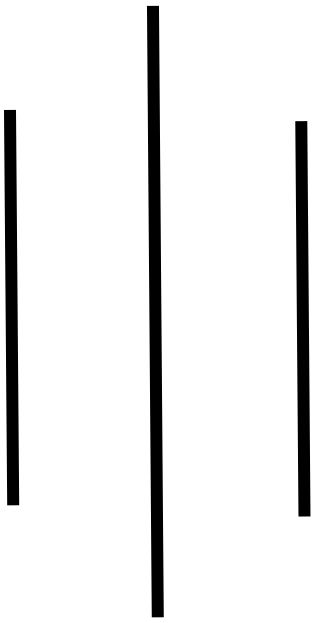
Assignment – II

Vacuum Cleaner World Simulation

(A Simple Reflex Agent-Based Model)



Submitted by: Bikram Giri

Submitted to: Samriddha Pathak

Submission Date: 2025/07/10

Table of Contents

Problem Statement

- Implement simple intelligent agent (Simple reflex agent for vacuum cleaner agent in our context.) Consider the vacuum cleaner world as shown in Figure1 and eight possible states in vacuum world as shown in Figure 2.

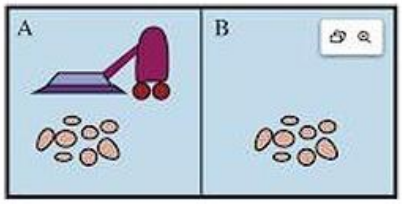


Figure 1: Vacuum Cleaner World

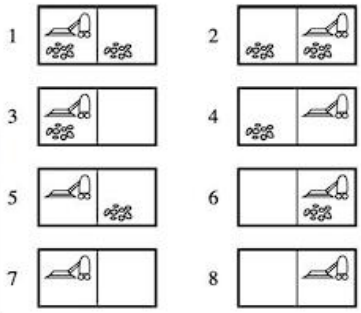


Figure 2: The eight possible states in vacuum world

- Draw the state space graph of the vacuum cleaner world domain that we have taken in to account. (Our goal is to reach either state 7 or state 8.)
- Formulate the appropriate algorithms and implement them in any high-level language as per convenient preferably python.

State Space Graph of the Vacuum Cleaner World

In this environment, the vacuum cleaner can be in:

- Room A or Room B
- Each room can be Clean or Dirty

This results in 2 (locations) × 2 (Room A states) × 2 (Room B states) = 8 states.

State Representation:

Each state is defined as:

(Vacuum_Location, RoomA_Status, RoomB_Status)

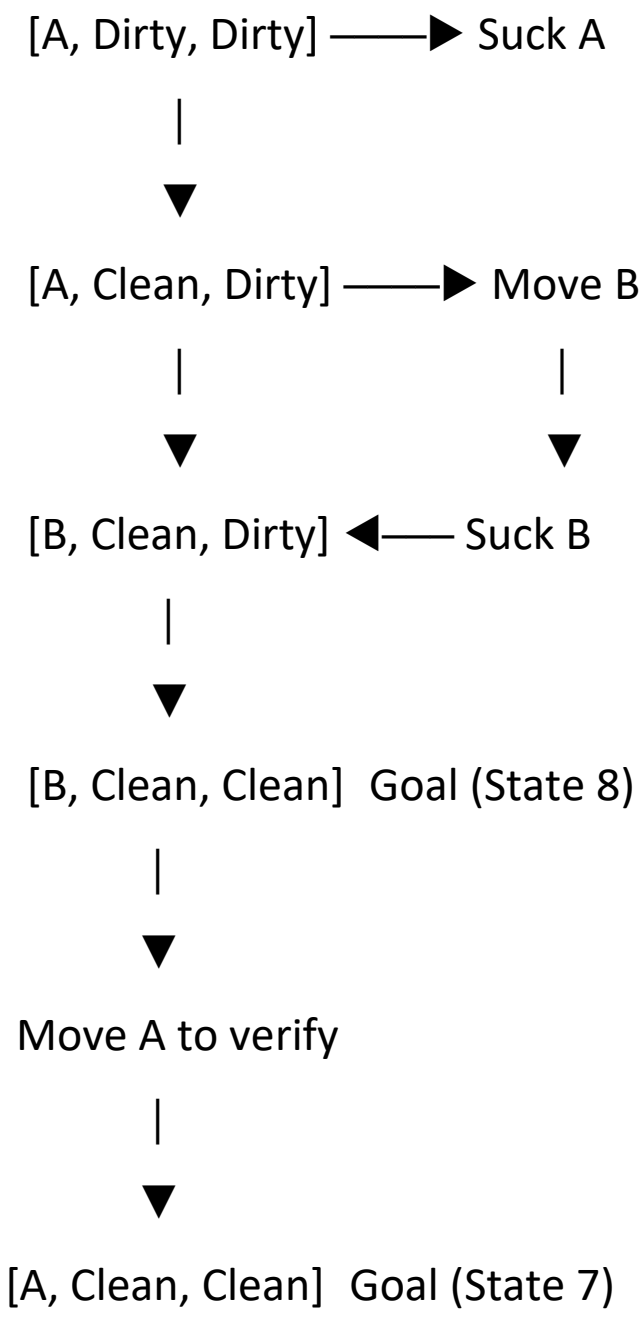
Where:

- Location: A or B
- Status: Dirty or Clean

All 8 States:

State No.	Vacuum Location	Room A	Room B	Description
1	A	Dirty	Dirty	Start at A, both rooms dirty
2	B	Dirty	Dirty	Start at B, both rooms dirty
3	A	Clean	Dirty	A is clean, B is dirty
4	B	Clean	Dirty	A is clean, B is dirty
5	A	Dirty	Clean	A is dirty, B is clean
6	B	Dirty	Clean	A is dirty, B is clean
7	A	Clean	Clean	Goal State
8	B	Clean	Clean	Goal State

State Space Graph



Similar transitions apply for starting from [B, Dirty, Dirty]

Algorithm Formulation

This is the Simple Reflex Agent Algorithm you're using:

Algorithm Steps:

1. Input current location and room statuses.
2. If current room is dirty → Clean it.
3. Else → Move to the other room.
4. Repeat steps 2–3 until both rooms are clean.
5. Confirm both rooms clean:
   - If not in the other room, visit and verify.
6. Return to initial location to confirm cleanliness.
7. End simulation.

Python Code

```
import time

class VacuumCleanerWorld:
    def __init__(self, location='A', status_A='Dirty',
status_B='Dirty'):
        self.initial_location = location
        self.location = location
        self.status = {'A': status_A, 'B': status_B}
        self.actions = []

    def display_status(self):
        print(f"\n[Current Status]")
        print(f"Vacuum Location: Room {self.location}")
        print(f"Room A: {self.status['A']}, Room B:
{self.status['B']}")
        time.sleep(1)

    def is_goal_reached(self):
        return self.status['A'] == 'Clean' and self.status['B'] ==
'Clean'

    def move_to(self, destination):
        if self.location != destination:
            direction = "Right" if destination == 'B' else "Left"
            print(f"Action: Moving {direction} to Room
{destination}...")
            time.sleep(1.5)
            self.location = destination
            self.actions.append(f"Move to Room {destination}")

    def simple_reflex_agent(self):
        print("\nStarting Vacuum Cleaner Simulation\n")
        time.sleep(1)

        # Step 1: Cleaning Phase
        while not self.is_goal_reached():
            self.display_status()

            if self.status[self.location] == 'Dirty':
```

```
                print(f"Action: Sucking dirt in Room
{self.location}...")
                time.sleep(2)
                self.status[self.location] = 'Clean'
                self.actions.append(f"Suck in Room {self.location}")
            else:
                next_room = 'B' if self.location == 'A' else 'A'
                self.move_to(next_room)

        # Step 2: Confirm cleanliness of other room only if vacuum is
not already there
        self.display_status()
        print("✅ Both rooms appear clean.")
        other_room = 'B' if self.initial_location == 'A' else 'A'

        if self.location != other_room:
            print(f"\n🔍 Visiting Room {other_room} to double-
check...")
            self.move_to(other_room)
            self.display_status()
            if self.status[other_room] == 'Clean':
                print(f"✅ Room {other_room} confirmed clean.")
            else:
                print(f"❌ Warning: Room {other_room} is dirty!
Cleaning again...")
                self.status[other_room] = 'Clean'
                self.actions.append(f"Suck in Room {other_room}")

        # Step 3: Return to initial location
        print(f"\n🔄 Returning to initial starting room: Room
{self.initial_location} to double-check...")
        self.move_to(self.initial_location)
        self.display_status()
        if self.status[self.initial_location] == 'Clean':
            print(f"✅ Verified: Room {self.initial_location} is
clean.")
        else:
            print(f"❌ Warning: Room {self.initial_location} is dirty!
Cleaning again...")
            self.status[self.initial_location] = 'Clean'
```

```
            self.actions.append(f"Suck in Room
{self.initial_location}")

        print("\n🏁 Simulation Complete: All rooms clean and
verified.")
        print("🧾 Action Path:", " → ".join(self.actions))


# Test: Start at Room A with both rooms clean
vacuum = VacuumCleanerWorld(location='A', status_A='Clean',
status_B='Dirty')
vacuum.simple_reflex_agent()
```