Summer School on Autonomous Vehicles 2017: Simulation Environments

Nikita Lyamin, Maytheewat Aramrattana

August 9, 2017

1 Lab Content

Below we split the lab contents on blocks.

1.1 Block 1 – Scenarios

In this block you will study various realistic platooning scenarios, i.e.:

- offramp scenario, when platoon creates a gap to allow vehicle on the left lane to take off-ramp;
- platooning under periodic disturbance scenario/emergency breaking;
- platoon merging scenario from GCDC 2016 competition;
- heterogeneous platoon, when platoon consists of different vehicles (passenger, trucks, trailers, etc.)

1.1.1 Platooning

In this scenario, you will observe a platoon on the rightmost lane on a straight highway. The leading vehicle varies the speed following sinusoidal pattern.

Exercise

1. The PlatoonsTrafficManager class has three configuration parameters which are `nCars`, `platoonSize`, and `nLanes`. The parameters represent the total number of vehicles to inject, the number of vehicles per platoon, and the number of lanes to use, respectively. Keep in mind that `nCars` must be a multiple of `platoonSize` $*$ `nLanes`. Change number of vehicles in the platoon to 4, 6, 8 in the platoon by changing these parameters. Then run the simulation.
2. Change the inter-vehicle distance by changing `CACCGap` to 6 and 20 meters. Check the results to verify that the gap is correct.
3. Change the **configuration** to `Braking`, reduce the `simulationDuration` to 20 seconds, and run the simulation with 8 vehicles in the platoon.
4. Change the `brakingDeceleration` parameters to 4 m/s^2. Check the results.

1.1.2 Off-ramp

In this scenario, you will observe a platoon on the rightmost lane passing by an offramp on a highway. On the left lane, a passenger vehicle is willing to take an exit from the highway. In order to let the passenger vehicle take the exit, the platoon has to increase inter-vehicle distance to make space. This space is defined by `GapSize` parameter in `omnetpp.ini` file in the > `examples`> `off_ramp` folder.

Exercise

1. Change the `GapSize` to 10 and 30 meters and run the simulation for each parameter. Examine the result files.
2. Change the size and type of platooning vehicles by:
   - In `omnetpp.ini` file, change `platoonVLength` parameter to 10 meters
   - In > `examples`> `off_ramp`> `sumocfg`> `off_ramp.rou.xml`, change `length` parameter of the `vtypeauto` vehicles to 10, and `vClass` to `trailer`
3. Change the `GapSize` to 15 meters.
4. Examine the results.

1.1.3 Heterogeneous platoon

This scenario allows us to study a platoon consists of different type of vehicles. For instance, the vehicles may have different size, length, type, width, and other attributes such as acceleration, deceleration capabilities. In order to run this scenario, it is required to switch the workspace. To switch the workspace, go to > `File`> `Switch Workspace`, then choose other workspace (`/home/summer/src/Heterogenous_platooning`).

Exercise

1. In the `platooning` example, take a look at > `sumocfg`> `freeway.rou.xml` file. Note down the difference.
2. Change the configuration to `Braking` and run the simulation
3. Now change `decel` in the > `sumocfg`> `freeway.rou.xml` file:
   - `trailer` vehicle class to 6
   - `bus` vehicle class to 4

   Then run the `Braking` configuration again. In the result, observe the difference in `distance` and `acceleration` parameters.
4. Please, remember to switch the workspace back to `/home/summer/src` before you proceed.

1.1.4 Merging scenario from GCDC 2016

In this scenario, you will observe a platoon merging scenario. Two platoons are running on a highway (Fig. 1, one of the lane (left lane) will be closed due to the roadwork ahead. The two platoons communicate with each other and merge into one platoon to pass through the roadwork area. The scenario is summarized in Fig. 2. The simulation will run for 200 seconds.

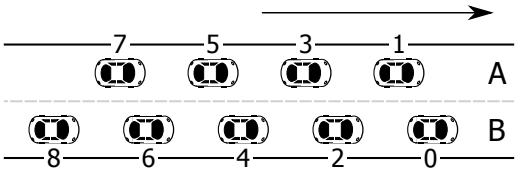

Figure 1: Arrangement of vehicles with their identification in the platooning phase.

In the pair-up and gap-making zone, platooning vehicles communicate and make a group with each other, and make gap with respect to their "forward pair" (gap_{FW} according to the Fig. 3). The desired gap is set to 20 meters by default.

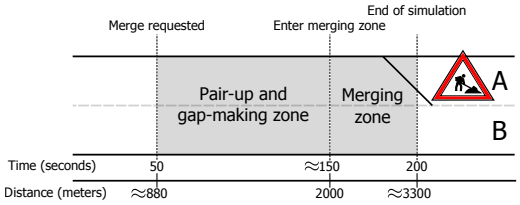

Figure 2: Overview of the simulation.

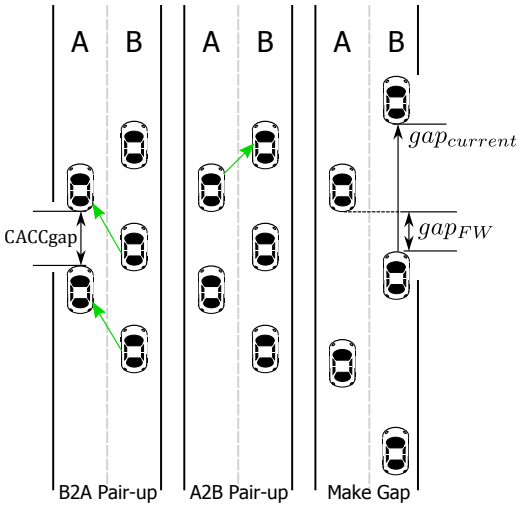

Figure 3:

Exercise

1. Run the simulation and observe the following parameters in the result:
    - `gap_to_fwd`
    - `distance`