

SYSC5103 Assignment 1 - Question 2

How to Read State Machine Config File:

File is in Krislet Folder. File Name: StateAgent_Table.csv

The file has the following format

- Line 1: Start State of the State Machine [Contains only one State name. Eg: Q1]
- Line 2: Acceptance States of the State Machine in CSV format [The states should be present in Line 3 of the file. Eg: Q2,Q3]
- Line 3: Set of States in CSV format [Eg: Q1,Q1,Q3,Q4]
- Line 4: Set of Alphabets in CSV format [Eg: a1,a2,a3]
- Line 5 - EOF: Contains the transitions result (Ac,Q) in tabulated format. Action and State should be separated by a ",". Transitions should be separated by a ";". [Eg: Ac1,Q1; Ac2,Q2; Ac3,Q3; ...]

Line 5 to EOF contains a table format for the transitions where each line corresponds to the specific index of the State defined in Line 3 and columns ("," separated in each line) corresponds to the specific index of alphabet defined in Line 4

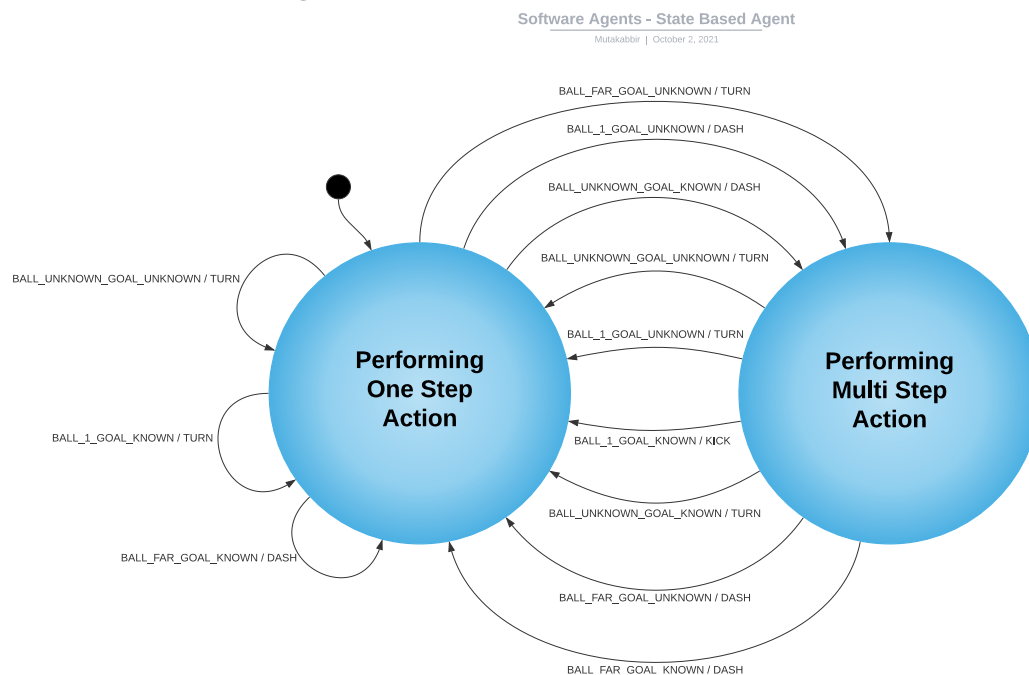
How to Edit State Machine Config File:

Points to remember during editing the file:

- There should be a minimum of 5 lines in the file.
- All States present in Line 1, Line 2 and from Line 5 onwards should be present in Line 3.
- All the Alphabets present from Line 5 onwards should be present in Line 4.
- Number of Lines from Line 5 onwards to the EOF should be equal to the number of states present in Line 3 and the number of transitions ("," separated) in each line should be equal to the number of alphabets defined in Line 4

Edit actions and States from Line 5 to EOF based on the above rules.

State Machine Diagram:



The State Machine has 2 States

- Performing One Step Action: **This state reminds the agent that the current action it will take is not dependent on its past actions.**

SYSC5103 Assignment 1 - Question 2

- Performing Multi Step Action: **This state reminds the agent that the current action it will take will be dependent on its past action.**

Agents can perform 3 actions:

- Turn: rotates the agent by a certain angle.
- Dash: moves the agent forward.
- Kick: agent kicks the ball.

The agent Senses 2 factors from the Environment:

- Ball: which can be in 3 possible situations
1) at a distance of "1", 2) "Far" from the agent, 3) the ball is "unknown"
- Goal: which can be in 2 possible situations
1) agent "Knows" where the goal is, 2) the goal is "UnKnown" to the agent

Based on the above 2 factors the environment can have the following 6 possibilities.

- Ball and Goal are Unknown
- Ball is Unknown but the goal is Known
- Ball is far and the goal is Unknown
- Ball is far and the goal is Known
- Ball is at a distance of 1 and the goal is Unknown
- Ball is at a distance of 1 and the goal is Known

Code Description:

- Abstract "Action" class structures the agent's actions. These are extended by "ActionTurn", "ActionDash", "ActionKick" classes which implement their own "do_action" method that performs the action by the agent.
- The "Environment" class stores the environment variables and returns the proper environment when passed the parameters to it.
- The "StateMachine" Class does the following tasks:
 - reads the config file for state machine.
 - sets the current state.
 - returns the action to perform while setting the new state when provided with the current environment.

Execution:

Exactly same as Krislet execution.

Expected Behaviour:

- Agent will turn to find the ball
- Once the agent finds the ball it will move towards it and align itself with the goal.
- If the agent comes near the ball it will try to kick it.

Proof of State Based Behaviour

