Assignment # III

IE502014 Artificial Intelligence: Part C, Spring 2020

Faculty of Information Technology and Electrical Engineering
NTNU in Ålesund

Saleh Alaliyat

March 09th, 2020

Deadline 29.03.2020 at 23:59

- This is the first mandatory assignment in the *Part C*. You must deliver a Unity package that include all the unity 3d parts (3d objects, codes and all the assets), and a report (PDF) that has all the theory parts and the explanation of the unity3d exercises as well.

- Name the file that you deliver in the Blackboard by your name and deliver one compressed file that includes all the files.

- Mange your file system in unity 3d before delivering (scenes, prefabs, and scripts).

- Test your unity 3d package before delivering.

- Remember to deliver your assignment before the deadline and remember that you have to pass all the mandatory assignments to get access to the final exam.

- *Note:* you are free to choose any implementation tool (i.e. software) to accomplish the given tasks, even if the assignment is designed to be implemented in Unity3d.

Good Luck

Part I: Goal-oriented agent:

Theory

1. Explain what is a movable-goal oriented agent?
2. What can a movable-goal oriented agent represent?
3. Explain how a movable-goal oriented agent can control its speed and direction toward a goal?
4. Explain how movable-goal oriented agents can choose one of several targets?

Unity3d:

5. Experiment with one goal: Create a movable-goal oriented agent with parameter and methods for service so that it can move with constant velocity until getting to a stationary target. Study the agent performance by measuring its ability to position itself to the target.
6. Experiment with a moving target: Create a new method that gives the agents the ability to read a movable target (i.e. moving in a circle). Study the agent performance by measuring its ability to position itself to the target.
7. Experiment with several targets: Set up a set of goals with random positions and values. Create a method to select targets with maximum value and study the agent's ability to optimise access of values.
8. Modify the previous tasks by using a PID controller.

Part II: Flocking behaviour:

Theory:

1. Explain Reybolds boids model (Cohesion, separation and alignment), write methods (equations) to calculate them also.

Unity3d:

Implement Reynolds boids model:

2. Create a goal-oriented movable agent and let it move in a circle around an object or an axis, or let it move between many defined waypoints, and then use it as a target that all agents (boids) will follow it in the model (Leader).
3. Create a swarm of agents in a landscape.
4. Create a method for agent cohesion rule and study how the agents move toward a common point. Adjust the cohesion force input to the agents to have a natural speed.
5. Create a method for agent separation rule and study how the agents stand in relation to each other (keeping distance between each other). Adjust the separation force parameter to the agents to get a suitable separation distance.
6. Create a method for alignment rule and study how the agents' directions stand in relation to each other. Adjust the alignment force parameter to the agents to get suitable effect on the agents' directions.

7. Change the different forces parameters (Cohesion force parameter, Separation force parameter, and alignment force parameter) between many values and see the effects on the whole swarm.
8. Discuss your algorithm and the results you achieved.

Part III: Flocking behaviour applications:

Theory:

1. What are queuing agents?
2. Give some examples of queuing agents?
3. What are the typical tasks of queuing agents?
4. Explain a method to model queuing agents?

Unity3d:

Implement a case of queuing agents:

5. Create a landscape where the agents must pass through a narrow path one after one.
6. Modify the swarm you have implemented in Unity 3D (Flocking behaviour), by letting the leader move through this narrow path, so the others will follow him.
7. Adjust the parameters of the moving vector (combination of all the steering behaviours) to have a queuing.
8. Agents supposed to have a natural speed.
9. Study how a good balance between different steering behaviours can form queuing.
10. Comment on the result of your experiment.

Theory:

11. Explain a method to train the agent's parameters with a Genetic algorithm in your model?

Part IV: Steering behaviours:

Theory

Explain the following simple steering behaviours for individuals and pairs (refer to Reynolds paper).

1. Seek and flee.
2. Pursue and evade.
3. Wander.
4. Arrival
5. Obstacle avoidance
6. Containment.
7. Wall following
8. Path following

Unity3d:

9. Implement arrival behaviour.
10. Implement leader following behaviour.
11. Implement wall following behaviour.
12. Implement path following behaviour.