Intelligent Traffic
Models and Simulations

Rik Claessens, Lukas Kang, Cliff Laschet, Tom Pepels

June 29, 2011

Abstract

In this article models for realistically simulating road-traffic are discussed and implemented in a microscopic traffic simulator. This implementation is validated based on a well-known traffic situation in Maastricht. After which the validated simulation is used to model, analyse and improve the situation. Different strategies are applied for optimizing traffic light phases and a comparison of different lane changing models is made.

1 Introduction

Traffic congestion is a major economical problem in developed countries. Developing realistic traffic simulations based on empirically tested and proven models can assist research in traffic congestion. Furthermore testing traffic light strategies within a simulation avoids the need for costly real life testing. In this article traffic simulation models are outlined. These models were implemented in a software tool that allows the generation of real-life traffic simulation as maps. These maps are configurable such that different traffic light strategies and traffic densities can be used throughout. The simulation of traffic flow is based on empirically tested models. Both a car-following model and lane-changing model are used to realistically mathematically simulate traffic flow within the generated maps. The simulated models are validated using real-life data generously provided by the Maastricht Municipality.

The article first gives an introduction to traffic simulation, then describes two traffic flow models in detail. After this the results for the simulation's validation based on real-life data is provided. Next some experiments with traffic light strategies are conducted and the effects of lane-changing are discussed. Finally, conclusions are outlined.

2 Simulation

For traffic simulation software a traffic model needs to be specified. In general there are two possible choices:

- A microscopic traffic model
- A macroscopic traffic model

Microscopic traffic models simulate each vehicle as an individual. All characteristics of the vehicle, for instance velocity, acceleration and position, are simulated at all time instances. The advantage of microscopic traffic models is the ability to show accelerating and breaking cars on the screen.

Macroscopic traffic models compute dynamic characteristics of the road, rather than considering individual vehicles. They calculate the average density of a road or averages of vehicle properties like velocity or overall standing time [5]. An advantage of macro traffic models is, for example, the ability to handle a lot of cars on a very large intersection.

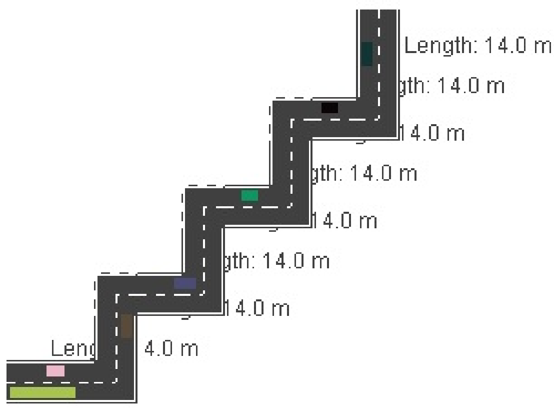

Figure 1: Model of a diagonal road

Now, to choose one of those model types for implementation a comparison of requirements and features is required. As this project involves the simulation and demonstration on screen of specific traffic situations, the microscopic model is chosen. In a microscopic model it is easily possible to calculate values like the averages of density and speed, since all vehicles are kept track of at every time instance. Additionally, since computation speed has increased rapidly over the past few years, it has become possible to simulate rather complex traffic situations with a microscopic model as well. This concludes and motivates the choice of the traffic model to be implemented.

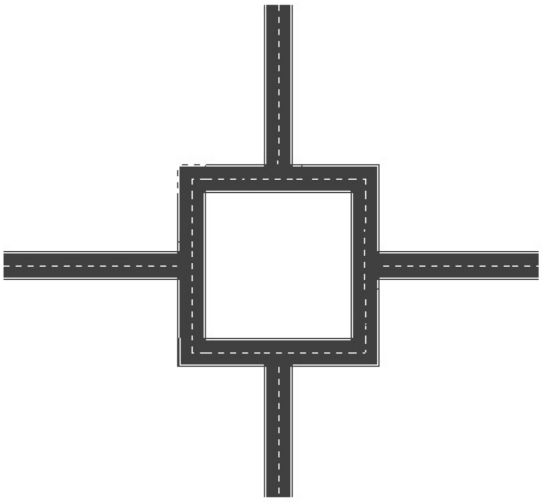

Figure 2: Model of a roundabout

After specifying the model type, additional restrictions and characteristics of the model need to be identified. For that purpose it is again necessary to take a look at the requirements and research questions. The program must be able to model specific situations from the city of Maastricht. In these situations it should perform simulations with different parameters and configurations of the lane changing model and the traffic light model. Most important is the ability to model these situations and not drawing the real map of it on the screen. Therefore the model has the restriction that roads can have four directions only: North, east, south and west. In other words it is not possible to draw diagonal roads on screen. However it is possible to model situations with diagonal roads (Figure 1). This restriction makes the implementation of the model easier, since vehicles and roads need to be drawn horizontally and vertically only. Another restriction of the model is the lack of roundabouts. This is also due to simplicity of implementation. However this restriction does also not impair the model. A roundabout can still be modeled using other intersections and road segments (Figure 2).

3 Models

As described in the previous section there are several possibilities to simulate traffic mathematically. Macroscopic models describe the dynamics of roads in terms of quantitatively, like density or flow. Microscopic models on the other hand describe the motion of individual vehicles. The simulation software outlined in this article uses a popular microscopic model called the Intelligent Driver Model (IDM) published by Treiber et al in 2000. Furthermore to simulate cars changing lanes (since IDM is a single-lane model) lane changing models are implemented [1] .

3.1 Intelligent Driver Model

The car-following model "Intelligent Driver Model" (IDM) was developed based on data from several German freeways that show different kinds of congested traffic [5]. As a microscopic model IDM describes the positions and velocities of single vehicles. IDM is a car-following model meaning that every car a bases his acceleration on the car in front $a-1$ at any given time based on a desired velocity v_0, approaching velocity $\Delta v_a = v_a - v_{a-1}$ and minimum spacing $s_a = x_{a-1} - x_a - l_{a-1}$ between itself and the leading car, where l_a is the length of the car. The acceleration of vehicle a is then updated using the differential equation.

$$\frac{\mathrm{d}v_a}{\mathrm{d}t} = a\left(1 - \left(\frac{v_a}{v_0}\right)^{\delta} - \left(\frac{s^*(v_a, \Delta v_a)}{s_a}\right)^2\right) \tag{1}$$

with

$$s^*(v_a, \Delta v_a) = s_0 + v_a T + \frac{v_a \Delta v_a}{2\sqrt{ab}} \tag{2}$$

v_0, s_0, T, a and b have the following meanings in the model: v_0 is the desired velocity that the car would drive at on a free road, s_0 is the minimum spacing that is kept between cars even at stand-still, T is the desired time headway to the vehicle in front which is kept constant, a is the acceleration of the car and b is the comfortable breaking deceleration. δ is a constant which is set to 4 according to Treiber et al. [5]. The values used in the developed software were obtained [4] and are listed in table 1.

Parameter	Value Cars	Value Trucks
Desired velocity v_0	Speed limit	Speed limit
Time headway T	1.5 s	1.7 s
Minimum gap s_0	2 m	2 m
Acceleration a	2 m/s^2	1.5 m/s^2
Deceleration b	3.0 m/s^2	2.0 m/s^2
Length	2.5 m	10.5 m

Table 1: IDM Parameter values

The v_0 parameter is set to the speed limit of the current road. The acceleration and length parameters have appropriate random variations to simulate different types of drivers and vehicles respectively. The acceleration expression can be split into a "free road" term

$a\left(1-(\frac{v_a}{v_0})^\delta\right)$ when a car drives at its desired velocity and an interaction term $-a\left(\left(\frac{s^*(v_a,\Delta v_a)}{s_a}\right)^2\right)$ which determines the interaction between the car and the leading car. On an empty road s_a will be large and therefore the free road term will approach its desired value. If s_a decreases the interaction term increases and thereby the acceleration decreases. The deceleration term depends on the ratio between the desired minimum gap s^* and the actual gap s_a. The variables used in IDM allow for variation in driver aggressiveness and allow different types of cars to be modelled such as trucks and cars of varying lengths and accelerations.

3.2 Lane Changing

Lane changing has a significant impact on the traffic flow and therefore it is an important component of a microscopic simulation [1]. The model discussed in this paper implements two variants of lane changing: discretionary lane changing (DLC) and mandatory lane changing (MLC). DLC behaviour occurs when a car switches to another lane because this lane has better traffic conditions. The condition on a lane is measured by its density and the average speed on that specific lane. The average speed $\bar{v}$ for lane n is calculated as follows:

$$\bar{v}_n = \frac{\sum_{n=0}^{k_x} v_n}{k_x} \tag{3}$$

Where k_x is the number of cars on lane x and v_n is the velocity of car n. Next, the density is calculated:

$$d_n = \frac{k_n}{l_n} \tag{4}$$

Where d_n denotes the density of lane n, k is the number of cars on lane n and l_n is the lenght of lane n. The completed evaluation function is then defined as as a weighted sum of the density and the average speed:

$$e_n = w_v * \frac{1}{\bar{v}_n} + w_d * d_n \tag{5}$$

Where e is the total evaluation value of lane n, w_v is the weight for the average speed, and d_n is the weight of the density. Hence, a lower value of a lane indicates a better traffic condition on that lane.

MLC occurs when a car needs to change lanes to follow a certain route, for example when it needs to leave a highway or changing to the most left lane for taking a left turn on a crossroad. Both variants have a different reason for lane changing but also a different method of determining the adjacent target lane for the lane change. MLC has a higher priority than DLC, hence a DLC can only occur when no MLC is required. Therefore, DLC is disabled for a car once it reaches a certain zone around a junction. Should the car be in a zone, then only MLC is executed, otherwise both variants can occur. When the target lane is determined by either of the two methods, the gap acceptance model is applied [1]. This checks whether there is enough space available on the adjacent target lane to perform a lane change. As shown in the image below, the total gap that is available is determined by the lead gap, the lag gap and the length of the car. When the total clear gap is acceptable, a lane change is performed immediately. It is assumed that performing a lane change is an instant process, consuming no time.

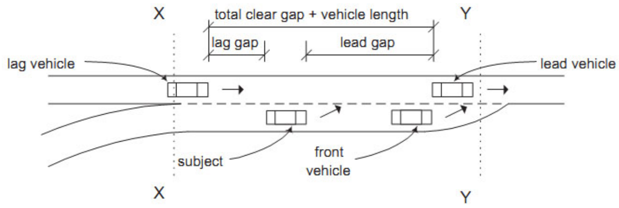

Figure 3: Gap Acceptance Model

However, a problem arises when a lane change is mandatory and there is no adjacent gap that allows a lane change to be executed. In that case, the car will not be able to follow the pre-determined route and a new route will be needed, which may be suboptimal compared to the previous route. Forced merging can solve this problem, by allowing cars to search for gaps that are not directly adjacent but are reachable by decelerating or accelerating the car. For example, in the image shown below a situation arises where the subject car needs to perform a MLC to the most left lane. There is however no immediate gap available for a lane change, and therefore it starts slowing down. By slowing down, an adjacent gap is found between the lag and lead vehicle depicted in the image. The subject car changes lanes and the route can be followed accordingly.

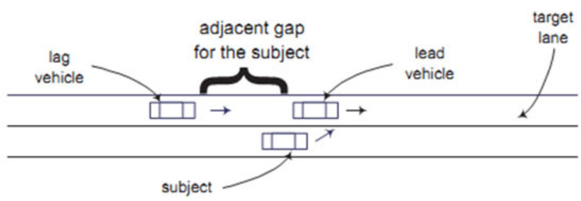

Figure 4: Forced Merging Model

4 Validation

One crucial factor of a traffic simulation model is its validity. This section deals with the validation process of the model. It is necessary to compare the model to a real-life situation. In this case empirical traffic data was collected from the Maastricht City Council. This data needs to be evaluated and then compared to the output of the computer program containing the traffic flow model. The comparison of these data reveals whether the traffic model is valid or, in other words, whether it can model traffic situations realistically.

4.1 Situation

For this project a specific situation in the city of Maastricht is chosen for validation purpose. The crossroads east of the river Maas having 'Noorderbrug' coming from west and turning to 'Viaductweg' in eastern direction and having 'Willem Alexanderweg" coming form northern direction turning to 'Franciscus Romanusweg' in the south (Latitude: 50.86041475239171, Longitude: 5.70314884185791). Figure 5).

Figure 5: Satellite picture of the crossroads

This crossroads is modelled using the software implemented for that purpose (Figure 6). Thereby all roads are modelled exactly as they look in reality. They have the same number of lanes and also the same speed limits. The only thing where they do not entirely correspond to their archetype is their exact postion. This, however, only is an optical drawback. The fact that they do not look exactly like in reality does not mean that they are wrong. In fact, characteristics like speed limits and number of lanes are far more important for a good model than a small changes in direction. Of course the traffic light phases are also implemented as on the real

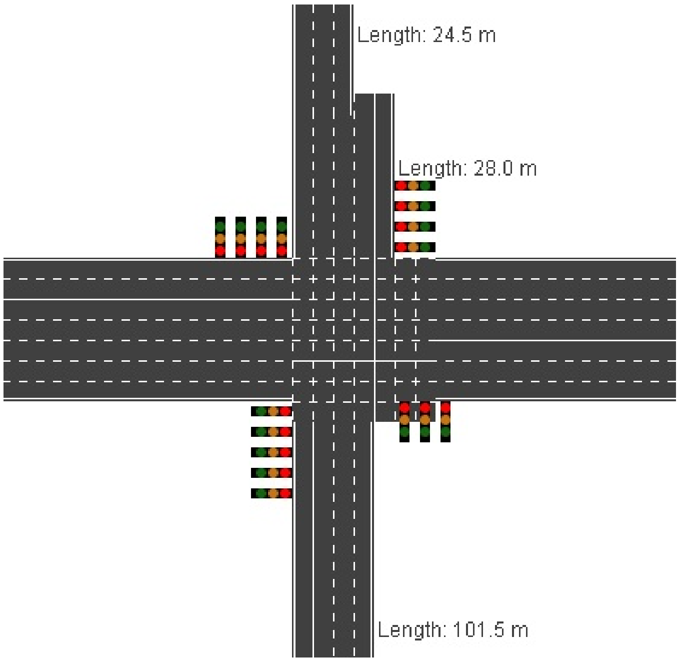

Figure 6: Screen shot of the software

crossroads. Finally one of the most important factor for validation is the cars. The collected data includes densities on every lane. Thus the more frequent routes can be determined and the number of cars coming from each road is known. Those parameters are also given to the program. The validation data is retrieved from the time between 8 am and 8.15 am. At this time there is most traffic on the junction and the traffic lights are set to a static phase control. The average waiting time of the program is compared to the average waiting times of the data form the Maastricht City Council.

4.2 Results

The average waiting time from the real crossroads between 8 am and 8.15 am is 29.54 seconds. The mean average waiting time of 20 simulation runs with all the parameters set according to the situation in the real world is 24.99 seconds, with a standard deviation of 3.65. The fact that the two values differ might have two reasons. First, it is not known when the City Council exactly considers a car as 'waiting'. In the program a car is considered as waiting, if it drives slower than 15 km/h or is braking while being slower than 30 km/h. These values have been determined by hand; by examining the driving cars on screen. Second, it is assumed that the standard deviation for the real life data is pretty high also. Thus the difference between the two values is really dependent on how many cars exactly drive on the roads. Since the cars are spawning random only considering the average densities and the probabilities for taking a certain route, it is almost impossible to have to very similar values. All in all, the values seem close enough to assume the model to be valid.

5 Optimizing traffic flow

Because modern traffic gets more busy each year, traffic control is getting more important as well. Optimizing traffic flow using different strategies throughout a city has numerous benefits and is essential for any big city. In the experiments in this section we are using the junction described in section 4.1. First different traffic light control strategies are introduced, as well as the different configurations tested and their results. Then, the experiments and results on the influence of lane changing are presented.

5.1 Traffic light control strategies

In this section we will delve into different traffic control strategies involving traffic lights. In detail this paper will discuss the characteristics of 3 different strategies, their advantages and disadvantages over the others and apply the strategies to a real-life traffic situation.

The first strategy is the most simple one: *Marching Control*. In this strategy all traffic lights on roads coming from the same direction will be green at the same time and all other directions will have a red light. This strategy is also called synchronized control as the traffic lights in 1 direction have a synchronized phase. This control strategy can easily be extended to be a more sophisticated controller using different phases for example.

The second strategy is called *Green Wave control* [2]. This control strategy aims that cars driving on the primary road through a set of traffic lights will have a green light at all following traffic lights from the moment they have their first green light on the primary road. This strategy could be combined with other strategies as long as the green light phase on the primary road is in sync with the other traffic lights on that road. The phase difference between two traffic lights on the primary road is calculated using the length and the maximum speed of the road connecting the traffic lights: $p = \frac{l_{\text{road}}}{v_{\max}}$. By summing over the phase difference over all preceding traffic lights you obtain the phase difference with the first traffic light in this green wave. This strategy is primarily used in traffic situations where there is a distinctive primary road going through a set of traffic lights. The primary road will have a much larger traffic density than the other roads so that the benefit of having a good traffic flow on this road is optimized.

The third traffic light control strategy is called *Self-organizing traffic light control* (Sotl-control) [2]. As the name suggest this control strategy does not have a fixed phase but organizes the green phases based on a set of rules. The first variation of the Sotl-control is *request control*. Traffic lights are set to green when a certain number of cars is waiting in front of the light. As this is the only rule, it is obvious that this variation has a major shortcoming: at a busy intersection, when a light switches to green and the treshold is reached on the crossing road, the green light will switch to that road. This will lead to a rather chaotic and annoying traffic situation as the light will have a short phase and therefore switch really fast as long as the traffic density is high.

To avoid this shortcoming an additional rule was developed for the second variation of Sotl-control, *phase control*. Now the light will not switch within a certain interval, avoiding the problem that the lights switches too fast. By avoiding this problem, a new problem arises. Since the light will stay red within a particular interval after a light has switched to green, the traffic can build up on another road where there is a red light.

The third variation, *platoon control*, has 2 more rules in addition to the one introduced in the second variation. This last variation handles large groups of cars, platoons, in order to allow them to stay together and improve the traffic flow. The light will not switch if there is at least one car within a certain distance of the traffic light. However to avoid the light staying on too long the first rule is not taken into account when there are more than a certain number of cars is approaching the intersection.

These Sotl-control strategies allow for a more sophisticated traffic control than the first two strategies. It allows the strategy to adapt to changing traffic situations. In the following section we will investigate the influence of the different control strategies on the traffic flow.

5.2 Experiments traffic light control strategies

In this section this paper will discuss the setup and the results of the experiments with different traffic light control strategies. For all configurations, 10 runs of 15 minutes each are made and only statistics after a warm-up period of two minutes are considered. After this time the density of the traffic has stabilised and is close to the density from the data obtained from the City Council of Maastricht. The average waiting time is based on all cars which have passed the junction investigated.

First of all the traffic flow is simulated using the traffic light control from the real situation in order to validate the model as discussed in section 4. Next, different configurations for marching control and one variant of the Self-organizing traffic light control, phase control were tested. Green wave was not tested as the simulation consisted of only one traffic light junction, making the principle of a green wave traffic light strategy useless. For the same reason platoon control was not tested since the aim of this control strategy was to keep a group of cars moving though a city and this does not occur in a simulation consisting of just one traffic light junction. Also the first variant of the Sotl-control, request control, was not tested based on the findings from [2], where it is shown that in busy traffic situations, this strategy causes the light to switch too fast.

5.3 Results traffic light control strategies

In the following table the results of the experiments with different traffic light control strategies are shown. The average waiting during 15 minutes is shown for each configuration tested. The real situation uses the actual phases from the junction in Maastricht during rush hour. The march configurations show the time the west- and eastbound directions get green and the greentime for the north- and southbound roads. The balanced march uses a phase of 20 seconds for the direction with the highest density, namely the eastbound road. The other directions get a phase proportinal to the average density from the traffic data obtained from the City Council.

Control type	Average waiting Time
20-10 march	12,39 sec
balanced march	13,35 sec
15-7.5 march	13,47 sec
17.5-10 march	16,34 sec
15-10 march	23,80 sec
real situation	24,99 sec
phase thr 40	43,52 sec
phase thr 30	44,42 sec
phase thr 20	47,78 sec

Table 2: Marching control results

From this results we can see that a simple marching control using different phases works best for this junction. In fact, all marching control configurations tested have a lower average waiting time than the real-life configuration. However it is very important to note that the simulation models only one intersection whereas the control strategy from the city of Maastricht looks at the traffic density and incoming traffic from other junctions and adjusts this strategy based on this information. Therefore we are not able to draw conclusions which strategy is better. Because the traffic light strategy of Maastricht is so complex you first have to model the complete city to be able to draw a decisive conclusion.

Also it is clear that phase control does not work as good as marching control for this junction. The main reason is that the density differs a lot from the different directions. The threshold for the number of cars waiting in front of the traffic light might be working well for traffic for the busy roads, but not for traffic coming from the north or the south.

As traffic light control is a very complex problem to solve, further investigation into different strategies and simulating a larger part of the city is required to draw a decisive conclusion about which strategy works best for the given traffic situation.

5.4 Experiments lane changing

As was stated before, DLC evaluates lanes using an evaluation function consisting of a weighted sum of both the average velocity on the lane and the density of the lane. The weights in this evaluation function can be optimized, increasing the performance of the lance changing system. Therefore, tests were conducted, using different weights for both the density and the average speed. Ten different compositions of weights were examined, where each value was used in a simulation of 15 minutes, taking average density, average speed, waiting time and total number of lane changes into consideration. The tests were executed on a 4 way crossroad with phase control trafficlights, where each road consisted of 4 lanes in each direction. MLC was disabled, since the evaluation function has no influence on MLC. The results are discussed in the next section.

5.5 Results lane changing

Results are shown in the tables below. For each composition of the speed weight and density weight, the total number of lanechanges and the average waiting time are indicated for a simulation of 15 minutes.

Weights		
Speed	Density	# Lane changes
0	1	1391
0.1	0.9	1387
0.2	0.8	1428
0.3	0.7	1448
0.4	0.6	1413
0.5	0.5	1403
0.6	0.4	1401
0.7	0.3	1391
0.8	0.2	1460
0.9	0.1	1497
1	0	1582

Table 3: Lane changes for different weights

Weights		
Speed	Density	Avg wait time (s)
0	1	12.90
0.1	0.9	12.50
0.2	0.8	14.60
0.3	0.7	13.40
0.4	0.6	13.30
0.5	0.5	14.10
0.6	0.4	13.30
0.7	0.3	14.00
0.8	0.2	15.10
0.9	0.1	12.90
1	0	12.50

Table 4: Average waiting times for different weights

As can be noted from table 4, the differences between average waiting time are only mere seconds. However, the average waiting time is lowest when weights 0 and 1 are used or 1 and 0 for speed weight and density weight respectively. It can also be noticed that when both weigts are chosen more evenly, the waiting time increases. For example, weights of 0.5 and 0.5 indicate more average waiting time. Hence, it would be better to consider just one factor for evaluating, either average speed or density, which will decrease waiting time, and certainly not both, since this will increase waiting time. However, when table 3 is also considered for this observation, it can be noted that the total number of lane changes is lower for weights 0 and 1 then for weights 1 and 0, for the speed weight and density weight respectively. A lower number of total lane changes indicates a more fluently and calm traffic flow, and therefore a lower number of lane changes is preferred. Hence, a speed weight of 0 and a density weight of 1 for the evaluation function of DLC achieves less average waiting time and smaller number of lane changes required to achieve this average waiting time.

There is however a side note that needs to be taken into consideration. A Dudewicz-Dalal process, that was performed after these results were obtained, indicated that the number of runs required to achieve trustworthy results was far greater than the runs that were actually performed. Therefore, more runs are required to be able to state definite conclusions.

6 Discussion

The implementation of the Intelligent Driver Model [5] and the Lane changing model [1] resulted in a model which can be considered as valid (Section 4). From the results of the traffic light control experiments it is concluded, that the marching control suits better than the reality based traffic light configuration. With regard to the fact that the program only models one intersection with its traffic parameters this conlcusion can only be drawn for the model and not for the real world. However the huge difference in waiting time (Table 2) implies that an improvement in average waiting time is probable in the real world as well. The results of the lane changing experiments show that the optimal values for average waiting time and number of lane changes is obtained by using a speed weight of 0 and a density weight of 1 for the evaluation function(Table 4). With these values the traffic will be most fluent and the average waiting time for the vehicles crossing the junction is minimized.

To be able to draw decisive conclusions about the best traffic control strategy the model has to be extended to a bigger part of the city of Maastricht and roads with a direction towards the city. Also different times of day need to be taken into account as now the model simulates 15 minutes of the morning rush hour. Furthermore the data corresponding to the roads in the extended model need to be obtained from the City Council in order to validate this model. When these criterias are met, more experiments can be set up to test whether the traffic control strategy can be improved upon.

As the model gets bigger, the complexity will also get higher fast. For this extended model, it is not feasible anymore to test different strategies by hand. Therefore, a machine learning algorithm needs to be implemented to develop a new control strategy or improve the existing strategy used in Maastricht.

7 Acknowledgements

The authors would like to thank Toby Leurs from the Maastricht Municipality for kindly providing the traffic light and densitydata used to validate the simulation.

References

[1] Ben-Akiva, M., Choudhury, C., and Toledo, T. (2006). Lane changing models. In *Proceedings of the International Symposium of Transport Simulation.*

[2] Cools, S. (2006). A realistic simulation for self-organizing traffic lights.

[3] Helbing, A. et al. (2002). Micro-and macro-simulation of freeway traffic. *Mathematical and computer modelling*, 35(5-6):517–547.

[4] Treiber, M. (2011). Martin treibers personal website. `http://www.vwi.tu-dresden.de/~treiber/`.

[5] Treiber, M., Hennecke, A., and Helbing, D. (2000). Congested traffic states in empirical observations and microscopic simulations. *Physical Review E*, 62(2):1805.

A Validation data

Wachttijden

Locatiecode: S050 Viaductweg - Willem Alexanderweg
Periode: ma 23-05-2011
Objecttype: Richtingen
De getoonde data wordt gefilterd getoond.

Van	Tot	0001	0002	0004	0005	0006	0007	0008	0010	0011	0012
07:00	07:15	14,9	28,6	2,7	21,6	25,3	19,2	28,7	12,1	16,4	23,5
07:15	07:30	15,6	31,6	3,5	16,8	15,2	18,8	31,1	11,7	24,6	29,9
07:30	07:45	38,3	62,9	3,8	15,1	16,0	37,8	25,0	24,5	27,8	35,3
07:45	08:00	42,4	57,0	6,5	12,2	25,8	41,3	41,4	25,9	23,6	42,6
	07:00-08:00	29,8	48,2	4,2	16,1	20,1	29,2	33,2	20,3	23,3	33,5
08:00	08:15	37,7	53,4	3,8	16,6	25,6	44,3	39,6	22,2	24,2	28,0
08:15	08:30	44,2	66,3	4,8	25,3	18,8	71,2	57,6	20,3	19,1	31,0
08:30	08:45	32,9	51,9	10,7	16,1	23,3	56,2	42,5	20,0	29,3	41,5
08:45	09:00	39,7	49,4	2,1	9,9	20,8	40,3	34,2	22,1	26,1	41,0
	08:00-09:00	38,5	55,6	5,1	17,1	22,1	52,8	42,3	21,0	24,6	35,1
09:00	09:15	37,3	47,9	2,5	7,1	25,9	45,4	42,5	16,7	32,8	39,8
09:15	09:30	32,6	45,2	3,3	13,4	16,9	34,6	41,4	26,1	17,6	31,0
09:30	09:45	24,9	54,8	3,2	9,4	8,9	21,8	35,1	13,8	22,8	34,3
09:45	10:00	21,1	42,3	9,0	13,2	19,0	16,5	35,2	16,6	27,7	33,1
	09:00-10:00	30,0	46,9	4,2	10,6	18,2	28,4	38,6	18,8	25,3	34,8
Periode :	07:00-10:00	33,2	50,4	4,5	14,7	20,4	37,8	39,1	20,1	24,3	34,5

Figure 7: Average waiting times of a working day of Noorderbrug junction

Figure 8: Trafficlight phases of Noorderbrug junction

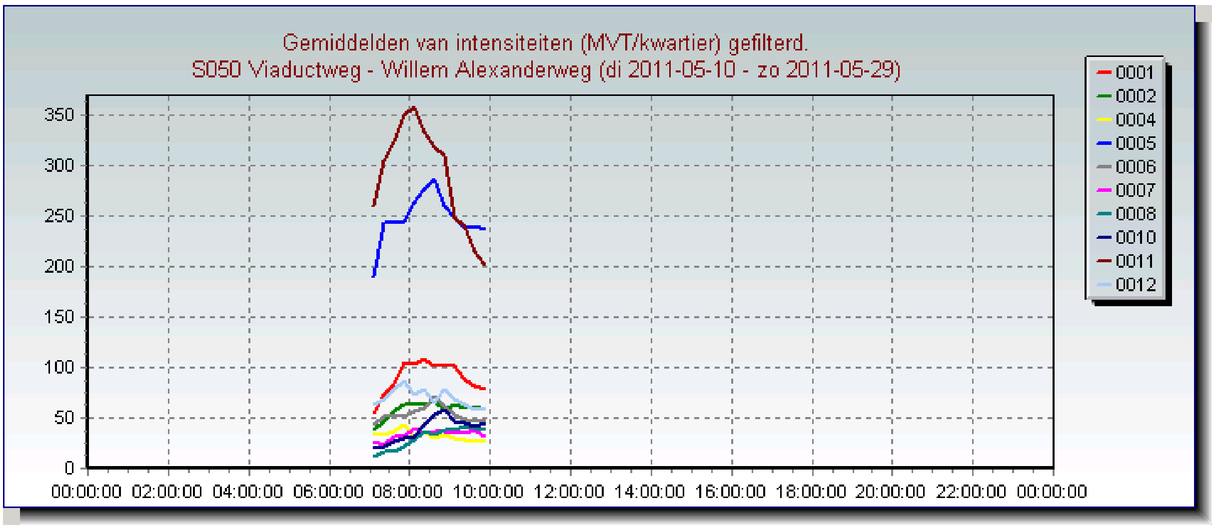

Figure 9: Traffic intensity of Noorderbrug junction