Car2X communications at intersections: Delay and emission minimizing algorithms implemented in VISSIM

Igor Moretti

Supervision:

Dr. Monica Menendez
Dr. Ilgin Guler

Master thesis
Civil Engineering **June 2014**

IVT Institut für Verkehrsplanung und Transportsysteme
Institute for Transport Planning and Systems

ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Acknowledgements

I would like to thank Dr. Monica Menendez for giving me the opportunity to write my master thesis under her supervision. Many thanks also to my supervisor Dr. Ilgin Guler, who gave me indispensable inputs, support and guidance for my thesis.

I am also very thankful to Qiao Ge, who gave me hints and support concerning the Car2X module in VISSIM. I thank also Joerg-Peter Lorch for giving me the possibility of using the server for running my simulations.

Contents

Master Thesis Civil Engineering

Car2X communications at intersections: Delay and emission minimizing algorithms implemented in VISSIM

Igor Moretti
Neumattstrasse 5
4133 Pratteln
morettii@student.ethz.ch

June 2014

Abstract

Emerging technologies such as Car-to-X wireless communications between vehicles and infrastructure have a high potential of improving efficiency at intersections compared to traditional traffic control systems. Information such as current speed and position exchanged between vehicles and infrastructure can be processed in an intelligent way in order to reduce delay or emissions. In this paper, algorithms that minimize total delay and emissions at an intersection consisting of two one-way-streets were implemented in the microscopic traffic simulation software VISSIM assuming that the vehicles are fully controlled by the system. These algorithms were compared to an adaptive traffic control system.

The results show a general improvement regarding delay and emissions compared to the adaptive traffic control system. The algorithms proposed are also suitable for more complex intersections and can be easily replaced with other optimizing criteria.

Keywords

Car2X, Adaptive Traffic Control, Emissions, VISSIM

Preferred citation style

Moretti, I. (2014) Car2X communications at intersections: Delay and emission minimizing algorithms implemented in VISSIM, *Master thesis*, IVT ETH Zürich, Zurich.

1 Introduction

A classic approach of controlling traffic at intersection is the use of traffic lights. At intersections where demand is relatively low, fixed-time traffic signals are often used, whereas at intersection with higher saturation degree, adaptive traffic control systems are applied to improve the efficiency. These well-established systems guarantee a high degree of safety, but they force a percentage of cars to stop. With every stopped car, the emissions and the delay of the intersection increase. So, safety and efficiency often contradict each other.
However, newer technologies such as Car-to-X communications are opening the door to solutions that could theoretically improve efficiency without a loss of safety. With the help of wireless communication, the vehicles can interact with each other (Car-to-Car, C2C) or with the infrastructure (Car-to-Infrastructure, C2I) and exchange detailed information like current position or speed. This information can be used to organize cars at intersections to increase the efficiency of traffic operations while maintaining a certain level of safety.

Some work has already been done towards this goal in the Traffic Engineering research group at the Swiss Federal Institute of Technology Zurich (ETHZ). However, the proposed methodology, although simple and insightful, has only been tested for deterministic driver behavior.

The aim of this thesis was to implement algorithms that minimize the delay and emissions at an isolated intersection with the help of C2X technology. These algorithms were tested with the microscopic traffic simulation software VISSIM simulating a stochastic driver behavior and compared to an adaptive traffic signal control.

The thesis is structured as follows: Section 2 gives a short literature review and explains the previous work done at the traffic engineering research group regarding this topic. Subsequently, section 3 illustrates the simulation framework and section 4 describes the implemented algorithms in detail. In section 5, the results are presented and discussed and last but not least in section 6, conclusions are drawn and recommendations for future research are given.

2 Literature review and previous work

Two papers were reviewed as basis for this work. Zohdy and Rahkha [1] proposed a heuristic optimization algorithm for automated vehicles at uncontrolled intersections using a game theory framework. The idea was to model the automated vehicles as reactive agents interacting and collaborating with the intersection controller to minimize the total delay. Two different intersection control scenarios were considered: a four-way stop control and the proposed intersection controller framework. Both simulations included four automated vehicles, i.e. a single vehicle per approach, that were simulated using a Monte Carlo simulation repeated 1000 times. The results showed that the optimization algorithm reduces the total delay by 70 percent compared to a traditional traffic control system.

Another algorithm has been developed by Lee and Park [2]. They propose a Cooperative Vehicle Intersection Control system using C2I technology for effective intersection operations assuming that all vehicles are automated. Thus, the algorithm does not require a traffic signal. The system manipulates the maneuvers of the vehicles in a way that all the vehicles can safely cross the intersection without collisions. Additionally, an algorithm that deals with system failure was designed. The simulation included a four-way single-lane approach intersection with different congestion conditions. The results showed that the algorithm improved the intersection performance by 99% and 33% of stop delay and total travel time reductions. Also, the system reduces the CO_2 emissions by 44%.

This thesis is based on the previous work done by Meier [3]. The goal of that project was to implement algorithms using Car2X communications in order to minimize total delay, stops and braking cars in an isolated intersection. The algorithms to optimize traffic are based on cumulative curves and the considered intersection consisted of two one-way streets without turning possibility. With the Car2X technology, the position and speed of each car can be determined and hence the arrival curves at the intersection can be calculated. The algorithms fit the departure curves to the arrivals to minimize delay, stops or braking cars. Since in continuous time and space there are an infinite amount of possible departure curves, the time is discretized into time steps.

The algorithms in [3] showed always an improvement in the specific property of traffic when compared to a fix-timed traffic light, regardless the traffic demand. Unfortunately, the algorithms were not compared to an adaptive traffic light. Also, the codes were implemented in MATLAB and hence the driver behavior was not stochastic. Out of the three implemented and tested algorithms, only the one minimizing delay produced satisfying results. In this thesis, only this algorithm was taken into account. Also, the algorithms were not compared in terms of emissions.

In summary, Zohdy and Rakha [1] used vehicle trajectories to eliminate conflicts and minimize delay explicitly using a game theory algorithm. Lee and Park [2] did not develop algorithms to minimize delay or emissions explicitly, although their approach resulted in an improvement. Very detailed kinematic equations were used, as their work was mainly concerned about traffic safety. Meier [3] implemented algorithms that find the best departure curve and discharging sequence. The time was discretized into time steps of 2 seconds and the vehicles were not controlled by the system and only send information about their current position and speed.

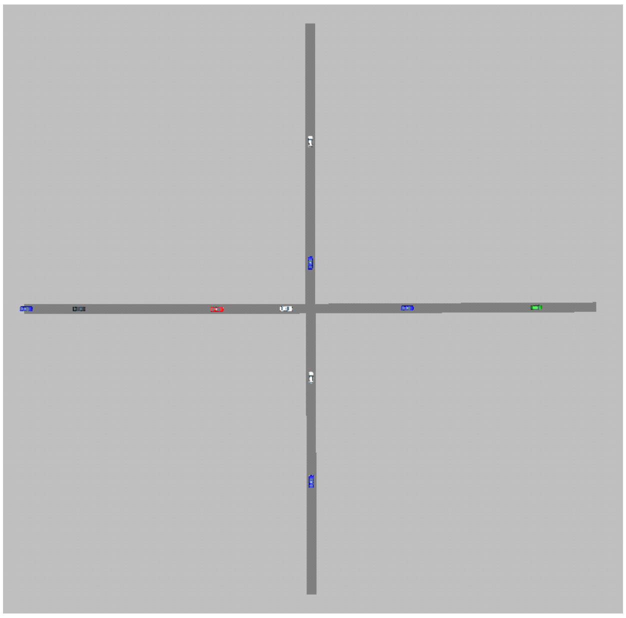
The idea in this thesis was to develop algorithms that minimize delay and emissions where the cars are controlled externally and thus allow calculating trajectories of the vehicles precisely enough without using very detailed kinematic equations like Lee and Park [2]. Also, by controlling the vehicles externally, exact arrival and departure times can be calculated in order to minimize delay or emissions.

3 Simulation framework

The intersection modeled in this thesis is exactly the same as the one designed in [3], consisting of two one-way streets without turning possibility. The symmetric geometry of the intersection is formed by 4 links each with a length of 100 m and a node with a quadratic 5 m^2 conflict zone. Hence, the range of reception reaches 100 m upstream and downstream of the intersection. The time measurement sections for the delay calculation are placed 100 m upstream and downstream of the intersection. The delay is the difference between the time a car needs to pass 200 m with free flow speed and the time it actually spends considering the intersection.

Three algorithms are compared in terms of delay and emissions: The Adaptive Traffic Control System, a minimizing delay and a minimizing emissions algorithm. An illustration of the intersection is given in Figure 1.

Figure 1 Simulated intersection



4 Algorithms

4.1 Adaptive traffic control strategy

4.1.1 Existing traffic control strategies

Overview

An overview of the different worldwide used traffic control strategies is given by Aguigui and Chan [4]. Common to every system is a central controller that processes the traffic data from detectors with a specific algorithm. The coordination is then performed with real time adjustments using a traffic model and degrees of saturation and the goals are typically minimization of queues, delays and stops. The variables that are adjusted are usually phase splits, cycle length and offset. Some of the control strategies may have a backup fixed plan. The adjustments are based on prediction of the arriving traffic (queues, size and approach of platoon, turn percentages, arrival times etc.). The most frequent used systems worldwide are summarized in Table 1.

As SCATS is one of the most used and best documented systems, it was chosen as a reference for the adaptive traffic controller used in this thesis. SCOOT works similarly concerning the optimization of the cycle length.

Table 1 Overview of adaptive systems [4]

Adaptive system	Goal	Process
SCOOT	Minimizes delay with relative importance on stops	Uses flow profiles, calculated delay and saturation level with regular small change
SCATS	Minimizes stops, delay and travel times	Uses Degree of Saturation and Car Equivalent flow for each approach lane
ATCS	Adjusts timing on a cycle-by-cycle Basis	Optimizes cycle, splits and offsets
OPAC	Minimizes a Weighted Performance function for The system based on Delays and Stops	Uses Flow Profiles
RHODES	Proactively responds to and utilize the natural Stochastic variations in traffic flow	Uses turning percentages based on O-D estimate and queue discharge rates
InSync	Services movement stages to minimize queues and delays	Uses queue lengths, volumes and occupancy to optimize time tunnels
ACS Lite	Adjusts splits and offsets on a cycle by cycle basis	Uses DS for split optimization and flow profiles for offset optimization

SCATS

SCATS (Sydney Coordinated Adaptive Traffic System) basically manages three main parameters to achieve traffic signal coordination [5]: Cycle time, phase split (proportion of cycle time allocated to each phase) and offset.
Using flow and occupancy data collected from stop line loop detectors [6], the system determines the optimum cycle length, phase splits and offsets to suit the actual traffic conditions. Therefore, green phases can be terminated early or omitted entirely from the sequence if the demand is low.
The adjustments are based on a measure called “Degree of Saturation”. It represents how effectively the road is being used. With the help of in-ground loop detectors at the intersection, flow and occupancy data is collected during the green phase. The data is then used to calculate the degree of saturation (effective used green time divided by total green time). If a lane approaches saturation, the system will respond to the situation with a change of the parameters.
The cycle length is increased or decreased to maintain the degree of saturation around 0.9 on the lane with the greatest saturation. The maximum range of the cycle time goes from 20 s to 240 s, but different lower and upper limits can be defined by the user. The cycle time can vary up to 21 s.
The green times are varied by a small amount each cycle in order to maintain equal degrees of saturation on competing approaches, respecting a minimum green time.
The calculated cycle times, phase splits and offsets are then compared to a library of possible combinations, where the system automatically chooses the best one [5].

SCOOT

SCOOT (Split, Cycle, Offset Optimization Technique) performs an optimization at three levels [7]. The system measures the amount of vehicles with a detector placed at least eight seconds of travel time upstream the intersection. With this detector, the profile of the arrivals can be estimated. The arrival profile is compared with a departure profile based on saturation occupancy from onset of green till the queue is cleared. The difference between the two profiles represents the vehicles delayed in a queue.
The three levels of optimization include split, cycle and offset optimization. The split optimizer evaluates the arrival and departure profiles every second. Five seconds before each change of signals within the cycle, the system adds the delay from all movements that will

end or begin at that change. Afterwards, this delay is compared with delay calculated with the change of signals. The scenario that produces the best balance of delay will be chosen.
The cycle optimizer checks the saturation levels of all intersection movements every 2.5 to 5 min. If the saturation of the heaviest movements at the intersection exceeds 90 %, the cycle optimizer will add an amount of seconds depending on the length of the cycle. If the saturation is much less than 90 %, the corresponding amount of seconds is subtracted from the cycle.

4.1.2 Implemented strategy

As in this thesis the intersection is basic and there are only two phases, it was decided to implement an own simplified SCATS-like algorithm using the principles of the system.

First, a fixed-time program is calculated manually in order to have a backup whenever the parameters reach too high or too low values. For the optimal cycle time, Webster [8] proposed an equation for the calculation that seeks to minimize the delay. The formula is the following:

$$C_{opt} = \frac{[(1.5L) + 5]}{\left(1.0 - \sum_{i=1}^{n} Y_i\right)}$$

Y_i is the arrival rate divided by the saturation rate of approach i. For all the simulations, a lost time of L = 5 s was assumed. The optimal green times $G_{i,opt}$ are then determined using the following relation:

$$G_{i,opt} = \frac{\lambda_i}{\mu} * C_{opt}$$

λ_i is the arrival rate of the approach. A saturation flow of μ = 1800 veh/h was used. With these relations, a fixed plan was established for all demand combinations by rounding the cycle and green times and considering the lost time. The plans are shown in Table 2.
During a green phase, a stop line detector determines the amount of discharging vehicles n_i. If the degree of saturation DS_i is exceeded (>90%), the green time is raised by 10% in the next cycle. If the degree of saturation is very low, the green time is reduced by 10% in the next cy-

cle (until the minimum green time of 10 s is reached). The new cycle length is then simply the sum of the green times plus a lost time of 5 s. The algorithm uses the following formulas:

$$DS_i = \frac{n_i}{G_i * \mu}$$

$$C = G_1 + G_2 + L$$

To keep the cycle length around its optimal value, the time can be modified by maximum 20% of its optimal value. To not reach this value too soon, the green times are modified by only 10% per cycle.
If this limit is reached, the system resets the cycle time and the green times to their values in the fixed-time program and the optimization restarts again.
An important feature is also the expansion of the green time over the whole cycle if there are no cars on the other approach.

Table 2 Fixed time plan

Total flow [veh/h]	Flow ratio [veh/h]	C [s]	G_1 [s]	G_2 [s]
	900/100	35	20	10
	800/200	35	20	10
1000	700/300	35	20	10
	600/400	35	20	10
	500/500	35	15	15
	1350/150	75	60	10
	1200/300	75	55	15
1500	1050/450	75	50	20
	900/600	75	40	30
	750/750	75	35	35

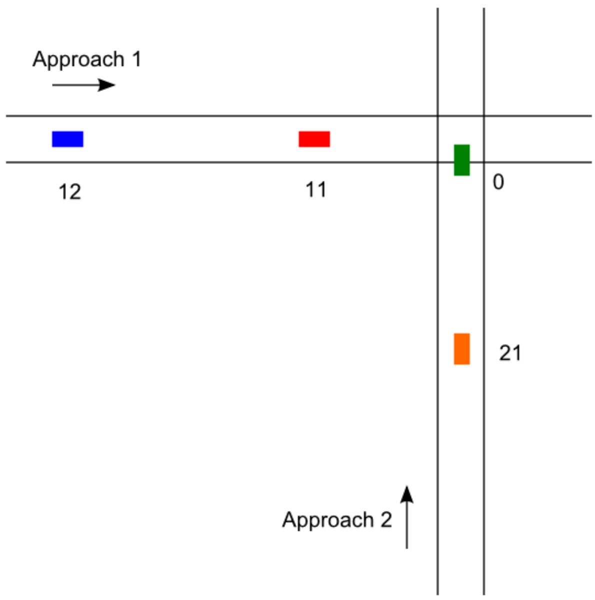
4.2 Delay minimizing algorithm

The delay minimizing algorithm basically tries to find the discharging combination which produces the smallest amount of total delay. With the Car2X module included in VISSIM, vehicles can send and receive information such as speed and position. Hence, the algorithm makes use of vehicle trajectories in order to find the best solution. The code, written in Python and read by the Car2X module in VISSIM, can be divided into 6 steps, which are executed in the listed order below every time step (in this case every 0.1 s):

Step 1: Scanning all vehicles on the network
Step 2: Updating the list of desired arrival times
Step 3: Producing all possible discharging combinations
Step 4: Calculating the new trajectories of each vehicle for each combination
Step 5: Calculating the total delay for each combination
Step 6: Select the best one and send the information of the updated speeds to each vehicle

Step 1: First, all the vehicles upstream the intersection and on the conflict zone are scanned and their actual speed $v_{a,mn}$, and actual distance to the stop line of the intersection $d_{a,mn}$ are determined, where m is the approaching direction (in this case 1 or 2) and n is the position of the car in the sequence of the respective approach. To make the explanation of the algorithm comprehensible, the steps are accompanied by an example. So let us say that in the first step, the situation illustrated in Figure 2 is recorded.

Figure 2 Approaching situation



The vehicle on the conflict zone has the number 0 and as it has already passed the stop line of the intersection, it is not considered in the combinations, but its speed is recorded for step 2.

Step 2: In the next step, a list of desired arrival times $t_{d,mn}$ of the vehicles is updated. As soon as the system detects a new vehicle (i.e. when it reaches the 100 m zone), its desired arrival time at the stop line of the intersection is calculated simply by:

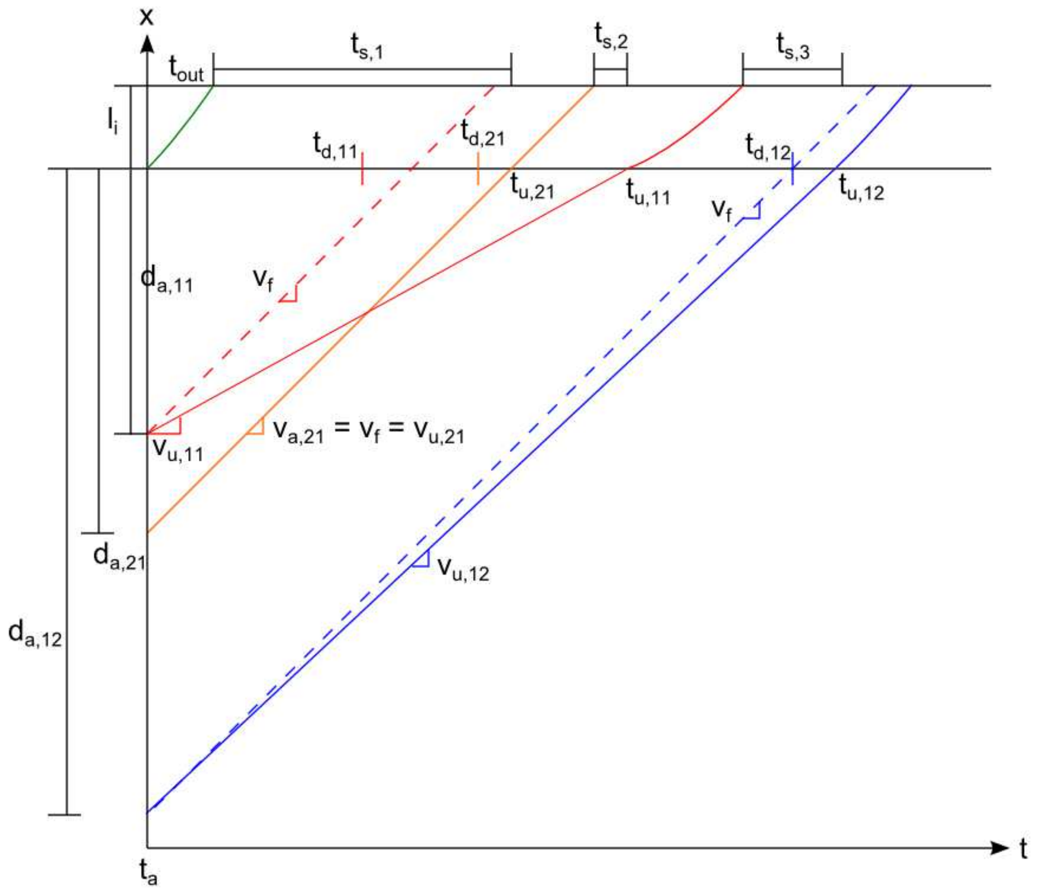
$$t_{d,mn} = t_a + \frac{d_0}{v_f}$$

Where t_a is the actual time, d_0 the distance from the stop line when the car is detected the first time and v_f the free flow speed (in this case 50 km/h).
Also, as soon as a car enters the conflict zone (see car number 0), its speed is used to calculate the time t_{out} it will leave the intersection knowing that the vehicle will accelerate with 4 m/s^2 till it reaches v_f.

Step 3: In step 3, all the possible discharging combinations are determined. In our example, these are [1, 1, 2], [1, 2, 1] and [2, 1, 1].

Step 4: The next step consists in calculating the new trajectories of each vehicle for each combination. Considering the situation depicted above, Figure 3 shows the time-space diagram for calculating the trajectories of the discharging combination [2, 1, 1].

Figure 3 Time-space diagram of combination [2,1,1]



The green car already passed the stop line and the time t_{out} when it leaves the intersection with the length l_i is known. The algorithm then checks if the orange car can travel with free flow speed v_f considering a safety time t_s. Let us say that in this case it is possible ($t_{s,1} > t_s$). Now, the red car comes after the orange one, but it cannot travel with free flow speed, but has to reduce the speed in order to enter the intersection after the orange vehicle has left the conflict zone and considering a safety time t_s . Its updated speed $v_{u,11}$ is set to arrive at the stop line at $t_{u,11}$. The blue car also has to update his speed. So for each combination, each car gets an updated speed $v_{u,mn}$ for the next time step.

A constraint was set that only one car can be in the conflict zone at a time. Plus, for a car discharging after a vehicle of the same approach has to consider a safety time t_s of 2 s to ensure a minimum headway. If the car crossing the intersection follows a vehicle of the opposite approach, the safety time t_s was set to 1 s, as the vehicles are going in different directions and thus the headway does not matter. This means that the capacity of the intersection depends on the discharging sequence and combinations with alternating approaching vehicles are favored. This was implemented on purpose, since the cars are controlled externally and the safety is theoretically ensured. The effects of this rule will be discussed later in the results section.
Note that the updated speeds are average speeds. For the delay, what matters is at the end the updated arrival time at the intersection. The algorithms also technically exclude stops, as $v_{u,mn}$ can never be zero.

Step 5: In step 5, for each combination, the total delay is calculated by subtracting the desired arrival times $t_{d,mn}$ (see Figure 3) from the updated arrival times $t_{u,mn}$ and summing them up. The delay generated downstream the intersection is also considered, as the vehicle has first to accelerate with 4 m/s^2 from the speed it arrives at the intersection to v_f.

Step 6: Knowing the total delay for each combination, the best one is selected and the information of the updated speeds is sent to the corresponding vehicles.

4.3 Emission minimizing algorithm

The emission minimizing algorithm uses the same framework as the delay minimizing algorithm. Basically only step 5 changes and step 2 is not necessary. After scanning all the vehicles on the network, it produces all the possible combinations. The new trajectories are determined in the same way as in the delay minimizing algorithm. In the following step, instead of calculating the total delay for each combination, the sum of the speed changes $\Delta v_{mn} = v_{u,mn} - v_{a,mn}$ is calculated.

Only decelerations are considered as penalizing. The idea of using speed changes as a criterion for minimizing emissions comes from the fact that reducing the speed is equal to losing kinetic energy that has to be restored once the car is accelerating to free flow speed again.

5 Results

5.1 Simulation

Each simulation was conducted five times with a simulated time of 30 min using different seeds and the values shown later represent the mean value of these five times. Different flow ratios were used for total flows of 1000, 1500 and 2000 veh/h. During the tests, it was assumed that each car was equipped with a Car2X module able to send and receive information.
The information between the cars and the system is exchanged with a frequency of 10 Hz, i.e. ten times per second.
It would be obviously more precise to measure the delay on a longer road section, but this would slow down considerable the computation speed, as more cars and combinations have to be considered in the algorithms. Only the section where the delay is effectively minimized is measured.

5.2 Delay

5.2.1 Total delay

The resulting total delays for a duration of 30 min are shown in Figures 4-6. Taking a look at the results of the simulations with a total flow of 1000 veh/h, it is recognizable that the delay increases with decreasing flow ratio in the case of the traffic signal and stays almost constant in the case of the algorithms.
One clear trend observable: The difference between the signalized intersection and the minimizing algorithms increases with increasing flow ratio. An explanation for that could be that the low demand on one approach only activates the green phase when a car is approaching. So, most of the time, the approach with the higher demand disposes of the whole cycle length to discharge. But the main reason is probably that the minimizing algorithms will alternate the discharging directions as explained in section 4.2, so the intersection has more capacity and the total delay does not increase with decreasing ratio.

The difference of delay between the delay minimizing algorithm and the emission minimizing algorithm is almost zero.

Looking at the simulation with 1500 veh/h, the delay also increases with increasing ratio for the signalized case. For the algorithms, a higher decrement than in the 1000 veh/h case is observable. The reason is the same as explained above. Note that the difference for the 1350/150 ratio is very small, meaning that the simpler adaptive traffic control system is almost as efficient as the minimizing algorithms The trend observed in the 1000 veh/h example is also noticeable here. The difference between the two minimizing algorithms is small also in this case.

In the simulation with 2000 veh/h, tests with the traffic signal were not conducted, as the intersection is saturated. As already explained above, the capacity of the intersection in the case of the algorithms can reach almost 3600 veh/h if the combination is always alternating. Thus, tests were performed for the two systems. Looking at the curves, the total delay decreases lightly in the last two ratios, but it can be considered as constant over all flow ratios.

For all three total flows, the delay generated by the emission minimizing algorithm is slightly higher than the delay minimizing algorithm, but the difference is almost negligible.

Figure 4

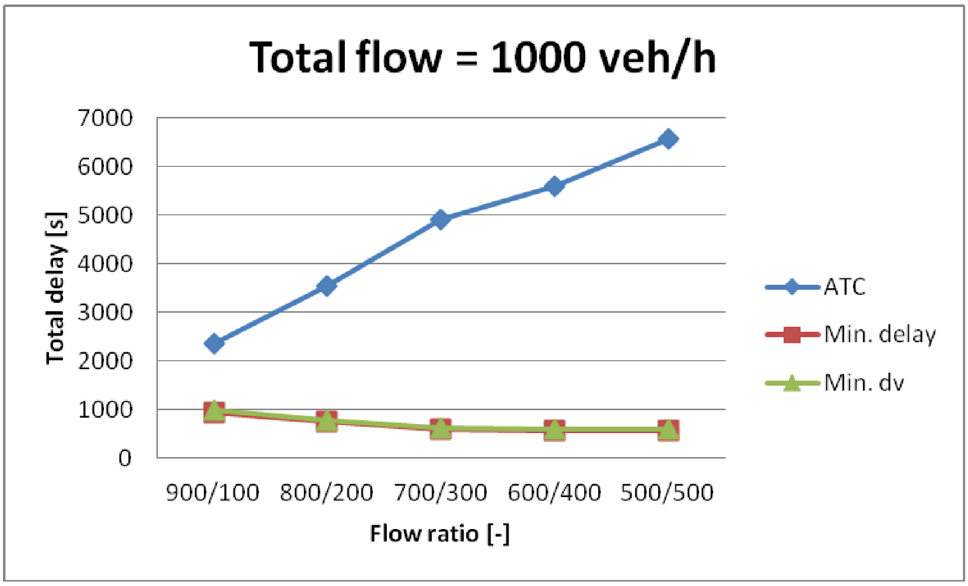


Figure 5

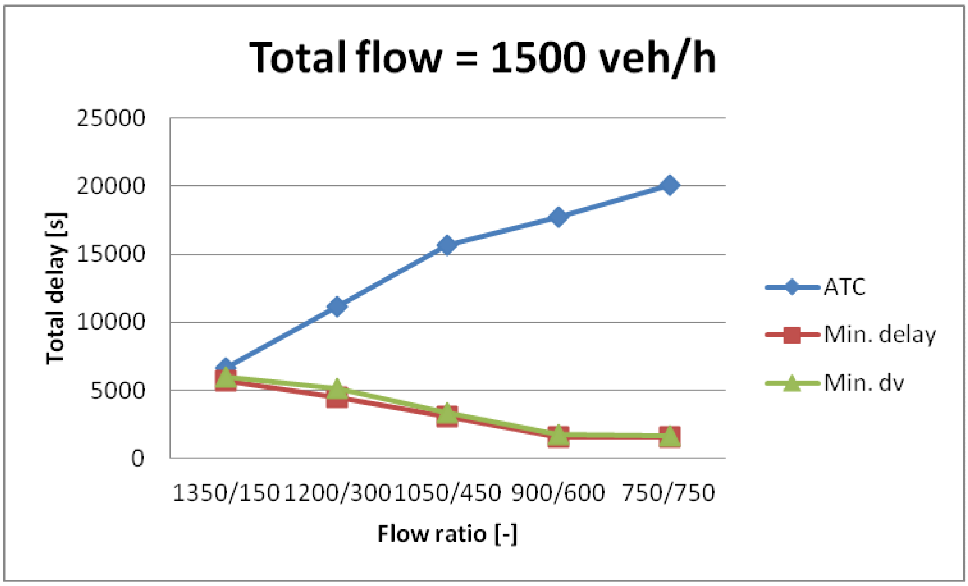
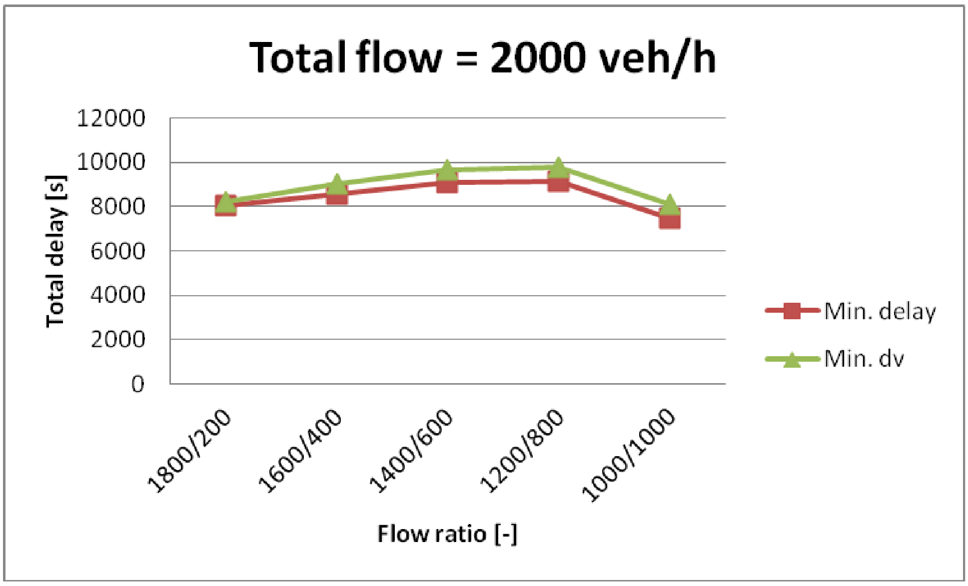


Figure 6



5.2.2 Average delay

By taking a look at the average delays per car (Figures 7-9), the shape is obviously the same as the one from the total delay. For the 1000 veh/h case, the average delay lies around 1.5 s for the controlled systems, which is very low and optimistic, but comprehensible since the car are controlled externally and the capacity increases with decreasing flow ratio.
The average delay for a total flow of 1500 veh/h lays between 10 s and 2 s for the algorithms and increases over 25 s in the ATC. For 2000 veh/h, it lays around 8 s to 12 s.

Figure 7

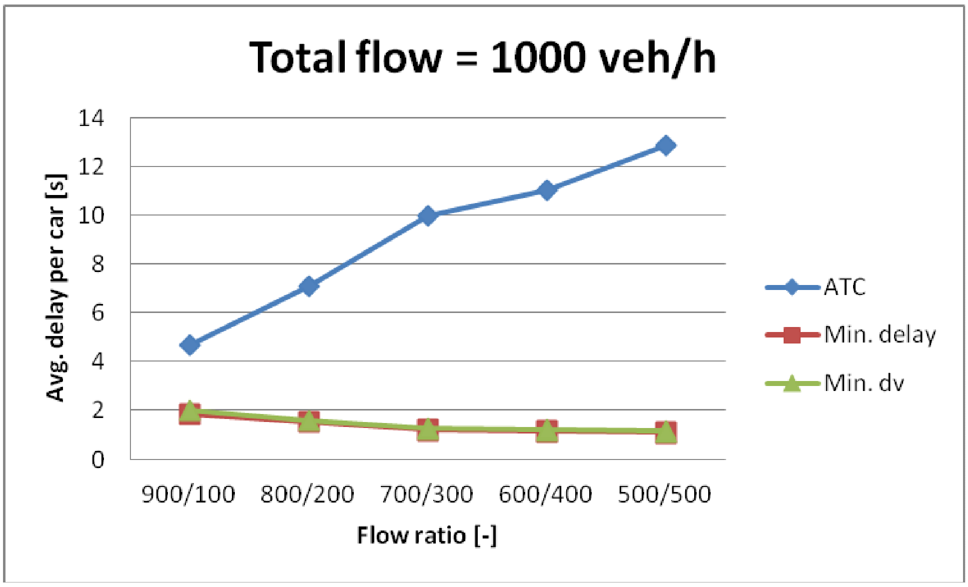


Figure 8

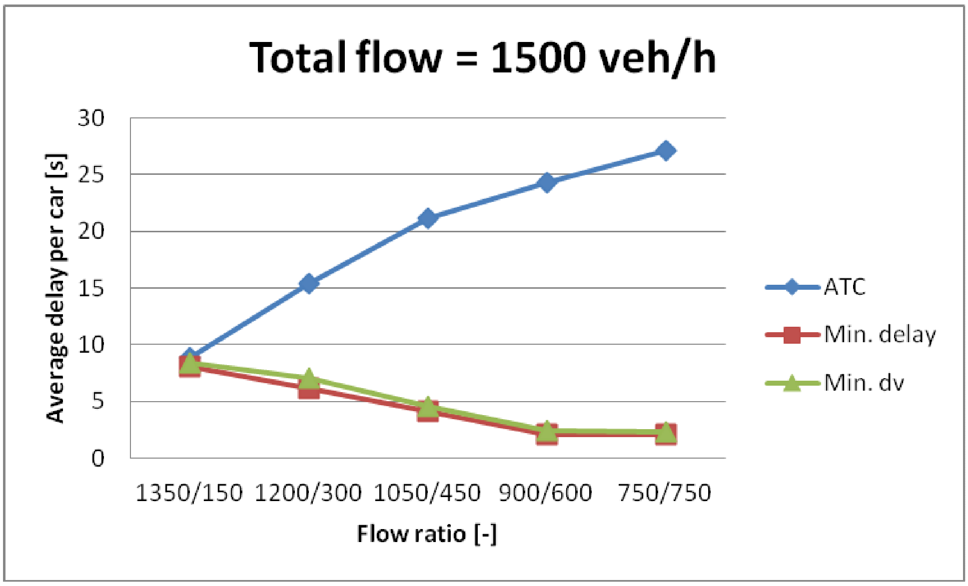
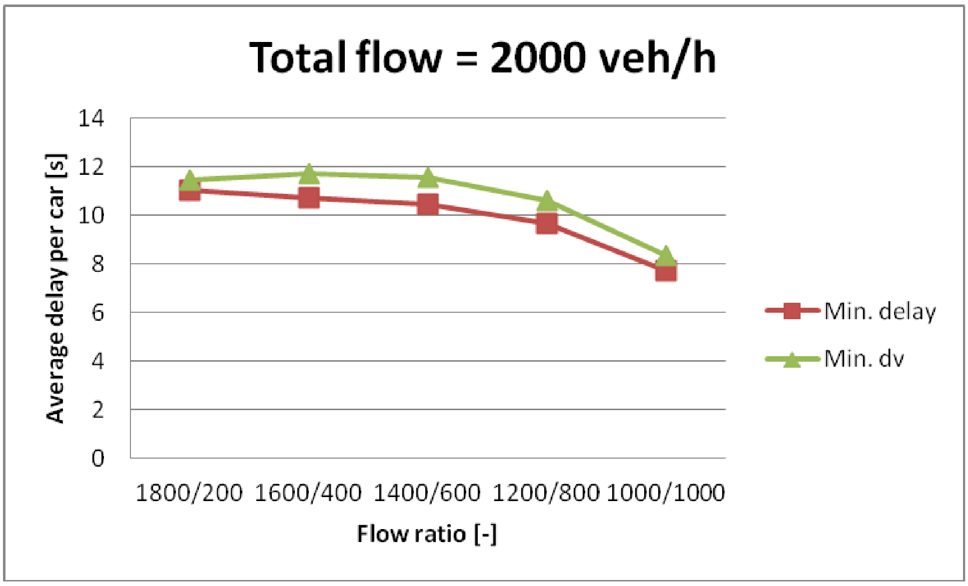


Figure 9



5.2.3 Standard deviation

The standard deviation (Figures 10-12) follows a similar trend as the average delay. The values are quite small for the controlled system and larger for the signal-controlled traffic. The controlled systems distribute the total delay very equally among all the vehicles, whereas the delay for a single vehicle at the traffic light depends on the arrival time of the car and the respective phase it runs across.

Figure 10

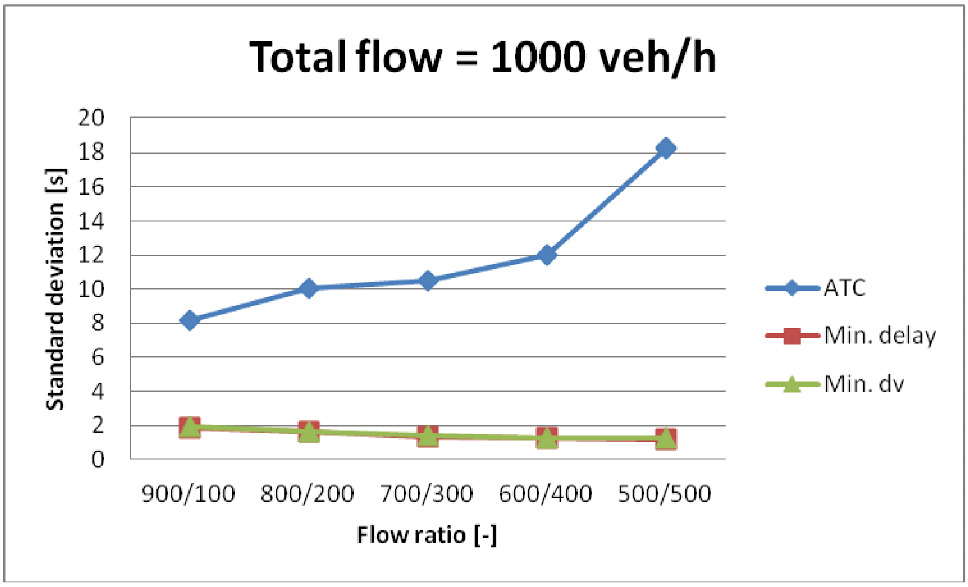


Figure 11

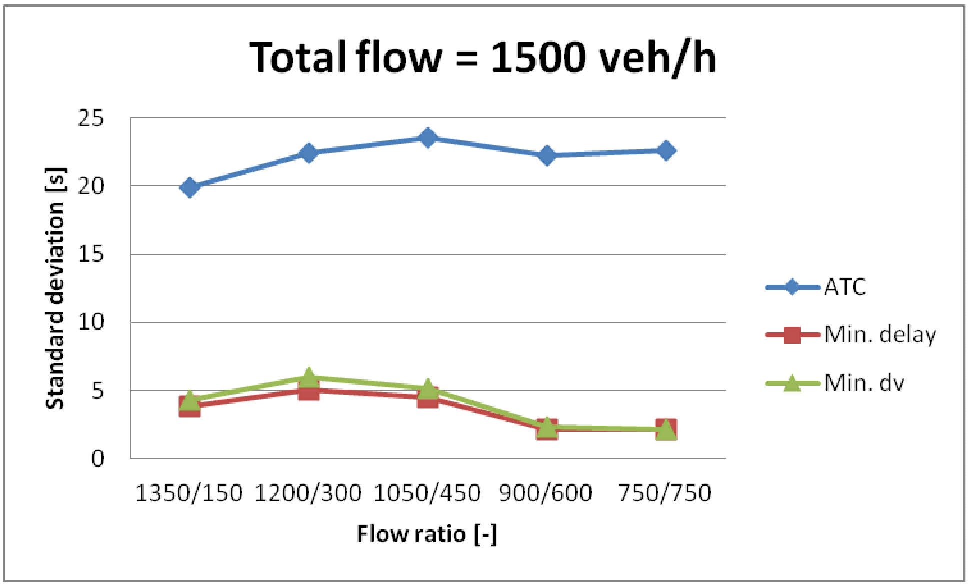
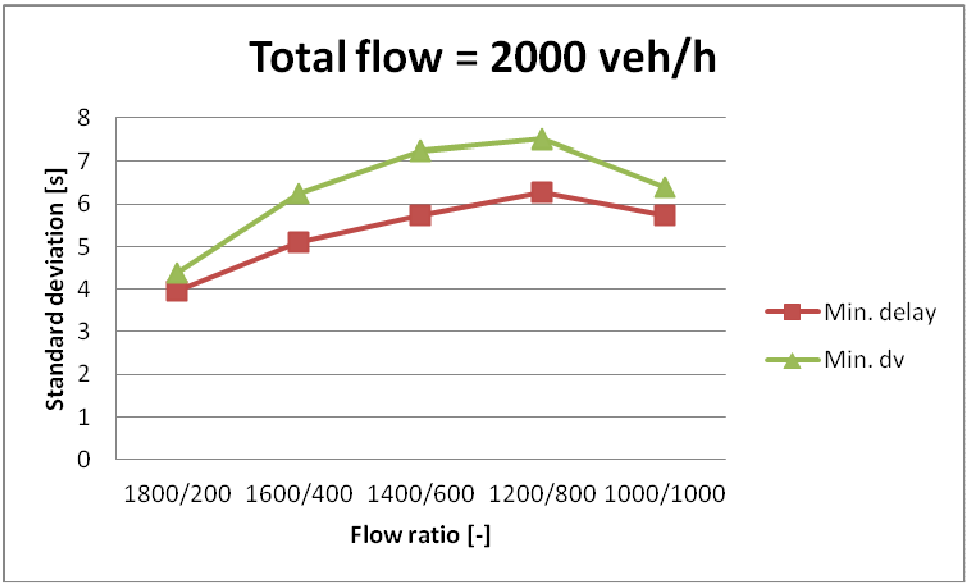


Figure 12



5.3 Emissions

The emissions were calculated using a tool called “EnViVer” developed by TNO, the organization for applied scientific research in Delft, Netherlands. The tool is based on the VERSIT+ emission model, which consists of different modules that calculate the emissions based on speed and acceleration of the simulated vehicles. The reference emissions per car are based on typical vehicle composition on European roads. The traffic emissions that can be calculated include CO_2, NO_x and PM_{10}. [9] Since CO_2 is by far the most important, only this emission is shown in the results.

The emission curves (Figures 13-15) follow also a similar trend as the delay. A relation between the two dimensions is traceable. The scale though is smaller than in the total delay comparison: Where the delay could be reduced till a factor of 10 in the extreme case, the emissions for the ATC are maximum twice the value of the minimizing algorithms.

The emission minimizing algorithm has almost the same values as the delay minimizing algorithm. The reason is that along with sharing the same framework, minimizing the Δv has in the majority of the cases the same effect as minimizing the delay. Obviously, it depends also on the position of the car, as the same speed difference causes a larger delay if applied early on the approach rather that just before the intersection. But in the most cases, the combinations picked are probably the same. So only the small amount of different combinations makes the negligible difference both in the delay and in the emissions. Nevertheless, compared to the adaptive system, a relevant improvement is observable.

Figure 13

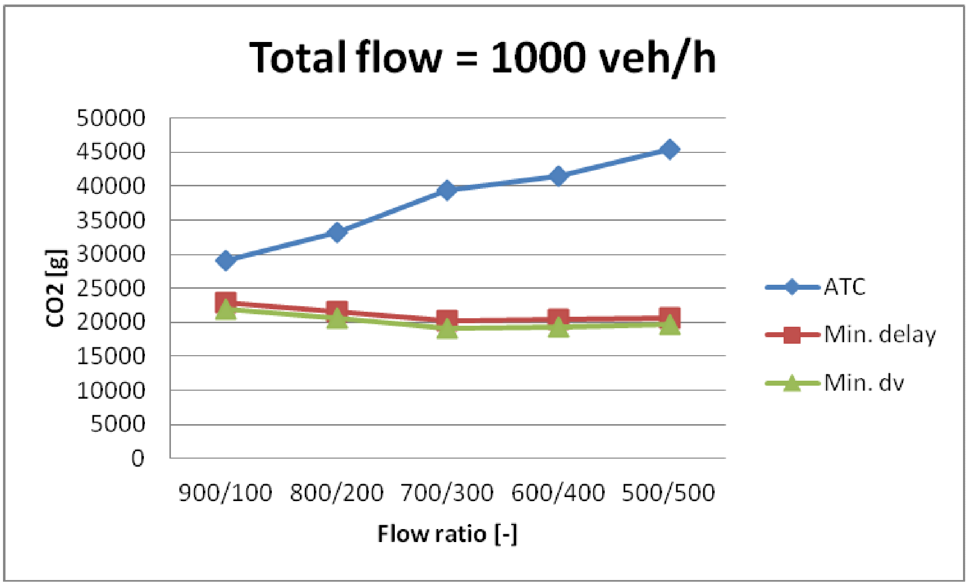


Figure 14

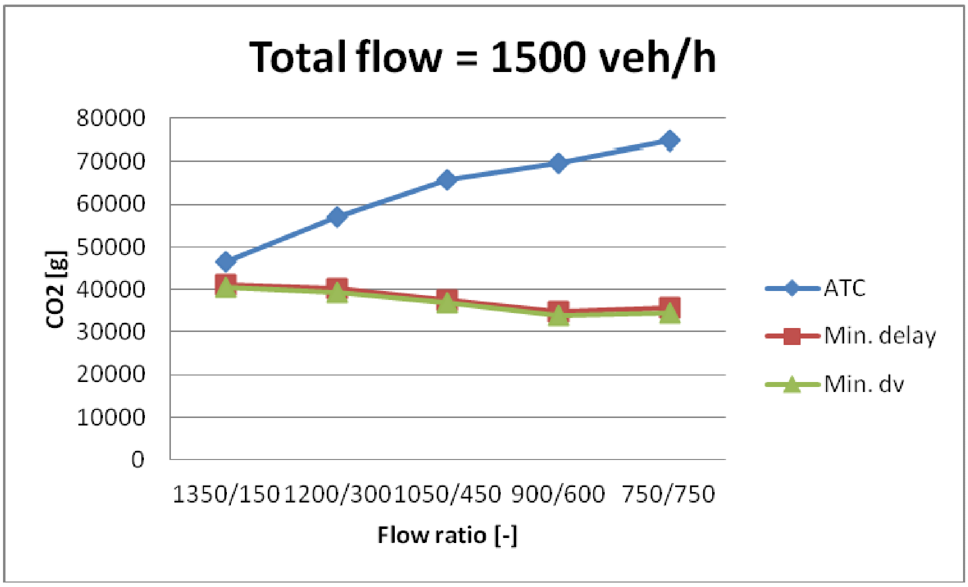
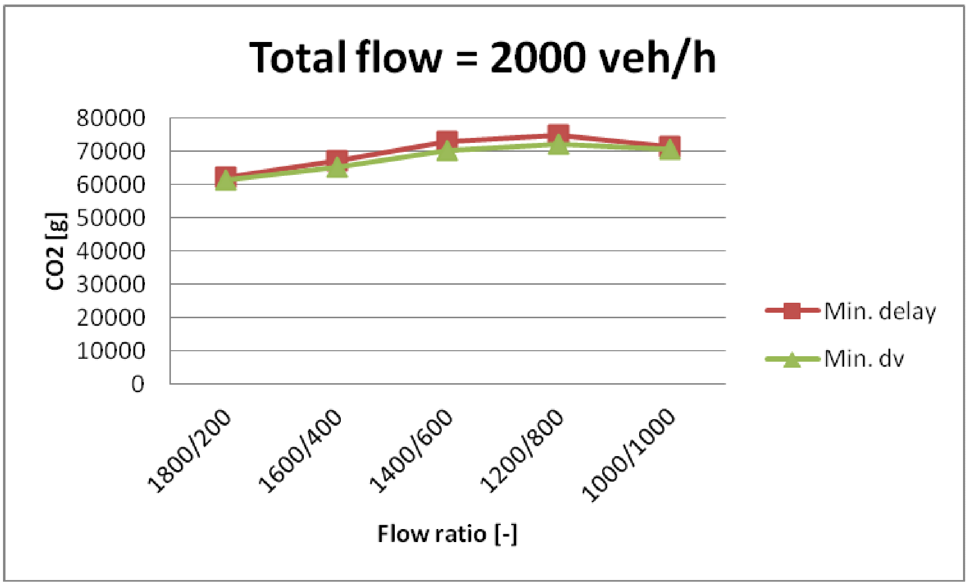


Figure 15


Total flow = 2000 veh/h
80000
70000
60000
50000
40000
30000
20000
10000
0
CO2 [g]
1800/200
1600/400
1400/600
1200/800
1000/1000
Flow ratio [-]
Min. delay
Min. dv

6 Conclusions and future work

The results show a general improvement regarding delay and emissions compared to an adaptive traffic control system. Especially with similar demand on both approaches, the presented algorithms produced better results. If the flow ratio is high, an adaptive traffic control system is probably the better choice.

However, many assumptions were made to generate such optimistic results: The capacity increasing with small flow ratios surely has the most relevant impact on the results and is also contestable, as the safety decreases by favoring a switching of discharging directions. By controlling the vehicles from the system, the safety was assumed to be guaranteed, but no backup plan if a system failure occurs has been implemented.
Comparing all the possible combinations is a very good approach as already shown in Meier's work [3], but might be computationally extensive for more complex intersections. In this case, since the intersection was very simple, the algorithm was repeated every 0.1 s to guarantee a high level of resolution. Obviously, such a high level is not necessary and with more complex intersections, it would be more reasonable to reduce the updating frequency.
Controlling vehicles externally improves surely the efficiency of an intersection, but it is still a state-of-art solution and all vehicles were equipped with a Car2X module during the simulation.

The emission minimizing algorithm was based on a too simple assumption and as the results showed, it was almost equally efficient as the delay minimizing algorithm. If further research is desired, it is recommended to use a more detailed scoring of the combinations. Theoretically, the vehicles equipped with a Car2X module could share the information about the specific emissions of the vehicle and thus facilitate the selection of the best combination, but this could mean that vehicles with low emissions could be penalized in terms of delay as they produce less CO_2. Also, existing emission models could be used in order to predict the sequence with the lowest emissions, but these detailed models often need more information such as mass of the vehicle etc. But looking at the results lets assume that minimizing the delay should also have a positive effect on the emissions in the majority of cases.

Nevertheless, the implemented algorithms are a good basis for further research. The algorithms should work also with more links, since the approach of using the trajectories is appli-

cable. Another advantage is that if tests with other criteria than minimizing delay or emissions are desired, only the section with the “scoring” has to be modified.
The idea of increasing the capacity might be risky, but can also be used as a “buffer” at intersections with changing demands. For example during rush hours, the capacity can be increased for a short-time by reducing t_s.

For further research, it is recommended to improve the emission minimizing algorithm and use more detailed criteria for selecting the best combination. Also, tests concerning computational speed and hardware requirements have to be done to check feasibility with more complex intersections.

7 References

[1] Zohdy, I. and H. Rakha (2012) Game theory algorithm for intersection-based cooperative adaptive cruise control (CACC) systems, *Research paper*, 15^{th} International IEEE Conference on Intelligent Transportation Systems, Anchorage.

[2] Lee, J. and B. Park (2012) Development and evaluation of a cooperative vehicle intersection control algorithm under the connected vehicle environment, *Research paper*, IEEE Transaction on Intelligent Transportation Systems Vol. 13.

[3] Meier L. (2013) Use of car2car communications to improve efficiency of intersections, *Master thesis*, Institut für Verkehrsplanung und Transportsysteme IVT ETH Zürich, Zurich.

[4] Aguigui K. and S. Chan (2009), Traffic Controller Technology, *Presentation*, Kimley-Horn and Associates Inc., Cary NC.

[5] Roads and Maritime Services (2012) SCATS 6 – Functional description, *Documentation*, Roads and Maritime Services, Sydney.

[6] McMillan S., G. Koorey and A. Nicholson (2010) Techniques for using SCATS as an incident management tool, *Research paper*, IPENZ Transportation Group Conference, Christchurch.

[7] Chan, P. (1997) Advanced Transportation Management Technologies, *Reference guide*, Federal Highway Administration, Washington DC.

[8] Webster, F.V. (1958) Traffic signal settings, *Research paper*, Great Britain Road Research Laboratory, London.

[9] Eijk, A., N. Ligterink and S. Inanc (2013) EnViVer 4.0 Pro and Enterprise Manual, *Manual*, TNO, Delft.