Capacity of cyclists at a signalized intersection

An analysis of scenarios that can happen during the queue discharge

by

Janine Timmerman

Bachelor Final Project
at the Delft University of Technology,

Student number:	4831578
Project duration:	April 19, 2022 – June 15, 2022
Supervisor and examiner:	Dr. Ir. Y. Yuan
Examiner:	Dr. D. C. Duives

TU Delft

Preface

This thesis is written for the bachelor final project at the end of the civil engineering bachelor. It is part of the transport and planning track and it will provide information about the microscopic traffic flow behaviour of cyclists before a signalized intersection. This thesis is suitable for anyone who wants to learn more about specific scenarios that can affect the maximum queue discharge rate of cyclists at an intersection.

To complete this project, I had help from several people. I want to thank my supervisor Dr. Ir. Y. Yuan for his constructive feedback every week and Dr. D. C. Duives for her feedback during the mid-term presentations. Lastly, I want to thank my peers who reviewed my work weekly.

Janine Timmerman
Delft, June 2022

Summary

The infrastructure in the Netherlands is becoming more and more focused on more sustainable forms of transport, like public transit or cycling. This increases the need for a good cycling infrastructure, especially around the intersections. Intersections are a bottleneck, so a high capacity at that point is crucial. Before it is possible to adapt the design of the intersection, it is import to know what factors affect the capacity of an intersection. At the moment, not much is known about the microscopic flow characteristics of cyclists, because most of the research done is focused on cars.

This thesis will aim to answer the following research question: *How can the capacity of cyclists be determined at the stop line of a signalized intersection?* This will be done with the help of a literature study and by analyzing an available data-set that contains the trajectories of cyclists during the queue discharge. Here, the queue discharge means the cyclists leaving the cycle path and passing the stop line when the light goes green. The capacity is influenced by several conditions. In this data-set, specific scenarios will be sought that might influence the maximum queue discharge rate. These scenarios do not totally represent any of the conditions the capacity is dependent on, so this thesis will be talk about the maximum queue discharge rate instead of the capacity when looking at those scenarios. The scenarios will be quantified and the maximum queue discharge rate of these scenarios will be estimated, to analyse the scenarios. The estimation of the maximum queue discharge rate will be done based on the composite headway model. The time headway is the time that passes between two cyclists when they cross a certain point, which is the stop line before the intersection in this case. The composite headway model states that the total headway distribution is the sum of the constrained headway distribution and the free flow headway distribution. A cyclist with free flow has all the space it needs, while a constrained cyclist is stuck behind another cyclist.

The capacity is dependent on several conditions: the base conditions, the roadway conditions, the traffic conditions and the control conditions. The scenarios analyzed in this thesis are part of the traffic conditions. Based on the available data-set, 6 scenarios were analysed. It was determined that the maximum queue discharge rate decreases when the front of the queue is very dense, but increases when the first cyclists take a long time before leaving. No definite conclusions could be made about cyclists overtaking each other, the density at the back of the queue and the effect of cyclists standing next to each other instead of behind each other.

The most interesting results are the importance of the first cyclists in the queue and the negative effect the density at the front of the queue seems to have on the maximum queue discharge rate. Even though previous studies found that increased density has a positive effect on the queue discharge.

These results help in understanding the randomness in capacity estimations and give more insight in how the microscopic traffic flow characteristics affect the maximum queue discharge rate.

Future research can focus on improving the design of intersections using these results. Other research options are to further analyze the scenarios to see if these results are representative for different intersections or to further analyze the effect of the first cyclists on the maximum queue discharge rate.

Contents

1
Introduction

1.1. Background information

Infrastructure in the Netherlands is becoming more and more focused on public transport, cyclists and pedestrians, instead of being centred around cars. This increases the need for a good infrastructure, that can handle the increasing number of cyclists. In other words, the capacity of the roads has to be increased. When cyclists approach an intersection, they first have to decelerate and stop for the red light. More cyclists arrive and they form a queue. After some time the light goes green and the cyclists have to accelerate again and cross the stop line. This last part is called the queue discharge. This takes more time than simply cycling on a straight road, making an intersection a large bottle neck in the road network. That is why it is especially important to increase the capacity of intersections. At the moment, not much is known about the microscopic flow characteristics of cyclists, because most of the previous research done is focused on cars. This means there is a lot of room for more research.

According to Hoogendoorn and Knoop (2013), the capacity is often defined as: “The maximum hourly rate at which people or vehicles can reasonably be expected to traverse a point or uniform section of a lane or roadway during a given time period (usually 15 minutes) under prevailing roadway, traffic and control conditions.” The capacity is not a constant, but varies due to several factors. It is a characteristic of the infrastructure in combination with several factors. The capacity can be calculated by taking the reciprocal of the mean time headway, in which the time headway is the time that passes between the rear bumpers of two vehicles, measured at a cross section. This is under the condition that all vehicles are following each other and there is no free flow. For cars, this can be done relatively easy because there will be only one car in each lane at a cross section. Cyclists, however, move a lot more heterogeneous. This makes it difficult to determine the capacity for cyclists.

1.2. Research question

The main research question will be:

How can the capacity of cyclists be determined at the stop line of a signalized intersection?

This question will be answered by looking at the microscopic and macroscopic flow characteristics of of the bicycle traffic flow before the stop line at a signalized intersection. The microscopic traffic flow characteristics give information about the individual behaviour of a cyclist, while the macroscopic traffic flow characteristics give information on the average state of all cyclists (Hoogendoorn & Knoop, 2013). A large focus will be placed on specific ’scenarios’ that can happen during the queue discharge and how these scenarios affect the maximum queue discharge rate. These scenarios will mainly have something to do with the microscopic flow characteristics.

To answer this research question, several sub-questions will be answered. When looking at the capacity, it must first be clear how the capacity is defined. For that it is also important to know which conditions influence the capacity at the signalized intersection. Based on literature study, a method must be found to determine an estimate of the capacity of the intersection. This first part of the study will provide a more general answer to the main research question and will be based on literature research.

Because much is still unknown about the microscopic traffic flow characteristics of cyclists, the research will mainly focus on specific scenarios that can happen during the queue discharge. Even though these scenarios will not give a complete idea of the all the factors that affect the capacity, they can help to better the understanding of all the different things that can affect the capacity at a signalized intersection. These scenarios will not accurately reflect all the different conditions that affect the capacity, but will be part of one of those conditions. This is done to limit the scope of this thesis. That is why this thesis will talk about the maximum queue discharge rate when looking at a specific scenario instead of talking about the capacity.

The literature study will also serve as an inspiration for different scenarios to analyze. After several scenarios are identified that can affect the maximum queue discharge rate, the scenarios will be looked for in the data-set and their effect on the maximum queue discharge rate will be analyzed with the help of the method that is found to determine an estimate for the capacity.
This plan is summarized in the following sub-questions:

- How can the capacity at a signalized intersection be defined?
- Which conditions influence the capacity at a signalized intersection?
- How can the capacity of cyclists at a signalized intersection be estimated?
- What scenarios could affect the maximum queue discharge rate?
- How do these scenarios affect the maximum queue discharge rate?

The first four questions will be answered using literature study. The last sub-questions will be answered using coding to analyze an available data-set. This data-set shows the individual trajectories of cyclists during the queue discharge phase.

1.3. Significance and stakeholders

This thesis will aim to increase the knowledge about the microscopic flow behaviour of cyclists. This research is relevant because there is not much knowledge about the microscopic traffic flow characteristics at the moment. Especially because the Netherlands is becoming less car dependent and focuses more on forms of transport like cycling. This thesis is relevant for other researchers that are doing a similar research or for city planners that are aiming to design new intersections for cyclists. The recommendations in section 6.2, will give a detailed list of possible subjects for further research.

1.4. Structure

In this subsection, a short overview will be given of the contents of the rest of the report. In chapter 2, a literature study is done to get some background information about the capacity, the flow characteristics of bicycles around an intersection and to get a clear idea of the research that has already been done. It will also give inspiration for possible scenarios to analyze In chapter 3 the methodology will be explained. This chapter will tell about the data-set and how this research will be conducted, including the equations and methods that will be used. Chapter 4 will give the results of the data processing after the methodology has been applied on the data-set. Finally, chapter 5 will contain the discussion and chapter 6 will give the conclusions and the recommendations for further research.

2

Literature research

2.1. Introduction

This chapter will provide some background information about this subject based on the available literature. First, the definition of the capacity will be given, including the different factors that influence the capacity. After that, some theory about capacity estimations will be given and finally, some literature that will serve as inspiration for the different scenarios will be covered.

2.2. Definition of capacity

In the introduction chapter, a general definition has been given for the capacity. Still, there are different ways to express the capacity of the road. Three different definitions for the capacity are described below (Minderhoud et al., 1997).

- **Design capacity**: The design capacity is the maximum traffic volume that can pass a cross-section under pre-defined road and weather conditions. The design capacity is very useful when designing roads and can be estimated based on existing guidelines and simulation models. This capacity is a value, that can be derived from a distribution;
- **Strategic capacity**: The strategic capacity is the maximum traffic volume a road section can handle. This is useful for analyzing roads and can be based on observed traffic flow data. The strategic capacity is also a value, which can be derived from a distribution. The strategic capacity is also the capacity that can be obtained based on the available data-set;
- **Operational capacity**: The operational capacity is a value which represents the actual maximum traffic volume of a road. It is a useful value for traffic control procedures.

Despite these definitions, it is not possible to give a quantitative definition of the capacity of a road, because the capacity can only be defined in relation to the traffic flow. The maximum capacity can be defined as: The "ability to achieve the maximum throughput under the full utilization of personal capabilities, means of transportation and available infrastructure." (Minderhoud et al., 1997) Most of the time, however, it is more useful to look at the capacity under predefined and common conditions. This will give values for the capacity, which are lower than the maximum. This is because of the assumption that not all the available resources are optimally used. In other words: Because the factors human drivers, the vehicles and the available infrastructure are not perfectly used, the capacity is a random variable. So all the different capacity values also form a distribution (Minderhoud et al., 1997). Some of the factors affecting the capacity, are shown in figure 2.1.

According to the highway capacity manual (Transportation Research Board, 2010) there are different types of conditions that affect the capacity. These conditions are fairly similar to the conditions depicted in figure 2.1. This is written for car traffic, but some of the theory about car traffic can also be applied on cyclists.

First there are base conditions which include, among other things, the weather, the condition of the pavement and the familiarity of users with the road. When making models, these conditions are often assumed to be 'favourable', to obtain a capacity as high as possible, but in reality that can be different.

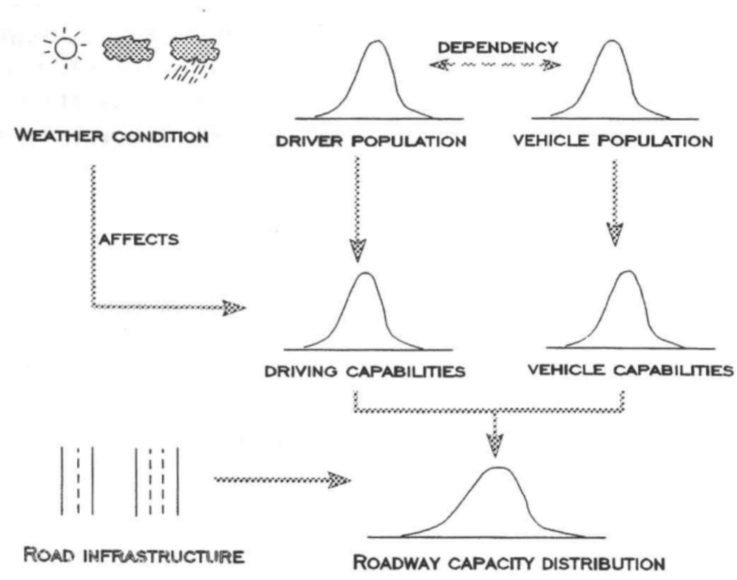

Figure 2.1: Factors affecting the Roadway capacity distribution (Minderhoud et al., 1997)

Besides the base conditions, there are also roadway conditions. This is about the existing infrastructure and includes things like the width of the lane and the horizontal and vertical alignments.

The third category is the traffic conditions. This includes vehicle types (on a cycle path this can be scooters, normal bikes, electric bikes or other special kind of bikes); the directional and lane distribution, which says something about how road users are spread over the road; and the driver population. When looking at car drivers, non-commuters and regular commuters display different behaviour. For cyclists this will probably be the same.

Lastly, there are the control conditions. This includes things like traffic lights (for example: the cycle length and the allocation of green time), but also normal stop signs or other types of traffic regulation.

2.3. Theories about capacity estimation

Hoogendoorn and Daamen (2016) did research on the bicycle capacity on a busy intersection, based on the headway of constrained cyclists. (Constrained cyclists are cyclists that are following another cyclist, whose speed is lower than their own desired speed.) This research calculates the capacity halfway on the intersection. A large difference with the data-set available for this bachelor final project is that the data only shows the trajectories of cyclists up to the stop line, so the data is mainly about cyclists who are stopping before a red light and then continuing again. This means that the speeds are very low and it will take some time for cyclists to start cycling again after the green light.

Yuan et al. (2019) did research on the same data-set as used for this thesis, and looked at the capacity at the intersection before the stop line. In this research, the capacity is determined using the saturation headway. The saturation flow can be defined as the inverse of the average headway when the queue is moving steadily. The saturation headway is the headway in a saturation flow. This is then multiplied with the number of virtual sublanes (the number of queues of cyclists than can cycle next to each other on the same cycle path). The difficult part with cyclists is that the width and the number of the virtual sublanes are not fixed values, which is the case with cars. The research proposes an improved method to calculate the saturation headway and the time lost when starting with cycling after a green light and it gives a new method to estimate the number of virtual sublanes on a cycle path. The research mainly focused on more average values, like the saturation flow and thus the macroscopic traffic flow characteristics. Its methods can be used to calculate several key variables of the traffic flow. To prevent repeating the same research, this bachelor final project will have more focus on the microscopic traffic flow characteristics and how specific scenarios impact the maximum queue discharge rate. To illustrate, Yuan et al. (2019) found that the saturation headway, the time lost starting up and the number of virtual sublanes are highly stochastic. By looking more at individual scenarios, it might give an explanation for some of the randomness. This information can then be used in future research and design questions to improve capacity of the intersection by anticipating on the different scenarios.

2.4. Possible scenarios during the queue discharge

This thesis will mainly focus on the different scenarios that can happen during the queue discharge with regard to the locations and movements of the individual cyclists. This section describes some peculiar things that can happen during the queue discharge, that might have an affect on the maximum queue discharge rate. This information will later be used to develop certain scenarios to look for in the data-set, as will be described in section 3.4.

An example of such a specific scenario is researched by Wierbos et al. (2020) where the impact of merging cyclists on the queue discharge rate was studied. This study found that cyclists who merged while overtaking had a positive influence on the discharge, while cyclists who merged from a perpendicular direction had a negative impact on the discharge. In the available dataset for this bachelor thesis there could also be looked for situations of overtaking cyclists.

Another example for a possible scenario can be found in the research from Wierbos et al. (2021) where the jam density was increased by giving queueing instructions in a controlled experiment. It was found that a higher queue density gave a higher discharge rate. This has not yet been verified in a field test, so looking for situations in the dataset with a significantly higher jam density, can also be a scenario to look for in the dataset.

A third scenario which is possible to look into is the effect the first cyclists have on the discharge of the total queue. According to Yuan et al. (2019), the first cyclists in a group tend to lose some time, because they have to react to the traffic light turning green. In other words, the time headway between the first cyclists is usually larger than the headway of later cyclists. The moment they cross the stop line, they are still accelerating and starting up, instead of having a constant speed like cyclists who start later in the queue. It can be interesting to look at the influence of the first cyclists on the maximum queue discharge rate.

3
Methodology

3.1. Introduction

This chapter will describe the methodology of this bachelor thesis. As described in section 2.2, there are different types of conditions that affect the capacity. From these conditions, this thesis will focus on the traffic conditions, and more specific: on certain scenarios that can happen during the queue discharge flow, regarding the locations and the movement of the cyclists. So the vehicle type and the population will not be a factor. As said in the introduction, because the different scenarios that will be described do not accurately reflect the traffic conditions, there will be spoken of the maximum queue discharge rate under a specific scenario. So first, the capacity of cyclists on the intersection will be estimated based on the whole data-set and after that, the maximum queue discharge rate will be determined when certain scenarios occur during the queue discharge period. For this, the same method will be used as for the capacity estimation for the whole data-set.

First, in section 3.2 some explanation about the available data-set will be given. After that, in section 3.3, some general methods will be explained. This includes a way to estimate the capacity based on the composite headway model, an explanation of the Pearson correlation coefficient and the p-value and a way to estimate some parameters that are needed for the capacity estimation. Finally, in section 3.4, the different scenarios will be given, including a way to find those scenarios in the data-set.

3.2. Explanation of the data-set

The data-set that is used for this thesis is provided and consists of 57 queueing phases of bicycles standing before the traffic light and leaving. The data is obtained from a signalized intersection in Amsterdam and the cycle path has a width of 2 meters. The data-set contains both cyclists and people riding scooters, but the percentage of scooters is very low (Yuan et al., 2019). Because of the low percentage of scooters, in previous research was chosen to refer as cyclists to both the people riding bicycles and the people riding scooters, with the assumption there will be no effect on the analysis. For this bachelor thesis, the same assumption will be made; both the people riding on bicycles and the people riding on scooters will be referred to as cyclists.

The trajectories of the cyclists were derived from top-view video images, with the help of two camera's on poles that covered a total of 20 meters of a cycle path. The footage came from June 6th in 2016 and the queue discharge periods that met several criteria were selected. The exact criteria and the procedure to get the data from the video images, is described in Yuan et al. (2019).

This data-set is initialized in Matlab, but for this thesis, python will be used to analyse the data-set. For some parts of the code and the modules used, see appendix A. The methodology will be described in more general terms, to make this research repeatable with other programming languages.

3.3. General methods

This will section will give the general methods used in this thesis. This includes the capacity estimation, an explanation of the Pearson correlation coefficient and the p-value and it will explain how to estimate some of the parameters needed for the capacity estimation.

The capacity

To estimate the capacity and the maximum queue discharge rate, a method using composite headway modeling and the mean empty zone from Hoogendoorn and Daamen (2016) will be used. The same composite headway estimation is explained a bit more elaborate in Hoogendoorn (2005). The exact equation are sometimes a bit different between the two mentioned papers. When this is the case, the equations from Hoogendoorn (2005) will be used. This is done because these versions of the equations were easier to implement in the coding.

The capacity can easily be estimated based on the empty zone. The empty zone is the minimum headway that a cyclist has when following another cyclist (i.e. the minimum headway of a constrained cyclist). Not every cyclist will have the same empty zone, so the empty zone is a distribution of a random variable X. The first step in determining the empty zone is to determine the time headway of a cyclist. As described before, this is the time that passes between two following cyclists crossing a cross section. For that, it has to be determined when a cyclist is following another cyclist. To determine the leader of cyclist i that passes the cross section at t_i, with lateral position $y_i(t_i)$, the leader can be found with equation 3.1, where $j < i$.

$$|y_j(t_j) - y_i(t_i)| \leq \frac{1}{2}a \tag{3.1}$$

In this equation $a > 0$ and equals the width of the cyclists handlebars and a small extra distance on either side. This means that a times the headway distance equals the free space in front of the cyclist. Using equation 3.1 to determine the leader, the time headway h_i of cyclist i can be determined by equation 3.2.

$$h_i = t_i - t_j \tag{3.2}$$

All these different time headways together form a distribution. To simplify, the headways of the cyclists without a leader will not be taken into account. These are the first or the first few cyclists of a queue. According to the composite headway model, the total distribution $f(h)$ is the sum of the distribution of the free headway $r(h)$ and the distribution of the empty zone $g(h)$. The fraction of constrained cyclists is denoted by ϕ. This gives equation 3.3.

$$f(h) = \phi g(h) + (1 - \phi) r(h) \tag{3.3}$$

When the capacity of an intersection is reached, all the cyclists are restrained so $\phi = 1$. Equation 3.4 gives the capacity C in cyclists per hour per meter.

$$C = \frac{1}{2aE(X)} \tag{3.4}$$

Here the $E(X)$ is the mean empty zone. To find the total capacity, C has to be multiplied with the width of the cycle path, which is 2.0 meters is this case. This gives equation 3.5.

$$C_{tot} = \frac{1}{aE(X)} \tag{3.5}$$

The main problem left is to determine the mean empty zone, from the total distribution $f(h)$. This can be done in several steps, by determining an estimation for the free headway distribution and subtracting that from the total distribution.

To get an estimation for the free headway distribution, the arrival rate, λ, and the normalization constant, A, have to be determined, based on a part of the distribution that has only free flow. This can be done by evaluating all headways that are greater than T^*, where $g(h) = 0$ for all $h > T^*$. This gives equation 3.6 for the estimation of λ and equation 3.7 for the estimation of A.

$$\hat{\lambda} = \left[\frac{1}{m} \sum_{i=1}^{\infty} \{h_i - T^* | h_i > T^*\} \right]^{-1} \tag{3.6}$$

$$\hat{A} = \frac{m}{n} e^{\hat{\lambda} T^*} \tag{3.7}$$

Here n is the total number of time headways and m the number of headways that are greater than T^*. The total equation for the estimate of the free headway distribution is given in equation 3.8 together with equation 3.9.

$$\hat{r}_1(h) = \hat{A}\hat{\lambda}e^{-\hat{\lambda}h}\left[1 - \frac{1}{\hat{\phi}}\int_h^\infty (\hat{f}_n(\tau) - \hat{r}_1(\tau))d\tau\right] \tag{3.8}$$

$$\hat{\phi} = \int_0^\infty (\hat{f}_n(\tau) - \hat{r}_1(\tau))d\tau \tag{3.9}$$

This can be solved iteratively, giving as a result $\hat{r}_1(h)$. This process is ready when the difference between $\hat{r}_i(h)$ and $\hat{r}_{i-1}$ is small enough. Most of the time, this doesn't take more than five iterations. According to Hoogendoorn (2005) equation 3.10 is a good initial solution.

$$\hat{r}_1^{(0)} = \hat{A}\hat{\lambda}e^{-\hat{\lambda}h};\ \ \hat{\phi}^{(0)} = 0.9 \tag{3.10}$$

Finally, an estimate for the empty zone is given by equation 3.12, the average of the constrained headway distribution, using equation 3.11. This can than be used in equation 3.5 to determine the capacity of the intersection.

$$\hat{g}(h) = \frac{\hat{g}_1(h)}{\hat{\phi}} = \frac{\hat{f}_n(h) - \hat{r}_1(h)}{\hat{\phi}} \tag{3.11}$$

$$E(x) = \overline{g}(h) \tag{3.12}$$

A more detailed methodology of this capacity estimation and the theory behind it can be found in Hoogendoorn and Daamen (2016) and Hoogendoorn (2005).

This capacity estimation will be used to determine capacity based on all available data. An individual discharge period does not contain enough headways to get a large enough distribution. To use this method to estimate the maximum queue discharge rate of smaller subgroups, individual phases that contain the same scenario can be grouped. This way it is also possible to compare the maximum queue discharge rate of the different scenarios with each other.

The Pearson correlation coefficient and the p-value

In Wierbos et al. (2020) the Pearson correlation coefficient and the p-value are used to see if there is some sort of relation and correlation between two variables. Those values can also be useful in this research to see if there is some sort of relation between the maximum queue discharge rate and the scenario.

The Pearson correlation coefficient (r) can be used to see how strong the linear correlation is between two variables and the direction of this relation (positive or negative). The coefficient gives a value between -1 and 1. Here a value of -1 signifies a perfect negative correlation, while a value of 1 signifies a perfect positive correlation. When the coefficient is 0, no correlation between the two variables can be identified based on the data. To use this coefficient it must be assumed the values come from a normal distribution and the relation between the variables is linear. It can be calculated by taking the covariance of the variables ($cov(x, y)$) and dividing that by the product of their respective standard deviations (S_x and S_y), see equation 3.13.

$$r = \frac{cov(x, y)}{S_x * S_y} \tag{3.13}$$

First, a null hypothesis and an alternative hypothesis must be specified. For this thesis the null hypothesis will be that there is no relation between the scenario and the maximum queue discharge flow and the alternative hypothesis is that there is some form of relation between those variables. After that, the t-test can be performed, which gives the t-value (t). This can be done with equation 3.14. Here n is the sample size.

$$t = \frac{r * \sqrt{n-2}}{\sqrt{1 - r^2}} \tag{3.14}$$

For every t-value is a corresponding p-value. This p-value can be found by looking it up in the so-called t-table or by using coding or a software. The p-value gives the probability that the null hypothesis is true, or in this case the probability that there is no relation between the maximum queue discharge rate and the scenario. Most of the time in research the threshold will be a p-value of 0.05. This means the null hypothesis will be rejected if the p-value is lower than 0.05, which means that there is less than 5 percent chance the two variables are not related. (Jaadi, 2021)

The p-value is very important here, because it takes the size of the sample in account. When using a small sample size, the chance is big the Pearson correlation coefficient indicates there is some correlation, while in reality there are not enough data points to really say something about the correlation. To summarize: The Pearson correlation coefficient quantifies the correlation and the p-value gives the chance this correlation is just by chance. So the Pearson correlation coefficient and the p-value should be used together. (OPEX resources, 2019)

Parameter estimates

For the capacity estimation, as described earlier in this subsection, some parameter estimations have to be made in advance, namely: a and T^*. The parameter a is the width of a cyclist and a bit of extra free space because of the fear of obstacles. According to CROW (2016) the width of a cyclist is about 0.75 meters and the distance between two cyclists about 0.25 meters. In other words a cyclist maintains about 0.125 meters of free space on either side. Those values are used while designing bicycle infrastructure. This thesis does not focus on determining the exact width of a virtual sublane (see chapter 2 for more information about these sublanes), so $a = 1$ meter is a reasonable assumption for now.

According to Hoogendoorn (2005) the parameter T^* can be easily estimated by plotting the empirical survival function $\hat{S}_n(h) = 1 - \hat{F}_n(h)$, where $\hat{F}_n(h)$ is the cumulative distribution function of $\hat{f}_n(h)$. This function must be plotted on a logarithmic scale. The free flow follows an exponential function, so the plotted line will be straight for $h > T^*$. The moment the line will bend, it means that there is no fully free flow anymore. The value for T^* will be determined in chapter 4. Based on the results, one value of T^* will be used for all the maximum queue discharge rate estimates or a new value of T^* will be estimated each time. Because T^* does not have to be very precise, but just big enough to only include free flow, it might not be needed to make a new estimate for every maximum queue discharge rate estimation.

3.4. Scenario methods

In this subsection, the different scenarios will be explained. As described earlier, the influence of a scenario on the maximum queue discharge rate will be analyzed. The scenarios described in this subsection will have something to do with the queue composition or the behaviour of the cyclists during the queue discharge. The scenarios have been chosen based on things that seemed relevant in the literature study and interesting situations that could be found in the data-set. These scenarios are by no means a complete list of all possible scenarios that can have an influence on the maximum queue discharge rate. First of all, off all the components that affect the capacity, there will be only looked at the traffic conditions and the scenarios will not represent all those traffic conditions, but are only part of these conditions. They can, however, give an idea of the different scenarios that can be present during the queue discharge and how these scenarios affect the maximum queue discharge rate. This will give more insight in the microscopic traffic flow behaviour of cyclists.

For the different scenarios, a motivation will be given why this scenario will be analyzed and the method to identify these scenarios in the data-set will be given. The different scenarios will be quantified. This means that every queue discharge phase will be given some quantity based on the presence of the scenario. After that, the queueing phases will be grouped based on those quantities and the maximum queue discharge rate for each of those groups will be estimated. The groups will be formed based on the results to make appropriate group sizes. When a group contains too few phases for a successful maximum queue discharge rate estimation, the group will be merged with another group. The exact requirements for this will be determined in chapter 4, based on the results. These different estimations can then be compared, to see if there is some sort of relation with the maximum queue discharge rate and the scenario. The Pearson correlation coefficient and the p-value can help with this. All the scenarios will be identified in the data-set with the help of coding.

The front or the back of the queue is very dense

As stated in chapter 2, the queue density can have a significant influence on the discharge rate. Based on those studies, the maximum queue discharge rate seems to get higher with a higher density. An interesting scenario is to look at the density at the front and the back of the queue and how this affects the maximum queue discharge rate. For this, the density can be calculated over the last n_q and the first n_q cyclists, where the number of cyclists (n_q) can vary to get more data.

According to Wierbos et al. (2021) the density of a queue can be calculated by dividing n_q by the dimensions of the waiting area (A). The number of cyclists waiting in the queue are not necessarily the same as the number of cyclists that pass the stop line. It might be possible that some cyclists join the queue later. The dimensions of the waiting area can be calculated by multiplying the width of the cycle path by the length of the queue. When looking at the density at the front of the queue, the length will be the distance form the stop line until the last cyclist. When the first cyclist has already past the stop line, the length of the queue will be calculated based on the position of that cyclist. The initial locations of the cyclists will be used, in other words: the location at the beginning of the measurements when all the cyclists are standing still. When looking at the density at the back of the queue, the distance from the last cyclist up to the first cyclist from the part the density will be calculated over, will be used. This gives equation 3.15.

$$k = \frac{n_q}{A} = \frac{n_q}{2.0 \times L} \tag{3.15}$$

Here k is the density and L the length of the queue.

Cyclists standing behind each other while waiting

As stated in the section above, the local density of the queue might have an affect on the maximum queue discharge rate. Density is more of a macroscopic traffic flow characteristic, so it might give some extra insight to look more at the queue configuration on a microscopic level. Sometimes, a cyclist might chose to stand behind another cyclist, while there is still enough enough space next to the other cyclist. In other words, the queue becomes longer and less dense. To identify this scenario there must be looked for places in the queue, before the discharge has started, where there is enough space for another cyclist. This will be done by looking at every cyclist, who is not the last cyclist in the queue, if there is enough space for another cyclist to stand next to it. The number of times there is enough space, will be counted. In other words, the number of times cyclists were standing behind each other instead of next to each other. Because not every cyclist has the same size and not every cyclist wants to keep the same space to another cyclist, this scenario can be tested with different widths of a cyclist.

To look for this scenario, the average dimensions of a cyclist must be known. The maximum width of the handlebars of a bike are 75 cm and adding some free space around the cyclist for movement adds up to 100 cm (CROW, 2016). The average bike has a handlebar width of 60 cm and the extra space adds up to 80 cm (Wierbos et al., 2021). These different values (with and without free space) can be used to determine if a space is wide enough to fit another cyclist. The average length of a cyclist is 195 cm (CROW, 2016), so rounding up gives a length of 200 cm. This length should be the minimum distance between two cyclists in the x-direction. Here, the x-direction describes the length of the cycle path and the y-direction the width of the cycle path. In the available data-set, the point of measurement is the middle of the head (Goñi-Ros et al., 2018), which is approximately in the middle of the total length of the cyclist. It will be counted as two cyclists standing next to each other when at least half of the cyclist overlaps, so the maximum distance in x-direction between a cyclist and the middle of the free space is 1 meter. This way there is enough space for another cyclist to stand next to the first cyclist. In figure 3.1 the requirements as described above are visualized. The figure shows the queue of one phase in the data-set, but it can also be applied on other data-sets. The purple lines

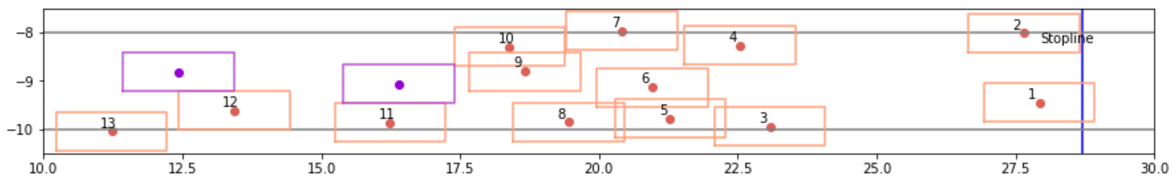

Figure 3.1: The figures shows two examples of possibles spaces where another cyclist can fit in purple, using a width of 80 cm. The red lines mark the spaces of the existing cyclists

are possible spaces that count as enough space for another cyclist to fit with a width of 80 cm. These are not the only possible spaces for another cyclist to stand. In this example, the imaginary cyclist next to cyclist 11 could stand more to the back and the imaginary cyclist next to cyclist 12 could stand more to the front. The space next to cyclist 13 does not count, because it is the last cyclist in the row. The empty space behind cyclist 1 and 2 is also not counted because the code will only look for spaces next to another cyclist.

Cyclists overtaking each other

In chapter 2 it was stated that according to a study, the overtaking of cyclists can have a positive influence on the queue discharge rate. That is why this scenario will also be looked for in the available data-set to see if the data shows that it affects the maximum queue discharge rate. To see if cyclists were overtaken in the data-set, there will be looked at the order in x-direction the cyclists were situated while waiting for the traffic light and the order the cyclists passed the stop line. Every time a cyclist would pass the stop line earlier than a cyclist that was standing closer to the stop line at the start of the queue discharge, the scenario overtaking took place. For each phase, the number of times the overtaking took place will be counted.

The first cyclists take a long time before leaving

As described in chapter 2 the first cyclists of the queue show a different behaviour in comparison with the cyclists further down in the queue. This might have an influence on the maximum queue discharge rate. For this scenario the discharge time of the first cyclist(s) in the queue will be looked at. According to Wierbos et al. (2021) this is the time interval between the first cyclist to leave the waiting area (in this case, passing the stop line) and the last cyclist to leave the area. This gives equation 3.16, where T_{dis} is the discharge time.

$$T_{dis} = T_{last} - T_{first} \tag{3.16}$$

In this particular case, T_{first} will be the time at which the light goes green and T_{last} will be the time that cyclist n crosses the stop line. This way, the time that is lost during the start-up time will also be included. Because the discharge time will be determined for several n number of cyclists, the resulting times can be pretty far from each other. (To clarify: the first cyclist to leave the stop line, will take considerably less time than the sixth cyclist to cross the stop line.) To get time stamps that lay closer to each other, it can also be interesting to divide the discharge time by the number of cyclists. This gives the inverse of the traffic flow. The traffic flow q is often defined as: “average number of vehicles (n) that pass a cross-section during a unit of time (T)” (Hoogendoorn & Knoop, 2013). Written as an equation this gives equation 3.17. Here $\overline{h}$ is the average time headway.

$$q = \frac{n}{T} = \frac{1}{\overline{h}} \tag{3.17}$$

In other words, by dividing the discharge time of the first cyclists by the number of cyclists, the average headway for those cyclists will be obtained. However, the average headway that will be obtained with this method is different than the average headway obtained with the method described in section 3.3. This is because for this calculation the headway of the first cyclist will also be taken into consideration. In this case, the headway of the first cyclist is the time that passes between the light that goes green and crossing the stop line.

4

Results

4.1. Introduction

This chapter shows the results of the methodology when applied on the dataset. First, the headway distribution is given, after that the estimation of T^* and then the capacity estimation. After that, the scenarios covered in the methodology will be worked out. In appendix A (parts of) the code that is used to get these results can be found.

4.2. Headway distribution

In figure 4.1 the headway distribution from all the data is shown. This gives an initial idea what the distribution looks like. For every cyclist the y-position and the time stamp when the cyclist crossed the stop line at $X = 28.7$m were determined. Because the exact time stamps when the cyclists crossed the stop line were not available, interpolation was used. This was done by getting the data points before and after the stop line and using linear interpolation to get an estimate of the exact position and time while crossing the stop line. After that, the leader of each cyclist was determined using equation 3.1 and then the time headway by using equation 3.2.

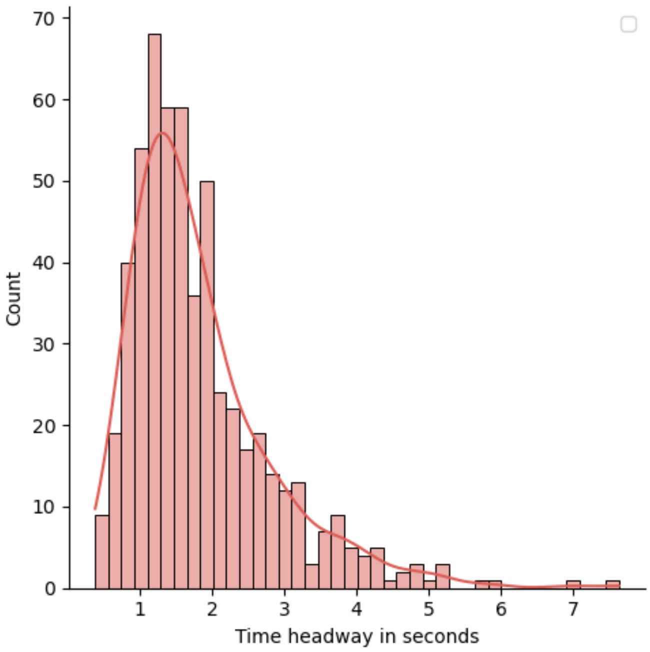

Figure 4.1: Time headway distribution from all the data

The average time headway is 1.83 seconds and the median is 1.57 seconds. When looking at the histogram most cyclists seem to have a headway between 1 and 2 seconds. There are not many cyclists with a headway lower than 1 second and the number of cyclists that have a headway larger than 2 seconds are rapidly decreasing.

4.3. Determining T^*

To determine an appropriate value for T^*, the survival function will be plotted on a logarithmic scale, as described in section 3.3. For the total headway distribution, this gives the plot in figure 4.2. The graph shows a bend for headway values lower than 1.5 seconds. Because the T^*-value is the threshold value, all headway values that are greater than T^* are part of the free flow. So it is better to choose a T^* that is a bit too large, than a T^* that is a bit too small. So to be on the safe side, a headway value of $T^* = 2$ seconds will be chosen as parameter. This value will be used in the estimation for the capacity for the whole data set and in all the estimations of the maximum queue discharge rate. This value can be used for all estimations, because the graph shows that all headways larger than 2 seconds are part of the free flow. When taking only a part of the total distribution, this fact will not change.

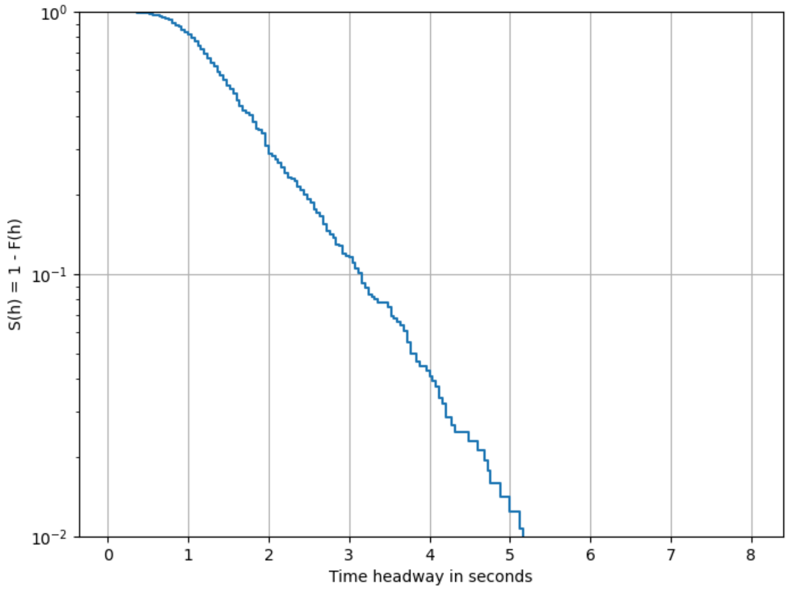

Figure 4.2: The survival function S(h) on a logarithmic scale

4.4. Capacity estimation

In figure 4.3 the total headway distribution and the estimate of the free flow distribution and the constrained distribution, according to the composite headway model is shown. This made it possible to estimate the capacity for the total headway distribution. The capacity, and several other parameters are shown below. All of these are determined according to the method described in the methodology in chapter 3.3.

- $E(X) = 1.25$ s
- $\hat{\phi} = 27.6\%$
- $\hat{\lambda} = 0.979$
- $\hat{A} = 2.16$
- $C = 2889$ cyclists / hour

Here $E(X)$ is the mean empty zone, or in other words the average headway value for constrained cyclists. $\hat{\phi}$ is the ratio of constrained cyclists, $\hat{\lambda}$ is a parameter called the arrival rate (equation 3.6) and $\hat{A}$ a parameter called the normalization constant (equation 3.7). Finally, C is the capacity estimated based on the total headway distribution with the help of equation 3.5 with $a = 1$.

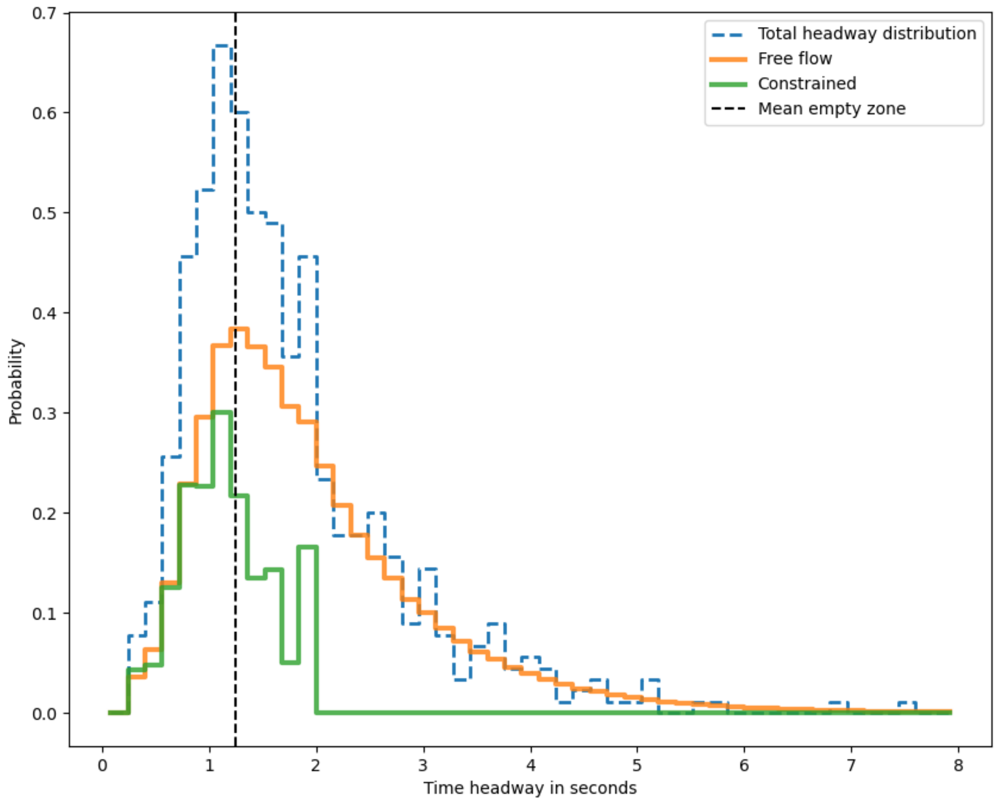

Figure 4.3: Time headway distribution from all the data

A large limitation when estimating the maximum queue discharge rate with this method is the relatively small sample size. When comparing the different scenario's a sample size that is too small would fail. However, the number of samples needed was not very consistent. Sometimes a sample size containing only 3 phases would succeed, while a sample size with more than 10 phases or even 20 phases would fail. When grouping the different phases based on the scenarios, groups with a minimum of 3 phases were formed. When the estimation of the maximum queue discharge rate would fail, the group was merged with another group and it was tried again. Because of the small sample size, especially when comparing different scenarios, the exact estimates of the maximum queue discharge rate might be not very accurate. However, because there will be mainly looked at the relations, the difference in the maximum queue discharge rate can still say something about the effect a scenario has on the maximum queue discharge rate.

4.5. Scenarios in the dataset

In this subsection, the results of the scenarios described in section 3.4 will be shown and the results will be further clarified and explained if possible. The exact numerical results of all the estimations of the maximum queue discharge rate and the sample sizes used for these estimations can be found in appendix B.

The front or the back of the queue is very dense

To calculate the density at the front and the back of the queue, there was looked at the density of the first n cyclists and the last n cyclists in the queue. When looking at the data, a cluster of cyclists at the beginning and the end, consisted most of the time of around 4 or 5 cyclists. To get a better idea how the maximum queue discharge rate changed when the density changed, the density of the front and the back of the queue was calculated for 2 to 6 cyclists. The density at the front of the queue is plotted against the maximum queue discharge rate in figure 4.4 and the density at the back of the queue is plotted against the maximum queue discharge rate in figure 4.5. In both figures, the change in the maximum queue discharge rate over the total density of the queue has also been taken into account.

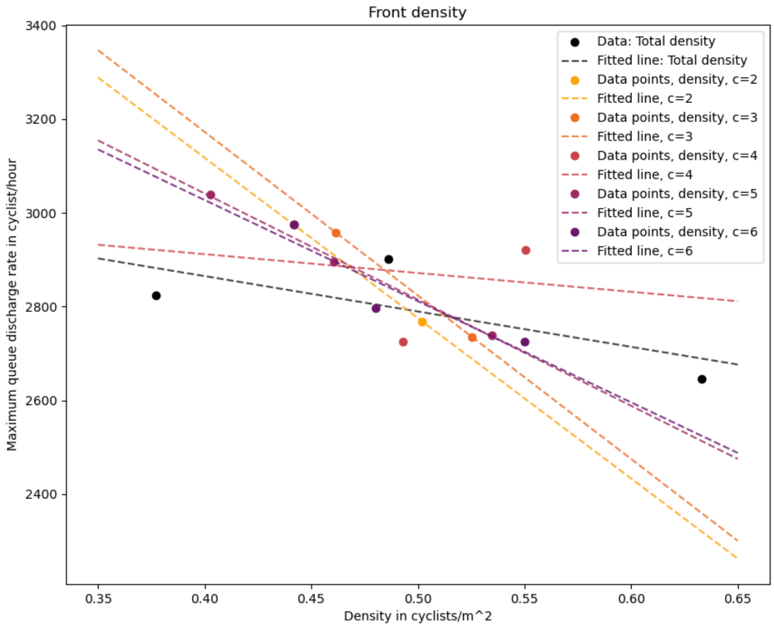

Figure 4.4: Estimations of the maximum queue discharge rate based on the front density of the queue. Here, c is the number of cyclists in the front of queue the density has been calculated for.

Looking at these graphs, the maximum queue discharge rate seems to go down when the front of the queue is dense. The Pearson correlation coefficient is -0.77 and the p-value 0.0022, when calculating these values for all the data points shown in the graph. This indicates that there is a pretty strong negative correlation between these variables. The p-value is below the threshold of 0.05 and indicates that there is more than 99% chance that the density at the front of the queue is somehow related with the maximum queue discharge rate.

This is an interesting result, because earlier studies found that the queue discharge was positively correlated with the total density of the queue (Wierbos et al., 2021)(Goñi-Ros et al., 2018). In this study the density at only the front of the queue is considered, which can explain part of the difference. However, also the total density and the densities when a larger part of the front queue was considered, seem to show a negative correlation with the maximum queue discharge rate. Though the number of data points for the total density is way less and thus is the correlation less reliable. Another difference between this study and the aforementioned ones, is that here a focus lies on the maximum queue discharge rate, instead of the observed queue discharge rate. A third interesting point is the difference in steepness of the several fitted lines. The negative correlation between the maximum queue discharge rate and the front density seems to be strongest when only the first 2 or 3 cyclists are considered. When more cyclists are considered the steepness becomes less. This might indicate that the position of the cyclists at the front have the largest effect on the maximum queue discharge rate.

When looking at the density in the back there does not seem to be any clear correlation. The lines

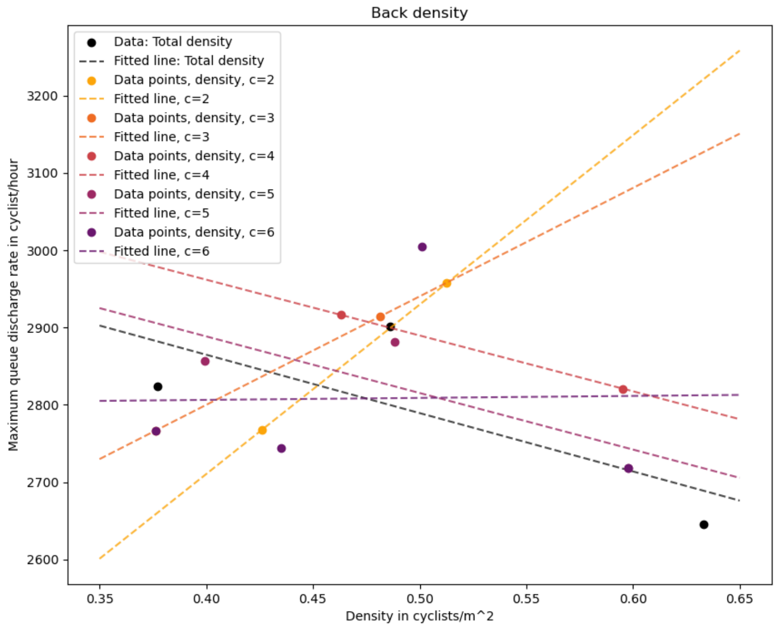

Figure 4.5: Estimations of the maximum queue discharge rate based on the back density of the queue. Here, c is the number of cyclists in the back of queue the density has been calculated for.

are all over the place. It can be noticed that when looking at the back density of the last 2 or 3 cyclists, there is a positive correlation with the maximum queue discharge rate, while there is none or a negative correlation when calculating the density of more cyclists at the back. The Pearson correlation coefficient for all data points in the graph is -0.026 and p-value 0.93. In other words, the chance that the density at the back is related with the maximum queue discharge rate is less than 7%, which makes this statistically insignificant.

When comparing the density at the back and the density at the front, it is not unexpected that the density at the front seems to have a larger influence on the maximum queue discharge rate than the density at the back. The back of the queue can only pass the stop line after the front of the queue has done so. When something happens at the front of the queue the cyclists at the back of the queue are also affected, while when something happens at the back of the queue the cyclists at the front will probably not notice the effects.

Cyclists standing behind each other while waiting

For this scenario, there was looped over each cyclist (except the last one in the queue) from each phase to determine if there was enough space for another cyclist to stand next to it. As an extra variable, the width of a cyclist was taken. These number of spaces were plotted against the maximum queue discharge rate, which can be seen in figure 4.6. The results for a cyclist width of 0.75 meters and of 0.8 meters are identical, which is not weird considering the width does not differ much. The Pearson correlation coefficient gives a value of -0.12, which indicates a very weak negative correlation. However, the p-value is 0.66, which is way too high to make any conclusions based on these results. If there is indeed no correlation, this indicates that even though the density has a large effect on the maximum queue discharge rate, it does not matter much if there are some wide spaces in the queue, as long as the rest of the density is high (or low) enough.

Cyclists overtaking each other

Each cyclist in the data-set had an initial number, which was given based on the x-coordinate of their start position. The higher the number, the farther away from the stop line. As described in the methodology, this scenario was counted when a cyclist crossed the stop line earlier than another cyclist with a

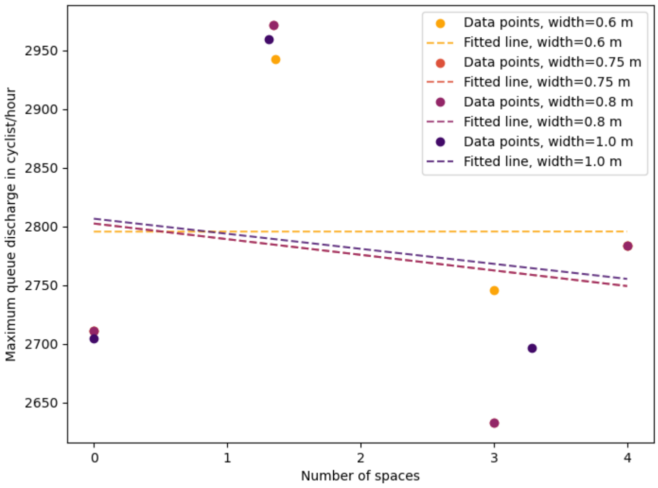

Figure 4.6: Estimations of the maximum queue discharge rate based on how many times cyclists are standing behind each other instead of next to each other. The capacity estimates are done for a width of the cyclist of 0.6 meters, 0.75 meters, 0.8 meters and 1.0 meters. Note: the results for 0.75 meters and 0.8 meters are exactly the same.

lower number. At most, 7 cyclists were overtaken in one phase and at least 0 cyclists. After all the times a cyclist was overtaken was found, the phases were grouped according to the number of 'overtakings' that took place in the phase. However, the estimations of the maximum queue discharge rate for all the groups with 4 or more overtakings failed, just like the estimation for the group with 1 overtaking. The groups of 0 and 1 were taken together, just as the groups with 3 and more overtakings. The group with 0 overtakings and 3 overtakings were still included in the graph to get more data points, so these groups are represented in two estimations of the maximum queue discharge. See figure 4.7 for the capacity estimation. The x-axis represents the (average) number of overtakings in the group.

Looking at the graph, there seems to be a positive correlation between the number of overtakings and the maximum queue discharge rate. The Pearson correlation coefficient is 0.84, which indicates a strong positive correlation. The results are not unexpected, when looking at previous research about merging and overtaking cyclists, which found that overtaking had a positive influence on the queue discharge rate (Wierbos et al., 2020). The p-value, however, is 0.077. This is just a bit above the threshold of 0.05. It is still likely that the number of overtakings has a positive effect on the maximum queue discharge rate, because of the results of previous studies and the fairly low p-value, but based on the results from this thesis alone, this can not be concluded.

The first cyclists are taking a long time before leaving

For this scenario, the discharge time for the first cyclist up to the discharge time for the fifth cyclist has been calculated. There has been chosen to do this calculation up to fifth cyclist, because when going through the dataset, the clusters of cyclist at the front seemed to be a around a maximum of 5 cyclists. A few times it occured that the first cyclist was already past the stop line before the light went green. In that case, the first cyclist was not counted and the second cyclist from that phase would be the first cyclist in the calculation. This has been done to avoid negative discharge times.

After the discharge time for each number of cyclists was calculated, the cyclist were grouped and the maximum queue discharge rate was estimated. The results can be seen in figure 4.8 Here, c is the number of cyclists the discharge time was calculated for, so each colored line represents another number of cyclists. In figure 4.9 the discharge time has been divided by the number of cyclists. In other words, the average headway of the first cyclist is plotted against the maximum queue discharge rate.

An interesting observation based on the aforementioned graphs is that the maximum queue discharge rate seems to increase with a higher discharge time. This is true for both graphs. So when the first cyclists take a longer time to leave, the maximum queue discharge time gets higher. The Pearson correlation coefficient for all the data points in the graph displaying the discharge time is 0.53 and the

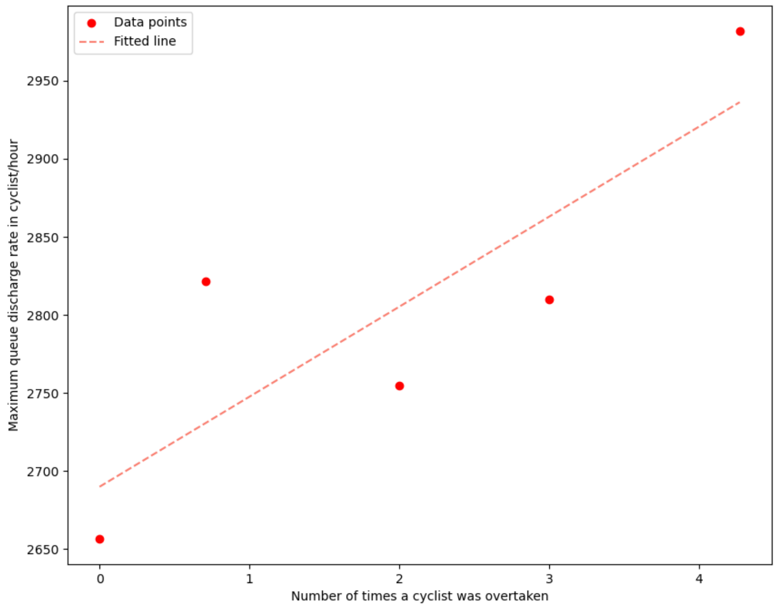

Figure 4.7: Estimations of the maximum queue discharge rate based on the number of times a cyclist was overtaken in one phase

coefficient for the graph with the average headway is 0.42. So in both cases the time shows a moderate positive correlation with the maximum queue discharge rate. The p-values are 0.0030 and 0.024, respectively. Both of these values are below the threshold, which indicates the presence of a relation between the discharge time of the first cyclists and the maximum queue discharge rate.

These results are counter intuitive. The capacity, or in this case the maximum queue discharge, can be estimated by taking the inverse of the headway of a constrained cyclist. When this mean empty zone gets higher, the maximum queue discharge rate gets less. In this case, the average headway of the total distribution gets higher, but the average headway of the constrained distribution should get lower for the maximum queue discharge rate to increase. This means that the ratio or the form of the constrained distribution changes when the discharge time of the first cyclists gets higher.

Plotting the discharge time of the first cyclists against the ratio of constrained cyclists gives a Pearson correlation coefficient of 0.15 and a p-value of 0.45, which seems to indicate that there is no relation between the discharge time and the ratio of constrained cyclists. However, when plotting the average headway of the first cyclists against the ratio of constrained cyclists a Pearson correlation coefficient of 0.36 is obtained and a p-value of 0.052. The correlation becomes weaker, but the p-value gets very low, just above the threshold, making it more likely there is some form of correlation between the average headway of the first cyclists and the ratio of constrained cyclists. If this is true, it means that when the average headway of the first cyclists are higher, so when the time distance is larger, more cyclists are constrained. A possible explanation for this could be that if the first cyclists take a longer time to leave, the cyclists behind cannot cross the stop line and are stuck behind the first cyclists, making them constrained. So a larger part of the later cyclists are constrained.

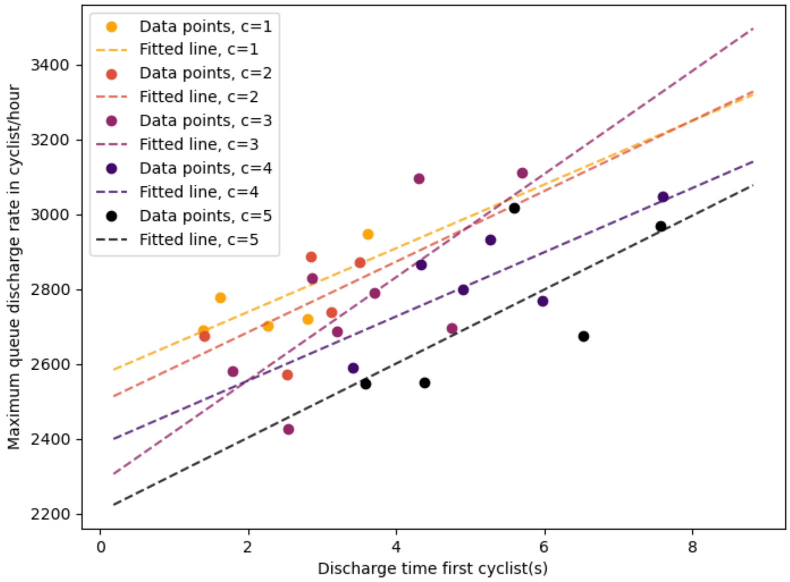

Figure 4.8: Estimations of the maximum queue discharge rate when cyclists based on the discharge time of the first cyclists. Here, c is the number of first cyclists the discharge time was calculated for.

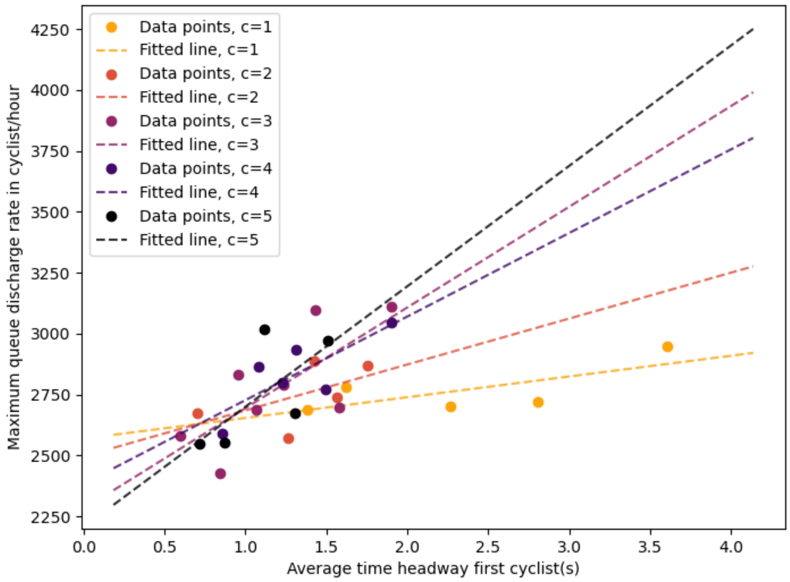

Figure 4.9: Estimations of the maximum queue discharge rate when cyclists based on the average time headway of the first cyclists. Here, c is the number of first cyclists the average headway was calculated for.

5 Discussion

This chapter contains the discussion. It will cover certain points that could be improved in further research.

This research covered several scenarios that could happen during the queue discharge phase, which were assumed to have some kind of effect on the maximum queue discharge rate. Because these scenarios did not accurately reflect the conditions that affect the capacity, this research focused on the maximum queue discharge rate, instead of on the capacity. However, the scenarios that were chosen to analyze were kind of random. The inspiration for the scenarios has been obtained from literature and the scenarios mostly focus on the microscopic traffic flow characteristics, but they do not have much in common other than that. Looking only at this research, nothing can be said about the whole capacity or all things that affect the maximum queue discharge rate, but only a few scenarios that can happen are briefly covered. A full answer to the research cannot be given based on this thesis alone.

To estimate the maximum queue discharge rate, the available data-set was split up in several smaller groups consisting of 3 or more queueing phases. On top of that, not every group had the same size. It is unclear to what extent this has influenced the maximum queue discharge rate estimations. Sometimes, the estimations failed even though the group consisted of more than 10 phases, while other times the estimation succeeded with a group of just 3 phases. This made the estimations less reliable. Furthermore, because the number of groups that could be formed was not very high, the number of estimations for the maximum queue discharge rate was also not very high. Because there were only a few data points in some scenarios, it was more difficult to judge if there was a relation between the scenario and the maximum queue discharge rate.

Another element that was not included in this research was a differentiation between the sizes of the different queue discharge phases. The smallest phase had a size of 6 cyclists, while the largest group had a size of 20 cyclists. For example, when looking at the density of the first and last cyclists, up to 6 cyclists were considered. For this smallest sample size, this meant that the density of the whole queue was considered, while when looking at the phase with the most cyclists, just a bit more than a quarter of the queue was considered. It has not been researched if this had any effect on the results. For future research it might be good to differentiate between the queue sizes and, for example, only uses queue sizes that are large enough.

6

Conclusions and recommendations

6.1. Conclusions

This thesis has the research question: *How can the capacity of cyclists be determined at the stop line of a signalized intersection?* The capacity of an intersection is dependent on several conditions: the base conditions like weather, pavement condition and the user's familiarity with the road; the roadway conditions which includes the existing infrastructure; the traffic conditions, which is about the vehicle type, lane distribution and the driver population; and the control conditions which includes traffic regulations. The capacity of an intersection could be estimated using the composite headway model. For this, only the headway distribution of the cyclists at an intersection was needed. Finding a general method to determine the capacity at a signalized intersection which includes all possible factors that could affect the capacity did not fall into the scope of this thesis. However, several scenarios that were part of the traffic conditions were analyzed to see how they affected the maximum queue discharge rate. A summary of these results is given below.

Based on the available data-set of cyclist trajectories, several scenarios were identified and plotted against estimates of the maximum queue discharge rate. In total 6 scenarios were analyzed, when the front and back density are treated as different scenarios, as well as the discharge time of the first cyclists and the average headway. The Pearson correlation coefficients (r) and the p-values (p) for each scenario can be found in table 6.1. When the absolute value of the Pearson correlation coefficient is larger, the correlation between the scenario and the maximum queue discharge rate was stronger. However, only when the corresponding p-value was very low this correlation is likely real and not just by chance. The threshold for this is a p-value of 0.05.

Scenario	r	p
Front density	-0.77	0.0022
Back density	-0.026	0.93
Cyclists standing behind each other	-0.12	0.66
Overtaking	0.84	0.077
Discharge time first cyclists	0.53	0.0030
Average time headway first cyclists	0.42	0.024

Table 6.1: The Pearson correlation coefficients and the p-values for each scenario

The density at the front of the queue seems to have a strong negative correlation with the maximum queue discharge rate. So when the density gets higher, the maximum queue discharge rate decreases. These findings seem to contradict earlier studies that showed that the the queue discharge rate was positively correlated with the density. The correlation is strongest when looking only at the density of the first 2 or 3 cyclists. This could indicate that the cyclists standing almost totally at the front have a high influence on the maximum queue discharge rate. When looking at the density at the back of the queue, there is no clear correlation with the maximum queue discharge rate based on these results. Interesting though, is the difference in correlation for the density of the last 2 or 3 cyclists and the density for more cyclists. The data-set used for research does not contain enough data points to make any conclusions about this. The number of times there is enough space next to a cyclist for another cyclist had a weak correlation and a very high p-value. This p-value was too high to make any conclusions. There seems to be a very strong positive correlation between the number of times cyclists are overtaking and the maximum queue discharge rate. However, the p-value is too high to make any definite conclusions. The strong correlation is line with earlier research that stated that overtaking had a positive affect on the queue discharge rate. So even though based on solely the numbers, no conclusion can be made, it would be in-line with the existing theory, to assume the overtaking has a positive effect on the maximum queue discharge rate. The discharge time of the first cyclists and the average time headway of the first cyclists seem to have a moderate positive correlation with the maximum queue discharge rate. The p-value is also low enough to back up this conclusion. There also seems to be a positive correlation between the average headway and the ratio of constrained cyclists, which indicates that with a higher average headway of the first cyclists, the total number of constrained cyclists goes up.

The most interesting results from this research is the effect the first cyclists have on the maximum queue discharge rate. Both the scenarios related with the behaviour and the positions of the first cyclists gave the most clear results. Both the scenario concerning the density of the first cyclists and the scenario concerning the discharge time of the first cyclists showed results that seemed counter intuitive at first.

The results from this research help in understanding the randomness in capacity estimations and give more insight into the effect of microscopic traffic flow characteristics on the maximum queue discharge rate. It can also give a better idea of the possible traffic conditions that happen on a microscopic level. This is important because the traffic conditions have an effect on the capacity. This can be used in further research to help designing better intersections with a higher capacity.

6.2. Recommendations

This Bachelor thesis leaves room for a lot more research. This thesis only briefly looked at several scenarios that can happen during the queue discharge. For future research, other scenarios could be identified to get a more complete view of the microscopic traffic flow characteristics during the queue discharge and to get more insight into the different factors that affect the maximum queue discharge. This can be done based on this data-set, but it is recommended to also use different data-sets. This way the presence of the different scenarios and the effect they have on the maximum queue discharge rate can verified. For this it would be advantageous to use a larger data-set. This way the different scenarios could be split up into more different groups, where each group still contains a reasonable number of phases. When the same conclusions can be obtained from a different data-set, the p-value will be lower and the conclusions will be more reliable.

Some scenarios that were covered in this thesis are also worth researching more in depth, in this data-set or on a different and larger data-set. Perhaps it is also worth to do a staged experiment.

For example, the estimations for maximum queue discharge rate for the different front densities showed a strong negative correlation. Because different studies that focused on the total density showed a positive correlation with the queue discharge, this scenario might be interesting to research more. Here it is possible to look at how the density in the front affects the queue discharge, instead of the maximum queue discharge or how different local densities affect the maximum queue discharge rate, like the density at the middle of the queue. Another interesting focus would be to look at the difference in densities between the front of the queue and the back of the queue. This research only looked at the density at the front of the queue (or the back), without taking into account differences of the density at the rest of same the queue, or the length of the total queue.

Another scenario that might be worth looking more into is the discharge time of the first cyclists and the average headway. As was concluded earlier, the discharge time and the average headway was positively correlated with the maximum queue discharge rate, even though the average headway of the constrained cyclist should become less for the maximum queue discharge rate to get higher. It would be an interesting focus to look more in depth into how the headway distribution (including the distribution for the constrained cyclists and the distribution for the cyclists with free flow) changes when the discharge time of the first cyclists becomes higher.

Combining the two scenarios mentioned above, another focus for future research would be to look more generally on the effect the first few cyclists have on the maximum queue discharge rate. Because both of these scenarios showed that the flow characteristics of the first cyclists can have a big influence on the maximum queue discharge rate.

The scenario ’overtaking’ is also worth researching more. The results showed a strong positive correlation with the maximum queue discharge, but the p-value was too high and there were very few data points to base this correlation on. So more research is needed to establish if there is a real relation between the scenario and the maximum queue discharge rate, or if the correlation was just a coincidence.

Finally, a last recommendation for future research is to look how the conclusions from this research can be used to design a new intersection or to improve the existing intersections.

Bibliography

CROW. (2016). *Ontwerpwijzer fietsverkeer*. CROW.

Goñi-Ros, B., Yuan, Y., Daamen, W., & Hoogendoorn, S. P. (2018). Empirical analysis of the macroscopic characteristics of bicycle flow during the queue discharge process at a signalized intersection. *Transportation Research Record: Journal of the Transportation Research Board*, *2672*(36), 51–62. https://doi.org/10.1177/0361198118790637

Hoogendoorn, S. P. (2005). Unified approach to estimating free speed distributions. *Transportation Research Part B: Methodological*, *39*(8), 709–727.

Hoogendoorn, S. P., & Daamen, W. (2016). Bicycle headway modeling and its applications. *Transportation Research Record: Journal of the Transportation Research Board*, *2587*(1), 34–40. https://doi.org/10.3141/2587-05

Hoogendoorn, S. P., & Knoop. (2013). Traffic flow theory and modelling. In B. van Wee, J. A. Annema, D. Banister, & van Wee (Eds.), *The transport system and transport policy* (pp. 125–159). Edward Elgar Publishing.

Jaadi, Z. (2021, December 12). *Everything you need to know about interpreting correlations*. Retrieved June 2, 2022, from https://towardsdatascience.com/eveything-you-need-to-know-about-interpreting-correlations-2c485841c0b8

Minderhoud, M. M., Botma, H., & Bovy, P. H. L. (1997). Assessment of roadway capacity estimation methods. *Transportation Research Record*, *1572*(1), 59–67. https://doi.org/10.3141/1572-08

OPEX resources. (2019, May 14). *Interpreting the pearson coefficient*. Retrieved June 2, 2022, from https://opexresources.com/interpreting-pearson-coefficient/

Transportation Research Board. (2010). *Hcm 2010: Concepts*. Transportation Research Board.

Wierbos, M. J., Knoop, V. L., Bertini, R. L., & Hoogendoorn, S. P. (2021). Influencing the queue configuration to increase bicycle jam density and discharge rate: An experimental study on a single path. *Transportation Research Part C: Emerging Technologies*, *122*, 102884. https://doi.org/10.1016/j.trc.2020.102884

Wierbos, M. J., Knoop, V. L., Goñi-Ros, B., & Hoogendoorn, S. P. (2020). The influence of jam density and merging cyclists on the queue discharge rate. *Journal of Advanced Transportation*, *2020*, 1–10. https://doi.org/10.1155/2020/9272845

Yuan, Y., Goñi-Ros, B., Poppe, M., Daamen, W., & Hoogendoorn, S. P. (2019). Analysis of bicycle headway distribution, saturation flow and capacity at a signalized intersection using empirical trajectory data. *Transportation Research Record: Journal of the Transportation Research Board*, *2673*(6), 10–21. https://doi.org/10.1177/0361198119839976

A

Coding

Introduction

This appendix will give more information about the programming parts of this thesis. First, the modules used will be described and after that some parts of the code will be displayed.

Modules

Because the data is initialized in matlab, it has to be converted to python. For this `scipy.io.loadmat` is used. This will then be processed using mainly `numpy` and `pandas`. The plots were made using `matplotlib.pyplot` and `scipy` was used for some specific functions, like calculating the Pearson correlation coefficient.The biggest part of the coding is done in the program PyCharm, however at the beginning some coding was also done in the program Jupyter Lab.

Code

In this section some important part of the code is displayed. The code is further elaborated with comments. A comment is preceded by a #.

Headway distribution

The following code is for converting the matlab code to python, getting the headway distribution and some other variables, and storing it in a separate file.

```
# Import modules
import scipy.io as sio
import pandas as pd
import numpy as np
import os

def get_headway():
    # Get the right directories for the matlab files.
    # This could probably have been done in a less complicated way
    absolute_path = os.path.abspath(__file__)
    file_directory = os.path.dirname(absolute_path)
    path_1 = os.path.join(file_directory, 'Matlab data')
    path = os.path.join(path_1, 'Trajectory_data_for_analysis')


    database = dict() # Create a dictionary to store the data in
    Xsl = 28.7  # X-position of the stop line

    # This is the dataframe all the data will be stored in.
    headway_database = pd.DataFrame(columns=['Phase', 'Cyclist', 'X', 'Y', 'Time', 'Leader',
                                             'Headway', 'X0', 'Y0'])

    # 'Phase' is the number of the phase (they have all been renumbered to get a number form
                                             1 to 59.
```

```
# 'Cyclist' is the number of the cyclist
# 'X', 'Y' and 'Time' are the variables when a cyclist crosses the stop line.
# 'Leader' is the number of the cyclist, the cyclist has been following and 'Headway' is
                                                the time headway.
# 'X0' and 'Y0' are the coordinates at the beginning of the phase.

# Store all the matlab data in a python dictionary
for i in range(1, 60):
    path_file = os.path.join(path, f'{i}')
    database[i] = sio.loadmat(path_file)

k = 0
x = Xsl

for key in database:

    # Loop over every cyclist
    for i in range(len(database[key]['Trajectories'][0])):
        y = 0
        t = 0

        # Get the start coordinates
        x0 = database[key]['Trajectories'][0][i][0][0]
        y0 = database[key]['Trajectories'][0][i][0][1]

        # Determine at which points the cyclists crosses the stop line, by looping over
                                                every point in the trajectory.
        for j in range(len(database[key]['Trajectories'][0][i])):
            if database[key]['Trajectories'][0][i][j][0] >= Xsl:

                # Interpolate between point before and after stop line
                x1 = database[key]['Trajectories'][0][i][j][0]
                x2 = database[key]['Trajectories'][0][i][j - 1][0]

                y1 = database[key]['Trajectories'][0][i][j][1]
                y2 = database[key]['Trajectories'][0][i][j - 1][1]

                t1 = database[key]['Trajectories'][0][i][j][2]
                t2 = database[key]['Trajectories'][0][i][j - 1][2]

                y = np.interp(Xsl, [x1, x2], [y1, y2])
                t = np.interp(Xsl, [x1, x2], [t1, t2])
                break

        # If a cyclist has never crossed the stop line, its trajectory will be
                                                interpolated up to the stop line,
                                                based on the last two data points

        if t == 0 and y == 0:
            x1 = database[key]['Trajectories'][0][i][j][0]
            x2 = database[key]['Trajectories'][0][i][j - 1][0]

            y1 = database[key]['Trajectories'][0][i][j][1]
            y2 = database[key]['Trajectories'][0][i][j - 1][1]

            t1 = database[key]['Trajectories'][0][i][j][2]
            t2 = database[key]['Trajectories'][0][i][j - 1][2]

            y = np.interp(Xsl, [x1, x2], [y1, y2])
            t = np.interp(Xsl, [x1, x2], [t1, t2])

        # Add data to dataframe
        headway_database.loc[k] = [key, i + 1, x, y, t, None, None, x0, y0]

        k += 1

k = 0

# Determine the leader and the headway of each cyclist. This is done in a separate loop
                                                to take into account that a cyclist can be
                                                overtaken by another cyclist, so the
                                                leader can be a cyclist with a higher
```

```
                                              number.
    for key in database:
        for i in range(len(database[key]['Trajectories'][0])):

            leader = None
            headway = None

            df = headway_database.loc[headway_database['Phase'] == key]

            for m in range(len(database[key]['Trajectories'][0])):
                # See equation 3.4
                if np.abs(df['Y'].iloc[m] - df['Y'].iloc[i]) <= 0.5 and df['Time'].iloc[i] -
                                                              df['Time'].iloc[m] >= 0 and m
                                                              != i:
                    leader = m + 1
                    headway = df['Time'].iloc[i] - df['Time'].iloc[m]  # See equation 3.5
                    headway_database['Leader'].loc[k] = leader
                    headway_database['Headway'].loc[k] = headway

            k += 1
    # Write dataframe to a file
    headway_database.to_csv('database', index=False)

    return headway_database
```

Capacity estimation

The following code was written to estimate the capacity, based on the equations given in 3.3. The capacity estimation is written in a so called `Class`, with several functions. This way it makes it easy to update variables and call functions, when needed. In a later section, it will be made clear how this class can be used to estimate the capacity.

```
# Import modules
import matplotlib.pyplot as plt
import numpy as np

class Capacity():

    # This is the initializer. It takes as input a dataframe with the headway distribution (
                                              as given by the get_headway() function)
                                              and the amount of bins that will be used.
    # The rest of the variables listed are initialized automatically, by the functions called
                                               in the initializer.
    def __init__(self, df, bins, ranges=(0,8)):

        self.df = df # Dataframe with total headway distribution
        self.bins = bins
        self.ranges = ranges
        self.tbins = 200
        self.T = 2

        self.lambda_hat = None
        self.A_hat = None
        self.ff = None # Dataframe with free flow headway distribution
        self.get_A_lambda()

        self.bins_count = None
        self.sum_count = None
        self.pdf = None
        self.size = None
        self.t = None
        self.get_histogram()

        self.succeed = None

    # This function gives a graph with the survival function, which is needed to estimate T*
    def t_graph(self):

        count, bins_count = np.histogram(self.df['Headway'].dropna(), bins=self.tbins, range=
                                                    (0, 8))
```

```
        pdf = count / sum(count)
        cdf = np.cumsum(pdf)
        surv = 1 - cdf

        plt.figure(figsize=(8, 6))
        plt.step(bins_count[1:], surv)
        plt.yscale('log')

        plt.ylim([10 ** -2, 1])
        plt.grid()
        plt.xlabel('Time headway in seconds')
        plt.ylabel('S(h) = 1 - F(h)')
        plt.show()

    # This function calculated lambda_hat and A_hat
    def get_A_lambda(self):
        self.ff = self.df.loc[self.df['Headway'] > self.T]
        m = len(self.ff)
        n = len(self.df['Headway'].dropna())

        self.lambda_hat = m / np.sum(self.ff['Headway'] - self.T)
        self.A_hat = m / n * np.exp(self.lambda_hat * self.T)

    # This function changes the distribution in a histogram
    def get_histogram(self):
        count, self.bins_count = np.histogram(self.df['Headway'].dropna(), bins=self.bins,
                                               range=self.ranges)
        self.size = self.bins_count[2] - self.bins_count[1]
        self.bins_count = self.bins_count - 0.5 * self.size
        self.t = self.bins_count[1:]
        self.sum_count = sum(count)
        self.pdf = count / self.sum_count / self.size

    # Get a first estimation for the free headway distribution
    def r0(self):
        r0_est = self.A_hat * self.lambda_hat * np.exp(-self.lambda_hat * self.t) / self.
                                               sum_count / self.size

        return r0_est

    # Get an estimate for r1. This function will be iterated.
    def r1(self, r_est, phi_hat):
        #  Calculate first part of product
        a = self.A_hat * self.lambda_hat * np.exp(-self.lambda_hat * self.t)

        # Calculate integrals
        f_tot = np.sum(self.pdf)
        r_tot = np.sum(r_est)
        f_int = np.cumsum(self.pdf)
        r_int = np.cumsum(r_est)

        # Calculate second part of the product
        b = ((f_tot - r_tot) - (f_int - r_int)) * self.size
        c = 1 - b / phi_hat

        return a * c

    # Get an estimation for phi, based on the estimation for r1
    def phi_est(self, r_est):
        r_est[r_est < 0] = 0   ## Maybe this line not needed??
        phi = np.sum(self.pdf - r_est) * self.size
        return phi

    # This is the function that estimates the capacity
    # It uses the functions and variables given above
    def get_capacity(self, phi=0.9, max_it=1000, plot=True):

        r0_est = self.r0()
        r_est = r0_est
        text = ”\nIterations failed”
```

```
        # Start iterations. The default is a maximum of 1000 iterations
        # Stop the iterations when the error gets very small
        for i in range(max_it):
            r_est = self.r1(r_est, phi)
            phi = self.phi_est(r_est)
            error = np.sum((r_est - r0_est) ** 2)
            r0_est = r_est
            if error < 10 ** -8:
                text = "\nIterations are successful"
                self.succeed = True
                break

        print(text) # Says is iterations failed or succeeded

        g = self.pdf - r_est # Get constrained headway distribution
        g[self.t > self.T] = 0 # Makes sure the the constrained headway distribution is zero
                                        when headway is above T*.
        g[g < 0] = 0 # Makes sure the constrained headway distribution does not get below 0.

        # Plots the capacity estimation, if wanted
        if plot is True:
            plt.figure(figsize=(15, 10))
            plt.step(self.t, self.pdf, '--', label='Total headway distribution', linewidth=2)
            plt.step(self.t, r_est, label='Free flow', linewidth=3, alpha=0.8)
            plt.step(self.t, g, label='Constrained', linewidth=3, alpha=0.8)

            if text == "\nIterations failed":
                plt.annotate('FAILED', (3, 0.5), color='red', fontsize=100)

        if text == "\nIterations failed":
            self.succeed = False

        # Calculate mean empty zone
        k = g * self.size * self.sum_count
        g_tot = np.sum(k)
        EX = np.sum(k * self.t) / g_tot
        capacity = 3600 / EX # Calculate the capacity per hour, by multiplying with 3600
        if plot is True:
            plt.axvline(x=EX, color='black', linestyle='--', label='Mean empty zone')
            plt.legend()
            plt.show()

        # Print the results
        print(f'Results:')
        print(f'Mean empty zone: {EX}s')
        print(f'Capacity: {capacity} cyclists / hour')
        print(f'Ratio constrained: {phi}')
        print(f'Lambda: {self.lambda_hat}')
        print(f'A: {self.A_hat}')

        return EX, capacity, phi
```

Density at the front and the back of the queue

First, a function was written to determine the density at the front and the back of the queue. Then said function was used, in combination with the function to estimate the capacity (or in this case, the maximum queue discharge rate) and code was written to make a plot.

```
import pandas as pd

# The variables front_c and back_c are for the number of cyclists the density calculation is
                                               done for
# The input is the dataframe with the headway distribution
def density(dataframe, front_c=5, back_c=5):

    # Create an empty dataframe to store the data
    data = pd.DataFrame(columns=['Phase', 'density', 'front_density', 'back_density', '
                                                   traffic flow'])

    # Loop over every phase
```

```
    for phase in range(1, 60):

        # Create for every phase a new dataframe, with only data from that specific phase
        dff = dataframe.loc[dataframe['Phase'] == phase]

        # Get number of cyclists and discharge time for all cyclists
        n = len(dff)
        T = dff.iloc[-1]['Time'] - dff.iloc[0]['Time']

        # Get density in the front
        # Locations of the start positions are used
        if dff.iloc[0]['X0'] <= 28.7:
            density_t = len(dff) / (2 * (28.7 - dff.iloc[-1]['X0']))
            density_f = front_c / (2 * (28.7 - dff.iloc[4]['X0']))

        else: # When first cyclist has passed the stop line, calculate the density from that
                                                      point, instead of from the stop line
            density_t = len(dff) / (2 * (dff.iloc[0]['X0'] - dff.iloc[-1]['X0']))
            density_f = front_c / (2 * (dff.iloc[0]['X0'] - dff.iloc[4]['X0']))

        # Get the density of the last cyclists
        density_b = back_c / (2 * (dff.iloc[-5]['X0'] - dff.iloc[-1]['X0']))

        # Store everything in the dataframe
        data.loc[phase] = [phase, density_t, density_f, density_b, n / T]

    phases = [[], [], [], [], [], [], [], []]
    phases_front = [[], [], [], [], [], [], [], [], []]
    phases_back = [[], [], [], [], [], [], [], []]

    # This loop stores every phase in a group, with an interval of 0.2 cyclists / m2
    for i in range(59):
        phases[int(data['density'].iloc[i] // 0.2 - 1)].append(i + 1)
        phases_front[int(data['front_density'].iloc[i] // 0.2 - 1)].append(i + 1)
        phases_back[int(data['back_density'].iloc[i] // 0.2 - 1)].append(i + 1)

    return data, phases, phases_front, phases_back
```

The code below shows the process for the maximum queue discharge estimation and the plotting of the graphs.

```
# Import modules, including the function to calculate the density and the Capacity class
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np
import scipy.stats as sst
from Capacity_estimation import Capacity
from Density import density

# Load the headway data from the earlier created file and store it in a dataframe
df = pd.read_csv('database')


xd = [[[], [], []] for i in range(5)]
cap = [[[], [], []] for i in range(5)]


for i in range(2, 7):
    # Get the density for the first and last i number of cyclists
    dff, dens1, dens2, dens3 = density(df, front_c=i, back_c=i)
    dens = [df, dens1, dens2, dens3]

    # Loop over total density, front density and back density
    for k in [1, 2, 3]:
        # This loop merges groups when they have 2 or less phases in them
        for l in range(1, len(dens[k]) + 1):
            if len(dens[k][-l]) <= 2 and l != len(dens[k]):
                dens[k][-l - 1].extend(dens[k][-l])
                dens[k][-l] = []
            elif len(dens[k][-l]) <= 2 and l == len(dens[k]):
                dens[k][-l + 1].extend(dens[k][-l])
```

```
                dens[k][-1] = []

        # Empty groups are deleted
        a = [x for x in dens[k] if x]
        dens[k] = a

        xd[i - 2][k - 1] = []
        cap[i - 2][k - 1] = []

        # Loop over every (new) group
        for m in range(len(dens[k])):

            # Calculate the capacity over one group
            capacity = Capacity(df.loc[df['Phase'].isin(dens[k][m])], 500)
            cap_new = capacity.get_capacity(plot=False)[1]

            # If capacity estimation succeeded, add results to a list
            if capacity.succeed is True:
                xd[i - 2][k - 1].append(dff.loc[dff['Phase'].isin(dens[k][m])]['density'].mean())
                cap[i - 2][k - 1].append(cap_new)

            # If estimation failed, try again while merging said group with the group before it.
            elif capacity.succeed is False and m > 0:
                new_dens = dens[k][m - 1] + dens[k][m]
                x_new = dff.loc[dff['Phase'].isin(new_dens)]['density'].mean()
                capacity = Capacity(df.loc[df['Phase'].isin(new_dens)], 500)
                cap_new = capacity.get_capacity(plot=False)[1]

                # If succeeded, the capacity estimation from the previous group is overwritten with this new estimation
                # If not, this group will not be taken into account
                if capacity.succeed is True:
                    dens[k][m - 1] = new_dens
                    xd[i - 2][k - 1][m - 1] = x_new
                    cap[i - 2][k - 1][m - 1] = cap_new

# The following code creates the plot
# Note: also a plot of the total density is made, but this one will not be used.
# However, the plots for the front and back density also include a line with the total density
densities = ['Total', 'Front', 'Back']
x_tot = np.array([0.35, 0.65])

data_color = plt.cm.inferno(np.linspace(0.8, 0.3, 5))
fit_color = plt.cm.inferno(np.linspace(0.8, 0.3, 5))
print('color', data_color)

for d in range(3):
    plt.figure(figsize=(10, 8))
    plt.xlabel('Density in cyclists/m^2')
    plt.ylabel('Capacity in cyclist/hour')
    plt.title(f'{densities[d]} density')

    plt.plot(xd[0][0], cap[0][0], 'o', color='black', label=f'Data: {densities[0]} density')

    p = np.polyfit(xd[0][0], cap[0][0], 1)
    cap_fit = p[0] * x_tot + p[1]
    plt.plot(x_tot, cap_fit, '--', color='black', alpha=0.7, label=f'Fitted line: {densities[0]} density')

    x_all = []
    cap_all = []

    for i in range(5):

        plt.plot(xd[i][d], cap[i][d], 'o', color=data_color[i], label=f'Data points, density, c={i + 2}')
```

```
        p = np.polyfit(xd[i][d], cap[i][d], 1)
        cap_fit = p[0] * x_tot + p[1]
        plt.plot(x_tot, cap_fit, '--', color=fit_color[i], alpha=0.8, label=f'Fitted line, c=
                                              {i + 2}')

        x_all += xd[i][d]
        cap_all += cap[i][d]

    # Create correlation coefficient and p-value for front and back density and print those
                                          values
    cor, p = sst.pearsonr(x_all, cap_all)
    print(f'Correlation {densities[d]}: {cor}')
    print(f'p-value {densities[d]}: {p}')


    plt.legend()

plt.show()
```

Cyclists standing behind each other or next to each other while waiting

The following code was used to determine if there was enough space next to a cyclist for another cyclist to stand. The code that was used for grouping the data, to estimate the maximum queue discharge rate and to make the plots is almost the same as the code used for the scenario about the density at the front and the back of the queue, so it will not be displayed again.

```
def row(df, width=1):

    # Create an empty dataframe to store the data in
    data = pd.DataFrame(columns=['Phase', 'space', 'cyclist'])

    # Loop over every phase
    for phase in range(1, 60):
        dff = df.loc[df['Phase'] == phase]

        count = 0
        front = False
        back = False
        space = True
        cyclist = '-'
        groups = [[], [], [], [], [], [], [], [], []]

        # This loops checks for every cyclist if there is enough space next to it.
        # If there is not enough space: the paramter "space" will be set to False
        for i in range(len(dff)):
            if ((dff.iloc[i]['Y0'] >= 8 + width or dff.iloc[i]['Y0'] >= 10 - width)
                    and i + 1 < len(dff)):
                for j in range(len(dff)):
                    if (np.abs(dff.iloc[i]['X0'] - dff.iloc[j]['X0']) < 1
                            and np.abs(dff.iloc[i]['Y0'] - dff.iloc[j]['Y0']) < 2 and i != j)
                                                                            :

                        space = False

                    elif np.abs(dff.iloc[j]['X0'] - dff.iloc[i]['X0'] < 2
                            and np.abs(dff.iloc[i]['Y0'] - dff.iloc[j]['Y0']) < 2 and i !
                                                                                = j):

                        front = True

                    elif np.abs(dff.iloc[i]['X0'] - dff.iloc[j]['X0'] < 2
                            and np.abs(dff.iloc[i]['Y0'] - dff.iloc[j]['Y0']) < 2 and i !
                                                                                = j):

                        back = False

                    if front is True and back is True:
                        space = False
```

```
                # If there is enough space, it will be counted
                # The number of the cyclist is also stored in the parameter "cyclist"
                # This is to make it easy to manually check the correctness of the function
                if space is True:
                    count += 1
                    cyclist += f'{i + 1}-'
                else:
                    space = True

        # The results are stored in the empty dataframe
        data.loc[phase] = [phase, count, cyclist]

    # Every phase is stored in the corresponding group
    for i in range(59):
        groups[int(data.iloc[i]['space'])].append(i + 1)

    return data, groups
```

Cyclists overtaking each other

The following code shows how the number of cyclists that were overtaking each other was counted. The code used to sort the groups, to estimate the maximum queue discharge rate and to make the plots will not be included. To get an idea of how this is done, see the code for the density at the front and the density at the back of the queue.

```
df = pd.read_csv('database')
phase_count = 0
groups = dict()

# Loop over every phase
for phase in range(1, 60):
    phase_count = 0
    dff = df.loc[df['Phase'] == phase]

    # Loop over every cyclist
    for i in range(1, len(dff) + 1):
        # Check for every other cyclist if cyclist i has overtaken cyclist j
        # This is done by checking the time stamp when each cyclist crossed the stop line
        for j in range(1, i):

            a = dff.loc[dff['Cyclist'] == i]['Time']
            b = dff.loc[dff['Cyclist'] == j]['Time']
            if float(a) < float(b):
                overtaken = True
                text = f'Phase: {phase}. Cyclist {i} has overtaken cyclist {j}'
                phase_count += 1

    # Add the results to a dictionary
    if phase_count in groups:
        groups[phase_count] = groups[phase_count] + [phase]
    else:
        groups[phase_count] = [phase]
```

The first cyclists take a long time before leaving

The following code shows the function that is used to determine the discharge time of the first cyclists. The code used to sort the groups, estimate the maximum queue discharge and to make the plots is not included. To get an idea of how this is done, see the code for the density at the front and the density at the back of the queue.

```
# The parameter "c" is the number of cyclists to do the calculation for
def get_time(df, c=1):
    # This code creates an absolute path to the original matlab file
    # This could probably have been done easier
    absolute_path = os.path.abspath(__file__)
    file_directory = os.path.dirname(absolute_path)
    path_1 = os.path.join(file_directory, 'Matlab data')
    path = os.path.join(path_1, 'Trajectory_data_for_analysis')
```

```
# Create an empty dictionary to store the matlab data in
database = dict()
# Create an empty dataframe to store the data
time_database = pd.DataFrame(columns=['Phase', 'Green_time', 'Leave_time', '
                                                Discharge_time'])
# Yes, this list of lists could have also been created in an easier way
groups = [[], [], [], [], [], [], [], [], [], [], [], [], [], [], [], [], [], [], [], [],
                                                 [], [], [], [], [], [], [], [], [], [], [
                                                ], [], []]

# Loop over every phase
for phase in range(1, 60):
    # Load the matlab data
    path_file = os.path.join(path, f'{phase}')
    database[phase] = sio.loadmat(path_file)

    dff = df.loc[df['Phase'] == phase]

    # Store the timestamp the light goes green
    green = database[phase]['greenPhaseStartTime'][0][0]

    # Check if the first cyclist crosses the stop line after the light went green
    # If not, count the second cyclist as the first cyclist
    if dff.iloc[0]['Time'] - green > 0:
        leave = dff.iloc[c - 1]['Time']
    else:
        leave = dff.iloc[c]['Time']

        if leave - green <= 0:
            leave = dff.iloc[c + 1]['Time']

    # Calculate the discharge time
    discharge = leave - green

    # Store the data in the empty dataframe
    time_database.loc[phase - 1] = [phase, green, leave, discharge]

# Sort all the phases in a group, corresponding to the the discharge time
for i in range(59):
    groups[int(time_database.iloc[i]['Discharge_time'] // 0.3)].append(i + 1)

return time_database, groups
```

Later on in the code, when all the groups are sorted the discharge time of each group will be calculated. The average headway can be easily calculated after that by dividing the discharge time by the number of cyclists.

B

Results: Tables

In this appendix the numeric results of the calculations for the maximum queue discharge rate are shown. Here n is the size of the group, so the number of phases in that specific group, C the maximum queue discharge rate in cyclists/hour, $\hat{\phi}$ the ratio of constrained cyclists.

The front or the back of the queue is very dense

Table B.1 shows the results of the maximum queue discharge rate of the density at the front. Here c is the number of cyclists the calculations has been done for and "density" was the average density of a group in cyclists / square meter. The results for the density at the back can be found in table B.2

c	n	density	C	$\hat{\phi}$
2	32	0.502	2769	0.307
	27	0.442	2974	0.297
3	47	0.461	2958	0.287
	12	0.525	2735	0.354
4	27	0.442	2974	0.297
	27	0.493	2725	0.315
	5	0.550	2931	0.251
5	9	0.403	3039	0.461
	32	0.461	2896	0.274
	18	0.535	2739	0.272
6	27	0.442	2974	0.297
	22	0.480	2796	0.298
	10	0.550	2724	0.324

Table B.1: The exact results of the maximum queue discharge estimation for the density at the front

Cyclists standing behind each other while waiting

The results from this scenario can be found in table B.3. Here w is the width in meters of the cyclist and "space" is the average number of free spaces in the group.

Cyclists overtaking each other

The results from this scenario can be found in table B.4. Here o is the average number of cyclists that was overtaken in one phase.

c	n	density	C	$\hat{\phi}$
2	33	0.513	2957	0.217
	26	0.426	2767	0.428
3	55	0.482	2914	0.300
	4	0.376	2767	0.333
4	54	0.463	2916	0.303
	5	0.595	2821	0.306
5	14	0.399	2857	0.400
	41	0.488	2881	0.289
	4	0.598	2718	0.279
6	4	0.376	2767	0.333
	22	0.435	2743	0.442
	29	0.501	3004	0.203
	4	0.598	2718	0.279

Table B.2: The exact results of the maximum queue discharge estimation for the density at the back

w	n	space	C	$\hat{\phi}$
0.6	9	0	2711	0.386
	39	1.36	2943	0.234
	6	3	2746	0.404
	5	4	2783	0.465
0.75	9	0	2711	0.386
	40	1.35	2971	0.222
	5	3	2633	0.451
	5	4	2783	0.465
0.8	9	0	2711	0.386
	40	1.35	2971	0.222
	5	3	2633	0.451
	5	4	2783	0.465
1.0	10	0	2704	0.418
	42	1.31	2959	0.219
	7	3.29	2697	0.578

Table B.3: The exact results of the maximum queue discharge estimation for the number of spaces in the queue

n	o	C	$\hat{\phi}$
7	0	2656	0.581
24	0.71	2822	0.340
13	2	2754	0.444
9	3	2810	0.454
22	4.27	2982	0.206

Table B.4: The exact results of the maximum queue discharge estimation for the number of times a cyclist was overtaken.

The first cyclists are taking a long time before leaving

The results from the discharge time of the first cyclists can be found in table B.5. Here c is the number of cyclist the calculation has been done for, d is the discharge time in seconds and h the average time headway.

c	n	d	h	C	$\hat{\phi}$
1	6	1.39	1.39	2689	0.529
	13	1.62	1.62	2780	0.346
	7	2.27	2.27	2702	0.447
	9	2.81	2.81	2720	0.731
	5	3.61	3.61	2948	0.583
2	5	1.40	0.70	2675	0.371
	3	2.52	1.26	2571	0.502
	9	2.84	1.42	2888	0.306
	7	3.12	1.56	2739	0.612
	5	3.51	1.75	2871	0.673
3	4	1.79	0.60	2580	0.290
	3	2.53	0.85	2427	0.403
	8	2.86	0.95	2830	0.245
	6	3.20	1.07	2687	0.381
	8	3.71	1.24	2790	0.267
	5	4.30	1.43	3095	0.773
	7	4.74	1.48	2695	0.485
	3	5.69	1.90	3113	0.233
4	7	3.41	0.85	2589	0.402
	13	4.33	1.08	2864	0.339
	5	4.90	1.23	2798	0.847
	5	5.26	1.32	2933	0.308
	4	5.97	1.49	2770	0.578
	3	7.60	1.90	3074	0.434
5	4	3.59	0.71	2547	0.418
	8	4.37	0.87	2551	0.295
	9	5.59	1.12	3018	0.574
	4	6.53	1.31	2674	0.622
	4	7.56	1.51	2970	0.433

Table B.5: The exact results of the maximum queue discharge estimation for discharge time and the average headway for the first cyclists