ST.FRANCIS INSTITUTE OF TECHNOLOGY

Mount Poinsur, S.V.P. Road, Borivli (West), Mumbai - 400103

Computer Engineering Department

Academic Year: 2021-2022 **Class/Branch**: BE CMPN

Subject: CSC 703 Artificial Intelligence & Soft Computing **Semester**: VII

Name: Selas Moro **Roll no:** 39 **PID:** 182074 **Class:** BE CMPN B

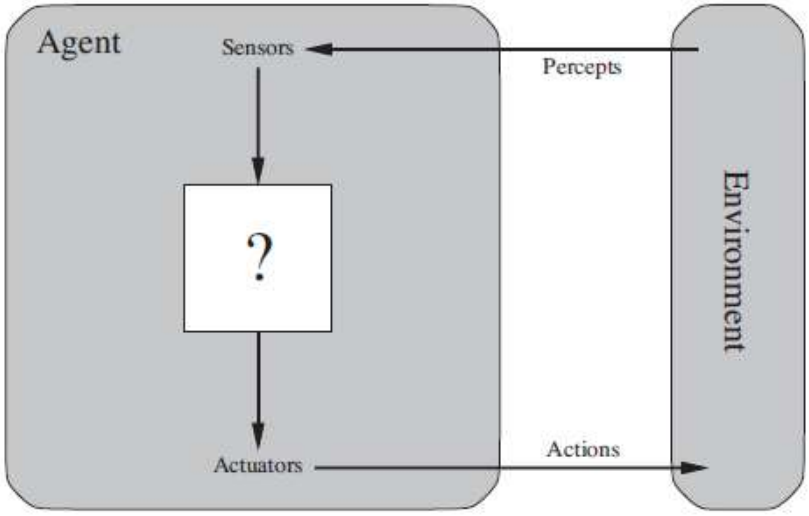
Experiment No. 2
Simple Reflex Agent Vacuum Cleaner

Aim: To develop a simple reflex agent program for the vacuum-cleaner world problem.

Theory:

Agents and Environment

An agent is anything that can be viewed as perceiving its environment through sensors and acting upon that environment through actuators. Agents can be hardware or software. Software agents are called softbots.



Types of Agents

1. Human Agent:

Sensors: eyes, ears, skin

Actuators: hands, legs, vocaltract and so on

2. Robotic Agent:

Sensors: camera, infrared finders

Actuators: motors

3. Software Agent:

Sensors: keystrokes, mouse clicks, files and network packets

Actuators: screen display, writing files, sending network packets and so on

Percept

- It is the complete set of input that an intelligent agent is perceiving at any given moment.
- The current percept or the sequence of percept can influence the actions of the agent

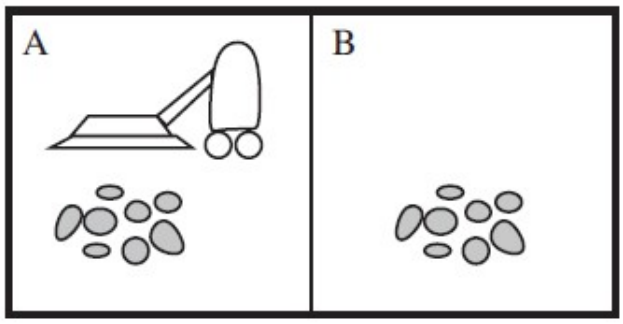
Percept Sequence

It is the complete history of everything the agent has perceived. An agent's choice of action at any given instant can depend on the entire percept sequence observed to date.

AGENT FUNCTION

- An agent's behavior is described by the **agent function** that maps any given percept sequence to an action.
- Tabulating the agent function is an external characterization and this would be a very large infinite table.

VACUUM CLEANER WORLD PROBLEM



- This particular world has just two locations: squares A and B.

- The vacuum agent perceives which square it is in and whether there is dirt in the square.
- It can choose to move left, move right, suck up the dirt, or do nothing.
- The simple agent function is :
    - If the current square is dirty, then suck
    - Else, move to the other square.

Agent Program

The agent function for an artificial agent will be implemented by an agent program. The agent function is an abstract mathematical description; the agent program is a concrete implementation, running within some physical system.

For Vacuum Cleaner

```
function REFLEX-VACUUM-AGENT([location,status]) returns an action
  if status = Dirty then return Suck
  else if location = A then return Right
  else if location = B then return Left
```

Simple Reflex Agents

- Simplest
- Acts on current percept
- Ignores history

Agent Program for Vacuum Cleaner

```
function REFLEX-VACUUM-AGENT([location,status]) returns an action
  if status = Dirty then return Suck
  else if location = A then return Right
  else if location = B then return Left
```

Figure 2.8 The agent program for a simple reflex agent in the two-state vacuum environment. This program implements the agent function tabulated in Figure 2.3.

- Vacuum agent program is very small because
    - Ignoring the percept history (cuts down no. of possibilities)

- when the current square is dirty, the action does not depend on the location.

- It can be implemented using condition–action rule
- E.g.
  - **if** car-in-front-is-braking
    then initiate-braking.
- Agent program: Build a general-purpose interpreter for condition–action rules and then create rule sets for specific task environments.

Task Environment

Task Environments are essentially the "problems" to which rational agents are "solutions."

Specifying the task Environment : PEAS Description

The task environment of an AI agent is specified by performance measure, environment, and the agent's actuators and sensors. This is acronymically given as the PEAS (Performance, Environment, Actuators, and Sensors) description. In designing an agent, the first step must always be to specify the task environment as fully as possible.

Example: Automated Taxi Driver

Agent Type	Performance Measure	Environment	Actuators	Sensors
Taxi driver	Safe, fast, legal, comfortable trip, maximize profits	Roads, other traffic, pedestrians, customers	Steering, accelerator, brake, signal, horn, display	Cameras, sonar, speedometer, GPS, odometer, accelerometer, engine sensors, keyboard

Implementation:

Experiment Exercise :

1. Develop a simple reflex agent program for the vacuum-cleaner world problem. The vacuum cleaner world is defined as follows: There are 2 states (left, right), initially a vacuum cleaner could be in any one state. There are 3 operations: **suck** if dirty, **move to right** if left state is clean and **move to left** if right state is clean). If both the states are clean then exit (**Nooperation**).

Code :

```
A=int(input("Enter 0 for clean and 1 for dirty for A:"))
B=int(input("Enter 0 for clean and 1 for dirty for B:"))
V=int(input("Enter vacuum location for A(0)(LEFT) and B(1)(RIGHT):"))
C=0

if V==0:
   print("Vacuum is at A")
   if A==0:
      print("Location A is cleaned ,Moving to Location B")
      C=C+1
      print("Cost is {}".format(C))
      if B==0:
         print("Location B is clean")
      else:
         print("Location B is dirty , so cleaning")
         C = C + 1
         print("Cost is {}".format(C))
         B=B-1
         print("Location B cleaned")
   else:
      print("Location A is dirty, so cleaning")
      C = C + 1
      print("Cost is {}".format(C))
      A=A-1
      print("Location A cleaned, Moving to Location B")
      C = C + 1
      print("Cost is {}".format(C))
      if B==0:
         print("Location B is clean")
      else:
         print("Location B is dirty , so cleaning")
         C = C + 1
         print("Cost is {}".format(C))
         B=B-1
         print("Location B cleaned")
else:
   print("Vacuum is at B")
   if B == 0:
      print("Location B is cleaned ,Moving to Location A")
      C = C + 1
      print("Cost is {}".format(C))
      if A == 0:
         print("Location A is clean")
```

```
        else:
            print("Location A is dirty , so cleaning")
            C = C + 1
            print("Cost is {}".format(C))
            A = A - 1
            print("Location A cleaned")
    else:
        print("Location B is dirty, so cleaning")
        C = C + 1
        print("Cost is {}".format(C))
        B = B - 1
        print("Location B cleaned, Moving to Location A")
        C = C + 1
        print("Cost is {}".format(C))
        if A == 0:
            print("Location A is clean")
        else:
            print("Location A is dirty , so cleaning")
            C = C + 1
            print("Cost is {}".format(C))
            A = A - 1
            print("Location A cleaned")
if A==0 & B==0:
    print("NO Operation")
    print("Cost is {}".format(C))
```

Output:

Case 1: A and B both are Dirty.

```
= RESTART: C:\Users\Administrator.MAHESHC\Documents\SEM 7\Airtificial intelligence\39_
exp2.py
Enter 0 for clean and 1 for dirty for A:1
Enter 0 for clean and 1 for dirty for B:1
Enter vacuum location for A(0)(LEFT) and B(1)(RIGHT):0
Vacuum is at A
Location A is dirty, so cleaning
Cost is 1
Location A cleaned, Moving to Location B
Cost is 2
Location B is dirty , so cleaning
Cost is 3
Location B cleaned
NO Operation
Cost is 3
```

Case 2 : A is clean and B is dirty.

```
= RESTART: C:\Users\Administrator.MAHESHC\Documents\SEM 7\Airtificial intelligence\39_exp2.py
Enter 0 for clean and 1 for dirty for A:0
Enter 0 for clean and 1 for dirty for B:1
Enter vacuum location for A(0)(LEFT) and B(1)(RIGHT):0
Vacuum is at A
Location A is cleaned ,Moving to Location B
Cost is 1
Location B is dirty , so cleaning
Cost is 2
Location B cleaned
NO Operation
Cost is 2
```

Case 3 : A is dirty and B is clean.

```
========== RESTART: C:\Users\Administrator.MAHESHC\Documents\SEM 7\Airtificial intelligence\39_exp2.py ==========
Enter 0 for clean and 1 for dirty for A:1
Enter 0 for clean and 1 for dirty for B:0
Enter vacuum location for A(0)(LEFT) and B(1)(RIGHT):0
Vacuum is at A
Location A is dirty, so cleaning
Cost is 1
Location A cleaned, Moving to Location B
Cost is 2
Location B is clean
NO Operation
Cost is 2
```

Case 4 : A and B both are clean.

```
========== RESTART: C:\Users\Administrator.MAHESHC\Documents\SEM 7\Airtificial intelligence\39_exp2.py ==========
Enter 0 for clean and 1 for dirty for A:0
Enter 0 for clean and 1 for dirty for B:0
Enter vacuum location for A(0)(LEFT) and B(1)(RIGHT):0
Vacuum is at A
Location A is cleaned ,Moving to Location B
Cost is 1
Location B is clean
NO Operation
Cost is 1
```

Post Experiment Exercise:

1. Identify a real robotic vacuum cleaner in the market and define the PEAS description for it.

Solution:

Mi Robot Vacuum-Mop P, 2100:

• Performance: cleanness, efficiency: distance traveled to clean, battery life, security.
• Environment: room, table, wood floor, carpet, different obstacles.
• Actuators: wheels, different brushes, vacuum extractor.
• Sensors: camera, dirt detection sensor, cliff sensor, bump sensors, infrared wall sensors.

Conclusion: In this experiment ,I have studied about an simple reflex agent for the Vacuum cleaner world problem. I have also understood the necessary concepts and have implemented them in the above experiment using python.