Design Reinforcement Learning Elements

<u>Instructions</u>

The goal of this assignment is to design reinforcement learning elements. It consists of two parts: a reading portion and a questions portion. You will earn points based on your efforts put into the assignment. The assignment will take 95 minutes.

Please read the following instructions carefully. Failure to comply to the instructions will result in your dismissal from the assignment.

Independent assignment: you are to work independently on the assignment, no collaboration allowed, and please keep quiet, avoid making noise. If you have questions, please raise your hand.

Materials: please keep any electronic devices away, use only provided materials. Do not use Google or other websites for help.

Integrity and confidentiality: don't disclose assignment information after you are done.

Legibility: please ensure that your handwriting is easily readable

<u>Please fill out the information below</u>

Name: ____

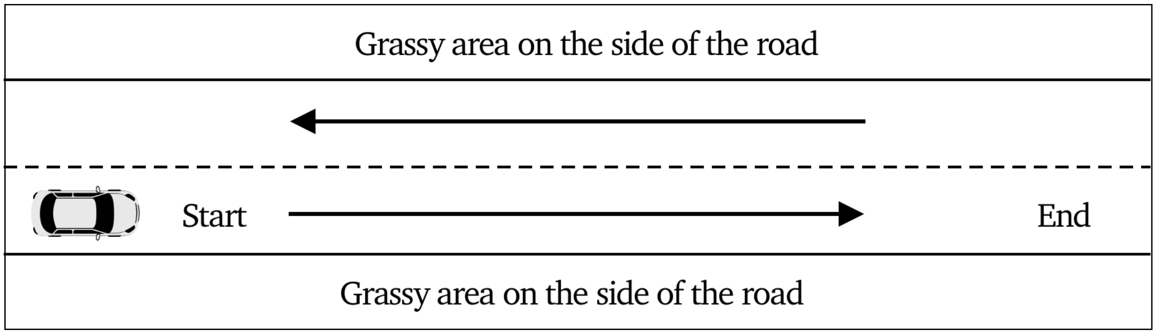
Sex: Male / Female Age: ______

Have driven a car before: Yes / No

If Yes, approximately for how many years?: ______

Reinforcement Learning is a popular machine learning approach where an agent (a robot vehicle in this assignment) learns to take optimal actions through trial and error.

<u>RL Driving Example</u>



Consider training a robot vehicle to drive on a two-way road environment, as shown above, with the goal of driving from "Start" to "End" while staying in its lane. The robot vehicle takes actions by accelerating and steering. As it drives, its position within the environment changes causing the environment to transition from its current state to the next state. The robot vehicle receives rewards for each action taken, earning positive rewards for good actions, such as driving in its lane towards “End” and negative rewards for bad actions, such as crossing the dotted line or driving in the grassy area.

We allow the robot to drive for an episode lasting 100 seconds, after which it stops and reflects on its performance. During this reflection, the robot vehicle identifies actions that led to good rewards and aims to repeat them while avoiding actions associated with bad rewards. After reflecting, the robot vehicle is reset to the "Start" position, and the 100-second episode begins again. Through multiple episodes of trial and reflection, the robot vehicle learns to successfully drive from "Start" to "End" while staying in lane.

<u>State Space</u>

The state space S, is the set of all environment states that can occur during an episode. To illustrate this, consider the road to be divided into a grid such that Column 2 is the start and Column 8 is the end with Row 3 being the lane the robot vehicle is to drive on. Rows 1, 2, and 4 represent areas we do not want the robot vehicle to drive in.

	1	2	3	4	5	6	7	8
1								
2								
3		Start						End
4								

If we represent each state by the row and column numbers the robot vehicle is in, the state space can be the robot vehicle's position, expressed by the following equation:

S = (x, y)

where x denotes the row and y denotes the column. For example, the robot vehicle's start state is (3, 2), and its end state is (3, 8).

Next, let's design a more complex state space. Assume the robot vehicle's position can be mapped on a coordinate plane with an x-axis of 30m and y-axis of 10m. Let's also assume that we can access the robot vehicle's speed. Now, we can represent the states like the following:

S = (robot_position, robot_speed)

with examples such as ((7.43m, 3.54m), 10m/s)) or ((26.20m, 8.52m), 16m/s)).

Reward Function

The rewards the robot vehicle receives helps it learn to take actions to achieve its goal. In our example above, the goal is to drive while staying in lane and reach "End." A reward function that encourages this behavior can be:

R = – dist_from_lane + reach_goal

where dist_from_lane is the robot vehicle's lateral distance from the lane we want it to drive in. This term grows larger as the robot vehicle strays from its lane, but we want to discourage such behavior, so we make it negative to punish this behavior. The term reach_goal is a positive value, let's say +10, if the robot reaches "End", and 0 otherwise.

The reward function can also encourage certain behaviors or introduce constraints, we can add a term to encourage the robot vehicle to drive under the speed limit:

R = – dist_from_lane + reach_goal + under_speed_limit

where the new term, under_speed_limit, is a +10 if the robot is under the speed limit, but a -10 if not.

It's fine for a reward to be a negative value when calculated; what matters is that as the agent learns to fulfill the goal, the reward gradually increases.

Assignment Instructions

This assignment has three parts. For each part, **you need to answer all questions**.

- A bank of metrics to choose from is provided for reference, however, you are not limited to the listed items, you are free to come up with anything new.

- You can transform your metrics with **any statistical computations** such as mean, median, mode, standard deviation, variance.

- Your state space should be something physically measurable within the road environment. Feel free to use any combination of metrics you believe is necessary; however, your state space **should not just be all the metrics**.

- Your reward function should be something physically measurable within the road environment and also **evaluates to a single numerical value**.

- Examples of an invalid answers
  - "No vehicle should move too fast": this answer does not provide a physically measurable value
  - "Every time-step the agent does not solve the problem, penalize with a -100. And every time-step the agent solves the problem, reward with a +1.": although this reward function evaluates to a numerical value, it leaves what it means to "solve a problem" open to interpretation.

<u>Bank of Metrics</u>

Vehicle:
- Position
- Speed
- Acceleration/ Deceleration
- Gap to vehicle's front or vehicle's rear
- Fuel consumption

Road environment:
- Number of vehicles that pass through a fixed point
- The density of vehicles in a given road length
- Speed limit
- Average speed
- Average acceleration
- Average time spent on not moving

Driving behavior:
- Distance/ time to come to complete stop
- Time to collide with vehicle in front
- Change in acceleration/ smoothness of acceleration
- Minimum speed

Again, you are not restricted to the metrics listed above.

Examples for S and R using above metrics including example explanations when using metrics not explicitly in the bank.

S = (speed_limit, avg_fuel_consumption)

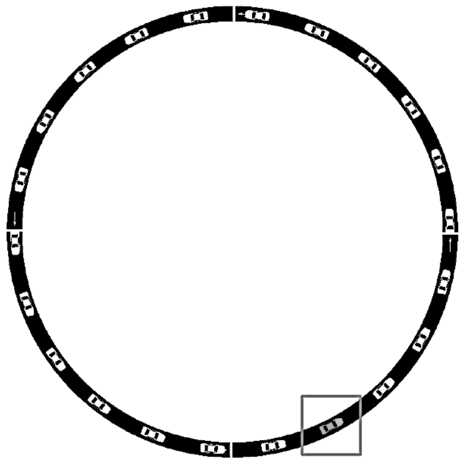
- avg_fuel_consumption is the average fuel consumption for every vehicle in the road environment.

R = distance_complete_stop - avg_time_to_collision

- avg_time_to_collision is the average time to collide with vehicle in front for all vehicles in the road environment.

These S and R are just examples and ineffective for any part.

Assignment Part 1. Ring Road



There are 21 human-driven vehicles and 1 robot vehicle (with box around it) driving in a ring road.

Problem: When there are only human-driven vehicles, drivers' imperfect behaviors will result in "stop-and-go" traffic patterns: a continuous pattern of vehicles stopping and moving again throughout the traffic flow. This decreases the average speed of vehicles in the road, increases fuel consumption, and increases the travel time.

Goal: The robot vehicle prevents stop-and-go traffic.

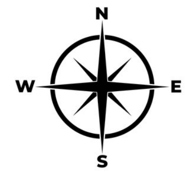
1. Give the state space as a tuple of metrics. If using metrics that are not in the bank of metrics, explain what they mean with the space provided.

S =

2. Give the reward function as equation that evaluates to a single numerical value. If using metrics that are not in the bank of metrics, explain what they mean in the space provided.

R =

3. Briefly explain (no more than 3 concise sentences) why this reward function helps the robot achieve its goal.



Assignment Part 2. Bottleneck

Human-driven vehicles and robot vehicles (with a box around them) flow onto this road traveling east with a ratio of human-driven vehicles to robot vehicles at 9:1 where after 9 human-driven vehicles drive on to the road, a robot vehicle drives onto the road. The number of lanes go from four to two to one, creating a bottleneck.

Problem: The vehicles need to wait to drive onto the single lane forming queues that build up quickly. This congestion causes vehicles to slow down significantly with many of them having to come to a complete stop. Additionally, this lowers the number of vehicles that can pass through on to the single lane.

Goal: Robot vehicles alleviate traffic congestion and increase the speed of vehicles on the road and the number of vehicles that pass through the singe lane/bottleneck point.

1. Give the state space as a tuple of metrics. If using metrics that are not in the bank of metrics, explain what they mean with the space provided.

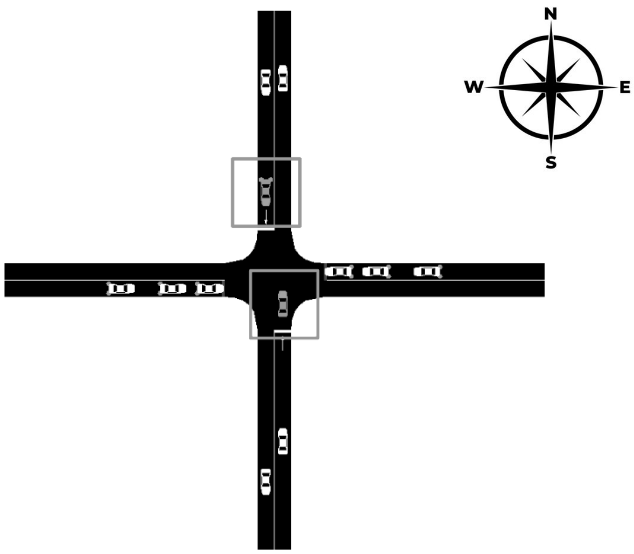
S =

2. Give the reward function as equation that evaluates to a single numerical value. If using metrics that are not in the bank of metrics, explain what they mean in the space provided.

R =

3. Briefly explain (no more than 3 concise sentences) why this reward function helps the robot achieve its goal.

Assignment Part 3. Four-Way Intersection



A mixture of human-driven vehicles and robot vehicles (with a box around them) drive in the north/south directions. After 8 human-driven vehicles drive on the road, 2 robot vehicles drive onto the road. Only human-driven vehicles drive in the east/west directions. Vehicles only drive straight through intersection. The traffic is higher in north/south directions than the east/west directions.

Problem: Higher traffic in the north/south directions makes it harder for the human-driven vehicles in the east/west directions to cross the intersection. Human-driven vehicles in the east/west directions typically have to wait some time, causing a queue of waiting vehicles to form in the east/west directions.

Goal: Robot vehicles control traffic flow in the north/south directions to allow for human-driven vehicles to cross through the intersection in the east-west directions.

1. Give the state space as a tuple of metrics. If using metrics that are not in the bank of metrics, explain what they mean with the space provided.

S =

2. Give the reward function as equation that evaluates to a single numerical value. If using metrics that are not in the bank of metrics, explain what they mean in the space provided.

R =

3. Briefly explain (no more than 3 concise sentences) why this reward function helps the robot achieve its goal.

Ending Questionnaire (OPTIONAL - Does not effect credit)

1. Was the introductory section on Reinforcement Learning clear and helpful? If not, explain what you think could be improved.

2. Do you have any other feedback that you would like to share?