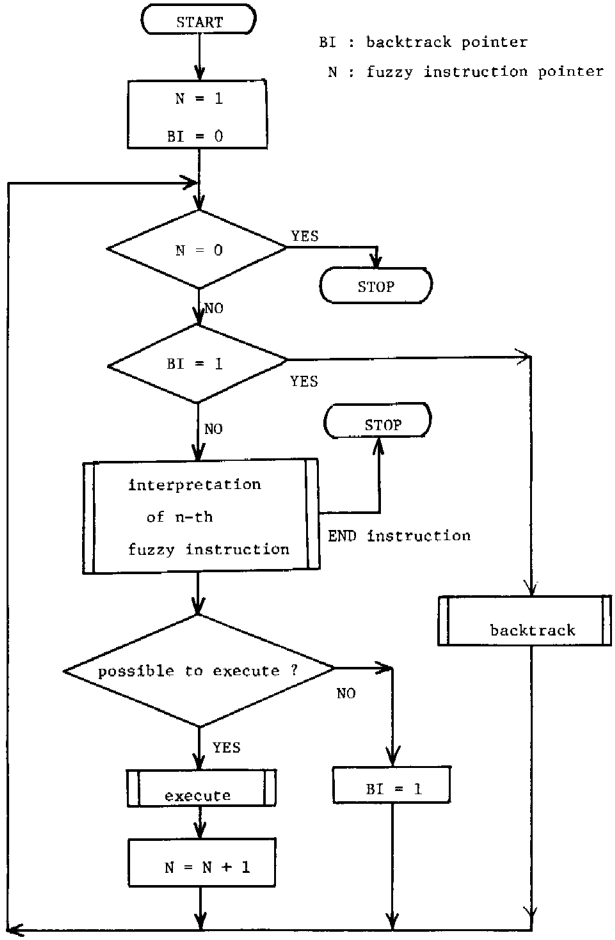

FIGURE 4. Flowchart of interpretation of fuzzy program.

the way forward. This function is added for our purpose to analyze the process of a wandering person.

We have shown the characters of fuzzy instructions and fuzzy programs in this section. In the next section, we shall examine the fuzzy instructions which are used in the system and their functions in detail.

FUZZY INSTRUCTIONS USED IN THE SYSTEM

Though there are many fuzzy instructions which we can use in the process of a wandering person, we shall select the following four fuzzy instructions as their representatives.

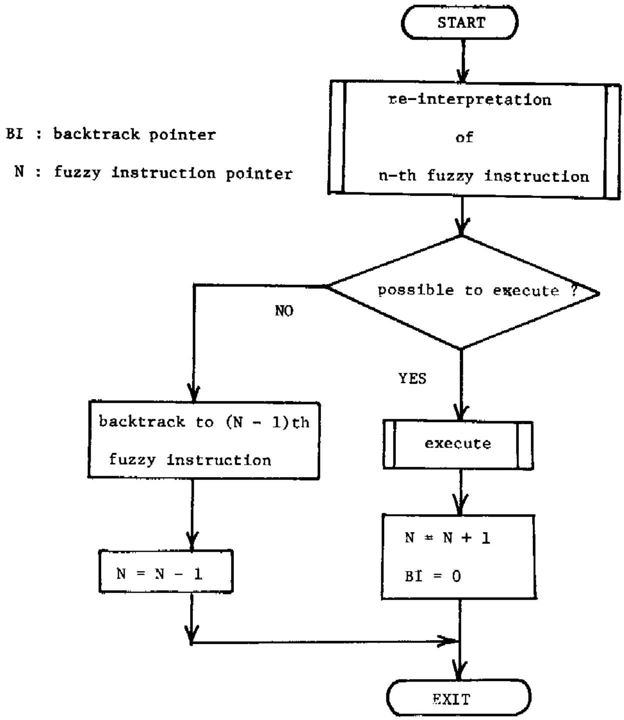

FIGURE 5. Flowchart of backtracking process.

1. GO ABOUT n STEPS. [abbreviated $G_A(n)$]
2. GO TO x. [abbreviated $G_T(x)$]
3. GO ABOUT n STEPS TO x. [abbreviated $G_{AT}(n,x)$]
4. TURN TO THE RIGHT. (abbreviated T_R)
   TURN TO THE LEFT. (abbreviated T_L)

and we use supplementary instructions as

5. HEAD EAST.
   HEAD WEST.
   HEAD SOUTH.
   HEAD NORTH.
6. START FROM x.

which give initial values to the system, and

7. END.

which shows the end of a fuzzy program.

Next we shall show the functions of these fuzzy instructions more precisely.

1. GO ABOUT n STEPS, $G_A(n)$: The membership function of distance for this fuzzy instruction is represented by

$$\mu(x) = \frac{1}{[(x-n)/a]^2 + 1} \tag{11}$$

where a is the relative distance parameter the value of which depends on the value of n.

Example 7: In the town shown in Figure 6, suppose that the robot is moving to the direction indicated by the symbol → in the figure, and a fuzzy instruction

GO ABOUT 20 STEPS $\quad G_A(20)$ (12)

is given to the robot. It is assumed that the membership function of distance for this fuzzy instruction is represented by Figure 7. With the threshold value θ given in Figure 7, we can read the following four points G, H, I, and J have the higher grades than the threshold value. Of the four eligible machine instructions, through MAX-METHOD, the machine

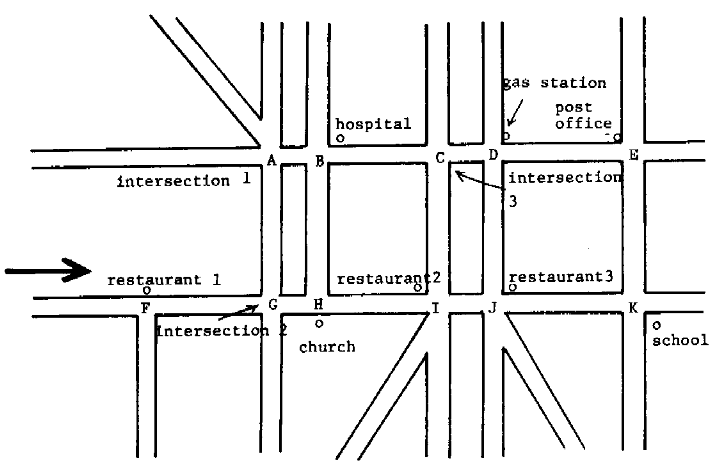

FIGURE 6. Town A.

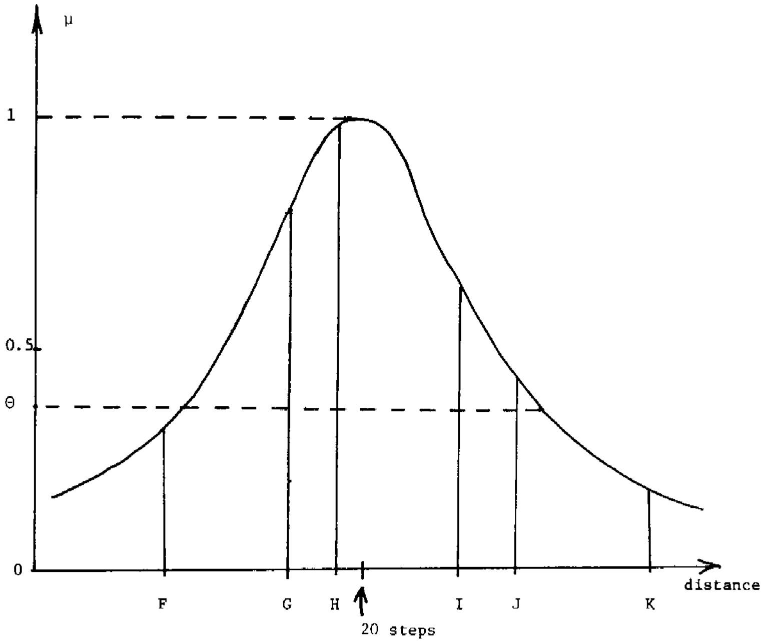

FIGURE 7. Membership function of distance for "GO ABOUT 20 STEPS."

instruction "GO TO H" is adopted first as the interpretation. If backtracking is performed, "GO TO G," "GO TO I," and "GO TO J" would be selected in this order as the machine instruction for the second, third, and fourth interpretation.

2. GO TO x, $G_T(x)$: When this fuzzy instruction is given, the robot goes to the destination x, which is either fuzzy or nonfuzzy.

Example 8: In the town shown by Figure 6, suppose that the robot is moving to the direction given in the figure, and a fuzzy instruction

GO TO RESTAURANT $\quad G_T(\text{REST})$ (13)

is given to the robot. The membership function is assumed to be represented by Figure 8. On the interpretation, the machine instruction "GO TO F" is selected. If the given fuzzy instruction is

GO TO RESTAURANT 2 $\quad G_T(\text{REST2})$ (14)

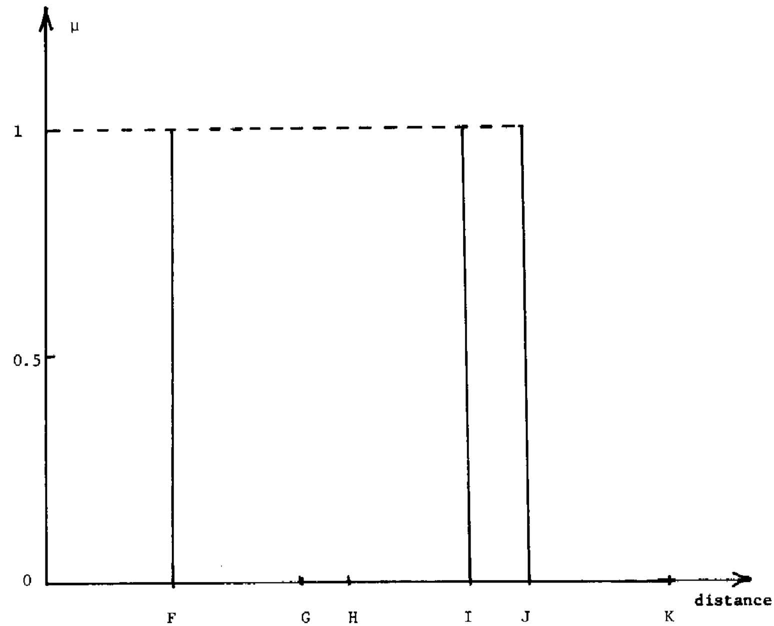

FIGURE 8. Membership function of "GO TO RESTAURANT."

then the machine instruction "GO TO I" is selected instead of the machine instruction "GO TO F" as the interpretation. If backtracking is performed and the fuzzy instruction (13) is reinterpreted, the machine instruction selected is "GO TO I." If the given fuzzy instruction is the one in (14), the interpretation process ends in a failure because of the lack of eligible machine instructions which can be selected as the interpretation.

3. GO ABOUT n STEPS TO x, $G_{AT}(n,x)$: This fuzzy instruction has compound characters of both of the fuzzy instructions $G_A(n)$ in 1 and $G_T(x)$ in 2.

Example 9: In the town and with the direction in Figure 6, the fuzzy instruction

GO ABOUT 20 STEPS TO RESTAURANT $\quad G_{AT}(20,\text{REST})$ (15)

is given to the robot. It is also assumed that the membership function of this fuzzy instruction is given by Figure 9. Namely, (15) is obtained from the composition (or intersection) of (12) and (13), that is,

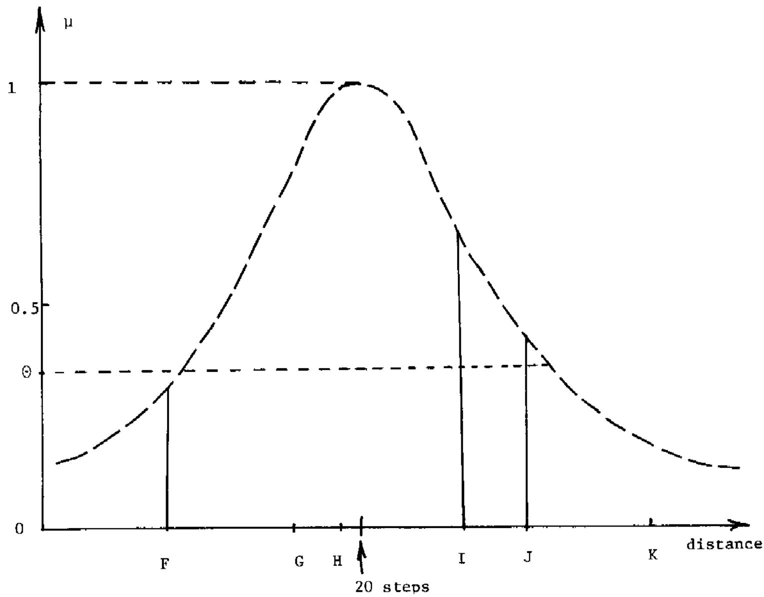

FIGURE 9. Membership function of "GO ABOUT 20 STEPS TO RESTAURANT."

$$G_{AT}(20,REST) = G_A(20) \cap G_T(REST) \tag{16}$$

In this case, the interpretation "GO TO F" adopted in Example 8 is abandoned because of the restriction of the threshold value. Instead, the machine instruction "GO TO I" is selected. On the first backtracking process, the machine instruction "GO TO J" is selected as the interpretation. If backtracking is performed again, the interpretation process ends in a failure because there do not exist any eligible machine instructions to be selected.

Here, we introduce a new fuzzy concept about size of objects. We use three fuzzy expressions of size, that is, "LARGE," "MIDDLE," and "SMALL." They are used in GO-instruction [GO-instruction means the following three fuzzy instructions, $G_T(x)$, $G_A(n)$, and $G_{AT}(n,x)$]. As the measure of size, we use volume of each object. They are stored in MAP as integers. We assume that the number means the volume of the object measured by such a scale as cubic meters or so. In the system, the membership functions of "LARGE" and "MIDDLE" are defined by Figure 10 and Figure 11. The membership function of "SMALL" is defined by

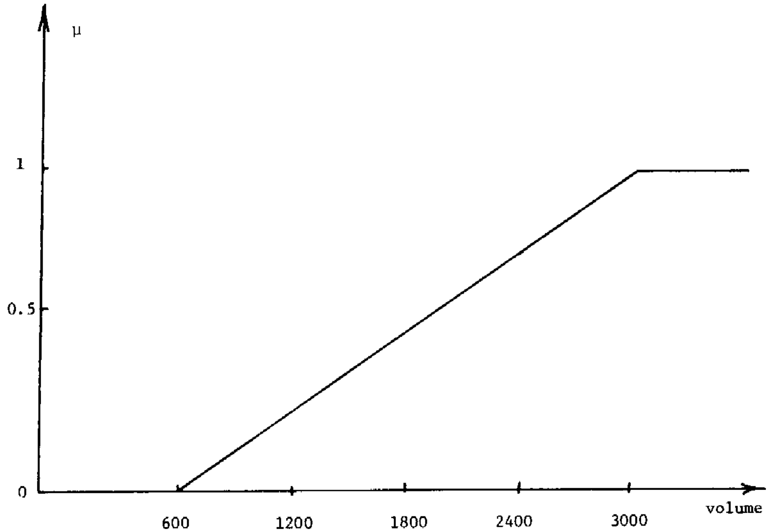

FIGURE 10. $\mu_{LARGE}(x)$.

$$\mu_{SMALL}(x) = 1 - \mu_{LARGE}(x) \tag{17}$$

Example 10: In the town and with the direction in Figure 6, we shall discuss the interpretation of the fuzzy instruction G_{AT} [20,SMALL(REST)], that is,

GO ABOUT 20 STEPS TO SMALL RESTAURANT (18)

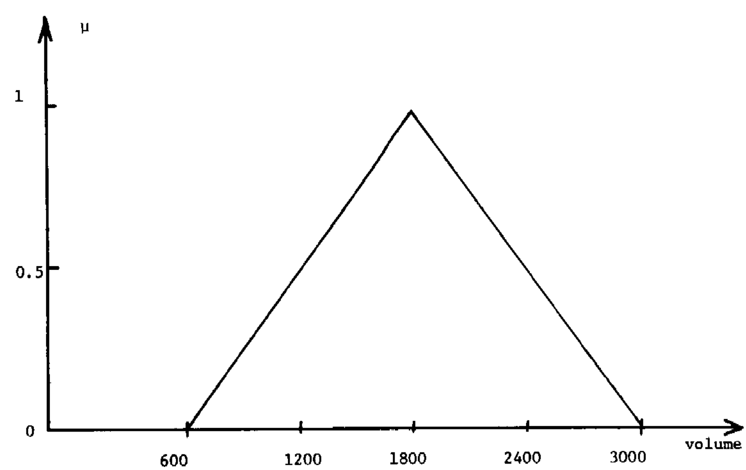

FIGURE 11. $\mu_{MIDDLE}(x)$.

(18) is thought to be composition of three fuzzy instructions, that is,

$$G_{AT}\,[20,\mathrm{SMALL(REST)}] = G_A(20) \cap G_T(\mathrm{REST}) \cap G_B(\mathrm{SMALL}) \qquad (19)$$

where G_B(SMALL) is defined by the membership function in (17). We assume that the volume of each object at the town is represented by Table 1. Then, by Figure 10, (17) and Table 1,

$$\mu_{G_B}(\mathrm{SMALL}) = 0.2/B + 0.5/D + 0.3/E + 0.6/F + 0.1/H + 0.8/I + 0.4/J + 0.2/K$$

And by (19), Figure 9, and μ_{G_B}(SMALL), we can get

$$\mu_{G_{AT}}\,[20,\mathrm{SMALL(REST)}] = 0.3/F + 0.7/I + 0.4/J \qquad (20)$$

Thus we get the interpretation of the fuzzy instruction, "GO TO I" which is characterized by the maximum grade 0.7.

4. TURN TO THE RIGHT (or LEFT), T_R (or T_L): The membership function of a turn-instruction (T_R and T_L) is represented by Figure 12. In Figure 12, the direction 0° means the way to which the robot is going to turn. The present direction of the robot plus 90° becomes the direction 0° in Figure 12 when the robot turns to the right. On the reinterpretation of a turn-instruction, the higher the grade of the course to be taken is, the earlier it is adopted as the interpretation of the fuzzy instruction. The threshold value of a turn-instruction is set to zero.

TABLE 1. Volume of Objectives at Town A

Point	Kind	Volume
A	Intersection 1	–
B	Hospital	2520
C	Intersection 3	–
D	Gas Station	1800
E	Post Office	2280
F	Restaurant 1	1560
G	Intersection 2	–
H	Church	2760
I	Restaurant 2	1080
J	Restaurant 3	2040
K	School	2520

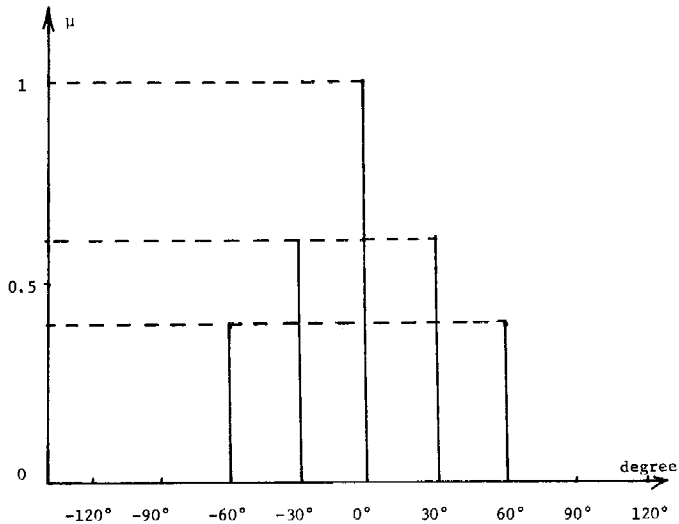

FIGURE 12. Membership function for direction.

Example 10: In the town shown by Figure 13, the robot is at the point G and is given the fuzzy instruction

TURN TO THE RIGHT (21)

The machine instruction adopted as the interpretation is "TURN TO THE G-L DIRECTION." If backtracking is performed, the machine instruction "TURN TO THE G-K DIRECTION" will be selected.

The fuzzy instruction "TURN TO THE LEFT" is processed in the same way as the case of "TURN TO THE RIGHT" instruction described above.

Other Functions

There are additional functions other than those previously described.

1. *At the wriggle*: On the interpretation of GO-instruction, the robot takes the course where the grade of direction holds the highest of all the alternatives. If the course which the robot is going to has a shape like the letter L, the robot goes along the road because the highest grade is zero and there are no other alternatives with zero grade.
2. *T shaped path*: On the interpretation of GO-instruction, if the robot

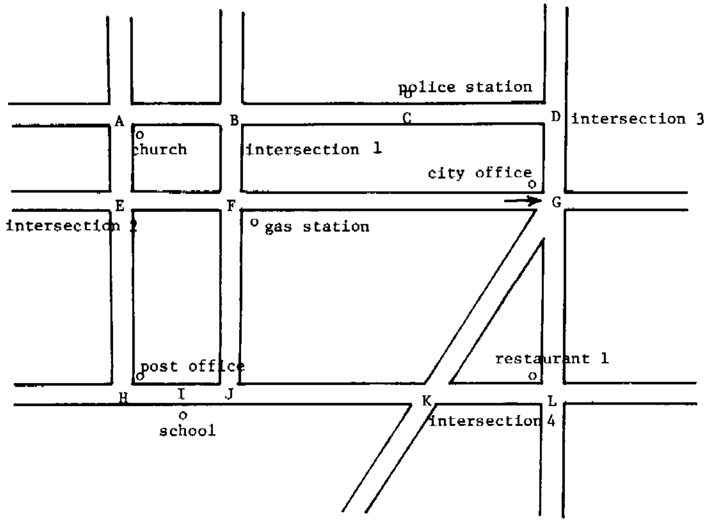

FIGURE 13. Town B.

runs into T shaped path, the process of interpretation of that fuzzy instruction ends. In this case, if the grade of the present point is lower than the threshold value, backtracking is performed. Otherwise, the interpretation of the next fuzzy instruction is performed.

3. *Loop*: If the robot's course crosses or corresponds to the one once passed through, the interpretation process of that fuzzy instruction ends (if the grade of the present point is lower than the threshold value, backtracking is performed). It owes to this function that the loop in robot's course is avoided in the system.
4. *The combination of GO and TURN instructions*: When a turn-instruction succeeds a go-instruction, the points where the successive turn-instruction cannot be interpreted are abandoned on the interpretation process. For example, let the following successive two fuzzy instructions

GO ABOUT 20 STEPS

(22)

TURN TO THE RIGHT

be given to the robot. In the town shown by Figure 6, the membership functions are assumed to be those given by Figure 7 and

Figure 12. This case differs from that of Example 7, and the machine instruction selected first is "GO TO G" because there is no possibility of turning to the right at the point H.

In this section we have explained each fuzzy instruction of the system and its function. In the following section, we shall show the inchworm robot which plays the role of a wandering person in the system.

INCHWORM ROBOT

The inchworm robot (Figure 14) moves on the real space. It is under the control of the computer through an interface unit. We have the following two commands for the inchworm robot.

1. Walk one step. (the length of one step is 15 cm)
2. Turn by one unit. (one unit is 30°)

From the computer, the inchworm robot is viewed as an output device. The controlling sequence of the robot is: The inchworm robot has a flag in its control unit. First the computer sets it to zero. Then it outputs a command of movement to the robot and waits until the flag becomes 1. In

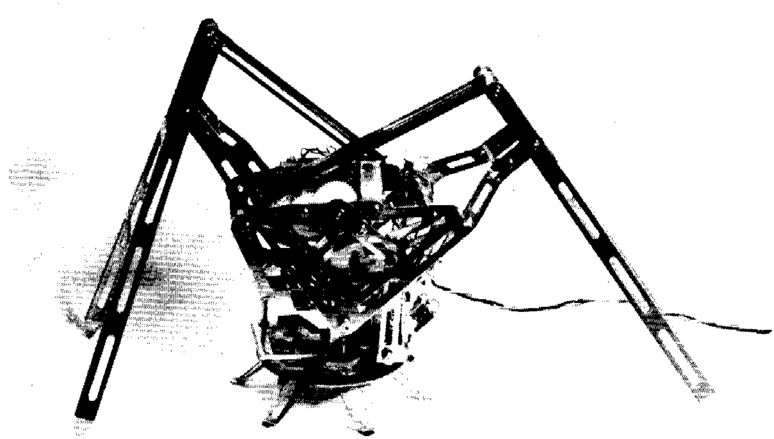

FIGURE 14. Inchworm robot.

return, the robot sets the flag to 1 when the execution of the command is finished. It is recognized as the end signal of the movement through the interface unit by the computer and the computer proceeds to the next task.

SOME EXAMPLES OF COMPUTER SIMULATIONS

In the map of the town where computer simulations are conducted (Figure 15), upward direction means the north. Lines of dots show roads. In the vertical lines two dots mean one step of the robot and in the horizontal lines does one dot one step. Also the symbol + means either an objective building or an intersection. Each objective building is represented by the following symbols (Table 2 and Figure 15),

S : school	R : restaurant
G : gas station	CO : city office
PL : police station	H : hospital
PS : post office	C : church

The number that follows the objective symbol makes it distinguishable from other same objectives (for example, the symbol G1 means the gas station 1).

From now on, we shall show a few examples of the computer simulations. In each simulation, the symbol *B* shows the locus of the robot's movement under backtracking. The symbol *C* means that the symbolized road was specially taken into consideration before backtracking. This function corresponds to the manner of a stranger when he has lost his bearings, and he cautiously looks over his way. The symbol @ indicates that the symbolized path was stepped on except under the preceding conditions.

Example 11: Figure 15 shows an example of the wandering robot starting from Restaurant 4 and going to School 3. The fuzzy program given to the robot is shown by Figure 15(a). The result obtained through the computer simulation is presented in Figure 15(b).

Example 12: Figure 16 shows an example of the wandering robot starting from Post Office 2 and finally arriving at Post Office 1. As shown in Figure 16(b), after lots of backtrackings it reached its destination. The given fuzzy instructions are listed in Figure 16(a).

```
HEAD NORTH
START FROM RESTAURANT 4.
GO ABOUT 5 UNITS.
TURN TO THE LEFT.
GO ABOUT 4 UNITS TO LARGE GAS STATION.
TURN TO THE RIGHT.
GO ABOUT 5 UNITS.
TURN TO THE LEFT.
GO TO LARGE SCHOOL.
END.
```

(a) Fuzzy program which shows the way from Restaurant 4 to School 3

```
QQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQ
Q                                     Q
Q +.......+...........+.......+.....+ Q
Q         .PS2        R3      .     . Q
Q         .                   .     . Q
Q         .                   .     . Q
Q         .                   .     . Q
Q +.......+.....+.....+.......+     . Q
Q ..      .     .     .       .   S2+ Q
Q . .     .     .     .       .    .  Q
Q .  .  G2+.....+     .       .   .   Q
Q .   .   .     .     .       .  .    Q
Q .    .  .     .     .       . .     Q
Q .     . .     .H1   .PL1    +.      Q
Q +      .+.....+.....+.......+       Q
Q .       ..    .     .       .       Q
Q .       . .   .     .   S1  .       Q
Q .       +  .  .     .   +   .       Q
Q .   +  @@   . .     .   .   .       Q
Q .   . @ @    .+CO1  .   .G1 .       Q
Q .   .@  @     ..    C...+...+.....+ Q
Q . C2+   @     . .   CPS1    .     . Q
Q .   @   @     .  .  C       .     . Q
Q .   @   @G3   .   . C       .     . Q
Q .   @   +@@@@@+@@@@@+.......+     . Q
Q .   @   .     G4    B       .     . Q
Q .   @   .           B       .     + Q
Q +  S3   +           B       .    R1 Q
Q .R2    .+BBBBBBBBBBB+.......+       Q
Q .     . .PL2        CC1     .       Q
Q .    .  .           C       .       Q
Q .   .   .           C       .       Q
Q .  .    .           C       .       Q
Q . .     .         R4+.......+CO2    Q
Q +.      +H2                         Q
Q                                     Q
QQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQ
```

(b) Locus of robot given fuzzy program in (a)

FIGURE 15. Simulation of Example 11 (starting point and destination are indicated by ◎ and ○).

TABLE 2. Position and Volume of Each Objective

Objective	Abbr.	Position	Volume	Objective	Abbr.	Position	Volume
School 1	S1	(12, 17)	3000	City office 1	CO1	(7, 15)	4000
2	S2	(17, 26)	1000	2	CO2	(14, 1)	1100
3	S3	(2, 8)	2600				
Gas station 1	G1	(12, 14)	500	Church 1	C1	(10, 6)	2500
2	G2	(4, 24)	1600	2	C2	(2, 13)	600
3	G3	(4, 10)	2500				
4	G4	(7, 10)	1800				
Police station 1	PL1	(10, 20)	1200	Hospital 1	H1	(7, 20)	3000
2	PL2	(4, 6)	2200	2	H2	(4, 0)	2200
Restaurant 1	R1	(17, 8)	700	Post office 1	PS1	(10, 14)	2000
2	R2	(0, 7)	800	2	PS2	(4, 32)	900
3	R3	(10, 32)	800				
4	R4	(10, 1)	2000				

```
HEAD SOUTH.
START FROM POST OFFICE 2.
GO ABOUT 10 UNITS.
TURN TO THE LEFT.
GO ABOUT 10 UNITS.
TURN TO THE LEFT.
GO TO POST OFFICE 1.
END.
```

(a) Fuzzy program which shows the way from Post Office 2 to Post Office 1.

```
QQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQ
Q                                     Q
Q +.......+...........+.......+.....+ Q
Q         CPS2        R3      B     . Q
Q         C                   B     . Q
Q         C                   B     . Q
Q         C                   B     . Q
Q +.......+.....+.....+.......+     . Q
Q ..      C     B     B       B   S2+ Q
Q . .     C     B     B       B    .  Q
Q .  .  G2+BBBBB+     B       B   .   Q
Q .   . B     B     B       B  .    Q
Q .    .  B     B     B       B .     Q
Q .     . B     BH1   BPL1    +.      Q
Q +      .+BBBBB+CCCCC+CCCCCCCC       Q
Q .       C@    .     .       .       Q
Q .       C @   .     .   S1  .       Q
Q .       C  @  .     .   +   .       Q
Q .   +  ..   @ .     .   .   .       Q
Q .   . . .    @+CO1  .   .G1 .       Q
Q .   ..  .     .@    @...+...+.....+ Q
Q . C2+   .     . @   @PS1    .     . Q
Q .   .   .     .  @  @       .     . Q
Q .   .   .G3   .   @ @       .     . Q
Q .   .   +.....+....@+BBBBBBB+     . Q
Q .   .   .     G4    .       .     . Q
Q .   +   .           .       .     + Q
Q +   S3  +           .       .    R1 Q
Q .R2    .+...........+.......+       Q
Q .     . .PL2        .C1     .       Q
Q .    .  .           .       .       Q
Q .   .   .           .       .       Q
Q .  .    .           .       .       Q
Q . .     .         R4+.......+CO2    Q
Q +.      +H2                         Q
Q                                     Q
QQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQ
```

(b) Locus of robot given fuzzy program in (a)

FIGURE 16. Simulation of Example 12 (starting point and destination are indicated by ◎ and ◯).

```
HEAD EAST.
START FROM RESTAURANT 3.
GO ABOUT 5 UNITS.
TURN TO THE RIGHT.
GO ABOUT 15 UNITS.
TURN TO THE LEFT.
GO ABOUT 10 UNITS TO SMALL RESTAURANT.
END.
```

(a) Fuzzy program which shows the way from Restaurant 3 to Restaurant 1

```
QQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQ
Q                                     Q
Q +.......+...........+@@@@@@@+.....+ Q
Q         .PS2        R3      C     B Q
Q         .                   C     B Q
Q         .                   C     B Q
Q         .                   C     B Q
Q +.......+.....+.....+.......+     B Q
Q ..      .     .     .       C   S2+ Q
Q . .     .     .     .       C    B  Q
Q .  .  G2+.....+     .       C   B   Q
Q .   .   .     .     .       C  B    Q
Q .    .  .     .     .       C B     Q
Q .     . .     .H1   .PL1    +B      Q
Q +      .+.....+.....+.......+       Q
Q .       ..    .     .       B       Q
Q .       . .   .     .   S1  B       Q
Q .       +  .  .     .   +   B       Q
Q .   +  ..   . .     .   .   B       Q
Q .   . . .    .+CO1  .   .G1 B       Q
Q .   ..  .     ..    +...+...+@@@@@+ Q
Q . C2+   .     . .   .PS1    C     @ Q
Q .   .   .     .  .  .       C     @ Q
Q .   .   .G3   .   . .       C     @ Q
Q .   .   +.....+.....+.......C     @ Q
Q .   .   .     G4    .       .     @ Q
Q .   +   .           .       .     @ Q
Q +   S3  +           .       .    R1 Q
Q .R2    .+...........+.......+       Q
Q .     . .PL2        .C1     .       Q
Q .    .  .           .       .       Q
Q .   .   .           .       .       Q
Q .  .    .           .       .       Q
Q . .     .         R4+.......+CO2    Q
Q +.      +H2                         Q
Q                                     Q
QQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQQ
```

(b) Locus of robot given fuzzy program in (a)

FIGURE 17. Simulation of Example 13 (starting point and destination are indicated by ◎ and ◯).

Example 13: Figure 17 shows an example of the wandering robot starting from Restaurant 3 and going to Restaurant 1. The fuzzy program and the locus of the robot are shown in Figure 17(a) and Figure 17(b).

CONCLUSION

We have discussed fuzzy programs which utilize the fuzzy conception and analyzed their processes. We have taken the process of a wandering person as an example and conducted computer simulations. In the system exploited we made a model of the human's process of wandering.

As application of this system we could see that the result of this system shows such robots could be used in exploration of space, where people handle the robot from a distant place. By using fuzzy instructions as commands, people will be able to handle the robot more easily, for almost all of the human commands and concepts are fuzzy.

REFERENCES

Chang, S. K. 1972. On the execution of fuzzy programs using finite state machines. *IEEE Trans. Electron. Comput.* C21:241–253.

Dimitrov, V. and Wechler, W. 1975. Optional fuzzy control of humanistic systems. *IFAC 6th Triennial World Congress*, Boston.

Fillmore, J. P. and Williamson, S. G. 1974. On backtracking: A combinatorial description of the algorithm. *SIAM J. Comput.* 3:41–55.

Golumb, S. W. and Baumert, L. 1965. Backtrack programming. *J. Assoc. Comp. Mach.* 12:516–524.

Tanaka, K. and Mizumoto, M. 1975. Fuzzy programs and their execution. In *Fuzzy sets and their applications to cognitive and decision processes*, ed. L. A. Zadeh, K. Tanaka, K. S. Fu and M. Shimura, pp. 41–76.

Zadeh, L. A. 1968. Fuzzy algorithms. *Inf. Control* 12:94–102.

Zadeh, L. A. 1973. Outline of a new approach to the analysis of complex systems and decision processes. *IEEE Trans. Syst. Man, Cybern.* SMC-3:28–44.

Received May 30, 1976

Request reprints from Masaharu Mizumoto, Department of Information and Computer Science, Faculty of Engineering Science, Osaka University, Toyonaka, Osaka 560, Japan.