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 problematic 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 Research Questions

This work aims to evaluate various traffic control strategies and compare their effect in different scenarios. In particular, we aim to investigate, whether different traffic signal control strategies are best suited for different street layouts or if there is the latter induce consistent rankings. Furthermore, experiments are conducted in order to compare the effectiveness of strategies controling the behaviour of traffic signals and such that are related to urban planning, e.g. introduction of new lanes or entirely new roads.

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. 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. 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). We make our code publicly available at GitHub[1].

1.3 Related Work

In the following, previous research on this topic will be briefly summarized. 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,

---

[1]https://github.com/weidler/traffic-simulation

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 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 research, this work uses the IDM. Hence, the simulation is time-continuous, rather than time-discrete.

Often utilizing such simulation environments for evaluation purposes, a lot of research exists investigating the strengths of different approaches to the automatic control of traffic signals or traffic lights. Papageorgiou et al. (2003) have reviewed such research and identified two key characteristics according to which strategies may be classified. They distinguish between *fixed-time* and *traffic-responsive* strategies, as well as *isolated* and *coordinated* strategies. Fixed-time strategies use historical data to determine (e.g. by optimization) static cycle lengths, while traffic-responsive strategies may use different techniques (e.g. heuristics or optimization algorithms) in order to derive optimal cycle lengths from real-time data that has to be measured in some way. Isolated strategies are focused on a single intersection, while coordinated strategies are able to communicate information between multiple intersections and thereby make potentially joint decisions. Coll et al. (2013) categorize strategies into four categories. They also distinguish between fixed-time and traffic-responsive approaches, but additionally differentiate *adaptive systems* that apply optimization techniques and *predictive strategies* that use both online and off-line information in order to predict future arrivals at the intersection (Coll et al., 2013). In this work, the focus will be laid on the former two categories, that is, fixed-time and responsive.

There has also been research on more advanced strategies using e.g. different machine learning techniques like neural networks (e.g. Srinivasan, Choy, and Cheu, 2006; Chao, Lee, and Wang, 2008) or linear programming (e.g. Coll et al., 2013; Lin and Wang, 2004) for optimizing the phase or cycle lengths of traffic lights. Coll et al. (2013) use predictions based on the queue size at red and green lights, flow capacity, and cycle length from the previous time step in order to optimize the cycle lengths with the help of linear programming. As the incoming traffic density from each road entering an intersection changes, the length of green and red phases for the corresponding traffic light is gradually adjusted to assure an optimal output flow. The objective function they optimize minimizes the queue lengths and maximizes the flow.

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.

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 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

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

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).

$$\dot{v}_a^{\text{int}}(t) = -a\left(\frac{s_0 + v_a T}{s_a} + \frac{v_a \Delta v_a}{2\sqrt{abs_a}}\right)^2 \quad (2)$$

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 $\dot{v}(t)$

$$\dot{v}(t) = \dot{v}_a^{\text{free}}(t) + \dot{v}_a^{\text{int}}(t) \quad (3)$$

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 $v(t + \Delta t) = v(t) + \dot{v}(t)$.

2.2 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.

2.3 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.
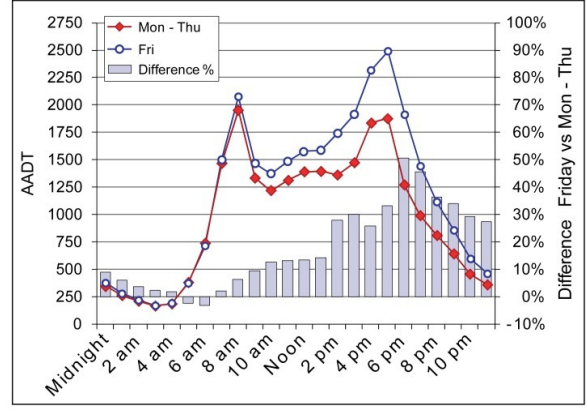

Figure 1: Traffic data from the U.S. Department of Transportation. Graphic received from https://www.fhwa.dot.gov/policyinformation/tmguide/tmg_2013/traffic-monitoring-methodologies.cfm

2.4 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. We use a non-stationary poisson process based on real world data from the U.S. Department of Transportation (*Office of Highway Policy Information - Policy | Federal Highway Administration*). The hourly data is used to draw IATs based on a poisson process that is then adjusted using the thinning algorithm by Lewis and Shedler (1979). The data used is shown in figure 1. As can be observed, this creates rush hours around the hours of 7 to 9 am and 5 to 7 pm.

2.5 Population

As Papageorgiou et al. (2003) state, most traffic signal strategies are only useful, when the roads are undersaturated with respect to traffic. For that reason but also for the sake of computational efficiency, the simulation prevents oversaturation by using a population count for each road. This population is initialized by determining how many cars would fit on the most outer lanes of that specific road. 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.
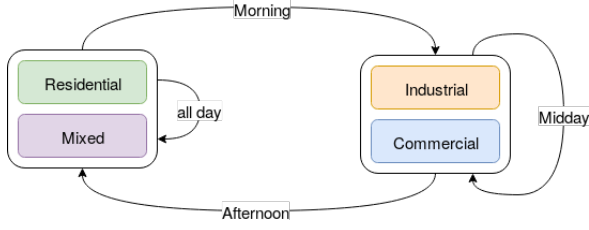
3

Figure 2: Traffic Flow between different zone types at different times of the day.

2.6 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. 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 separated by its different areas. Figure 2 sketches how the traffic behaves between the zones based on the time of day.

2.7 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. This is done by using two scores for each node: H and G score. The H score is equal to the Euclidean distance from the node to the destination. The G score is the total distance travelled from the start node to the current node. Based on these two scores, the algorithm finds the shortest path between two points in a graph.

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.

Early experiments have shown that the advanced A* algorithm provides more safety against oversaturation of roads and therefore creates a more realistic simulation in which drivers are able to adjust their path based on their knowledge about the traffic.

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** and 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** and refers to a similar strategy, in which a road's green phase is skipped if there are no cars waiting on it.

3.2 Responsive Strategies

These methods 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.).

Three strategies are investigated that, similarly to *Basic Cycling*, choose the next green light in a given order. Crucially though, they do not use
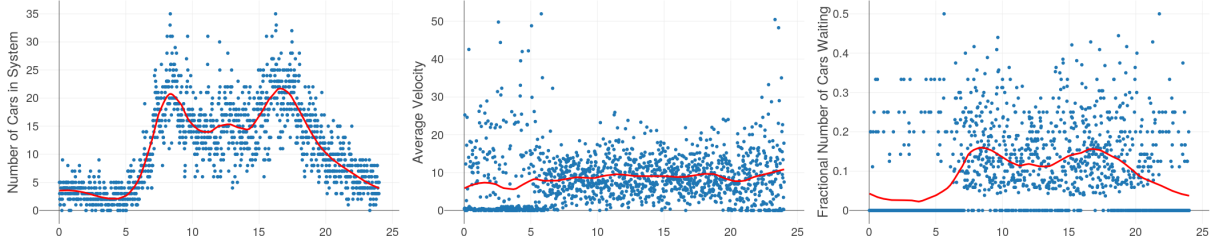
Figure 3: Number of cars (left), average velocity of cars (middle) and percentage of cars waiting (right) in the system simulated for one day. IATs are generated from the empirical distribution shown in figure 1.

fixed times for the green phase. Instead, the phase-length is determined by some strategy-specific measure of traffic, normalised by the total traffic surrounding the intersection.

Density Weighted Cycling The most simple of such strategies uses the total number of cars that come from a road to determine the phaselength of the traffic signal. It therefore assigns longer phaselengths to trafficlights that have more cars approaching the intersection relative to all other incoming roads.

Queue Weighted Cycling Aiming to more precisely judge the severity of traffic load incoming from a road, this second weighting approach only takes into account cars that are actually waiting at a traffic light. A car is classified as waiting if it is slower than 5 km/h and has a negative acceleration.

Flow Weighted Cycling Finally, the last of such strategies weights by taking into account the traffic flow coming from a road during its previous green phase. Thereby, roads are preferred that predictively enqueue the most cars when turned green. The most apparent advantage of this approach is the fact that it will not give too much attention to overfilled roads that can actually not be relieved when lights turn green because the following traffic disallows it.

Dynamic strategies are strategies that instead of deciding how long a light has to stay green decides which light to turn green based on the information that the intersection gathers. Currently there are two strategies that do this.

Dynamic Cycling The first strategy that determines which light of an intersection should turn green is Dynamic Cycling. This strategy measures the amount of cares that are moving towards the

intersection for each connected road and chooses to turn the light green for the road which has the most incoming traffic. It makes the next decision at a set interval.

Dynamic Cycling+ The second strategy is Dynaminc Cycling+. This strategy is a more advanced version of the regular Dynamic Cycling. The differenc is that Dynamic Cycling+ does not only measure the incoming traffic for each road but also measures their average speed. it chooses to turn the light green for the road with the lowest average speed. Should the average speed of two or more roads be close to each other, then the intersection will choose the road with teh most incoming traffic. Another feature this strategy has is that at every traffic light has to turn green at least once within a certain amount of time. This is to prevent some roads from never turning green and thus causing some cars to never be able to drive.

4 Methodology & Experiments

4.1 Measured Statistics

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:

$$v = \frac{s}{t} \qquad (4)$$

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

5

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

---

of time each car spends 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.

4.2 Simulated Maps

Experiments are conducted on three different maps.They differ in the challenges that they impose on the traffic light systems and are therefore supposed to test their strengths and weaknesses. In the following, the three maps are described and motivated. Images can be found in the Appendix.

Two Districts "Two Districts" is a small and simple map, designed to test the effects of modifying the most crucial roads in a map. The map has a residential district on the left and an indus-

trial district on the right, connected by only one road, in order to force most vehicles to take it during the rush hours. This road is then modified in various ways, including adding lanes, changing it to a highway and also replaced by multiple one-way streets.

Brusselsepoort This map is modelled after a part of Maastricht called Brusselsepoort. It includes lots of residential areas, and a small commercial zone. This map was included in the testing as to have a "real world" scenario. This map was used to find the optimal phase lengths for the strategies.

Tie Fighter The "Tie Fighter" map has three sections. In the middle, there is a residential area, spawning most of the traffic in the morning. The left and right side of the map are industrial and commercial areas. The three sections are connected by three lane highways. The map is not the most realistic, but it can be a challenging case for the traffic light strategies, due to the often-saturated mid-section.

4.3 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.

4.4 Methodology

In order to perform statistical calculations on the data, it is often necessary but at least useful to have IID observations. It is obvious that this independence can not be assumed in a traffic simulation between individual cars, since the delay of one car often influences that of others in shared roads and queues. While restarting the simulation n times in order to get n observations that can be combined to a grand mean seems an obvious choice, the day-based simulation also allows to run multiple days and assume independence between days, since traffic resets during night to a reasonable extend.

Since this is a non-terminating simulation, it is furthermore necessary to take a so called *warm-up*

period or *transient phase* into account. It is the result of starting the simulation in an initial state that might not represent an actual state of the simulation in steady-state or inside a cycle. Therefore, the period where the simulation enters the steady-state needs to be removed from any statistical calculation such that it does not bias their results to the initial situation. Arguably, simulating traffic on the basis of days results in a cyclic simulation. Importantly though, this will not necessarily mean that the simulation entirely resets to zero cars in the system at midnight. For those reasons and the sake of convenience, in any experiment, the first simulated day will be removed from the simulation.

It is the objective of this work to compare multiple systems (i.e. strategies) under different conditions. In order to rank these strategies, a convenient framework for such a comparison would be needed. An all-pairwise comparison is not desirable, since the total of systems would require 30 comparisons, which would result in tiny confidence levels due to Bonferroni inequality (see e.g. Law, Kelton, and Kelton, 2007). Instead, the sample size required for each system in order to achieve a confidence level of α for the complete ranking can be determined by the approach of Dudewicz and Dalal (1975). Though, given the number of strategies to compare and the fact that we care about relatively small differences in the systems, fairly high numbers of simulated days would be necessary to guarantee a significant ranking with e.g. an significance level of 0.05 (about 4000 days for e.g. flow weighted cycling). The computationally expensive simulation experiment used in this work does not allow us to run such large experiments in feasible time. We therefore fall back to a comparison with benchmark strategies and report significance of the differences.

Three measures are employed to compare the performance strategies. Similarly to prior work (Srinivasan, Choy, and Cheu, 2006), the average velocity of cars in the system is measured. Additionally, we report the number of cars on the map and the percentage of cars waiting (as defined in section 3.2).

5 Results

In the following section, results for all experiments are presented. Firstly, the outcome of the optimization of the phase length parameter are reported and shortly discussed. Secondly, for each street layout, the performances of all strategies will be used to rank them. In the last subsection, the influence of some urban planning measures will be investigated.

5.1 Phase Length Optimization

Experiments for the optimization of phase lengths in all strategies were conducted on the Brusselse-poort map with 5 different configurations each and a simulation duration of 21 days (where the first day was removed as the transient phase. As the measure to be optimized on, we choose the average speed, since this is one of the most common metrics in the literature. Results of those experiments for the basic, informed and flow weighted strategy can be found in the appendix in figure 6 in the form of box plots as an average during the rush hour between 7 and 9 o'clock.

In both basic and informed cycling, 10 seconds and 25 seconds are respectively the best and second best performing strategies numerically. Since 10 seconds is a rather low phase length which might get its benefits from the specific design of the simulation environment, we choose 25 seconds as the phase length for these two strategies in the following experiments. For the flow weighted cycling (and it will be assumed here, that this generalizes to the similar other responsive strategies) higher phase lengths induce worse performance. Again, for the sake of realism we choose a cycle length of 30 seconds and do not consider lower values.

5.2 Relative Ranking of Strategies

Using the phase lengths chosen in the above section, experiments for all strategies on all maps were conducted. Each experiments ran 51 days long, such that the final score is the average of 50 days after the removal of the transient phase. In the following, tables contain averages for the first rush hour. Numbers for the afternoon's rush hour can be found in the Appendix.

5.3 Experiments on the Two-Districts Map

On the first of three maps, the informed cycling strategy outperforms all other strategies significantly in average velocity. Except for Dynamic and Dynamic+, all strategies outperform the Basic Cycling benchmark system, though. All results can be found in table 1
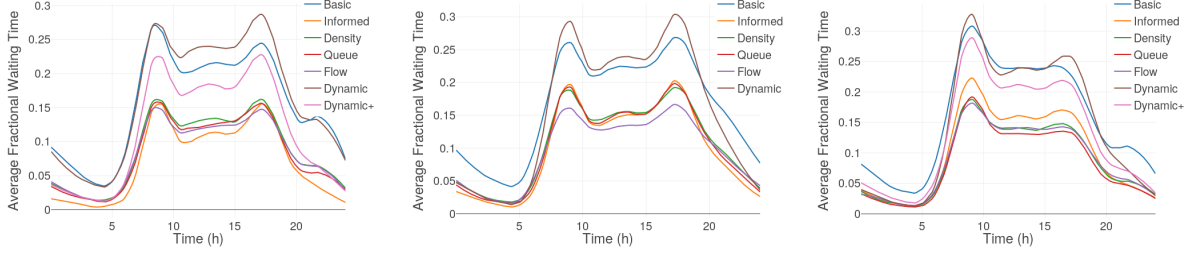
Figure 4: Fraction of cars waiting in the experiments for Two-Districts (left), Brusselsepoort (middle) and the Tie-Fighter (right), as a development over a day.

Table 1: Average Velocity of Cars during first rush hour (between 7am and 9am) on the Two-Districts map.

	Mean	$S^2(n)$	$S^2_{Means}(n)$
Basic	6.323	46.36	0.139
Informed	**11.58**	133.15	0.596
Density W.	9.231*†	34.61	0.434
Queue W.	9.344*†	36.62	1.288
Flow W.	9.536*†	35.81	0.3752
Dynamic	6.015*†	36.75	0.0943
Dynamic+	2.461*†	8.5459	0.104

Table 3: Average Velocity of Cars during first rush hour (between 7am and 9am) on the Brusselsepoort map.

	Mean	$S^2(n)$	$S^2_{Means}(n)$
Basic	6.34	41.65	0.064
Informed	**9.86**	90.13	0.240
Density W.	8.46*†	20.80	0.304
Queue W.	7.55*†	20.07	0.649
Flow W.	9.00*†	23.02	0.239
Dynamic	1.99*†	3.997	0.073
Dynamic+	-	-	-

Table 2: Fraction of Cars Waiting during first rush hour (between 7am and 9am) on the Two-District map.

	Mean	$S^2(n)$	$S^2_{Means}(n)$
Basic	0.2435	0.017	0.0004
Informed	**0.1229**	0.0166	0.0004
Density W.	0.1409*†	0.014	0.0002
Queue W.	0.1332*†	0.0137	0.0004
Flow W.	0.1326*†	0.0139	0.0001
Dynamic	0.242†	0.242	0.0004
Dynamic+	0.189*†	0.0149	0.001

Table 4: Fraction of Cars Waiting during first rush hour (between 7am and 9am) on the Brusselsepoort map.

	Mean	$S^2(n)$	$S^2_{Means}(n)$
Basic	0.226	0.015	0.0005
Informed	0.156	0.017	0.0004
Density W.	0.1607*†	0.0119	0.0002
Queue W.	0.159*	0.0123	0.0005
Flow W.	**0.139***†	0.010	0.0002
Dynamic	0.2387*†	0.0156	0.0019
Dynamic+	-	-	-

With respect to the fraction of cars waiting (table 2), informed cycling is the best performing strategy again, but flow and queue weighted cycling are close runner-ups. All strategies but the dynamic ones can outperform the basic cylcing benchmark.

5.4 Experiments on the Brusselsepoort Map

On Brusselsepoort, Informed Cycling is significantly the best performing strategy for average speed, but is closely followed by flow weighted cycling for average speed (see table 4. Dynamic cycling again performs worst, and Dynamic+ even

had to be terminated due to performance issues. Except for these two, all strategies can outperform basic cycling.

Measured in the fraction of waiting cars (table 4, flow weighted cycling is the best performing system, significantly better than both basic and informed cycling. Close runner-ups are informed cycling and queue weighted cycling. As was observed before, all strategies except for the dynamic ones, outperform the basic cycling benchmark.

8

Table 5: Average Velocity of Cars during first rush hour (between 7am and 9am) on the Tie-Fighter map.

	Mean	$S^2(n)$	$S^2_{Means}(n)$
Basic	5.588	49.31	0.060
Informed	8.710	91.42	0.1881
Density W.	8.482*†	28.14	0.365
Queue W.	8.182*†	29.41	1.596
Flow W.	**9.002***†	31.77	0.328
Dynamic	1.503*†	2.983	0.054
Dynamic+	1.574*†	3.251	0.078

Table 6: Fraction of Cars Waiting during first rush hour (between 7am and 9am) on the Tie-Fighter map.

	Mean	$S^2(n)$	$S^2_{Means}(n)$
Basic	0.256	0.016	0.0006
Informed	0.1742	0.0215	0.0006
Density W.	0.1520*†	0.0127	0.0004
Queue W.	**0.1467***†	0.0132	0.0006
Flow W.	0.1475†	0.01287	0.0002
Dynamic	0.2565*†	0.01699	0.0035
Dynamic+	0.2284*†	0.014	0.0040

5.5 Experiments on the Tie-Fighter Map

The Tie-Fighter map yields similar results to those of Brusselsepoort. Best-prforming system measured in average speed (table 5) is flow-weighted cycling, followed by informed cycling. Except for the dynamic strategies, all strategies significantly outperform basic cycling. The average fraction of cars waiting on this map (table 6 has its best score for queue weighted cycling, but flow weighted cycling follows iwth a difference of only 0.0008. For both measures, all systems except for the dynamic strategies outperform basic cycling significantly.

5.6 Urban Planning

To investigate the effect of measures related to urban planning, three ways of changing the Two-Districts map were tested. Firstly, the number of lanes of the road connecting the districts was increased. Secondly, instead of one connecting road, multiple one-way roads were used to make moving from one district to the other possible. Finally, the connecting road is replaced by a highway. In all cases, informed cycling is the underlying traffic light strategy. Figure 5 shows the results in the form of boxplots for the morning rush hour. All differences are significant with a significance level of 0.05.

As can be observer, all measures improve the fraction of cars waiting. The smallest effect comes from one-way roads, while highways have the largest effect. There is no difference in the mean between two and three lanes, but for three lanes, the quantiles are shifted downwards. For the average velocity, only the introduction of a highway improved the result, while the other measures even slightly decrease the average velocity.

6 Discussion

On the simple map Two-Districts, Informed is best strategy by some margin. More complicated maps result in flow and sometimes queue weighted cycling to be superior, in particular when measured in fraction of cars waiting. This shows that the map design is an important factor for the choice of strategy. It can not be assumed that there is a definite ranking in performance for such strategies.

Dynamic Cycling+, even though it is the most advanced strategy, does not perform well. Likely the cause is that it is always trying to absolve queues but in doing so it actually creates at least two new ones depending on the number of roads connected to the intersection. This happens because the algorithm decides on which light to turn green based on the average speed and amount of cars on the road. However if for example we have an intersection with 4 connected roads and one road has a big queue, this road will continuously be selected as the traffic light that needs to be green. Unfortunately this causes the other roads to clutter up until they are weighted heavier that the currently most cluttered road. It was attempted to avoid this from happening by making every traffic light turn green once in a while but this impact was clearly not enough to counteract this problem.

6.1 Limitations

6.2 Future Work

- test more maps and find reasons for differneces in maps: are grid maps resulting in informed to be better? - see if that also applies to more advanced strategies

7 Conclusion

References

Behrisch, Michael et al. (2011). "SUMO–simulation of urban mobility: an overview". In: *Proceedings of SIMUL 2011,*
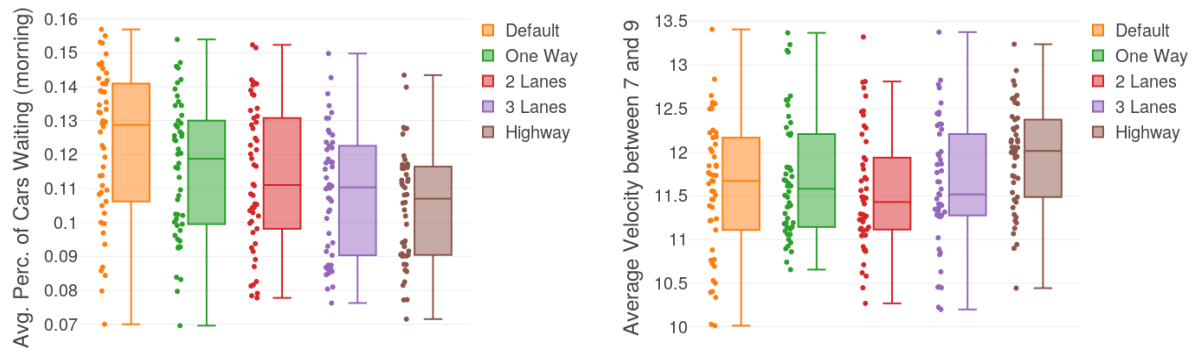
Figure 5: Fraction of cars waiting and average velocity for different map configurations in the form of boxplots. Default map is Two-Districts, the strategy used is informed cycling.

The Third International Conference on Advances in System Simulation. ThinkMind.

Chao, Kuei-Hsiang, Ren-Hao Lee, and Meng-Hui Wang (2008). "An intelligent traffic light control based on extension neural network". In: *International Conference on Knowledge-Based and Intelligent Information and Engineering Systems*. Springer, pp. 17–24.

Coll, Pablo et al. (2013). "A linear programming approach for adaptive synchronization of traffic signals". In: *International Transactions in Operational Research* 20.5, pp. 667–679.

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

Dudewicz, Edward J and Siddhartha R Dalal (1975). "Allocation of observations in ranking and selection with unequal variances". In: *Sankhyā: The Indian Journal of Statistics, Series B*, pp. 28–78.

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.

Law, Averill M, W David Kelton, and W David Kelton (2007). *Simulation modeling and analysis*. Vol. 3. McGraw-Hill New York.

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.

Lin, Wei-Hua and Chenghong Wang (2004). "An enhanced 0-1 mixed-integer LP formulation for traffic signal control". In: *IEEE Transactions on Intelligent transportation systems* 5.4, pp. 238–245.

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).

Papageorgiou, Markos et al. (2003). "Review of road traffic control strategies". In: *Proceedings of the IEEE* 91.12, pp. 2043–2067.

Srinivasan, Dipti, Min Chee Choy, and Ruey Long Cheu (2006). "Neural networks for real-time traffic signal control". In: *IEEE Transactions on Intelligent Transportation Systems* 7.3, pp. 261–272.

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.

Appendix

Table 7: Average Velocity of Cars during second rush hour (between 17am and 19am) on the Two-Districts map.

	Mean	$S^2(n)$	$S^2_{Means}(n)$
Basic	6.900	43.806	0.06631
Informed	**10.2690**	106.758	0.2388
Density W.	8.9365*†	26.867	0.2742
Queue W.	8.902*†	28.91	0.811
Flow W.	9.105*†	28.58	0.2642
Dynamic	5.9756*†	27.924	0.0824
Dynamic+	9.9371*†	110.38	0.1399

Table 8: Fraction of Cars Waiting during first rush hour (between 17am and 19am) on the Two-District map.

	Mean	$S^2(n)$	$S^2_{Means}(n)$
Basic	0.221	0.012	0.0001
Informed	0.135	0.01492	0.0002
Density W.	0.1383*	0.0123	0.0001
Queue W.	0.1327*	0.0118	0.0001
Flow W.	0.1310*	0.0119	0.0001
Dynamic	0.252*†	0.0128	0.0002
Dynamic+	**0.10620**ial*†	0.0121	0.0000

Table 9: Average Velocity of Cars during second rush hour (between 17am and 19am) on the Brusselsepoort map.

	Mean	$S^2(n)$	$S^2_{Means}(n)$
Basic	6.190	33.74	0.040
Informed	**9.524**	74.75	0.156
Density W.	8.471*†	15.78	0.263
Queue W.	7.534*†	15.55	0.352
Flow W.	9.198*†	17.50	0.177
Dynamic	1.868*†	2.564	0.032
Dynamic+	-	-	-

Table 10: Fraction of Cars Waiting during second rush hour (between 17am and 19am) on the Brusselsepoort map.

	Mean	$S^2(n)$	$S^2_{Means}(n)$
Basic	0.2553	0.014	0.0001
Informed	0.182	0.016	0.0001
Density W.	0.1808*	0.0104	0.0001
Queue W.	0.1817*	0.011	0.0001
Flow W.	**0.154**al*†	0.009	0.0001
Dynamic	0.2814*†	0.012	0.0001
Dynamic+	-	-	-

Table 11: Average Velocity of Cars during first rush hour (between 17am and 19am) on the Tie-Fighter map.

	Mean	$S^2(n)$	$S^2_{Means}(n)$
Basic	5.109	40.03	0.091
Informed	7.273	75.76	0.108
Density W.	7.481*†	22.69	0.295
Queue W.	7.803*†	23.47	1.735
Flow W.	**7.826**al*†	23.13	0.261
Dynamic	1.399*†	1.932	0.030
Dynamic+	1.745*†	3.031	0.038

Table 12: Fraction of Cars Waiting during second rush hour (between 17am and 19am) on the Tie-Fighter map.

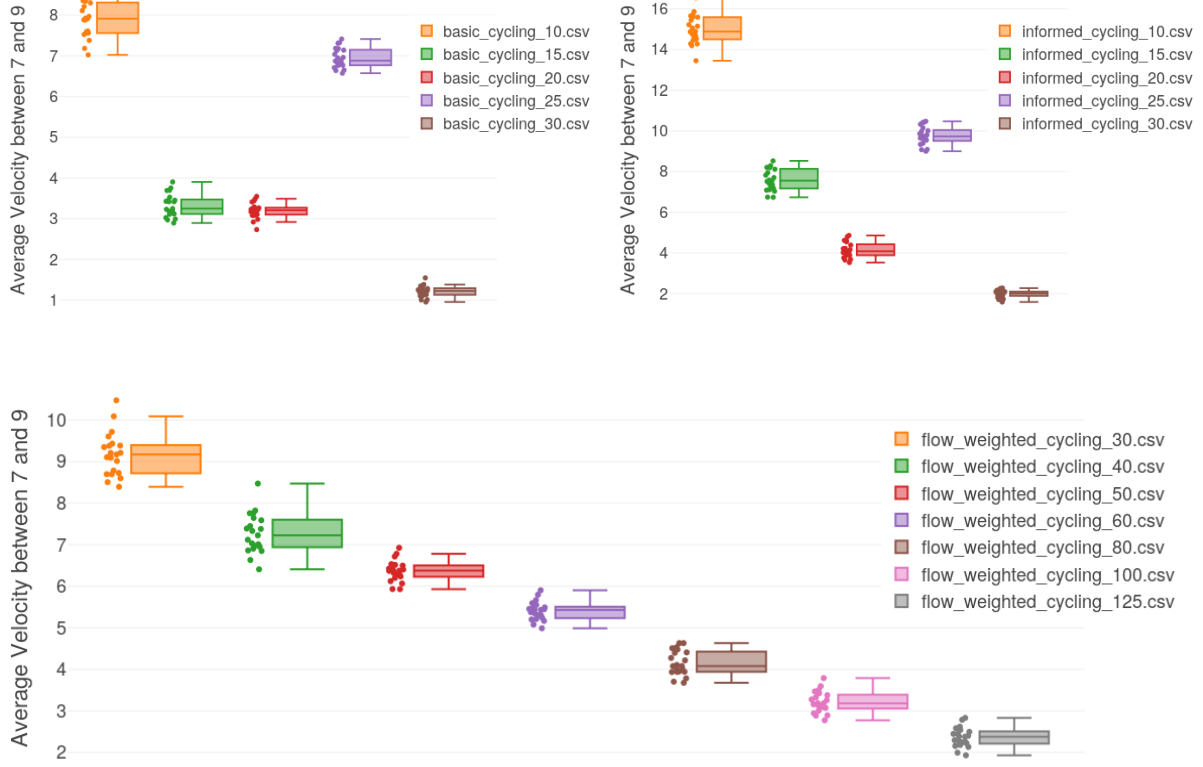|	Mean	$S^2(n)$	$S^2_{Means}(n)$
Basic	0.1934	0.0108	0.0001
Informed	0.1402	0.0140	0.0001
Density W.	0.1221*†	0.009	0.0001
Queue W.	**0.1119**al*†	0.0094	0.0001
Flow W.	0.1202*†	0.010	0.0001
Dynamic	0.2193*†	0.0108	0.0001
Dynamic+	0.1762*†	0.0100	0.0001

Figure 6: Results for the phaselength optimization experiments for basic cycling, informed cycling and flow weighted cycling.
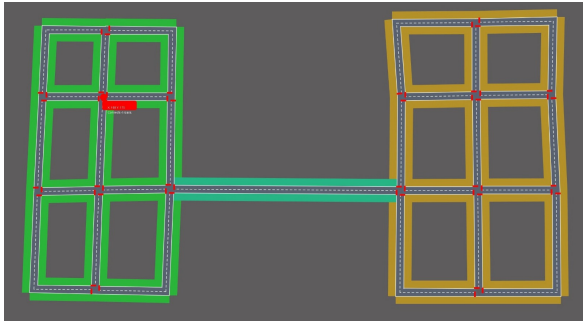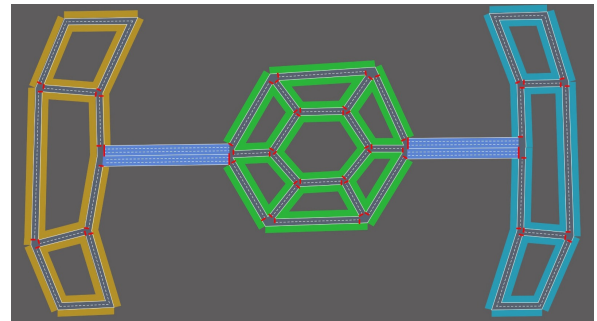

Figure 7: Map Two-Districts.

Figure 9: Map Tie-Fighter.

Figure 8: Map Brusselsepoort.