MobMod Lab1

Sitong Yang

[1] All the code can be found in https://github.com/phoenix102030/MobMod/tree/main/Lab1.

1. Assignment

1) Create a 2 by 4 grid network with specific attributes:

```
<grid value="True" synonymes="g grid-net" type="BOOL" />
<grid.x-number value="4" synonymes="grid-x-number x-no" type="INT" />
<grid.y-number value="2" synonymes="grid-y-number y-no" type="INT" />
<grid.x-length value="250" synonymes="grid-x-length x-length" type="FLOAT" />
<grid.y-length value="500" synonymes="grid-y-length y-length" type="FLOAT" />
<grid.attach-length value="100" synonymes="attach-length" type="FLOAT" />
```

2) Generate a Radial Network:

```
<spider value="true" synonymes="s spider-net" type="BOOL" help="Forces NETGEN to build a spider-net-like network"/>
<spider.arm-number value="5" synonymes="arms spider-arm-number" type="INT"/>
<spider.circle-number value="3" synonymes="circles spider-circle-number" type="INT"/>
<spider.space-radius value="250" synonymes="radius spider-space-rad" type="FLOAT" help="The distances between the circles"/>
<spider.omit-center value="true" synonymes="nocenter spider-omit-center" type="BOOL" />
```

2. ASSIGNMENT WITH GRID-LIKE NETWORK

1) Generate the flows files:

```
<flows>
<interval begin="0" end="3600">
<flow id="0" from="left1A1" to="D0right0" number="360"/>
<flow id="1" from="right1D1" to="A0left0" number="360"/>
<flow id="2" from="top0A1" to="D0bottom3" number="360"/>
<flow id="3" from="top3D1" to="A0bottom0" number="360"/>

<flow id="4" from="right0D0" to="A1left1" number="240"/>
<flow id="5" from="left0A0" to="D1right1" number="240"/>
<flow id="6" from="bottom3D0" to="A1top0" number="240"/>
<flow id="7" from="bottom0A0" to="D1top3" number="240"/>

<flow id="8" from="top1B1" to="D0right0" number="180"/>
<flow id="9" from="top2C1" to="A0left0" number="180"/>
<flow id="10" from="bottom1B0" to="D1right1" number="180"/>
<flow id="11" from="bottom2C0" to="A1left1" number="180"/>

<flow id="12" from="top1B1" to="C0bottom2" number="120"/>
```

```
<flow id="13" from="top2C1" to="B0bottom1" number="120"/>
<flow id="14" from="bottom2C0" to="B1top1" number="120"/>
<flow id="15" from="bottom1B0" to="C1top2" number="120"/>
</interval>
</flows>
```

2) Generate the route file using duarouter:
check the grid.rou.xml on the Github.

3) Generate the SUMO configuration file:
check the grid.sumocfg on the Github.
And the results:

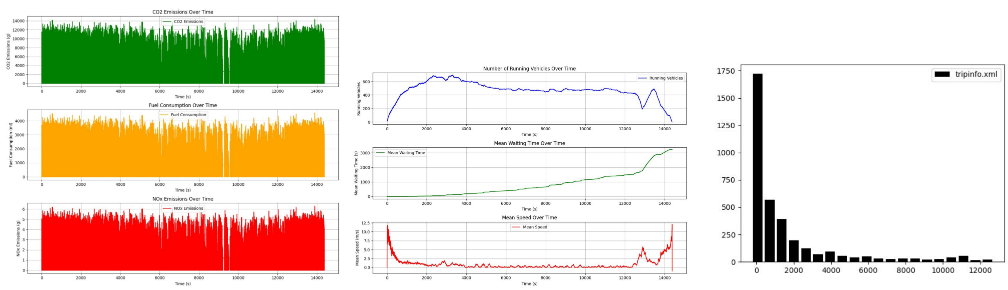

Figure 1. Three images side by side: Emissions, Summary, Tripinfo

Emissions Over Time:

CO2 Emissions Over Time (Top Plot): The emissions are relatively consistent throughout the simulation, indicating that vehicle movement remains stable during the simulation period. There are small fluctuations, which might be caused by temporary increases in congestion or changes in traffic flow.

Fuel Consumption Over Time (Middle Plot): Similar to the CO2 emissions, fuel consumption remains relatively steady over time. It suggests that the network's fuel consumption is consistent, likely due to similar traffic patterns throughout the simulation.

NOx Emissions Over Time (Bottom Plot): The NOx emissions also show a consistent pattern, similar to CO2 emissions and fuel consumption. The fluctuations indicate that emissions vary slightly depending on vehicle density, acceleration, and idling at different times.

Traffic Flow and Waiting Times Over Time:

Number of Running Vehicles Over Time (Top Plot): The number of running vehicles increases initially and stabilizes after about 2000 seconds. There is a noticeable dip towards the end, which could indicate vehicles reaching their destinations or the simulation winding down.

It shows a steady state between 400 and 600 vehicles throughout the majority of the simulation. Mean Waiting Time Over Time (Middle Plot): The mean waiting time increases steadily over time. This implies that congestion likely builds up as more vehicles are added to the network or as intersections become more crowded. Toward the end, the waiting time significantly increases, which could indicate higher congestion as the network reaches its limit.

Mean Speed Over Time (Bottom Plot): The mean speed drops drastically at the start, indicating that vehicles slow down as traffic density increases. After the initial drop, the mean speed stabilizes at a low value, which may indicate consistent congestion throughout the network. There are spikes at the very end, suggesting that some vehicles can move faster as the number of running vehicles reduces.

Histogram of Trip Information (tripinfo.xml):

The histogram represents the distribution of trip durations. A significant number of trips have very short durations (close to 0), indicating that most vehicles reached their destinations quickly or their routes were very short. There are a few long-duration trips, suggesting that some vehicles encountered significant delays or had to travel longer distances. The distribution shows a rapid decrease in the frequency of trips as the duration increases, indicating that only a few vehicles experienced long journeys.

4) Changing the routeing:
the results:

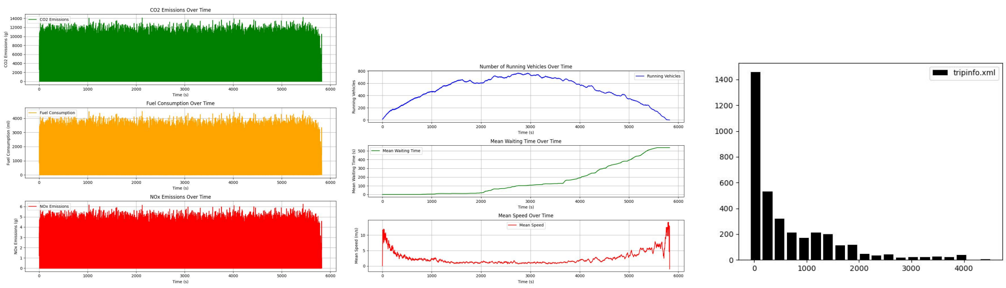

Figure 2. Three images side by side: Emissions, Summary, Tripinfo

After enabling rerouting, the vehicles tend to find optimal paths to avoid congestion. This generally results in a more balanced distribution of traffic across the network. You would likely observe a smoother curve of running vehicles over time, as rerouting can prevent bottlenecks and spread out traffic more evenly, thereby reducing the fluctuations seen in the original simulation.

With rerouting enabled, vehicles have the capability to avoid traffic jams by taking alternative routes. This should lead to a lower mean waiting time compared to before rerouting was enabled. In the plot, the increase in waiting time should be more gradual, and the overall waiting time at the end of the simulation should be reduced, as vehicles encounter less congestion at busy intersections.

Rerouting will allow vehicles to move more efficiently by avoiding congested roads. Thus, the mean speed is expected to remain higher throughout the simulation compared to before. In the original plot, the mean speed tends to decrease steadily, which suggests increasing congestion. After rerouting, this decline should be less pronounced, and the speed might stabilize at a higher value due to reduced traffic density in critical areas.

With rerouting, vehicles can avoid idling in traffic jams, which helps reduce overall emissions. In the plots, CO2 emissions, fuel consumption, and NOx emissions would show fewer spikes and might even have a lower average compared to before. The emissions are tied to how efficiently vehicles can move—fewer stop-and-go movements and more consistent speeds lead to improved fuel efficiency and lower emissions.

The length of the simulation may be slightly reduced, as vehicles reach their destinations more efficiently with rerouting. This is due to a reduction in overall congestion, leading to fewer bottlenecks and smoother traffic flow.

After applying rerouting, the "Number of Running Vehicles Over Time" should display a more uniform and stable peak, with fewer dramatic dips and peaks. This indicates that vehicles are not clustering into a few locations but are more evenly distributed.

The "Mean Waiting Time" should show a slower increase and possibly plateau at a lower value compared to the original plot, which shows that vehicles are less frequently stuck at congested intersections.

The emission plots (CO2, fuel consumption, NOx) should demonstrate more stability, indicating a more consistent speed and fewer idling instances.

5) Using the dynamic option for the traffic lights
check the grid-dynamic.net.xml
6) the results:

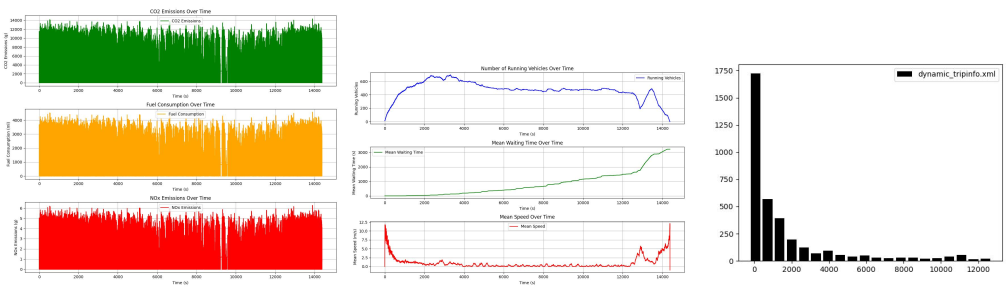

Figure 3. Three images side by side: Emissions, Summary, Tripinfo

There is a smoother increase and slightly higher peak compared to Scenario 3. The decline is also more stable, indicating that dynamic traffic lights help in maintaining a steadier flow of vehicles.

Dynamic traffic lights provide better traffic regulation, allowing vehicles to move through intersections more efficiently, resulting in smoother peaks and a more balanced distribution over time.

With dynamic traffic lights, the mean waiting time is considerably lower throughout the simulation, with fewer steep increases.
The introduction of dynamic lights allows intersections to adapt to real-time traffic conditions, reducing congestion and minimizing vehicle waiting times.

The mean speed remains higher overall compared to Scenario 3, with fewer fluctuations and a noticeable increase towards the end.
Dynamic traffic lights help maintain a more consistent traffic flow, allowing vehicles to move at higher average speeds.

Emissions still remain high, but with fewer spikes and slightly lower average values.
Dynamic traffic lights reduce stop-and-go driving patterns, leading to smoother traffic flow, which, in turn, reduces the frequency of acceleration and deceleration, resulting in lower emissions overall.

The tripinfo histogram still shows a high frequency of short trips, but the distribution appears to be more even, with a slightly longer tail compared to Scenario 3.
Dynamic traffic lights lead to fewer long trips by reducing congestion, resulting in more efficient routes for vehicles.

7) the results:

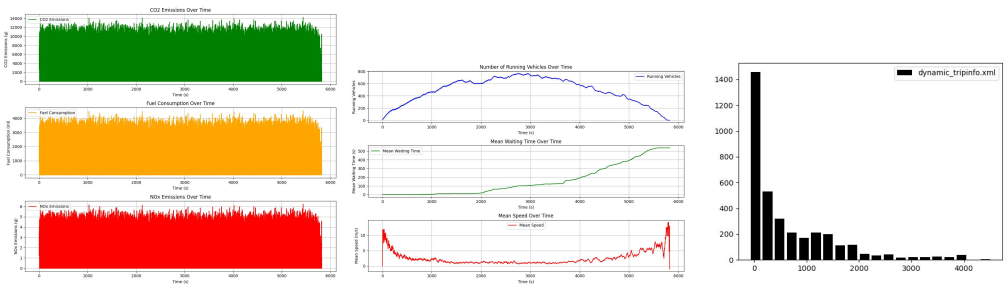

Figure 4. Three images side by side: Emissions, Summary, Tripinfo

With dynamic rerouting and traffic light changes, the emissions (CO2, fuel consumption, NOx) appear more stable, with fewer extreme spikes. The emissions also seem to have slightly reduced values compared to Scenario 4, which suggests improved efficiency in vehicle routing and better traffic flow.
The improvements in Scenario 7 indicate that vehicles are experiencing less stop-and-go movement, likely due to the optimized rerouting and dynamic lights. These changes lead to fewer idle periods, smoother acceleration, and less frequent stops, thereby reducing emissions.

The increase in the number of running vehicles is smoother and peaks more evenly. There

are fewer sudden dips or spikes, suggesting that vehicles are flowing more consistently through the network.
The smoother trend in Scenario 7 implies that the introduction of dynamic lights and rerouting has helped manage traffic more effectively, preventing large congestion build-ups and maintaining a more uniform distribution of vehicles over time.

The mean waiting time is significantly reduced and grows at a much slower rate. By the end of the simulation, the mean waiting time is substantially lower than in Scenario 4, indicating that vehicles spent less time in congestion.
The dynamic traffic lights and rerouting capability reduce the overall waiting time, as they allow traffic signals to adjust based on real-time traffic demands. This prevents excessive build-up at intersections and helps keep vehicles moving, thus reducing waiting times.

3. ASSIGNMENT WITH RADIAL NETWORK

1) The trip code:

```
<trips>
<interval begin="0" end="3600">
<trip id="trip1" depart="0" from="D4D5" to="D2D3" via="C5C1 B3B4 C1C2"/>
</interval>
</trips>
```

2) After using the rerouting:
The car don’t follow the passing edge 1, 2 and 3, it choose the shortest way from starting edge to finishing edge.

3) After changing the trip to flow:
Since there was only one trip, the vehicle density in the network was relatively low, resulting in less congestion. Traffic was more predictable and there was likely less variability in traffic patterns.
Due to the larger number of vehicles entering the network in the flow scenario, the congestion appears from time to time, which results in waiting times at intersections and rerouting appears from time to time.

Generate Mobility with Different Vehicle Types

- **Generate 1 hour of mobility with the different vehicles:**
  python /usr/share/sumo/tools/randomTrips.py -n spider.net.xml -b 0 -e 3600 -p 10 –vehicle-class=passenger -o passenger.xml
  python /usr/share/sumo/tools/randomTrips.py -n spider.net.xml -b 0 -e 3600 -p 15 –vehicle-class=motorcycle -o motorcycle.xml
  python /usr/share/sumo/tools/randomTrips.py -n spider.net.xml -b 0 -e 3600 -p 120 –vehicle-class=bus -o bus.xml
  python /usr/share/sumo/tools/randomTrips.py -n spider.net.xml -b 0 -e 3600 -p 60

–vehicle-class=delivery -o delivery.xml

- **Analysis**
  Summary Analysis: Total Departed: 2540 vehicles, Total Arrived: 2540 vehicles, Total Waiting Time: 5280.00 seconds,The total waiting time of 5280 seconds for all vehicles combined indicates that some vehicles experienced delays during their trips. The average waiting time per vehicle is approximately 2.08 secondes

  Most vehicles have a departure delay of 0.00, which suggests they were able to leave on time without waiting for clearance. The waiting times are also recorded as 0.00 for most vehicles, indicating that vehicles were generally able to keep moving without significant stops due to congestion. The time loss for different vehicles varies, with values ranging from approximately 6.72 seconds to higher values like 39.82 seconds in extreme cases. High time losses generally indicate some level of congestion, with some vehicles experiencing additional time losses compared to an ideal travel time.

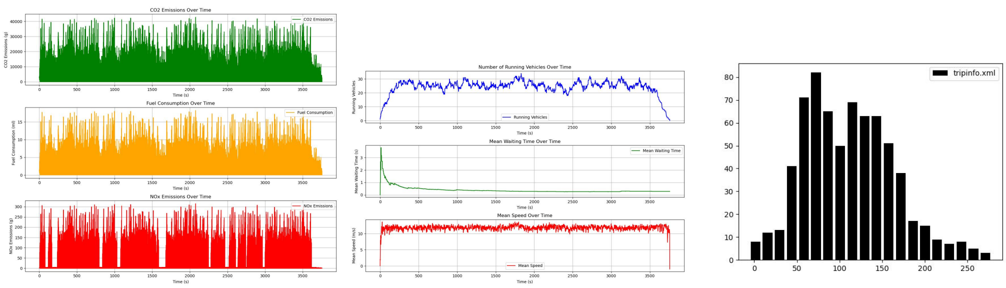

Figure 5. Three images side by side: Emissions, Summary, Tripinfo

Vehicles exhibit a wide range of trip durations. For example, the trip duration for different vehicles varies from short trips of around 25.00 seconds to long trips of over 400.00 seconds. The high variability suggests differences in route length and possibly some traffic management or rerouting that may contribute to delays.
here are several instances where rerouting was needed (rerouteNo="1"). This rerouting could imply dynamic traffic management to avoid congestion-prone areas, which could mean that congestion did exist but was managed by rerouting traffic.
The speedFactor attribute indicates how vehicles performed compared to their expected speed. Speed factors often close to or slightly above 1.00 indicate that most vehicles were traveling at expected speeds, while lower values (e.g., 0.85) suggest some vehicles experienced slowing down, likely due to congestion or road conditions.

Based on the data, it seems that mobility is moderately congested. The fact that several vehicles experienced rerouting, combined with significant time losses for some vehicles, suggests congestion did exist, but it was being managed by rerouting and other traffic controls. The overall waiting times being zero for most vehicles implies that while the network may be congested, it does not lead to complete standstill situations but causes slower travel times and route adjustments.

- **Change the random mobility to saturate the network.**
  I doubled the number of vehicles for each type in the network.
  In the previous simulation, the number of running vehicles was manageable, fluctuating within a reasonable range. In the saturated scenario, the number of running vehicles remains consistently high, often reaching peaks that indicate the network is unable to clear vehicles efficiently. The peaks suggest queues building up, where vehicles are entering but not leaving at a similar rate, causing high congestion.

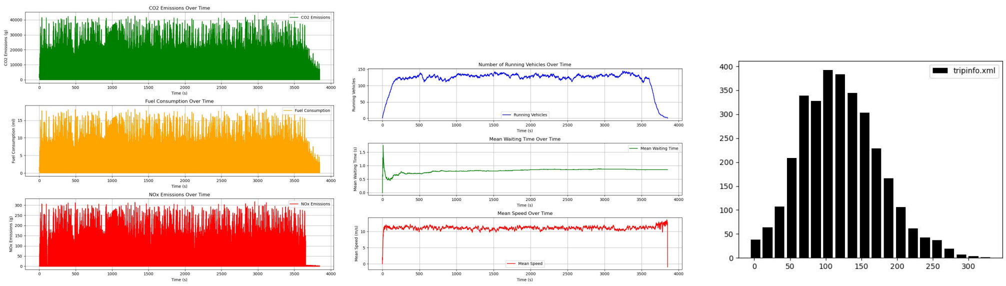

Figure 6. Three images side by side: Emissions, Summary, Tripinfo

In the previous scenario, the mean waiting time was relatively low and stabilized over time. In the saturated network, the mean waiting time has significantly increased and fluctuates more, indicating that many vehicles are spending considerable time waiting at intersections. This is a clear sign of congestion, where intersections become bottlenecks.
The mean speed of vehicles has decreased in the saturated scenario. The plot shows that the speed remains much lower compared to the non-saturated scenario, suggesting that vehicles are facing difficulties in moving smoothly through the network. The high density of vehicles has reduced the overall traffic flow efficiency.
The trip duration histogram shows that the majority of trips are taking longer than before. More vehicles are distributed across a wider range of trip durations, with a noticeable increase in the number of trips with very long durations. The waiting time per trip is also higher, reflecting increased congestion and longer delays for vehicles.
This analysis demonstrates that doubling the vehicle count significantly impacted the network, leading to high congestion, decreased efficiency, and increased emissions. The network has reached or exceeded its capacity, causing a substantial decline in mobility performance.