Team Tetrahedron

Milestone #5 - Simulation Program

Team members:

Lauro Fialho Müller

Chandan Radhakrishna

Raghava Vinaykanth Mushunuri

Kavya Vajja

Arnab Das

Anjan Chatterjee

02.06.2020

Table of contents

1. Overview

This document aims at providing a detailed account of the simulation program currently implemented by Team Tetrahedron, the differences between the program and the conceptual model outlined in Milestone 3, as well as difficulties and limitations of the program.

2. Program concept and structure

4.1 Road Network

We have started building our model with Roads. We used forward and backward lanes and built a road network similar to our node along with the intersection.

4.2 Populating Agents

Cars, Trams (long trucks in this case), Pedestrians are the different types of Agents used in the model. These Agents were made to interact with each other on the road network. We use sources, sink, descriptors, blocks, areas. Measurements from Data Analysis were used in these blocks to make the model scale to real world scenarios.

4.3 Controlling the behaviour of Agents

Stop lines on the road control the movement of cars and trams on the roads. Stop lines were added to necessary points on the road. Traffic lights were modeled using these stop lines.

4.3 Using individual roads rather than a single road with multiple lanes

On running the model we found that to anylogic software controls lane change. So we used individual roads and populated the agents according to their probability distribution on each road.

4.4 Model Pedestrian Crossing

Main task is to make cars and pedestrians interact with each other and make sure pedestrians are not run over by cars.

4.5 Make decisions about the directions

Use some special elements to assign directions to each car according to their probabilities. Also model the cars / trams to move into particular lanes according to assigned directions.

4.5 Run the Model and verify

Once the model is built we need to verify whether it is compatible and working as expected. Make sure it runs till the end time without exceptions and errors.

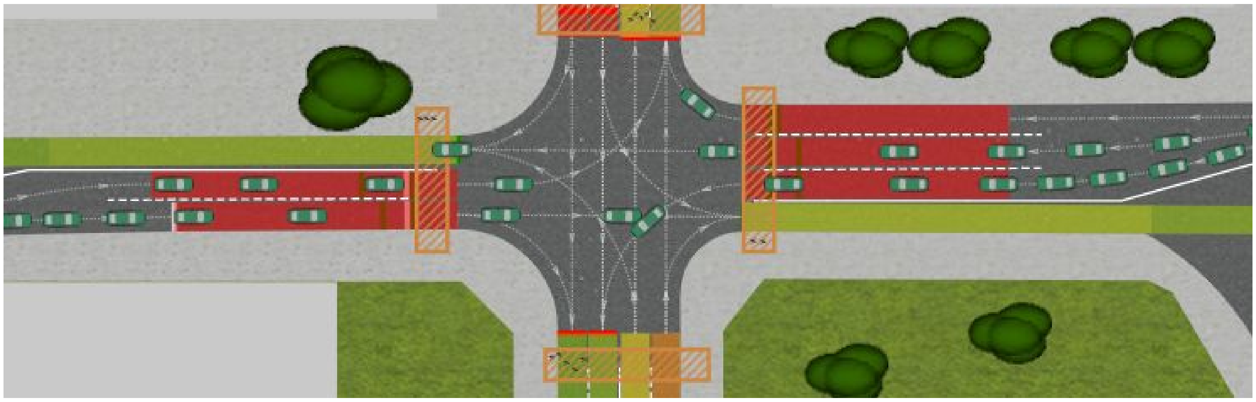
3. Modularization or other special issues

Abstract car entity which is then extended by Tram and Car so that all have required properties.

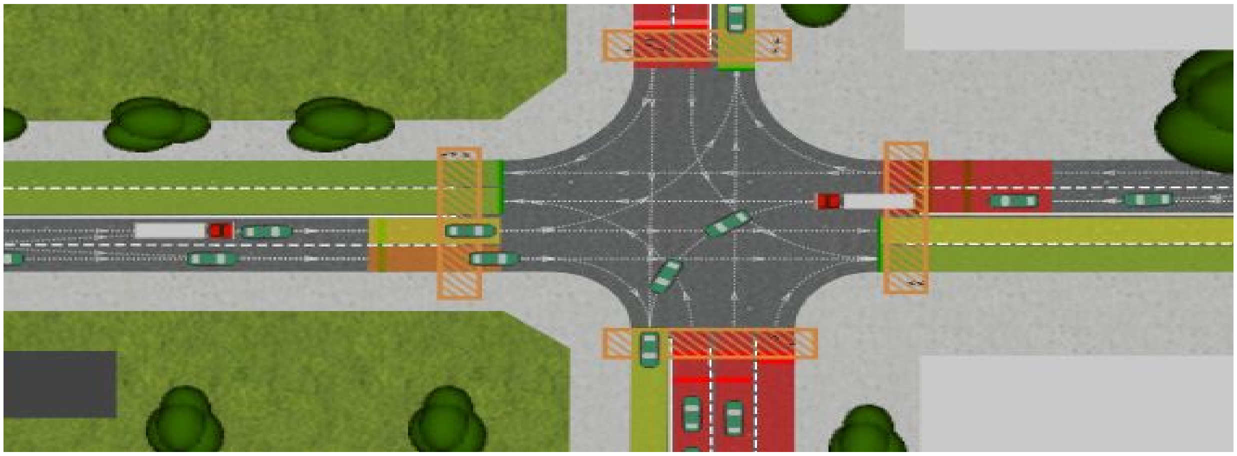
Since the node is complicated we worked on different sections of the road and integrated the work.

3.1 Roads and Lanes

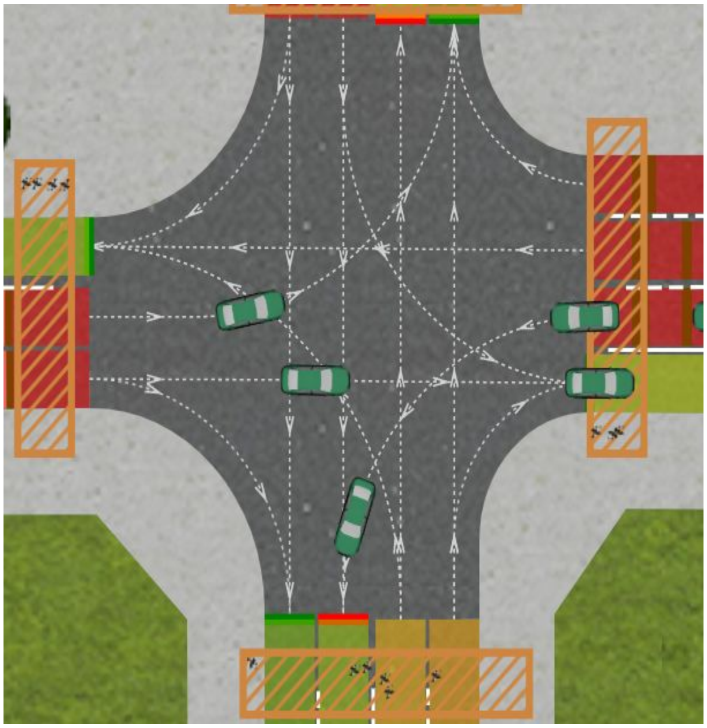
Am Fuchsberg - Erich-Weinert Straße(S)

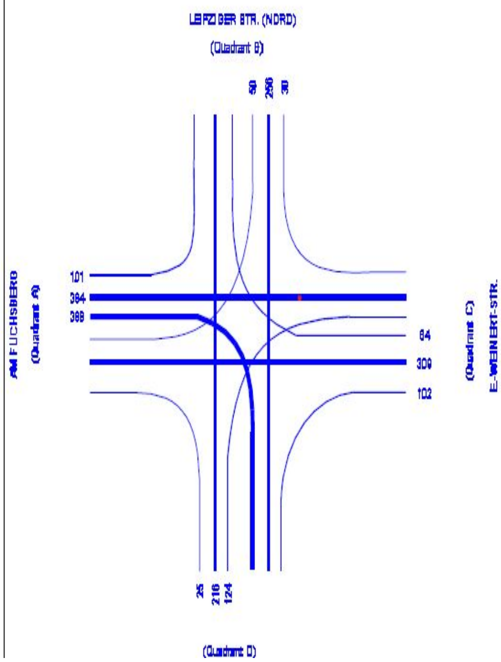


Leipziger Straße(N)-Leipziger Straße(S)



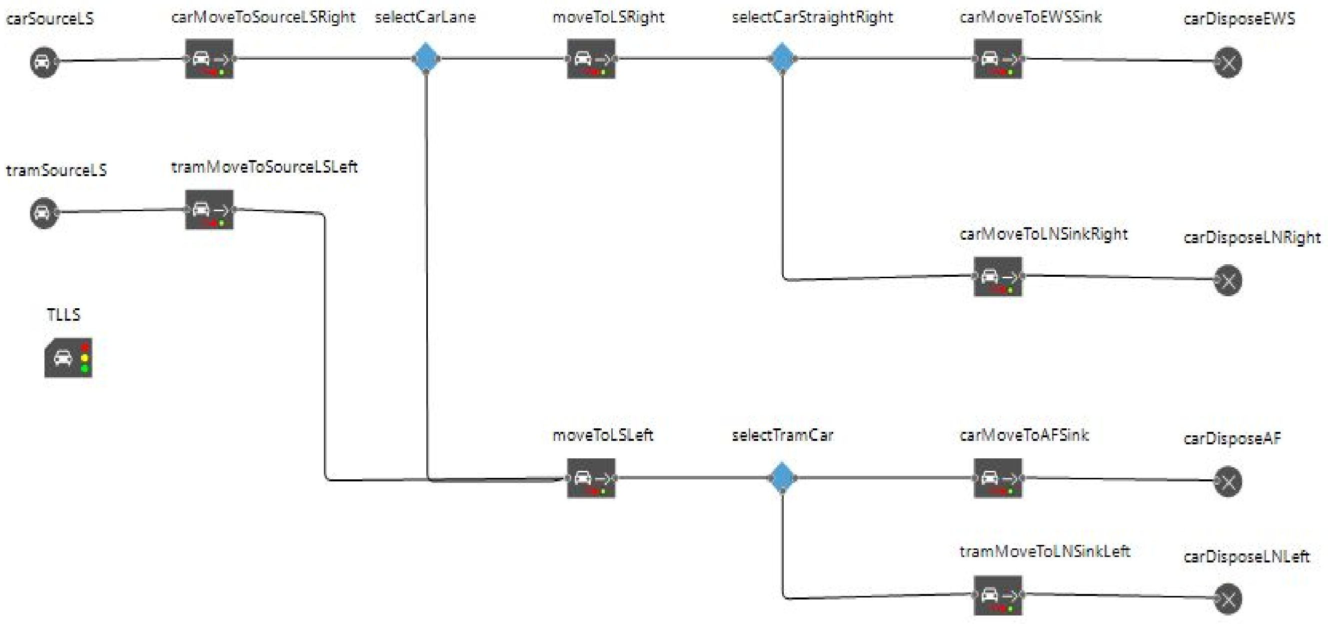
Node Intersection



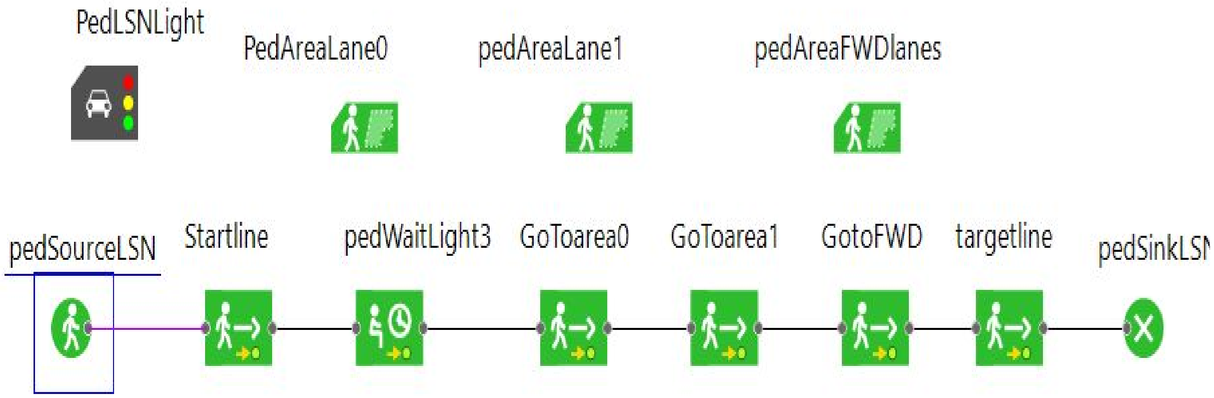


3.2 Modelling Agents on the Road

Cars/Trams Leipziger Straße(N)

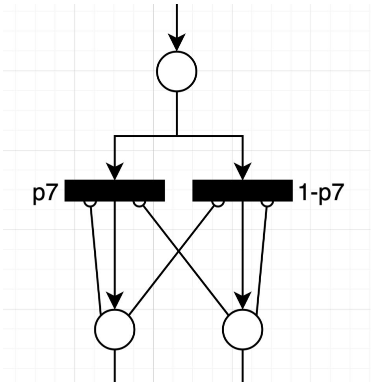


Pedestrian Crossing - Leipziger Straße(N)



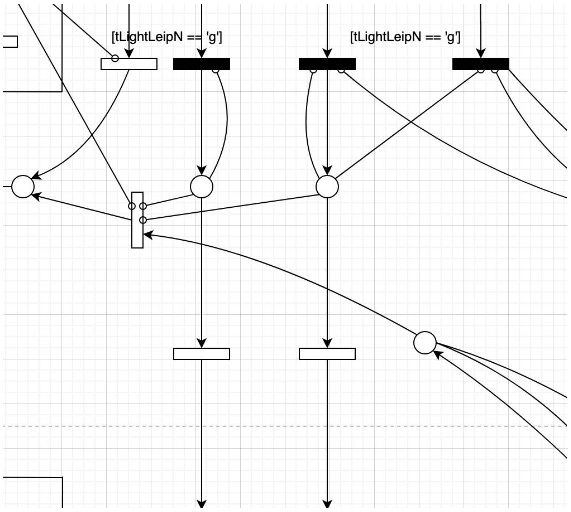
4. Differences to the conceptual model

4.1 Cars sharing the same lane



In the conceptual model, we defined several of the following constructs to represent a single traffic lane where cars could go in either of two directions. Due to the facilities of the road traffic library in any logic, such representation is not necessary: the underlying program requires only a probability of going in each direction and the correct setup of the streets, and it takes care of organizing the cars in a single lane.

4.2 Intermediary stop places for intersections



Another difference with respect to the conceptual model is the absence of previously modelled intermediary stop places for cars waiting to turn left on the intersection. This is again a result of AnyLogic's collision detection facility, which allows us to abstract the complex logic of place handling and transition modelling also when considering more sophisticated intersections such as the one present in the model.

4.3 Timed transitions to represent car movement

A third difference to the conceptual model is the absence of timed transitions to simulate car movement. This is done by modelling acceleration, deceleration and maximum speed in AnyLogic, which allows us to obtain reliable estimates of car movement.

4.4 Traffic lights

Once again, it is possible to model fairly complex cycles of traffic phases using AnyLogic's traffic library, which has allowed us to model a more complex system than the single-phase model used in the conceptual model.

5. Steps taken to verify the program

- Running until the end of run time.
- Checked for different inter-arrival time distributions and checked how the agents are behaving for different inputs.
- Visually verified collisions of cars with pedestrians and the behaviour of their interactions are as expected.
- Checked for the differences between conceptual model and the simulation model and it is observed that most of the design requirements are met.
- Verified the syntax of the code with the best of our knowledge for any possible errors.

6. Any particular difficulties encountered

Despite the many advantages of using AnyLogic's traffic library, implementing the model was not without difficulties. Among the many challenges faced, we would like to highlight the following three.

6.1 Collision detection

Collision detection has been and still is a present challenge in our model. While the road traffic library provides collision detection out of the box, the lack of customization possibilities and access to the underlying algorithm prevents us from fine-tuning it to perfectly represent the traffic rules of our intersection. Additionally, we have no access to the underlying numbers or flags from the algorithm (for example, whether a car is in danger of collision), making the calculation of statistics a somewhat more cumbersome work. We are looking at integrating it with different agent-based techniques as to obtain better insight into the numbers and the model behavior.

6.2 Pedestrian modelling

Pedestrian modelling has also been a challenge, and has required a considerable amount of time as to prevent collisions between the cars and pedestrians crossing the streets.

6.3 Model validation

Due to the lack of output data, as well as the atypical traffic situation due to the Corona pandemic, model validation is not fully possible. We are mainly considering face validation and ensuring that the model works as expected: cars on a given lane are able to stop on time, initial speed is within reasonable limits, cars do not drive over pedestrians, cars respect the lane restrictions regarding directions, among other conditions.

7. Cost overview

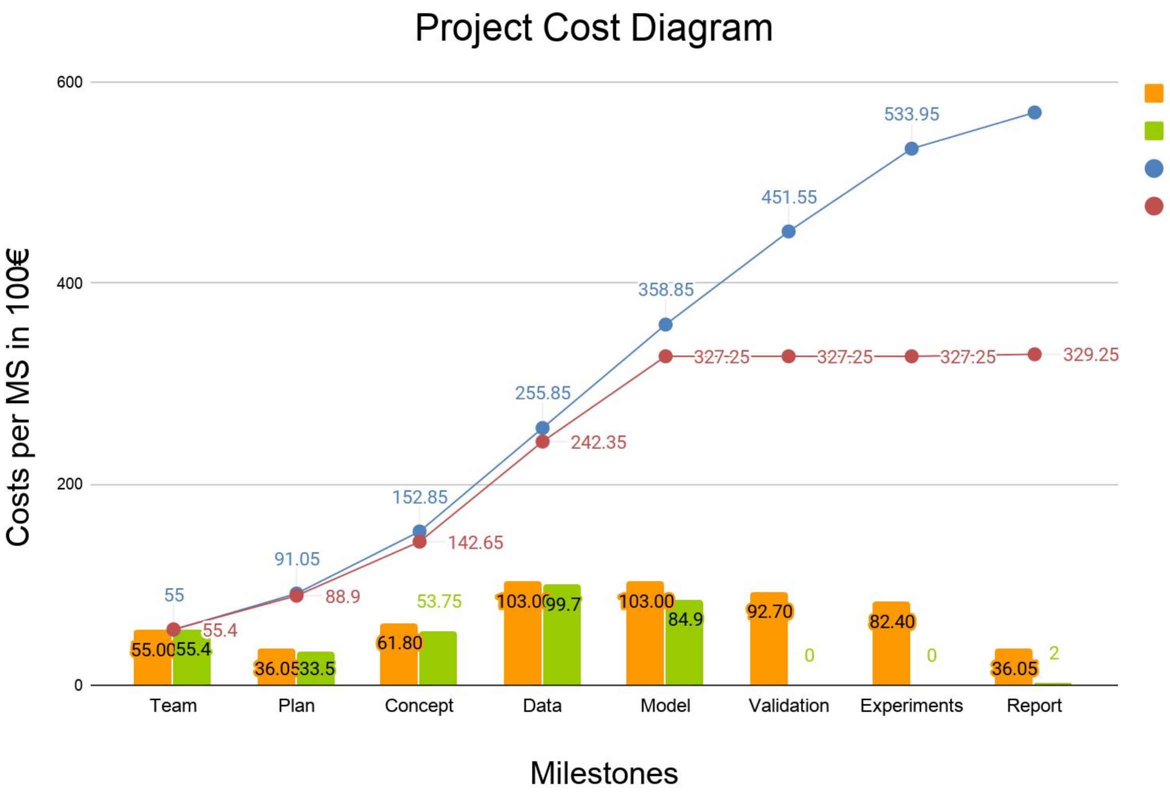
The chart in the next page presents a two-dimensional breakdown of costs, aggregated by both milestones and individual members. As planned during Milestone 2, we have already started working on the final documentation to keep it updated with the developments of each individual milestone.

Name

Team Tetrahedron

	Lauro	Chandan	Vinay	Kavya	Arnab	Anjan	Total
Milestone 1 (hrs)	12.60	8.10	8.40	8.70	9.05	8.55	**55.40**
Milestone 2 (hrs)	5.75	3.50	4.00	5.05	4.25	10.95	**33.50**
Milestone 3 (hrs)	12.25	17.00	4.00	7.30	8.50	4.70	**53.75**
Milestone 4 (hrs)	19.50	9.30	41.00	8.30	12.50	9.10	**99.70**
Milestone 5 (hrs)	17.25	9.00	11.00	26.60	12.00	9.05	**84.90**
Milestone 6 (hrs)							**0.00**
Milestone 7 (hrs)							**0.00**
Milestone 8 (hrs)	2.00						**2.00**
Total hrs	69.35	46.90	68.40	55.95	46.30	42.35	**329.25**
Billing rate (hourly)							**€100.00**
							€32,925.00

Additionally, the chart below shows the cumulative cost of the project so far. The orange bars represent the planned milestone costs, and the blue line the planned cumulative cost. The green bars represent the actual milestone costs, and the red line the actual cumulative cost.



8. Future work

Our focus from now on is to (1) fine-tune the model during the validation phase as to ensure that all its parts work as expected, and (2) implement changes to the model and run experiments in order to check whether it is possible to improve the node towards a safer and more efficient intersection.