Intelligent Traffic Control

Thomas van den Broek, Vincent Van Driel, Zsolt Harsányi, Tonio Weidler

Department of Data Science & Knowledge Engineering

Maastricht, The Netherlands

`vincevandriel@gmail.com, thomasbuo@gmail.com,`

`zsharsany@gmail.com, uni@tonioweidler.de`

Abstract

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Keywords: traffic simulation; traffic control; strategies; linear programming; machine learning; realistic environment; IDM; MOBIL;

1 Introduction

An intelligent traffic control system adjusts traffic in order to assure all people reach their destinations in the most optimal time and distance. These systems are important for the daily workings of major cities by alleviating traffic congestion and identifying problem areas. It is important to constantly keep these systems updated to assure optimal performance and safety of the general public. With advancements in technology and artificial intelligence, new and more sophisticated strategies for traffic control become possible.

1.1 Goal

In this work, we aim to evaluate different traffic control strategies and compare their effect. We hope to identify the best working solutions to traffic jams and other occurrences (such as closing of roads/tunnels) in cities and to show their workings. We attempt to recreate a city's traffic flow in order to see how these systems would work in a realistic setting. Our simulation aims to model the dynamics of a realistic environment as close as possible within the scope of the project, so that the comparison is meaningful and provides valuable data regarding the potential application of the strategies.

Our main contributions are as follows:

1.2 Approach

Hence, in order to test the effect of different intelligent traffic control strategies, an appropriate simulation is required. For this purpose, we created a simulation environment which allows the incorporation of different such strategies into a dynamic traffic model. Maps are realised as undirected graphs in which vertices represent intersections and roads appear as edges between those. The simulation is *microscopic*. That is, instead of globally controlling traffic (*macroscopic*), the atomic parts of the simulation are locally controlled cars (see also Krajzewicz et al., 2002). Driving behaviour is modelled using the time- and space-continuous *Intelligent Driver Model* (IDM) (Treiber, Hennecke, and Helbing, 2000).

1.3 Related Work

There has been extensive work on the simulation of traffic flow as well as the development of traffic control strategies. Following up on different approaches on modelling car following behaviour (e.g. Gipps, 1981), Treiber, Hennecke, and Helbing (2000) developed the influential Intelligent Driver Model for the simulation of urban traffic. Similarly to the former, it creates a collision free environment where cars mind the a spacial and time-wise gap to the leading vehicle. The SUMO package (Krajzewicz et al., 2002; Behrisch et al., 2011) utilizes the model by (Gipps, 1981) in an extended version (Krauß, 1998) in a complex simulation software. In contrast to their work, we use the IDM. Hence, our simulation is (quasi) time-continuous, rather than time-discrete.

In the following sections we further describe our approach and the results of our experiments. We begin by elaborating on the implementation of the simulation environment (section 2), followed by a description of different control strategies (section 3). Following, we describe the experiments we conducted for those strategies and our evaluation methodology (section 4). In section 5 we will report the results of these experiments. We conclude this work by discussing our results and proposing future directions.
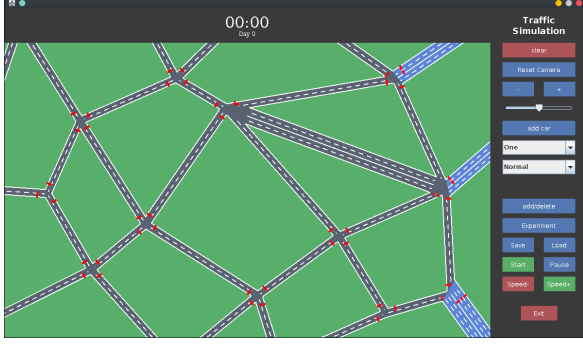
Figure 1: Interface of the interactive simulation environment. Different road colors indicate different road types (e.g. streets or highways).

2 Simulation Environment

In the following subsections we describe the implementation and design choices of the simulation environment in which we evaluated the effect of different traffic control strategies.

2.1 Graphical User Interface

The Graphical User Interface (GUI) is designed for simplicity. It consists of three panels. On the left side the main panel is located. Here, maps can be created and navigated through and the simulation process is visualized. Maps can be interactively constructed by clicking and dragging on the panel. Above the main panel there is an info panel that indicates different values such as the current time and day of the simulation. Finally, the right panel contains the controls for different functionalities such as the creation of maps, setting of experimental parameters and controlling the simulation. Figure 1 shows this interface with a small map.

2.2 Traffic Flow Simulation

As mentioned beforehand, the IDM (Treiber, Hennecke, and Helbing, 2000) is applied for the simulation of car dynamics. It models traffic flow time- and space-continuous as a combination of *free-road* and *interaction* behaviour. The *free-road term* is governed by a car's intention to reach its desired speed. The acceleration for this behaviour is calculated (Treiber, Hennecke, and Helbing, 2000) as

$$\dot{v}_a^{free}(t) = a(1 - (\frac{v_a}{v_0})^\delta), \tag{1}$$

where a refers to a cars maximum acceleration, v_a is its current velocity and v_0 the desired velocity. When a car approaches a leading vehicle, it is supposed to slow down in order to avoid collision. This behaviour is modelled by an *interaction term* which incorporates the distance to the leading vehicle and its speed (Treiber, Hennecke, and Helbing, 2000).

$$\dot{v}_a^{int}(t) = -a(\frac{s_0 + v_a T}{s_a} + \frac{v_a \Delta v_a}{2\sqrt{ab}s_a})^2 \tag{2}$$

In the equation above, s_0 and T restrict the cars minimum distance in space and time respectively. The interaction term will hence attenuate the free road term when approaching other cars, given by the complete equation for acceleration $\dot{v}(t)$

$$\dot{v}(t) = \dot{v}_a^{int}(t) + \dot{v}_a^{int}(t) \tag{3}$$

In order to simulate a time-continuous model, we need to numerically approximate the integration of the differential equations in 1 and 2. For that, we choose a small time step Δt and repetitively update the velocity as $v(t + \Delta t) = v(t) + \dot{v}(t)$.

2.3 Traffic Rule Compliance

Due to the large scale maps used in this work, drivers not only need to behave according to their own desires and other traffic participants. They additionally need to comply to a set of traffic rules as given by speed limits or traffic lights. Following, we briefly discuss the approaches we use in our simulation.

Approaching Traffic Lights When approaching red traffic lights, drivers will stop in a predefined distance from the intersection, comparable to stop bars in real world streets. Cars that have already exceeded this line will drive through the intersection.

Speed Limits The IDM's free road term incorporates a driver's desired velocity as his intended maximum speed. We use an additional parameter, a driver's favoured velocity v_{fav}, and determine the desired velocity by taking the minimum of v_{fav} and the roads speed limit.

2.4 Lane Changing

In order to incorporate lane changing behaviour into our traffic model, we apply the MOBIL model (Treiber and Helbing, 2002; Kesting, Treiber, and Helbing, 2007). MOBIL makes decisions for lane changing based on two questions: Is the lane change *incentive* and is it *safe*? The former question is answered based on the potential gain in acceleration on the new lane, while the latter decision is made based on the speed of and distance of the approaching car on the new lane.

2.5 Arrival Times in a continuous Simulation

Rather than predefining arrival times for a fixed simulation duration, we dynamically generate inter-arrival times (IAT) during runtime. These IATs are drawn from a predefined distribution independently for each road in the map. Therefore the amount of traffic on a map grows automatically with its size. The IAT can be drawn from the following distributions

- Exponential (Poisson Process)

- Gaussian
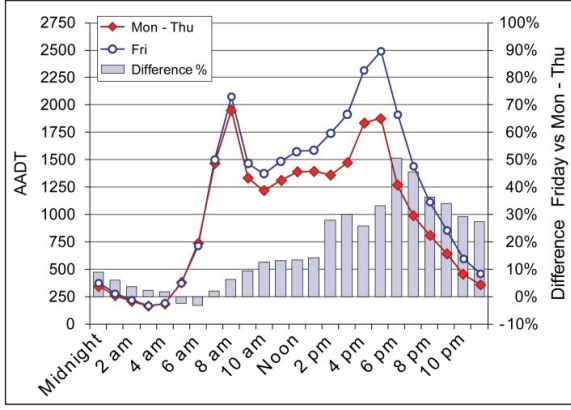
- Realistic (Non stationary Poisson Process)

2

Figure 2: Traffic data from the U.S. Department of Transportation. Graphic received from https://www.fhwa.dot.gov/policyinformation/tmguide/ tmg_2013/traffic-monitoring-methodologies.cfm

While the Gaussian and Exponential distribution are standard distributions with adjustable paramters, the realistic distribution is based on real world data from the U.S. Department of Transportation (*Office of Highway Policy Information - Policy | Federal Highway Administration*). We use hourly data and draw IATs based on a poisson process that is then converted to the data using the thinning algorithm by Lewis and Shedler (1979). The data used is shown in figure 2. As can be observed, this creates rush hours around the hours of 8 to 9 am and 5 to 6 pm.

2.6 Path Finding

Whenever a car enters the traffic, it needs to get a predefined route to take to its destination. In the simulation, each car runs an instance of a path finding algorithm, two of which were implemented.

A* This popular path finding algorithm, based on Dijkstra's algorithm (Dijkstra, 1959), was first introduced in 1968, by Hart, Nilsson, and Raphael (1968). It finds the shortest path between the origin and the destination of the cars, given the map only taking the length of the path into account.

Advanced A* The advanced A* path finding algorithm is very similar to the original version, but it also takes the traffic density and average speed of the roads into account. This is realized by calculating the traffic density and average speed of each road whenever the route of a car is created. The density of a road equals the number of cars on it divided by the length of the road, to get the number of cars per meter of road. The algorithm then adds a weighted density value to the G and H scores of the road to calculate the current best route. The average speed for the roads is taken from the continuously calculated statistics. The value of the average speed is taken away from the score of the road, in order to incentivize the use of roads with a higher speed.
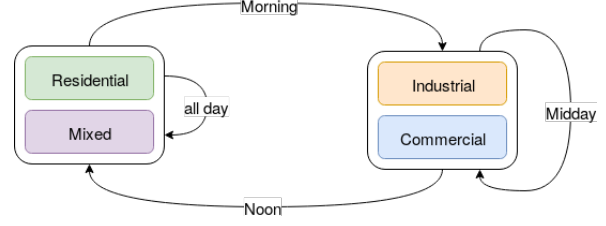

Figure 3: Traffic Flow between different zone types at different times of the day.

2.7 Zone-based Scheduling

Aiming to increase the level of realism, roads on the simulated map have different types of zones that influence the spawn rates and routing of cars. Distinguished zones are *residential*, *mixed*, *commercial* and *industrial*. A road's zone restricts its *available population* in the beginning of the simulation. For instance, commercial zones start with a lower overall population than residential zones. Cars will only spawn from roads as long as there is population available. Whenever a car leaves or arrives at a road, its population is respectively decremented or incremented. Additionally, the zone type determines the traffic flow between different areas of the map. Based on the time of day, cars either mostly move from residential and mixed zones to industrial and commercial zone or vice versa, when they are on their way back home. This way, the simulation also incorporates the difficulties of heavy traffic in one direction, as faced in cities where living and working places are seperated by its different areas. Figure 3 sketches how the traffic behaves between the zones based on the time of day.

2.8 Simulation Model

Putting the above parts together, the simulation procedure depicted by algorithm 1 is run.

3 Traffic Control Strategies

In this section, different traffic control strategies are highlighted as well as their effectiveness in different environments.

3.1 Benchmark Strategies

In order to provide a baseline of what more advanced strategies should be able to outperform, we implemented two simple benchmark strategies. The first one is Basic Cycling, this refers to a traffic light control strategy based on a simple, fixed-time cycle in which one traffic light at a time is green. The second is Informed Cycling refers to a similar strategy, in which a road's green phase is skipped if there are no cars waiting on it. Finally, priority Cycling adjusts the green phases linearly based on the density of the roads going into the intersection.
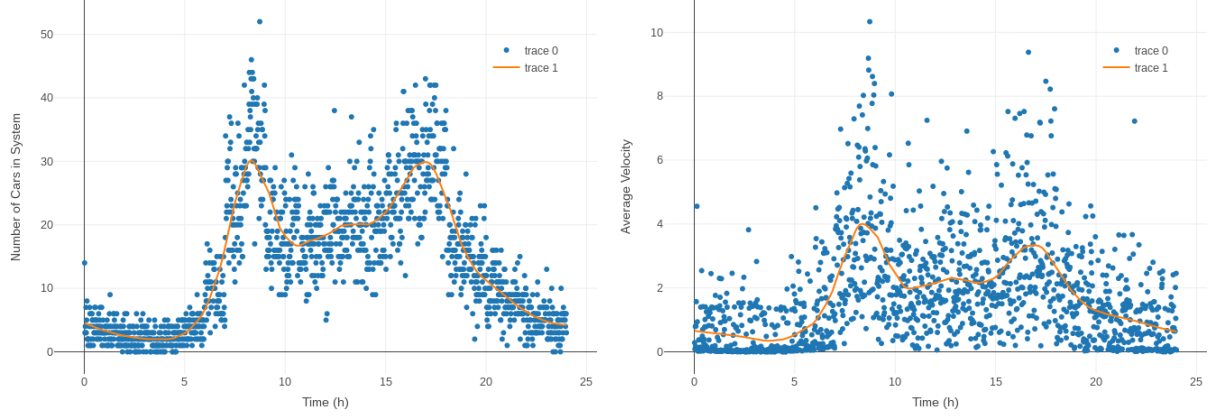
3

Figure 4: Number of cars (left) and average velocity of cars (right) in the system simulated for one day. IATs are generated from the empirical distribution shown in figure 2.

3.2 Responsive Strategies

These strategies should outperform the benchmark strategies as they are more dynamic and change their actions based on their surroundings. The most simple version of such a responsive strategy is priority Cycling, this strategy adjusts the green phases linearly based on the density of the roads going into the intersection. Weighted Cycling This strategy was chosen to be implemented to reduce the time people spend waiting at a light. Simply put, the traffic lights give priority to those who are waiting at the light, and if no one is waiting, it gives priority to the street with the most cars incoming. This method seems to mimic real-world examples of traffic control systems by acting similar to a traffic light with some form of traffic sensors (pressure plates, traffic cameras, etc.). It also allows the most amount of people possible through fastest, which is extremely effective in situations where a large amount of cars approach from one side.

3.3 Advanced Strategies

The advanced strategies are different from responsive strategies in that they either compare many different inputs to make a decision as to which traffic light should be turned green at which time or they use linear programming. Coordinated traffic lights are traffic lights that exchange information with each other in order to predict future traffic patterns and assure the least amount of congestion possible. Each intersection is outfitted with different types of sensors, depending on the environment in which they are needed. These sensors can include inductive loops, cameras, or microwave radar systems, each of which has its own pros and cons. The coordinated traffic light strategy currently implemented uses features resembling that of a microwave motion sensor. The reason of using this type of sensor is mainly because the detection range can be altered to each one depending on the area it is installed, the ability to accurately detect distance of vehicles, and also it is able to tell if cars are moving to-

wards or away from it. Other reasons for implementing this type of detection system is that in real-world situations, this device can be installed above ground, which greatly reduces cost of installation, and can also be installed very quickly at very low charge and cause minimal traffic disturbance. This type of traffic control system is preferred for areas with minimal obstructions (such as road signs or buildings). Since our current traffic simulation does not have any external obstructions other than cars, this is the best strategy to use. In cases where many obstructions are in play, it would be more recommended to use inductive loops (commonly thought of as 'pressure plates' installed into the road, but quite costly). The way these microwave motion sensors are used in the simulation is by recognizing how many cars are currently coming towards an intersection from each road and measures the average speed of those cars per road. The road with the lowest average speed will be set to green. If it is close to being a tie than the decision is made by selecting the road with the most amount of cars waiting. In order to prevent certain roads from never turning green, every road has to turn green at least once in a certain amount of time.

4 Methodology & Experiments

In order to be able to measure and compare the efficiency of the different strategies applied, various measurements are taken while the simulation is running. One of these is the average speed of the vehicles for each road. Measuring this is done by starting a timer for a car when it enters a road and stopping it when it leaves it. Then, by using the following formula, the average speed is calculated:

$$v = \frac{s}{t} \tag{4}$$

where v is the velocity, s the road length and t refers to the time spent on a road. Using this, we can also easily derive the average speed on each road. The second statistic measured is the amount of time each car spends

4

Algorithm 1: Simulation procedure.

Data: Map m, CarList c, Schedule s, days d, delta_t

Initialize trafficlights
Initialize tracking variables

while *simulated_days < d* **do**
    update time_parameters

    foreach *Road r in m* **do**
        if *r.nextArrival == now* **then**
            spawn car
            r.availablePopulation–
            generate nextIAT
        end
    end

    update trafficlights
    update car positions *and* velocities
    remove arrived_cars *and* adjust population

    visualize
    track statistics
end

report statistics

---

waiting at traffic lights. This is achieved by checking at each time step if a car is under a certain speed and is not accelerating. Whenever the previous conditions are fulfilled, the total amount of waiting time is increased by the size of the time step. The purpose of the control strategies is to maximize the average speeds and minimize the time spent waiting in congestions.

4.1 Simulation Validation

In order to validate both the IA generation as well as the general behaviour of different measures and dynamics, we validated the simulation environment using the average speed of cars, their number on the map as well as fractional waiting times in the simulation as a development over one day. Figure 4 shows the results in three graphs. As can be observed, not only the density, but also both other dependent measures resemble the rush hour pattern from the original data.

5 Results

6 Discussion

7 Conclusion

- Often times, better road layout makes way more of a difference than different strategies (tiefighter example)

References

Behrisch, Michael et al. (2011). "SUMO–simulation of urban mobility: an overview". In: *Proceedings of SIMUL 2011, The Third International Conference on Advances in System Simulation*. ThinkMind.

Dijkstra, Edsger W (1959). "A note on two problems in connexion with graphs". In: *Numerische mathematik* 1.1, pp. 269–271.

Gipps, Peter G (1981). "A behavioural car-following model for computer simulation". In: *Transportation Research Part B: Methodological* 15.2, pp. 105–111.

Hart, Peter E, Nils J Nilsson, and Bertram Raphael (1968). "A formal basis for the heuristic determination of minimum cost paths". In: *IEEE transactions on Systems Science and Cybernetics* 4.2, pp. 100–107.

Kesting, Arne, Martin Treiber, and Dirk Helbing (2007). "General lane-changing model MOBIL for car-following models". In: *Transportation Research Record: Journal of the Transportation Research Board* 1999, pp. 86–94.

Krajzewicz, Daniel et al. (2002). "SUMO (Simulation of Urban MObility)-an open-source traffic simulation". In: *Proceedings of the 4th middle East Symposium on Simulation and Modelling (MESM20002)*, pp. 183–187.

Krauß, Stefan (1998). "Microscopic modeling of traffic flow: Investigation of collision free vehicle dynamics". PhD thesis.

Lewis, Peter A and Gerald S Shedler (1979). "Simulation of nonhomogeneous Poisson processes by thinning". In: *Naval Research Logistics (NRL)* 26.3, pp. 403–413.

Office of Highway Policy Information - Policy | Federal Highway Administration. URL: https : / / www . fhwa . dot.gov/policyinformation/tmguide/tmg_2013/traffic-monitoring-methodologies. cfm (visited on 05/28/2018).

Treiber, Martin and Dirk Helbing (2002). "Realistische Mikrosimulation von Strassenverkehr mit einem einfachen Modell". In: *16th Symposium Simulationstechnik ASIM*. Vol. 2002, p. 80.

Treiber, Martin, Ansgar Hennecke, and Dirk Helbing (2000). "Congested traffic states in empirical observations and microscopic simulations". In: *Physical review E* 62.2, p. 1805.