ModernComputerGames — A4

Marcus Friis Klausen

Wednesday 27th November, 2024

NOTE

When I open the project on my PC after zipping, it opens a blank scene and the correct scene (/Assets/Scenes/SampleScene) needs to be reimported. I assume this will have to be done for the person grading the assignment as well.

Parameters x and v

Paramater x is sat to 0.75, and can be found as a public field in Cars.cs attached to the "Cars" GameObject.

Parameter v, called nCars in the project, is sat to 12 and can also be found as a public field in Cars.cs attatched to the "Cars" GameObject.

HTN-Based AI

(a) Detail the world state representation

1. **Frog Position**: Vector3 of frog coordinates
2. **Car Positions**: List of List of Vector3 for coordinates of cars in each respective lanes
3. **Lane Speeds**: List of int for speed in each lane
4. **Car Approaching Lane**: Bool indicating if a car is getting close, dependent on speed, in the lane of the frog, given that it's in a lane
5. **Car Approaching Above**: Bool indicating if a car is getting close, dependent on speed, in the lane above the frog
6. **Car Approaching Below**: Bool indicating if a car is getting close, dependent on speed, in the lane below the frog
7. **Distance To Fly**: Float of distance to fly from frog
8. **Flies Eaten**: Int for number of flies currently eaten

9. **Jump Status**: Bool for if frog is currently jumping
10. **Move Status**: Bool for if frog is currently moving
11. **Cooldown Status**: Bool for if frogs actions are currently on cooldown
12. **Time Remaining**: Float for seconds left of the game
13. **Valid Movement Direction**: Set of Direction for the possible directions possible to move given the frogs position

(b) draw the HTN structure showing all composite and primitive tasks

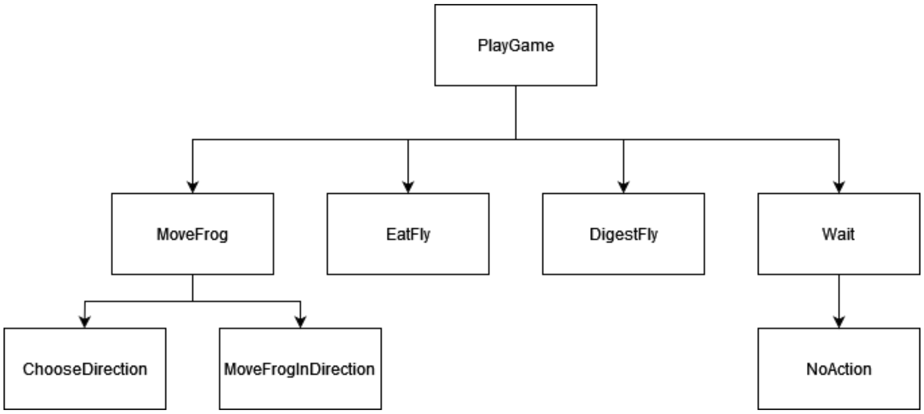

Figure 1: HTN Structure

(c) describe relevant pre/post conditions

1. **MoveFrog**:
   - **Preconditions**: Jump Status = false, Move Status = false
   - **Postconditons**: Frog Position updated dependent on direction and previous position, Move Status = true during movement
2. **ChooseValidPosition**:
   - **Preconditions**: Frog Position valid for moving to direction
   - **Postconditions**: Valid direction is selected
3. **MoveFrogInDirection**:
   - **Preconditions**: Valid direction chosen

- **Postconditions**: Frog Position updated dependent on direction and previous position, Move Status = true during movement

4. **EatFly**:
   - **Preconditions**: Flies Eaten < 2, Distance To Fly < 10
   - **Postconditions**: Flies Eaten incremented by 1

5. **DigestFly**:
   - **Preconditions**: Flies Eaten > 0, Move Status = false, Jump Status = false, Car Approaching = true
   - **Postconditions**: Flies Eaten decremented by 1, Jump Status = true during jump

6. **Wait**:
   - **Preconditions**: Car Approaching Above = true, Car Approaching Below = true, Car Approaching Lane = false
   - **Postconditions**: None

(d) include a list of of all possible plans your HTN can generate

1. MoveFrog
2. MoveFrog ⇒ EatFly
3. MoveFrog ⇒ EatFly ⇒ Wait
4. MoveFrog ⇒ EatFly ⇒ Wait ⇒ DigestFly
5. MoveFrog ⇒ EatFly ⇒ DigestFly
6. MoveFrog ⇒ EatFly ⇒ DigestFly ⇒ Wait
7. MoveFrog ⇒ DigestFly
8. MoveFrog ⇒ DigestFly ⇒ Wait
9. MoveFrog ⇒ Wait
10. EatFly
11. EatFly ⇒ DigestFly
12. EatFly ⇒ MoveFrog
13. EatFly ⇒ MoveFrog ⇒ DigestFrog
14. EatFly ⇒ Wait
15. DigestFly
16. Wait