

COSC 3P91 – Assignment 1 – 7376726

PARKER TENBROECK, Brock University, Canada

Traffic simulation layout and design

1 DESIGN OVERVIEWS

1.1 RoadMap

The RoadMap contains all information about the Intersections, Roads, and Vehicles that are apart of the map. The map is modeled as a directed graph where nodes are Intersections and edges are Roads.

1.1.1 Road. Roads act as edges in the graph and consist of one or more Lanes, each Lane holds the Vehicles that are currently on it. Roads exist only as a child to two intersections, one being the incoming and the other being the outgoing. this also implicitly determines the 'direction' of the road. Vehicles can only turn onto a road from the incoming end, and turn off a road when at the outgoing end. Vehicles can only travel in a single direction along the road. Vehicles must always start at the back most position on the incoming end of the road. The Road is responsible for updating all Vehicles currently on it.

1.1.2 Intersection. Intersections act as the nodes in the graph and can have none to many outgoing and incoming Roads. Intersections have a position x,y that specify where is exists in the RoadMap. Every Intersection is responsible of keeping track of what Turns are possible. Each Turn consists of a Lane from an incoming road, a name (used to indicate Left, Right, Forward, etc.) and a Lane from an outgoing road to turn onto. A turn also has a flag indicating if the turn is enabled. An arbitrary number of Turns are possible in any direction from any given Intersection.

Intersection can be extended to provide additional functionality. Three subclasses of Intersection are provided as an example. SourceIntersection, DrainIntersection, and TimedIntersection. SourceIntersection and DrainIntersection provide ways to add/remove Vehicles to the road system respectively. and TimedIntersection will act like a timed traffic light, preventing Vehicles from making turns until the 'light' turns green for them.

1.2 Vehicle

A Vehicle is some entity that can exist on a Lane on a Road in a RoadMap. A Vehicle can contain a Controller that influences how the Vehicle makes navigation decisions. Vehicles are abstract because they can represent many different forms of transportation however, navigation decisions should be left up to the Controller and not overridden. Other aspects of Vehicle should be modified like how it is drawn, and other specialized behavior related to the specific kind of Vehicle being implemented.

Two implementations of Vehicle are examples of how a Vehicle can be made, Car is a standard vehicle and Truck is a larger and slower vehicle.

Author's address: Parker TenBroeck, Brock University, 1812 Sir Isaac Brock Way, St. Catharines, ON, L2S 3A1, Canada.

1.3 IO

1.3.1 Display. The `Display` is responsible for drawing and displaying simulation visuals, it is constructed with `Input` so it can attach `Input`'s listeners to its window.

1.3.2 Input. `Input` listens to user input present on `Display` and provides easy access to event data. `Storage` holds all event data per frame, two are used to record new input for the next frame saving old input from the previous frame for consistency.

1.3.3 View. The `View` holds a `Display` and `Input`, it provides a 'view' into the simulation that can be zoomed / moved around. It also provides functionality to provide mouse events in the same coordinate space as the `Simulation`. A `View` can optionally follow a `Vehicle`.

1.4 Simulation

`Simulation` ties all these components together allowing the simulation to actually be ran. it allows access to all the different parts of the simulation alongside other information / configuration allowing things like pausing the simulation to allow players to choose a turn they want to take.

2 LOOKING

The player will be able to see the surrounding cars by looking at the `Display` output. When a `Vehicle` is a non player it can query surrounding `Vehicles` in its `Lane` and its surrounding `Lanes` or the `Lane` it may or may not turn onto. Additionally any `Vehicle` can be followed by the `View` allowing you to focus on the `Vehicle` being controlled by the player

3 MOVING

The `Input` handles all user interactions that occur to `Display`. `PlayerControllers` can be attached to `Vehicles` allowing the user to take control over the actions any `Vehicles` performs. Players will be able to select turns, make lane changes and accelerate / decelerate with keyboard input.

4 ASSERTING

When turning a list of all possible turns are provided, There are some turns that are 'disabled' because of intersection traffic control or the turn is full because of traffic but invalid turns in the sense of ones that don't exist on the graph aren't possible to return simply because they are never provided in the first place. If a turn is chosen that is disabled one possible solution is to apply a gambling event that calculates if the `Vehicle` blows a red without crashing, etc. A similar event could happen for an invalid lane change.

5 GAMBLING

Different Lane change decisions can be made that differ in how risky they are. The options are `Nothing`, `Wait`, `Nudge`, `Force` (with the latter three having left and right variants). `Nothing` means to make no lane change, `Wait` will wait until it is clear to make a lane change, `Nudge` will prevent oncoming traffic in the desired lane from passing and wait till it is clear to change lanes, `Force` will simply change lanes no matter what potentially causing damage. `Nudge` can potentially decrease a vehicles reputation due to the fact its annoying.

There are other ways that Vehicles can gamble, as previously mentioned some solutions to invalid choices can be punished with gambling.

6 MOVEMENT DECISIONS

6.1 Intersection Decision

When a `PlayerController` controlled `Vehicles` arrive at a `Intersection` the `Simulation` is paused and a menu of available turns the player can take will be displayed.

6.2 Lane Changing

Lane changes can be made at any point, keys for left/right lane changes with modifiers for doing `Wait`, `Nudge`, and `Force` lane changes.

6.3 Challenge

6.4 Reputation Value

6.5 Damage

Challenges will happen automatically when any `Vehicle` makes a decision that is deemed to be 'unsafe' like a `Forced` lane change. `Damage` and `reputation` will be updated accordingly depending on the outcome of the challenge that occurred.