Traffic Load Balancing to improve Urban Air Quality

Master thesis

by

P. L. de Goffau

to obtain the degree of Master of Science
in Embedded Systems
at Delft University of Technology, faculty EEMCS,
to be defended publicly on Thursday January 19, 2017 at 15:00.

Student number:	4098064	
Project duration:	November 1, 2015 – January 19, 2017	
Thesis committee:	Prof. dr. C. Witteveen,	Faculty EEMCS, TU Delft
	Dr. M. T. J. Spaan,	Faculty EEMCS, TU Delft, daily supervisor
	Dr. V. L. Knoop,	Faculty CEG, TU Delft
	Dr. P. B. Bakker	Cygnify BV

An electronic version of this thesis is available at http://repository.tudelft.nl/.

TU Delft
Delft University of Technology

Preface

The Road Not Taken
Two roads diverged in a yellow wood,
And sorry I could not travel both
And be one traveler, long I stood
And looked down one as far as I could
To where it bent in the undergrowth;

Then took the other, as just as fair,
And having perhaps the better claim,
Because it was grassy and wanted wear;
Though as for that the passing there
Had worn them really about the same,

And both that morning equally lay
In leaves no step had trodden black.
Oh, I kept the first for another day!
Yet knowing how way leads on to way,
I doubted if I should ever come back.

I shall be telling this with a sigh
Somewhere ages and ages hence:
Two roads diverged in a wood, and I—
I took the one less traveled by,
And that has made all the difference
Robert Frost (1874 – 1963)

In his poem 'The Road Not Taken' Robert Frost described the unevitable choice between two paths. A traveller makes a route choice based on the signs that he is observing. The traveller decides to take the road that is less traveled by. The better the traveller is able to observe which path is the most interesting, the better its journey would be. Each traveller decides which path will serve him well, based on tracks that have been left by previous travellers. In this research the traveller represents a driver that has to navigate from a point of departure to a destination. The goal of the research is to disperse traffic, such that drivers will follow the roads less traveled by – just like the traveller.

The main theme of the poem is making a choice. The choice the traveller made, determined the further course. The second stanza shows that the choice was made rather impulsive, as if he could come back to the crossing if the path would have disappointed eventually. In the third verse however, the traveller realizes that one choice initiated a set of new choices, such that it becomes unlikely that he will return to the crossing. This concatenation of choices is also present in my life, where choices in my study program have led eventually to this graduation report.

Fortunately, several people were around to help me find my way. I am especially thankful that I had such a great supervisor around, Matthijs Spaan. He has spent a lot of time and put a lot of effort in me and this project. I am grateful for the trust that he and the departement showed in me right from the start. Furthermore, I would like to thank Bram Bakker, whose valuable feedback helped me staying critical to my own work. It is very special to me that we could work together on an interesting idea with an unrestrained mind.

Finally, I would like to thank my wife, family and friends, who were essential for the completion of my study, because their support helped me reaching my objectives. It was refreshing to discuss my ideas with them in common language, which hopefully has led to a comprehensive graduation report.

P. L. de Goffau
Dordrecht, January 12, 2017

Contents

Abstract

The growth of traffic densities has increased the concentrations of air pollutants, especially in city centres. Although the natural immune system of the human body can withstand low levels of exposure to air pollutants, exposure to high levels of air pollutants can lead to adverse health effects and even death. These high exposure levels are often present at city centres, due to high traffic densities at these areas. Therefore, in this research is explored if exposure levels to air pollutants can be decreased by improving traffic distributions.

In this research the benefits and challenges are explored of a traffic load balancing system that has the objective to improve the air quality locally, while bounding the length of the detours that are needed to achieve this. Traffic dispersion is achieved by influencing routes of individual drivers by a collective navigation system. A weighted combination of travel time and air pollutant emissions is minimized to achieve lower air pollution exposure levels to citizens, while drivers maintain short travel times. Nitrogen Oxide (NOx) is used to bound the vehicle emissions locally, while the emission of Carbon Dioxide (CO_2) is minimized globally. For each car an emission trace is constructed using an emission model. The emission traces are combined to form a time-dependent emission map. This time-dependent emission map is used for the exploration of routes that have a lower impact on the health of people that live near these routes.

The approach is evaluated by simulation on real-life maps with an estimation of the traffic demand. The weighted combination approach is configured and compared to a base case where the drivers would only optimize their travel time. Using the simulation, it is shown that the exceedance of local thresholds of the NOx concentrations can be reduced, with only a small increase in travel time.

Furthermore is explored what the effect is if only a part of the drivers would participate in the constructed load balancing system, while the other drivers would optimize their own travel time. By simulation is shown that even if only a part of the drivers would participate in this traffic load balancing system, the air quality can still be improved with a marginal cost of additional travel time.

1 Introduction

One of the main environmental problems is the problem of growing concentrations of air pollutants in urban areas. Since the Industrial Revolution the air quality has become worse very rapidly [1], especially in urban areas [2]. High levels of exposure to air pollutants lead to health problems and shorter life expectancies [3]. Brunekreef and Holgate observed that exposure to ozone, particulate matter (PM) and a mixture of Nitrogen oxides have been associated with increases in mortality and hospital admissions due to respiratory and cardiovascular disease [4]. Therefore, it is desirable to avoid high exposure levels to these air pollutants.

A large part of the emission of air pollutants in cities stems from the combustion engines of cars and trucks. Together, the traffic flows have a large impact on the local concentrations of air pollutants.

Societies struggle with the air pollution problem, because on the one hand people want to reach all their destinations in the shortest possible travel time, while on the other hand – as local residents – they want to inhale fresh air. Because people interact and have relations with each other, they tend to gather together, which increases traffic densities, travel times and decreases the air quality caused by vehicle emissions.

1.1. Objective

This section describes the objective of this research, and the way in which the research is divided. Figure 1.1 shows that a driver often has several short routes at his disposal, which run through different areas with different air qualities. If drivers would base their route choices not only on travel time, but also on the local air quality, the local air quality could be improved. However, if all drivers would choose routes which run through the same area, the concentration of air pollutants becomes too high in those areas. This research explores if a method could prevent this by influencing route choices based on estimations of local air pollutant concentrations, global vehicle emissions and the individual travel time.

The objective of this research is to evaluate if the air quality can be improved by influencing the individual routes of drivers. Therefore, the main research question is formulated as follows.

- Can collective routing algorithms improve the air quality in cities, while preserving low travel times for individual drivers?

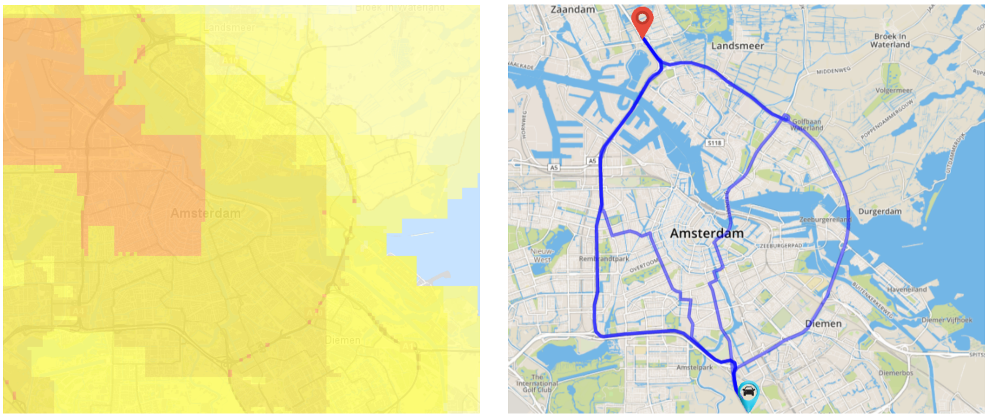
To solve this problem, the problem is divided into several subquestions. First a metric must be determined for the air quality. It must be specified which air pollutants are considered, and at which scale.

1. What is a good expression for the air pollution of a single vehicle, dependent on its vehicle type and speed?

From the emissions of single vehicles, a global emission model can be constructed, which can be used to compare different routing algorithms with each other.

2. What is a good emission model that can be used to compare different routing methods with each other?

An algorithm can be developed to solve the Shortest Path Problem for a single vehicle, where the weights of edges are calculated using a database that contains traffic information based on statistics and on the routes planned by other cars.



(a) The prevailing local estimations of the Air Quality Index from the RIVM in Amsterdam at 30 december 2016, 10:00 AM [5].

(b) A driver from South to North has 4 different routes at his disposal of which the travel times differ at most 20% from each other.

Figure 1.1: If drivers from South to North would choose routes that run through the eastern part of the city, the air quality at the western part of the city is improved, while the additional travel time for drivers is limited.

3. How can an efficient algorithm be created that solves the Shortest Path Routing problem on a graph of which the weights are dependent on emission information which is updated during the routing process according to the planned routes of other vehicles?

The algorithms that have been created must be evaluated on the objective to improve the air quality, while offering short routes for individual drivers.

4. What is the relative improvement of the air quality if these algorithms, which take the local air quality into account, have been used for the calculation of routes compared to the situation where each driver chooses its fastest possible route?

The algorithms may improve the air quality, but most presumably, this improvement of the air quality comes with the cost of longer travel times of drivers.

5. What is the relative growth of the average and maximum travel times for drivers that use these algorithms compared to the situation where each driver chooses its fastest possible route?

Most presumably, not all drivers are willingly to participate in a project like this, because the time investments would be regarded to high. Therefore, also the impact of lower participation rates on the air quality is explored.

6. What is the impact of lower participation rates on the growth of the total travel time and the improvement of the air quality?

1.2. RELATED WORK

To reduce the traffic densities in the heart of their cities and metropoles, governments have taken several measures. These measures are described in Section 1.2.1. Scientists are interested in improving the air quality and traffic densities in cities as well. Related research that has been performed on improving traffic distributions by the manipulation of routes of individual drivers is described in Section 1.2.2, for methods which focus on emissions, and in Section 1.2.3, for methods which focus on traffic densities.

1.2.1. GOVERNMENTAL TRAFFIC MANAGEMENT IN CITIES

The methods that have been used by governments to influence the traffic densities in cities can be categorized into two groups: banning all or part of the traffic from certain areas, or pricing of certain road segments or areas. Subsequently, road pricing can be divided into two different scopes: setting a penalty for entering an area with a vehicle with high emission rates (generally called low emission zones), or by pricing per road segment (road pricing).

TRAFFIC BAN

Several cities have set thresholds for the exposure of air pollutants to their citizens, with bounds on the number of days that these thresholds may be exceeded. Last year, the cities Beijing[1] and Rome[2] were closed for (part of the) motorized traffic for several days, because the number of days that the thresholds were exceeded, was higher than permitted. A traffic ban is a very unpopular measure, because people are restricted in their freedom, and unable to reach their destinations, like working places. Therefore, this solution cannot be regarded as a sustainable method to reduce the concentrations of air pollutants in cities.

LOW EMISSION ZONES

A less strict measure is the constitution of a zone where motorized traffic with high emissions pay a price for entering that zone. For instance, in the Netherlands, the local government of Rotterdam constituted such a low emission zone at the center of the city. Vehicles that have large emission rates are not allowed to drive in that region [6], and will be fined if they neglect the prohibition. The same measure was introduced to the center of the city of London, to exclude vehicles with large emissions from the city center. All vehicles that have been marked as highly pollutant are fined for every day they travel through the low emission zone. Since the introduction of the low-emission zone, the concentrations of particulate matter within the low emission zone have dropped by 2.5 to 3%, compared to just over 1% for areas just outside the zone [7].

Although low emission zones have been proven to be feasible implementations of a road pricing system in cities, they do not charge vehicles closely according to road usage, because the price is the same regardless of the time or road space that is spent by drivers in the zone.

ROAD PRICING

The idea of a low emission zone in London is based on the road pricing system that is introduced in Singapore [8]. Road pricing is a method that assigns a price to road segments that expresses their value. Road segments that are very popular are assigned higher prices. In Singapore, one of the main objectives that was put forward for the introduction of a road pricing system is the increasing average traffic speed and traffic flow. The traffic flow is increased by reducing the demand of driving at certain road sections by pricing them. Encouraged by lower prices drivers would change their time of departure, their routes or their means of transport. Based on the economical decisions of travellers, the traffic would be spread out over the infrastructure. Road pricing can then consequently also lead to a better air quality in busy city centres.

Singapore has used the Electronic Road Pricing (ERP) system since 1998 [8]. ERP gantries are placed at the most crucial road sections in the city center. Each driver is obliged to have an In-vehicle Unit (IU) built in at the dashboard of a vehicle. Every time the IU passes a gantry, it charges the price of a road section from a credit card. The time, the vehicle type and the expected traffic density determine the price of a road section.

Such a system could easily lead to unwanted driver behaviour. Drivers tend to look for certain loopholes. In Singapore, large queues were formed just after the time period in which road sections were more expensive. In order to discourage this behaviour, the time period with high prices is extended, after which this behaviour has vanished.

Although the traffic volumes have been decreased by 15% by the ERP system in Singapore, there is still plenty of room for improvement, especially by combining real-time traffic information into the prices of road sections. The price of road sections can be made dynamic, based on the current traffic density. The drawback of such a system is that drivers do not know the prices of different trajectories in advance.

May [9] formulated guidelines for the introduction of a road pricing system[3]. It is very complicated and expensive to construct a road pricing system that complies with all these guidelines. Therefore, in many countries, road pricing is only used for highways, by a toll booth system. Such implementations of road pricing systems are not easily expandable to city centres, and do therefore not comply with the guidelines of May [9].

1.2.2. ECO-ROUTING

Fortunately, there are other methods that can reduce traffic emissions (in cities). Global Positioning System (GPS) based navigation systems have been used increasingly by travellers. These devices are used to calculate

[1] BBC, China pollution: Factories closed by beijing smog, http://www.bbc.com/news/world-asia-china-35141188 (Accessed 19 december 2015).

[2] BBC, Italy smog: Milan and rome ban cars as pollution rises, http://www.bbc.com/news/world-europe-35188685 (Accessed 28 december 2015).

[3] Santos et al. [8] have formulated a summary of these guidelines.

the shortest path (mostly in time, or in distance) from a source to a destination. Furthermore, the navigation devices can also be configured to minimize emissions. Several methods have been developed for the global minimization of emissions, of which some are described in this Section.

Yunlong Zhang et al. [10] presented a methodology of assigning traffic in a network with the consideration of air quality. The traffic assignment problem is formulated as an optimization problem of an objective function that contains both the travel cost and on-road emissions. The on-road emissions have been stored cell-based, where emissions in one cell have influence on the nearby cells, assuming a Gaussian distribution. By means of a simple case study it is shown that minimizing the travel cost and reducing the air pollutants may not always be achieved at the same time. However, the proposed traffic assignment procedure can effectively reduce emission concentrations at locations with the worst air quality conditions. This approach does not take the current air quality into account while routing. It computes routes that globally have lower emissions, but locally the concentrations of air pollutants may still exceed the legal bounds.

Patil [11] constructed several static traffic assignment models in order to reduce tailpipe emissions in road transportation systems. Based on its speed, an approximation for every road section can be made of the driver's emission of air pollutants. Patil constructed two model-types, based on Wardrop's principles for the User Equilibrium (UE) and System Optimum (SO) [12]. In the environmental UE (E-UE), users are assumed to minimize their own vehicle emissions or fuel consumption. In the environmental SO (E-SO) the sum of the emissions of all drivers is minimized, while individual drivers could get a route assigned that does not have the minimum emissions. The objective function is a summation over all edges, where for each edge the traffic flow is multiplied with a factor that represents either the fuel consumption or the CO_2 emission. To analyse the systems, both models have been solved on two hypothetical networks. The travel time, CO_2 emissions and fuel consumptions have been compared for these two models, and the equivalent models that only consider travel time cost per edge. For the second network, which is the most realistic of the two hypothetical networks, the CO_2-based E-UE model increases the travel time, CO_2 and the fuel consumption with 23, 8 and 8% respectively. This increase in fuel-based UE is about 22, 3, and 3%. The total emissions are increased, because of decreased traffic flows. Reduced traffic flows leads to higher emissions per driver. Therefore, the author cannot guarantee that the system level emissions can be reduced when the travellers' aim is to reduce their own CO_2 or fuel. Further analysis needs to be carried out to find out under what situations the system level emissions increase. However, a reduction in emission level can be achieved if the drivers are routed to reach environmental SO. For this network, the reduction of CO_2 in the E-SO is about 4% compared to corresponding pollutants in Travel Time-UE; about the same percentage of reduction is observed for fuel. In this approach the emissions are minimized globally, while locally high concentrations of (in this case) CO_2 may occur.

Aziz et al. [13] chose CO as a representative for the air pollutants stemming from traffic. They combine CO emission rates, system-wide travel time and speed profiles into a new traffic assignment model, and compare this model with a traditional dynamic assignment model (where the only objective is to minimize the travel time). The model is formulated as a minimalisation problem with constraints. Using small hypothetical standard networks they show the effect of considering vehicle emissions on the route choice, for three different objective functions. First only travel time is considered in the objective function. The second model aims only for the minimization of the CO emission. In the main model both the travel time and the CO emission are reduced by expressing both objectives as monetary values. A CPLEX-solver is used to solve the mathematical programs. If both the CO emission and travel time have been expressed in monetary values, the CO emissions have been decreased by a factor of two, while the increase of the travel time is marginal. Like with Patil [11], this approach does not consider the local effects of high concentrations of air pollutants.

Fan Zhang et al. [14] have performed research on multi-criteria routing dynamic traffic assignment models. They have presented a simulation-based bi-level optimization method that solves the multi-criteria routing dynamic traffic assignment model iteratively using two different loops. In the inner loop, the travel times are updated, while the emission costs are considered to be fixed. Then, in the outer loop, the emission costs are calculated, after which a new traffic assignment is made. A multi-criteria user equalibrium is found when the solutions converge to an equilibrium. By using the traffic simulation model Dynasmart-P, the authors show that if drivers choose routes based on both travel time and emissions, the average travel time and emissions per vehicle decrease. The average travel time is reduced, because the number of congested links is reduced. Congested links have a higher impact on vehicle emissions and travel time. The study shows what the positive impact of presenting more routing information to drivers can be on the average travel time and the environment. However, concentrations of air pollutants may locally still be very high.

1.2.3. TRAFFIC DISPERSION ALGORITHMS

Besides the research that has been performed on eco-friendly routes, research has also been done on traffic dispersion. In this section, some traffic dispersion methods are described that have the objective to minimize the number and the length of traffic jams. These methods were an inspiration for the idea of this research, where the emission of vehicles is stored at a common graph, such that the emissions can be bound locally. Improved traffic distributions over the road network will usually reduce the number and length of traffic jams. Traffic dispersion can therefore lead to lower travel times and maybe also lower concentrations of air pollutants, which makes these methods interesting.

ANT-INSPIRED DYNAMIC TRAFFIC ROUTING

Alves et al. [15] used an Ant Colony Optimization technique to disperse the traffic. For multiple source-destination pairs the shortest routes have been determined by an ant-inspired algorithm, which finds short routes on the pheromone trace that is left by each ant. The pheromone traces are used to predict the traffic load during the pre-processing of a graph. The predicted load at edges is translated to a probability to choose a certain edge on every vertex. During the initialisation round the probabilities are set, after which new vehicle routing requests can be made. The algorithm proved to work accurately for a case study network involving the Singapore Expressway Network.

Dallmeyer, Schumann, Lattner and Timm investigated ant-inspired Traffic Routing in urban environments with a changing problem structure [16]. Dynamic traffic routing differs from static routing in the sense that the problem structure changes while solving the problem. The aim is to minimize the traveling time, instead of minimizing the traveling distance. The authors assume that cars can be enabled to communicate with each other and also with their environment. The information that they will share will help for the routing decisions. Each car determines its own route. Traffic congestion is caused by a too high number of vehicles that enter the same road segments. In order to avoid congestion an ant-based routing algorithm is used in each car that will improve the traffic distribution over the network. Each car acts as an ant by virtually storing a pheromone trace of their trajectory in a shared map. Based on the cumulative pheromone that is dropped by other cars the routing algorithms decide which route to take. The more pheromone on a certain path, the less the probability that the routing algorithm would select this path. Using the MAINSIM traffic simulator the authors show that the ant-based algorithm spreads the traffic over different routes, while an individual algorithm manages the traffic through a single path which will lead to huge traffic jams on those paths.

TRAFFIC ROUTING BASED ON BACKPRESSURE

Rui Zhang et al. [17] constructed an algorithm that disperses traffic based on the backpressure algorithm that is known from wireless communication networks. The authors have assumed in their research that an infrastructure was present which makes it possible to share information about the traffic density on edges between cars and intersections. Each car sends the edge on which it currently is to the intersections which are near to that car. Based on the traffic flow of the adjacent edges the intersection returns a dispersion rate for each adjacent edge the car may take. The car decides which edge he will take next based on this rate, and on its urgency to reach the destination as fast as possible. The dispersion rate can be used within a 'roulette wheel selection' method, where the dispersion rate indicates the probability of the edge that the car may take next. The urgency parameter is dependent on the difference between the current time and the preferred time of arrival of the car. The authors show that this approach can lead to a balance between user satisfaction and a system optimum.

1.3. APPROACH AND SCIENTIFIC CONTRIBUTION OF THIS RESEARCH

Eco-routing methods can lead to lower emissions, but locally high exposure levels to air pollutants may still be present. Therefore, in this research an emission model will not only be used for the comparison of routing methods, but the routing methods will also emphatically make use of this emission model, by sharing emission traces of drivers that have already calculated their paths. By this collective routing method the concentrations of air pollutants are kept low locally, while preserving small travel times of individual drivers. Besides the objective of low local concentrations, a global minimization of the emission of CO_2 is pursued as well.

The main difference with Yunlong Zhang et al. [10], Patil [11], Aziz et al. [13], Fan Zhang et al. [14] is that in this method the objective is to improve the air quality locally, while in the other researches only a global improvement of the air quality is pursued.

The method that is constructed in this research is comparable to the traffic dispersion methods that are suggested by [16] and [17], but the objective of this method is different, because vehicle emissions are considered as well. This different objective may lead to other route choices compared to the objective of congestion-minimization. Traffic dispersion methods may prefer the fastest routes, which are usually road sections with higher speeds. For higher speeds the vehicle emissions per kilometer are higher. Therefore, it is interesting to explore methods that disperse traffic based on the local air quality.

This method stores the emission of planned routes in a common map. The users calculate the fitness of a certain road segment based on the travel time of a certain road segment, and the local emissions of routes that have already been planned to this (or a nearby) road segment. By means of this method, road segments that usually are very busy, become less attractive when other routes have already been planned on these segments.

Most scientists use traffic simulation applications which are able to calculate the average travel time and total emissions of a certain traffic demand. In these systems it is not possible to implement a collective routing method that takes the local air quality into account. However, improving the air quality locally by such a method is the main objective of this research. Therefore, in this research another approach is used for the construction and evaluation of the multi-objective routing method.

In this research a new simulator is constructed, which makes it possible to have full control on the selection of information that is important for this research. The simulator needs a specific area as input for which the simulation must be performed. Based on this area, a traffic demand is generated which represents the actual traffic demand at that area. The traffic demand consists of a list of driver profiles that have a start and end location, a start time and a vehicle type. Based on the trajectories of individual drivers an emission model is constructed. This model is used for the evaluation of different routing methods, but it is also used as a data source for the traffic dispersion method that is constructed in this research. That is the reason why a new simulator is constructed, such that the emission of other vehicles has influence on the route choices of successive drivers.

The simulator is route-based, which makes it not feasible to implement traffic congestion. The implementation of traffic dependency (and thus congestion) would require multiple iterations, which would cost much run-time. Considering the fact that avoiding traffic congestion is not an explicit objective of this research, traffic dependency is not implemented in the simulation setup.

The studies by Patil [11] and Aziz [13] did not evaluate their methods on real life networks. We consider it essential to evaluate the constructed methods on real life networks, because the effects of air pollutants are measured locally. Generalization of networks would probably remove local problems present in every city.

1.4. OUTLINE OF THE REPORT

In Chapter 2 the problem is formalized. Chapter 3 describes the emission model that is used for the comparison of the methods and algorithms. A description of the constructed methods and algorithms can be found in Chapter 4. A description of the experimental setup can be found in Chapter 5. Chapter 6 compares the outcome of the different methods with each other. Suggestions for other methods to solve this problem can be found in Chapter 7. A conclusion of the results and an answer to the research questions is presented in Chapter 8.

2

Problem Description

It is necessary to formalize the problem, such that the methods and algorithms can be evaluated in an unambiguous way. The main objective of this research is to improve the air quality in cities. Section 2.1 describes how a healthy air quality can be defined. Then, in Section 2.2 the problem is defined for finding routes for drivers while minimizing the local air pollution. Section 2.3 describes which assumptions are made in this research and why these are reasonable.

2.1. Representation of the Air Quality

Exposure to air pollutants has locally a bad influence on the health of citizens (Section 2.1.1), while globally high levels of greenhouse gasses lead to higher temperatures, which lead amongst other things to increased water levels (Section 2.1.2).

2.1.1. Local air quality

As already mentioned in Chapter 1, Brunekreef and Holgate indicated that exposure to Ozone, Particulate Matter (PM) and a mixture of Nitrogen Oxides (NOx) can lead to increases in mortality and hospital admissions due to respiratory and cardiovascular disease [4].

The concentrations of ozone and NOx are strongly correlated. This correlation stems from the fact that Nitrogen oxides react very easily with ozone, because NOx belongs to the group of radicals. Similarly, the World Health Organisation (WHO) observed a very strong relation between the concentrations of NOx and PM [18].

Because of those correlations between the air pollutants, and the fact that it would make the problem unnecessarily complicated if all three air pollutants would be considered, the decision has been made to use the emission of NOx as an indicator of the air quality of traffic distributions that are composed by the traffic routing methods. Based on the guidelines from the WHO and the European Union (EU), the hourly average of NOx may not exceed 200 $\mu g/m^3$ [4].

2.1.2. Global air quality

Besides the local effect of traffic emissions on the health of people that live nearby, commonly the emission of CO_2 is assumed to contribute to the global warming of the earth, by partly blocking the emission of heat from the earth to the atmosphere. Therefore, most countries strive to decrease the emissions of CO_2[1]. Decreasing the concentrations of NOx locally might lead to an increase of the total CO_2 emission. Therefore, the reduction of the total emission of CO_2 is also part of the problem.

2.2. Formal Problem

In Section 2.1 is described that for a healthy air quality, the concentrations of NOx may not exceed the threshold of 200 $\mu g/m^3$. However, drivers want to be able to reach their destinations. Therefore, in this research, drivers must be assigned routes such that the exceedance of the NOx threshold by vehicle emissions is minimized, while they will reach their destinations in time. The problem is defined as a minimization problem

[1] This is a consequence of the global climate agreement in Paris.

with constraints in Equation 2.1.

$$\begin{array}{lllll} \min\limits_{S} & h(A,S,\theta) & & & \\ \text{s.t.} & l(s) - l(s_{fast}) & \leq & \rho \times l(s_{fast}) & \forall s \in S \\ & \sum\limits_{s \in S} c(s) & \leq & \sum\limits_{s_{fast} \in S_{fast}} c(s_{fast}) & \end{array} \tag{2.1}$$

In this formulation, set S contains the routes s of a traffic demand on an area A. s_{fast} denotes the fastest route of a driver, while S_{fast} denotes the set of fastest routes for all vehicles in area A. $c(s)$ denotes the CO_2 emission of a route s. ρ represents the ratio between a route length and the length of the fastest route.

The set of routes must be calculated in such a way that the exceedance of the NOx threshold θ (Equation 2.2) is minimized. The set S of routes is used to construct a time-dependent emission map M for area A. This is described further in Chapter 3. The emission map is partitioned into a grid with a granularity of 350 by 350 m^2. The NOx concentration at grid point g at time t is denoted by $NOx(g,t)$. For every timestep and for every gridpoint g of the emission map M a summation is taken over the concentrations that exceed the threshold θ (200 $\mu g/m^3$).

$$\text{h(A,S,}\theta\text{)} = \sum_{t} \sum_{g:M} \begin{cases} 0 & \text{if } NOx(t,g) \leq \theta \\ NOx(t,g) - \theta & \text{otherwise} \end{cases} \tag{2.2}$$

To encourage drivers to follow prescribed routes, it is essential to limit the length of potential detours. Therefore, for every driver the additional travel time of $l(s)$ may only be as large as ρ times the travel time of its fastest possible route $l(s_{fast})$.

According to Section 2.1.2 the total emission of CO_2 must be reduced. In this research is assumed that if no routes are prescribed by a system, each driver will take its fastest possible route s_{fast}. The calculation of the set of routes S is therefore constrained by the requirement that the total CO_2 emission of the routes of set S must be less than the total CO_2 emission of the set S_{fast} (denoted by $c(S_{fast})$), which contains only the routes with the minimum individual travel time.

2.3. ASSUMPTIONS

This research is intended to prove the benefit of the concept of collective routing. Therefore, some significant assumptions have been made, mainly for the simulation model. These assumptions are reasonable, because the assumptions are also applied to the original traffic distributions, before the comparison between the several methods will be performed.

KNOWN ROUTE REQUESTS OF ALL DRIVERS IN THE AREA A

To be able to calculate a set of routes for area A, it is assumed that for every driver in area A the source, the destination, the time of departure and the vehicle type are known. This set of route requests could for instance be retrieved from in-vehicle navigation devices. In the experiment that is set up for this research, a set of route requests is generated using statistical data.

MANAGEABILITY OF DRIVERS

For the methods in this research the assumption is made that drivers will follow the proposed route of the navigation system strictly. This assumption is reasonable if we consider that inhabitants are willingly to participate if they know what the risks are of long-time exposure to air pollutants. Drivers will follow the advice of the routeplanner as long as the detour does not become too long, and as long as other drivers are also participating in the project. This is similar to participating in waste separation programs. Waste processing or recycling can be performed more effectively if it is separated in appropriate groups. Trust in the separation system, the number of years that citizens are living in the community and the reciprocity in the model are the three important incentives for citizens to participate in waste separation programs [19].

From this we may conclude that citizens would be willingly to participate in a project that improves the environment clearly, as long as the additional effort is marginal and others in their surroundings do also participate. The participation rate can be increased by introducing rewards for participants, or fines for citizens that do not participate (reciprocity). As a consequence of traffic dispersion through collective routing, the number of traffic jams can also be reduced, which makes it advantageous to participate in such a project. Finally, it is important that the system is clearly explained and that the advantages of the system are put in the spotlight.

The assumption that is made in this research becomes even more feasible if it is considered that in the future, vehicles will drive increasingly in an automated way. Route choices will be taken no longer by drivers on run-time, but routes will be prescribed by the controlling software of the vehicle. As a result of this, the choice to refuse to participate in the traffic routing program has to be made deliberately, and the routing preferences must be configured in the controlling software of the car.

FIXED SPEED OF DRIVERS

It is also assumed that all drivers will keep at every road segment the corresponding maximum speed. With this fixed speed the time that it would take to drive through edges can be approximated, such that the emission concentrations can be retrieved at a fixed timepoint of a route. This assumption is reasonable because only small maps are taken into consideration. As a result of that, the travel time is maximally 1 hour – measured from one end of the map to the other end of the map. The emissions are stored in timeframes of minutes, so the impact of divergent speeds to the maximum speed is marginal.

CONGESTION-FREE NETWORK

Traffic is not moving independent from each other, because the addition of an extra vehicle to an edge that is already heavily occupied will limit the speed of all vehicles at that edge. After planning a route for one driver, the routes of drivers that have already been planned must be re-evaluated. This has a large impact on the run-time of the simulation.

Therefore, the precision of the simulation model is further reduced by assuming also a congestion free network. That means that no matter how many vehicles are planned on the same edge at the same time, the speed of vehicles will stay the same. The main objective of this research is not to reduce the number of traffic jams, but to improve the local air quality by means of traffic routing. Although traffic jams can have an adverse effect on the travel times and vehicle emissions, it is assumed that traffic jams will occur more often in the base case where every driver optimizes its own travel time, than in the case where traffic is routed by means of a method that uses a traffic dispersion algorithm. The actual travel times will therefore be larger than the travel times that are presented in the simulation in this research.

3

EMISSION MODEL

This section describes the construction of an emission model that is used for the evaluation of models and partially for the methods. First a description is given of the model that is used for the approximation of vehicle emissions in Section 3.1. From this model the emission per road segment (edge) can be calculated (Section 3.2). Furthermore, the construction of an accumulative emission map is described in Section 3.3.

3.1. EMISSION CHARACTERISTICS PER VEHICLE

The MEET project [20] formulated speed-dependent functions for the emissions per vehicle type, based on measurement of the emissions on predetermined test cycles. Although more exact emission models are available, the fact that the emissions are expressed as speed-dependent functions suits the application in this simulation model very well under the adjusted assumptions. Considering that the objective of this research is not to construct extremely realistic predictions of the air quality, it is reasonable to use a model that makes it possible to construct an accumulated emission map from a set of routes.

As described in Section 5.2 the traffic has been divided into four vehicle classes. For Petrol and Diesel passenger cars the speed dependent model of the EURO I classes is used[1]. For the truck emissions a heavy truck with 7.5 to 16 tonnes of gross is assumed. The assumption is made that electric cars have zero emission, although it could be the case that in the whole energy chain pollutants are exhausted during the energy conversion to electricity. It is not necessarily needed to use fossil fuels as original energy sources, because wind and sun power can also be used to generate electricity. Furthermore, in this research not the complete energy chain has been considered for the other vehicle classes. The emissions are modelled as expressions that express the concentration of the individual pollutants in g/km, dependent on the speed of the vehicle in km/h (Table 3.1 and Figure 3.1).

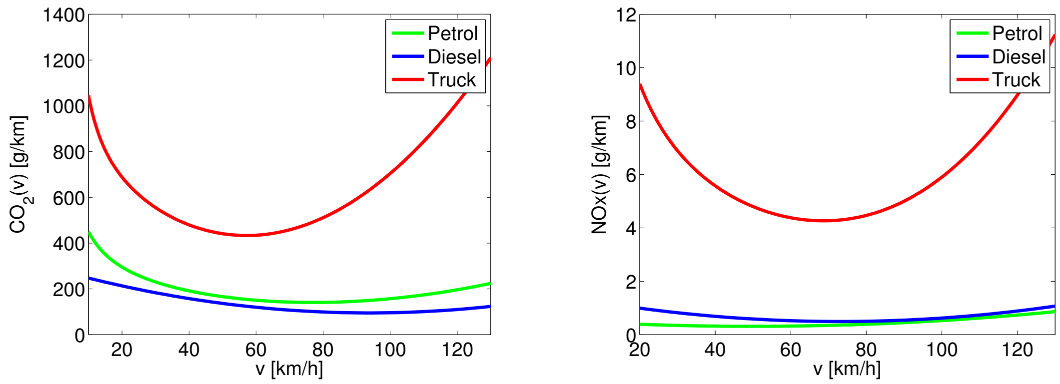
3.2. EMISSION PER EDGE

From the estimation of the emission of a single vehicle the total emission of a vehicle on a certain edge on a graph can be computed, by assuming an average speed on that edge. The speed- (v) and vehicle type (c)

[1] Recently, several scandals about the emission rates were in the news. In order to use more realistic emission rates, the old EURO I class is used, although newer cars are complying to new standards.

Table 3.1: The speed-dependent emission model per vehicle type. Because the main objective of this research is to improve the air quality locally, these equations do only contain the actual emissions of the vehicle. If the complete energy chain would be taken into account, the emission characteristics would be higher, especially for electric vehicles that are charged by electricity that is generated using fossil fuels.

Vehicle type	**CO_2(v) [g/km]**	**NOx(v) [g/km]**
Petrol	$231 - 3.62v + 0.0263v^2 + \frac{2526}{/}v$	$0.526 - 0.0085v + 8.54 * 10^{-5}v^2$
Diesel Passenger Car	$286 - 4.07v + 0.0217v^2$	$1.4435 - 0.026v + 1.78 * 10^-4v^2$
Diesel Truck	$871 - 16 * v + 0.143v^2 + \frac{32031}{v^2}$	$2.59 - 6.65 * 10^4 v^2 + 8.56 * 10^{-6}v^3 + \frac{140}{v}$
Electric	0	0



(a) The speed dependent emission of CO_2.

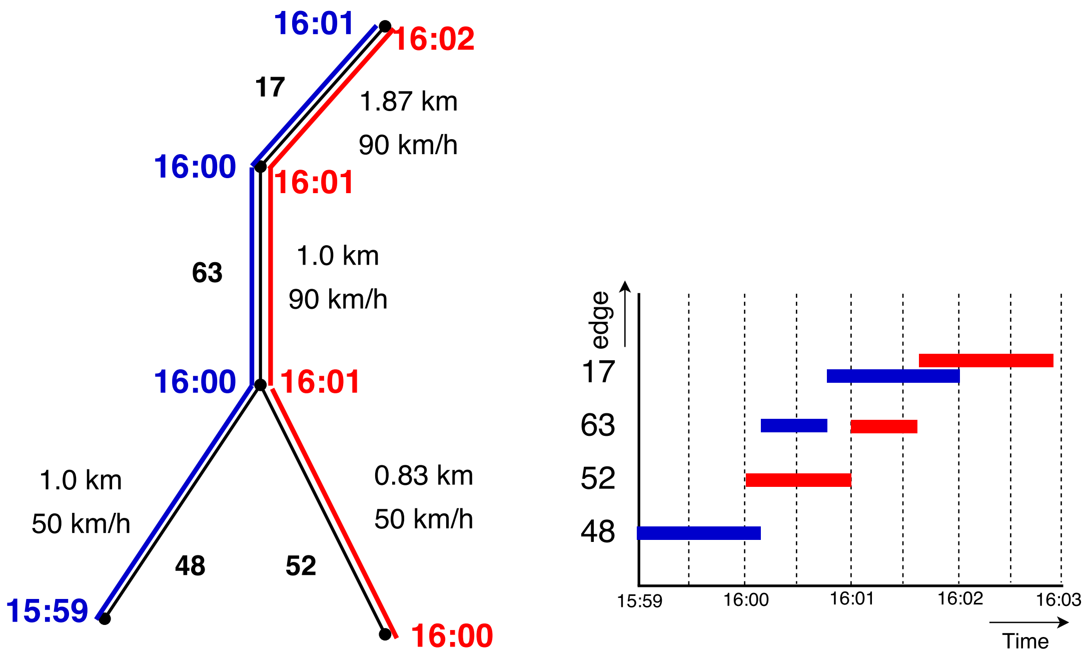
(b) The speed dependent emission of NOx.

Figure 3.1: The estimation of emissions per vehicle type based on their speed in [g/km].

dependent emission characteristic ($R(v,c)$) is multiplied by the speed of the driver (v) [km/h] and the duration (d) [h] that the vehicle spends at that edge, which results in a NOx and CO_2 emission per edge (Equation 3.1).

$$\text{Emission per edge } = R(v,c) \times v \times d \tag{3.1}$$

Figure 3.2 shows the conversion from two routes to the occupation per edge. Figure 3.2a shows an elementary graph consisting of 5 nodes and 4 edges, which are labelled with bold numbers. In this graph, two routes are displayed. The blue route is taken by a driver of a Petrol car, who enters edge 48 at 15:59. The driver with the red route starts from edge 52 at 16:00, with a Diesel passenger car. The trajectories of both drivers are abstracted per edge, which leads to Figure 3.2b.

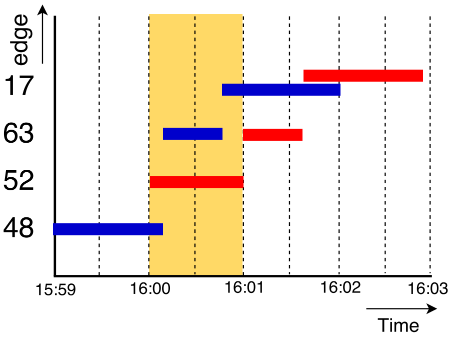


(a) Graph with routes and arrival times.

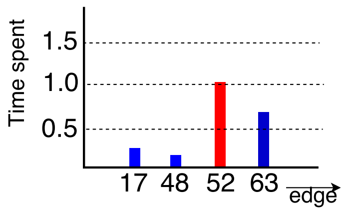
(b) Time-dependent edge occupation.

Figure 3.2: Conversion of routes to the time-dependent edge occupation rates. The blue route is driven by a petrol car, while the red route is driven by a diesel passenger car.

From the occupation per edge (Figure 3.3a), the cumulative time that is spent per edge (Figure 3.3b) can be calculated. Combining the vehicle emission rates, the maximum allowed speed per edge and the time spent by each car the cumulative emission per edge can be calculated, using Equation 3.1. The results of these



(a) Edge occupation at 16:00.

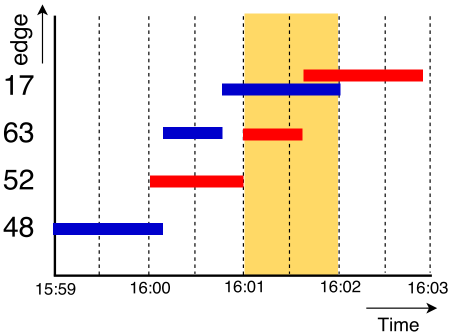


(b) Duration per edge at 16:00.

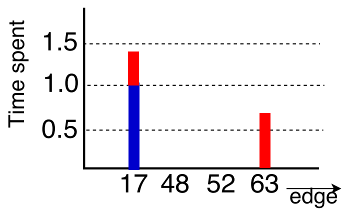
Edge	NOx [g]	CO_2 [g]
17	10.17	3294
48	1.55	830
52	29.5	6840
63	30.42	9873

(c) Emission per edge at 16:00

Figure 3.3: The conversion from edge occupations to the total emission of NOx and CO_2 at 16:00.



(a) Edge occupation at 16:01.



(b) Duration per edge at 16:01.

Edge	NOx [g]	CO_2 [g]
17	60.12	16605
48	0	0
52	0	0
63	34.4	6021

(c) Emission per edge at 16:01

Figure 3.4: The conversion from edge occupations to the total emission of NOx and CO_2 at 16:01.

calculations is shown in Figure 3.3c. The concentrations are calculated and stored per timestep. Figure 3.4 shows therefore the same procedure for calculating the emissions per edge for the timeframe from 16:01 to 16:02.

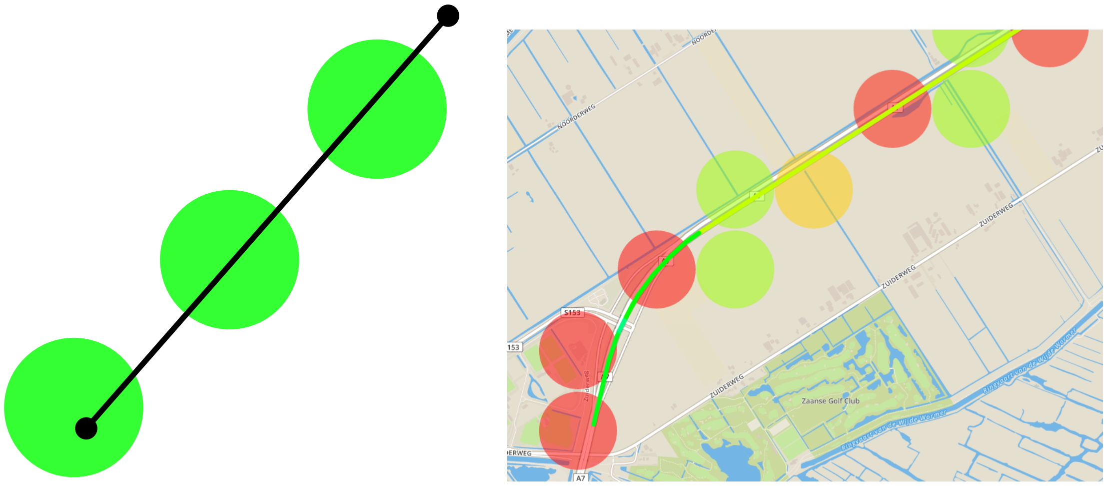
3.3. CONSTRUCTION OF AN ACCUMULATIVE EMISSION MAP

The formulation of the emission per edge per timeframe (Equation 3.1) makes it possible to calculate the contribution of a route to the NOx concentrations in the simulation area. A route is the concatenation of edges with an arrival time per edge. The total NOx concentrations could be stored per edge at the right timeframe. However, if two edges are close together, the NOx concentrations at these road segments are not independent anymore. Therefore, it is needed to translate the emission characteristics of a set of routes to an emission map with NOx concentrations. The area that is used in the simulation is divided into a grid with a granularity of 350 by 350 m^2, as required in Chapter 2.

For every edge the closest grid points are located. Figure 3.5 shows the mapping of edges to grid points. The emission per edge is distributed over each surrounding grid point of the edge, and divided by the surface area of each grid point, which results in an emission contribution $[\mu g/m^2]$. It is assumed that the emission of NOx stays within a height range of 1 meter during one timeframe. For short timeframes this assumption is reasonable, if we also consider that the tailpipes of vehicles are located near the ground. This induces that the contribution of a route to the NOx concentrations on every grid point that is near the route is stored as $[\mu g/m^3]$.

Furthermore, it is assumed that the NOx concentration does not vaporize within one timeframe. As a consequence of this assumption, the emission of multiple vehicles on the same grid point at the same timeframe can be added independently. Moreover, the assumption is made that the cumulative NOx emission at a grid point vaporizes completely at the end of a timeframe. As a consequence, the NOx emissions at one timeframe do not have influence on the NOx concentrations of other timeframes.

In Figure 3.5a the closest grid points are located for edge 17 from the example of Figure 3.2. In this case,



(a) Conversion from edge to grid points.

(b) Conversion from edge to grid points at a part of an OSM.

Figure 3.5: The spatial conversion from emission on an edge to concentrations in grid points. The trajectory of the edge is converted to a set of grid points (visualized as filled circles). The differences in colour display the difference of the NOx concentrations, where red points represent higher concentrations than green points. When a grid point is located nearby multiple edges, the concentration at that grid point is increased by multiple edges. That results in higher NOx concentrations on grid points that are near intersections.

the emission per edge is distributed over 3 grid points. Therefore, the contribution of edge 17 to the concentrations of the nearby grid points is equal to 163.6 $\mu g/m^3$ at the timeframe from 16:01 to 16:02. This concentration does only depend on the emission of the vehicles from 16:01 to 16:02. The emissions during the timeframe 16:00 to 16:01 do not have influence on the concentration from 16:01 to 16:02.

The concentration per grid point might be higher than it is calculated in the example of Figure 3.5a, because some of the grid points may be located near other edges. The contributions to the NOx concentrations of all routes are accumulated per grid point and per timeframe which results in an accumulated emission model that is used for the evaluation of methods. This model is also used for the determination of optimal routes.

4

Methods and algorithms

In this chapter is described how a set of routes S is calculated, such that the exceedance of the NOx threshold θ is minimized, while the individual travel times of drivers are kept short. The total emission of CO_2 of all routes from set S together is also reduced. This is in less formal language the problem definition as described in Chapter 2.

A route can be seen as the concatenation of roads and intersections on area A. These roads and intersections can be transformed to edges and nodes on a graph G. The construction of the graph G for the area A is described in Section 4.1. After that, in Section 4.2 an algorithm is constructed for calculating the lowest-cost paths from an arbitrary source to an arbitrary destination on graph G. In Section 4.3 is described how the constructed algorithm for calculating one path can be applied to calculate the routes of a whole set of drivers, which results in a set of routes S.

4.1. Construction of time-dependent graph G

The objective of this research is to calculate a set of routes S on an area A for a set of route requests. A route request consists of a source, a destination, a start time and a vehicle type, like it is described in Chapter 2.

To calculate a route from a source to a destination, it is first necessary to transform a real-world area A (Figure 4.1a) with roads and intersections to a graph G of edges and nodes (Figure 4.1b). In graphs, weights are assigned to edges, such that the weight expresses the cost for traveling across that edge. The optimal path from a source to a destination is the combination of edges in a graph which have together the lowest sum of the weights. The feasibility (cost) of a path is determined by the costs of the individual edges that are part of that path.

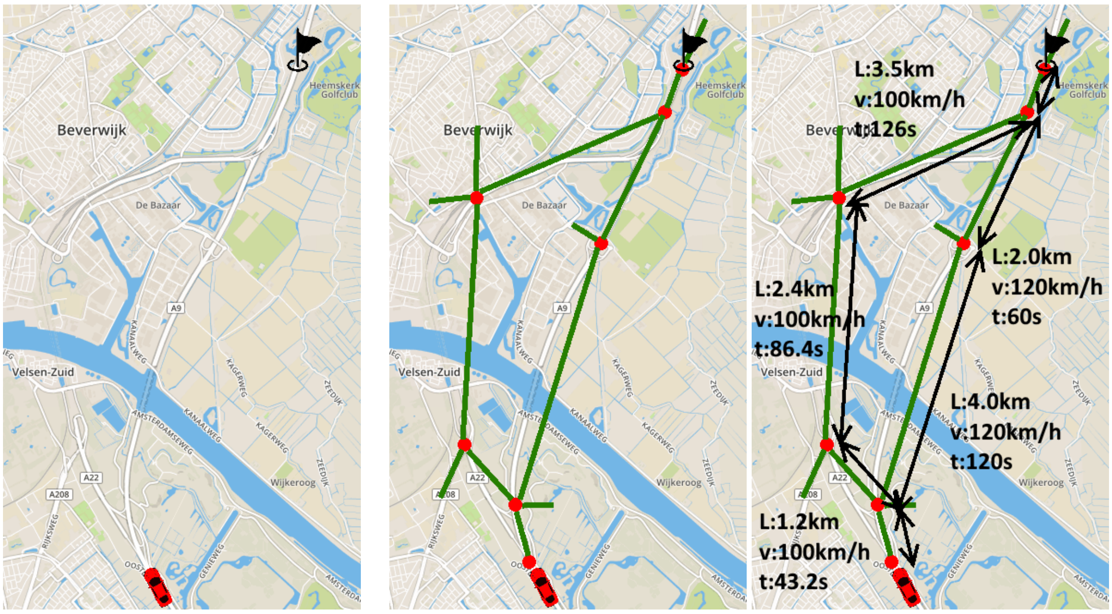
For routing problems which strive for the shortest routes of drivers, the weights of edges are defined as the distance between the nodes on the two ends of the edge. If the objective of a routing problem is to find the shortest travel time, the weight of an edge represents the distance of the edge, divided by the average speed that can be driven across that edge (Figure 4.1c).

The objective of this research is to find routes for drivers, such that the exceedance of the NOx threshold caused by the cumulative emission of vehicles is minimized. In addition to that, the travel time of routes is bounded above, as well as the CO_2 emission of all vehicles together. Therefore, in this research, the weight of edges is expressed as a weighted combination of the travel time and the environmental impact.

One part of the weight of an edge is the travel time. The travel time is approximated by dividing the length of the edge l [m] by the expected speed v [m/s] that a driver can drive across that edge (Equation 4.1 and Figure 4.1c).

$$\text{Travel time} = \frac{l}{v} \tag{4.1}$$

An important part of the weight is a factor that represents the local NOx concentrations. Low concentrations of NOx are permitted, so the weights of edges with low concentrations of NOx nearby should depend mostly on the other factors (travel time and CO_2 emission). Therefore, the share of NOx on the weight of an edge is expressed as the cumulative exceedance of threshold θ of every grid point nearby an edge. In Section 3.3 is described how it is determined which grid points are located nearby an edge. Grid points that have



(a) Original map with origin and destination. (b) Translate map to graph of nodes and edges. (c) Calculate distance, speed and travel time.

Figure 4.1: The conversion from an area A to a graph G, where the weights represent the travel time.

a NOx concentration lower than the threshold are not considered. If all the grid points around an edge have a NOx concentration below the threshold, the NOx part of the weight of that edge is equal to zero. The NOx concentrations fluctuate over time, like it is described in Section 3.3. The NOx concentrations around an edge ($NOx(t)$) are therefore requested for a certain time period. At the corresponding time frame, the exceedance of the threshold θ is calculated per grid point g of the set of points around edge E, and the summation over all points is stored as $\phi(t)$ (Equation 4.2). The cumulative exceedance is scaled by a factor C_{nox}, which leads to Equation 4.3, which is visualized in Figure 4.2. A visualisation of the values for $\phi(t)$ for different edges can be found in Figure 4.3a.

$$\phi(t) = \sum_{g \in E} \max(0, NOx(t) - \theta) \tag{4.2}$$

$$\text{NOx part } = \frac{\phi(t)}{C_{nox}} \tag{4.3}$$

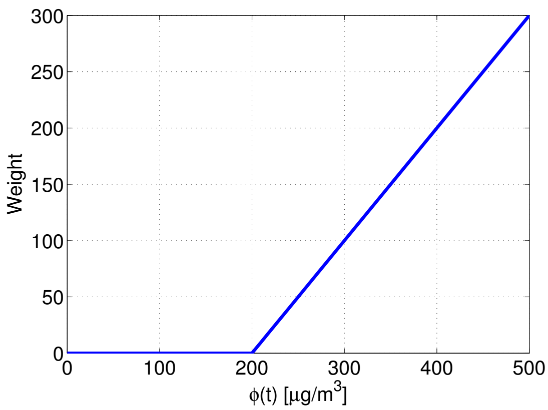
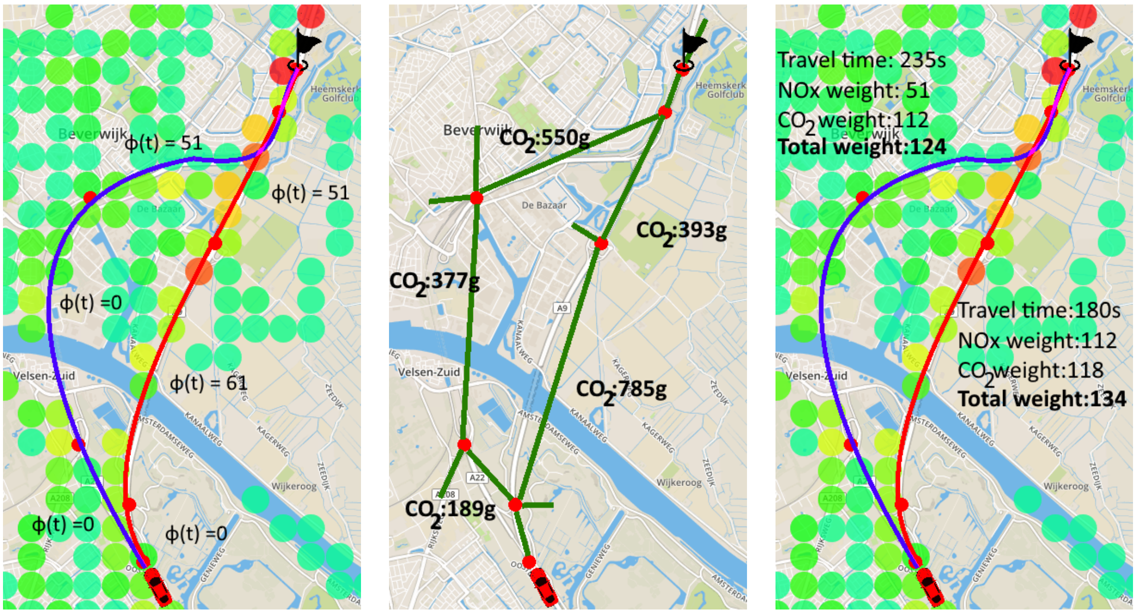


Figure 4.2: A plot of Equation 4.3, as a function of the cumulative concentrations $\phi(t)$ at that edge. In this case the weighting factor C_{nox} is set to 1 and the NOx threshold θ to 200 $\mu g/m^3$.



(a) Retrieve the exceedance $\phi(t)$ $[\mu g/m^3]$ around every edge.

(b) Calculate own expected CO_2 emission.

(c) Calculate the combined weight of the edges.

Figure 4.3: A visualisation of the routing procedure for one driver who navigates in an existing traffic distribution. The NOx concentrations are visualised as colored circles. Green colors represent low NOx concentrations, while orange colors represent high NOx concentrations. In Figure 4.3a $\phi(t)$ is greater than zero for edges nearby orange grid points. In Figure 4.3c the total weight of the blue and the red route is calculated, with α = 0.3, β =0.4 and γ=0.3. The red route is faster, but the blue route is preferred, to improve the air quality near to the red route.

Another objective is to limit the total emission of CO_2, so the expected CO_2 emission of a vehicle (R(CO_2) is another part of the weight of an edge (Equation 4.4 and Figure 4.3b). The expected CO_2 emission is calculated according to the emission model that is described in Section 3.2. The expected CO_2 emission is divided by a scaling factor (C_{eco}), such that this part does not dominate the weight of an edge.

$$\mathrm{CO_2\ part} = \frac{R(CO_2)}{C_{eco}} \tag{4.4}$$

As we stated earlier, the weight of an edge is expressed as a weighted combination of travel time, the nearby NOx concentrations and the expected emission of CO_2 of a vehicle for that edge. The complete expression of the edge weights is formulated by Equation 4.5. The three different parts are weighted by factors α, β and γ. Equation 4.6 states that the weighting factors of each part of the formula should be equal to one, while they individually should have values between zero and one.

For vehicles with zero emissions, like electric vehicles, α is set to 1, and β and γ to zero. In this way, vehicles which do not contribute to higher NOx concentrations or to a higher CO_2 emission, are assigned the fastest routes.

Figure 4.3 shows that this weighting procedure may lead to decide to take a small detour (the blue route) in order to reduce the peak NOx concentrations nearby the red route.

$$\mathrm{W(t)} = \alpha T + \beta \frac{\phi(t)}{C_{nox}} + \gamma \frac{R(CO_2)}{C_{eco}} \tag{4.5}$$

$$\alpha + \beta + \gamma = 1 \tag{4.6}$$

4.2. CONSTRUCTION OF AN ALGORITHM FOR FINDING THE SHORTEST PATH IN A TIME-DEPENDENT GRAPH

In Section 4.1 a time-dependent graph G = (V,E) is defined, where the weights of edges express the requirements on routes: minimal exceedance of the NOx threshold, with low travel times and reduced emissions.

The weights of edges are time-dependent in this graph, so in this section a time-dependent algorithm is constructed that calculates a path from a source to a destination, with a certain time of departure. For the construction of this algorithm, first some of the best-known algorithms for finding shortest paths in static graphs are described in Section 4.2.1. We examine which algorithms can be transformed to time-dependent versions for the constructed graph G of Section 4.1. After that, the time-dependent algorithm that is constructed is described in Section 4.2.2.

4.2.1. ALGORITHMS FOR FINDING THE SHORTEST PATH IN A STATIC GRAPH

Several algorithms have been developed that find a solution to the static point-to-point version of the shortest path problem, of which one will be used for the construction of a time-dependent shortest path algorithm.

DIJKSTRA'S ALGORITHM

In 1959 Edsger Dijkstra [21] proposed a greedy algorithm to solve the single source shortest-path problem (Algorithm 1). For a graph G=(V,E) the shortest route between start node *source* to destination node *target* is calculated by the construction of a solution tree from the source node to all other nodes in the graph G, until a path to the destination node is found. The path from the source node to the destination node can be retrieved by traversing the solution tree from the destination node back to the source node.

The algorithm can also easily be used to calculate the distance of one node to all the other nodes in the graph. In that case the algorithm should stop only if the set of unvisited nodes Q is empty. Without further modifications the worst-case runtime for connected graphs is $O(|E|\log(|V|))$.

Because Dijkstra constructs the shortest path by computing a tree until the destination node is found, it is relatively easy to obtain the edge weights at the correct expected time.

A* SEARCH ALGORITHM

An extension of Dijkstra's algorithm is proposed by Peter Hart, Nils Nilsson and Bertram Raphael in 1968 [23] (Algorithm 2). They have added a heuristic to Dijkstra's algorithm. Dijkstra's algorithm traverses the graph in all directions, considering also nodes that will lead the traveller further away from his destination. Off course, in advance it is not possible to know exactly which nodes should be considered. However, a estimations can be made on the minimum distance between each node and the destination node ($\hat{h}(n)$). This estimation can be based on the geographical distance between the nodes. Nodes that have the smallest minimal distance to the destination will be evaluated first. If the function that estimates the minimal distance is admissible, which means that it never overestimates the minimal distance, then the optimal path will be found by this algorithm.

The time complexity of the algorithm depends on the chosen heuristic for the estimation of the distance to the endpoint. If the search space is unbounded, the complexity is exponential in the depth d of the path from node u to node v, equal to $O(b^d)$, where b is the average number of succeeding nodes for each node [24]. Although the worst-case runtime of the A* search algorithm is worse than the complexity of the original Dijkstra algorithm, in most cases the A* search algorithm will outperform the algorithm of Dijkstra if a good heuristic is chosen for the estimation of the minimal distance to the endpoint.

If the heuristic is consistent (also called monotone), each node will be evaluated at most once by the A* algorithm. A consistent heuristic estimates the minimal distance from a node s to the destination always less than or equal to the estimated distance from any neighboring vertex v to the destination, plus the distance $d(s,v)$ of reaching that neighbor (Equation 4.7). Then the complexity becomes linear in the number of vertices (O(V)) [22]. Usually the solution trees of the A* algorithm are cut off earlier than the solution trees of Dijkstra's algorithm, such that the shortest route is found faster.

$$\hat{h}(s) \leq \hat{h}(v) + d(s,v) \tag{4.7}$$

Like Dijkstra's algorithm, the A* algorithm constructs the shortest path as a tree with the source as the root node. Therefore, the time at which an edge is traveled can be calculated using the distance from the source node to the node that is connected to that edge. The strength of the A* algorithm is determined by the strength of the heuristic. However, for the constructed graph G it is not worthwhile constructing a heuristic that estimates the minimal cost to a destination, because the weights are dominated by the NOx concentrations, which are not as predictable as geographical distances.

Algorithm 1 Dijkstra's algorithm for finding the shortest path between *source* and *target* in a graph G=(V,E).

```
Q = ∅
for all v ∈ V do
    d[v] ← ∞
    prev[v] ← null
    add v to Q
end for

d[source] ← 0
while Q ≠ ∅ and u ≠ target do
    u ← vertex in Q with min d[u]
    remove u from Q
    for all neighbours v of u do
        d_estimated ← d[u] + c(u, v)
        if d_estimated < d[v] then
            d[v] ← d_estimated
            prev[v] ← u
        end if
    end for
end while
return d[], prev[]
```

Algorithm 2 A* search algorithm for finding the shortest path between *source* and *target* in a graph G=(V,E) according to the formulation of Martelli [22].

```
CLOSED = ∅
OPEN = {source}
g(s) ← 0
f̂(s) ← ĥ(s)

while OPEN ≠ ∅ and current ≠ target do
    current = min_{n∈OPEN}(f̂(n))
    remove current from OPEN
    add current to CLOSED

    for all neighbours v of current do
        g(v) ← g(current) + c(current, v)
        if v ∉ OPEN and v ∉ CLOSED then
            ĝ(v) ← g(v)
            f̂(v) ← g(v) + ĥ(v)
        else
            if ĝ(v) > g(v) then
                ĝ(v) ← g(v)
                f̂(v) ← g(v) + ĥ(v)
                if v ∈ CLOSED then
                    remove v from CLOSED
                    add v to OPEN
                end if
            end if
        end if
    end for
end while
```

Contraction Hierarchy

For large graphs the Dijkstra and A* algorithms might be too slow to be used in practice. To decrease the number of nodes that must be considered, a contraction hierarchy can be used. This technique as described by Sanders, Schultes and Delling [25] will pre-process the graph, such that future requests of paths can be treated faster. During this pre-processing stage, a hierarchy is constructed in the graph. Multiple edges will be combined into one imaginary edge with a weight that is the sum of the contracted edges. The selection of the edges is based on some predetermined nodes that will be considered as most important. After the pre-processing stage the routing will be performed between the depart node to the destination node by searching through the hierarchy. In this way route requests can be handled much faster.

For time-dependent graphs the implementation of this procedure is much more complicated. The pre-processing of the graph must be performed for multiple time-frames. If after the pre-processing stage one of the NOx concentrations would change, the whole pre-processing stage must be performed again. Transforming this algorithm to a time-dependent version is therefore regarded as infeasible for this problem.

4.2.2. Algorithm for finding the shortest path in a time-dependent graph

In Section 4.2.1 is described that transforming the Contraction Hierarchy to a time-dependent algorithm is regarded as infeasible. Furthermore, the run-time of the shortest-path algorithm can be reduced by using the A* algorithm with a heuristic for the estimation of the minimal distance from a node to the destination node. However, the construction of such a heuristic is considered meaningless for graph G, due to the strong dependency of the weights of edges with the local NOx concentrations which are hard to predict. Therefore, we choose to transform Dijkstra's algorithm to a time-dependent version.

Time-dependent version of Dijkstra's Algorithm

The time-dependent version of Dijkstra's algorithm is formulated in Algorithm 3. Off course, this formulation is broadly the same as the original Dijkstra's algorithm. The difference is that for each edge the weight is requested at a specific time (described at Section 4.1). To be able to request weights at the correct time ($c(u,v;t[u])$), the travel time is stored separately as $t[u]$. The weight of an edge $w[u]$ is used to calculate the optimal route, in the same way as it is computed for static graphs. The travel time of an edge $t[u]$ is used to estimate the arrival time at the next edge, such that the weight of the next edge can be requested at the right time frame.

Evaluation of Constraints

Before calculating a route with the time-dependent Dijkstra's algorithm, certain values are chosen for α, β and γ, such that they sum up to 1, while they have individually a value between 0 and 1 (Equation 4.6). These weighting factors determine the influence of the travel time, the local NOx concentrations and the expected CO_2 emission of a vehicle on the calculation of the path.

In Chapter 2 the problem of this research is formulated as a minimization problem with constraints. One of the constraints is that the length of a route between a certain source and a destination must not be longer than ρ% of the fastest possible route between that source and that destination. To satisfy this constraint, the determination of a route is performed by Algorithm 4. As long as the constraint is not satisfied, the weighting parameters are adjusted, until a short enough path is found. In the worst case, Algorithm 3 would be executed 10 times, because it would take 10 iterations to reduce β or γ from 1 to 0.

4.3. Methods

Algorithm 4, which is constructed in Section 4.2, calculates an optimal route for one driver on a graph G that is constructed in Section 4.1. The objective of this research is to calculate a set of routes, such that the exceedance of the NOx threshold is minimized locally. This can only be achieved with Algorithm 4 if the graph G is an accurate reflection of the real distances and traffic densities. In order to achieve such accurate reflections of the actual NOx concentrations, two methods are constructed, which are described in Sections 4.3.1 and 4.3.2.

4.3.1. Method 1: First come, first served

One approach is to let the drivers calculate their optimal path in a 'First come - first serve' manner (Algorithm 5). The set of drivers is stored as a list with random order. Each driver in this list has a certain source, a destination, a time of departure and a vehicle type (Section 2.3). Then, after the list of drivers is retrieved,

Algorithm 3 The time-dependent version of Dijkstra's algorithm for finding the shortest path between *source* and *target* in a graph G=(V,E).

```
Q = ∅
for all v ∈ V do
    w[v] ← ∞
    t[v] ← ∞
    prev[v] ← null
    add v to Q
end for
w[source] ← 0
t[source] ← start time
while Q ≠ ∅ and u ≠ target do
    u ← vertex in Q with min w[u]
    remove u from Q
    for all neighbours v of u do
        w_estimated ← w[u] + c(u, v; t[u])
        if w_estimated < w[v] then
            w[v] ← w_estimated
            t[v] ← t[u] + t(u, v)
            prev[v] ← u
        end if
    end for
end while
return w[], t[], prev[]
```

Algorithm 4 The determination of a route between *source* and *target* in a graph G=(V,E), satisfying the maximum route length constraint.

```
α ← 1
β ← 0
γ ← 0
t[source, target]_base ← Algorithm 3 with α, β and γ.
t[source, target] ← ∞
counter ← 0
0 ≤ α_orig ≤ 1
0 ≤ β_orig ≤ 1
0 ≤ γ_orig ≤ 1
while t[source, target] − t[source, target]_base > ρ × t[source, target]_base do
    α ← α_orig + counter × 0.1
    if α > 1 then
        α ← 0
    end if
    β ← β_orig - counter × 0.1
    if β<0 then
        β ← 0
    end if
    γ ← γ_orig - counter × 0.1
    if γ<0 then
        γ ← 0
    end if

    w[], t[source, target], prev[] ← Algorithm 3 with the adjusted α, β and γ
    counter ← counter + 1
end while
return w[], t[], prev[]
```

the routes for drivers are computed sequentially. After the computation of one route, the emission map is updated according to the model that is described in Section 3.3, using Algorithm 6. The addition per grid point is calculated by dividing the expected emission at that edge (NOx(t)) by the number of grid points (|P|) times the surface area of a grid point (x^2), where x denotes the granularity of the grid.

Algorithm 5 Approach where routes are calculated one after another

```
for all Drivers do
    Compute route by Algorithm 4
    Store emission trace of computed route at collective map (Algorithm 6)
end for
```

Algorithm 6 Procedure for increasing NOx concentrations around the computed route

```
prev[], t[] from Algorithm 4
u ← destination
while prev[u] ≠ ∅ do
    v ← prev[u]
    for t:=t[v] to t[u] step 1 do
        P ← grid points around edge (v,u)
        for all p ∈ P do
            p(NOx(t)) ← p(NOx(t)) + R(NOx)/(|P|x^2)
        end for
    end for
    u ← v
end while
```

Algorithm 4 returns a set of connected nodes from the destination to the source (prev[]). This set is the optimal path between the source and the destination. In addition to that, Algorithm 4 does also return a set of arrival times for every node, relative to the start time at the source. These two sets are used to store an estimation of the emission trace of a vehicle in the graph G. For every edge between any nodes between the source and the destination, the NOx concentrations are updated for every grid point nearby that edge (as described in Section 3.3). The next driver in the list has therefore a slightly different graph compared to the previous drivers.

The emission map that is updated by the last driver of the list of drivers is an estimation of the actual emission map that is present due to the constructed set of routes of all drivers in the list.

The advantage of this method is that a set of routes can be constructed by calculating for every driver one route sequentially. In this way, the peak NOx concentrations and the total CO_2 emission will be reduced, while the routes of individual drivers are bounded above.

Another advantage is that this method would accept additions to the list of drivers during run-time, because the routes are computed sequentially, with graph updates in between.

A drawback of this approach is that a driver that computes his route can not take into account the emission traces of successive drivers. As a result of that, his route may lead through areas that have already large exceedances of the NOx threshold θ, while he possibly has an alternative route at his disposal through an area with low NOx concentrations which would take only slightly longer travel time.

4.3.2. METHOD 2: ROUTING WITH RESCHEDULING

Another approach is to allow drivers to recompute their route, after all other drivers have stored their routes (Algorithm 8). This procedure is roughly the same as for method 1 (Section 4.3.1), but the difference is that the calculation of the set of routes S is performed in iterations. In the first method, the first drivers of the list have computed their routes on a graph that did not contain the estimated emissions of the routes of the successive drivers of the list. Potentially, the routes of the first drivers of the list are no longer optimal. Therefore, in this method the set of routes is constructed using multiple iterations. Before a driver computes its route, he removes his former emission trace from the graph, according to Algorithm 7. Then, after the determination of his optimal route by Algorithm 4, he stores his new emission trace according to Algorithm 6.

After several iterations this approach would lead to a user equilibrium, where no driver has the advantage to change its route. This is equivalent to Wardrop's principle [12], that if each driver in a traffic network seeks

Algorithm 7 Procedure for reducing NOx concentrations around the former route

```
prev[], t[] from Algorithm 4
u ← destination
while prev[u] ≠ ∅ do
    v ← prev[u]
    for t:=t[v] to t[u] step 1 do
        P ← grid points around edge (v,u)
        for all p ∈ P do
            p(NOx(t)) ← p(NOx(t)) - R(NOx)/(|P|x^2)
        end for
    end for
    u ← v
end while
```

Algorithm 8 Approach where recomputing routes is permitted

```
for all Iterations do
    for all Drivers do
        Remove (if present) old emission trace from collective map (Algorithm 7)
        Compute route by Algorithm 4
        Store emission trace of computed route at collective map (Algorithm 6)
    end for
end for
```

routes to benefit himself most, the User Equilibrium (UE) state is reached when no driver has the incentive to change his current route. In Wardrop's UE every driver optimized his route on the travel time only, while in this method the local NOx concentrations are incorporated in the edge cost function of the graph.

One of the drawbacks of this method is that the run-time can be too large for large areas. Due to the addition of iterations, the run-time is scaled up linearly. Although this would normally not be considered as a problem, in this case the linear growth of the run-time would make the algorithm infeasible for the application in metropolitan areas. The number of nodes, edges and vehicles is large in metropolitan areas. Due to the sequential computation of a set of routes, the run-time of one iteration would already take quite some run-time.

5

EXPERIMENTAL SETUP

In this chapter the setup of the experiment is described. Figure 5.1 shows the components of the system, and the relation between them. According to Chapter 2 the set of routes must be calculated for a certain area. For a certain area A both a map and a corresponding set of route requests is retrieved. For this traffic demand a set of routes must be constructed, which lead to certain emissions. In this research, the emissions are modeled by the emission model that is described in Chapter 3. After the set of routes is calculated, the total travel time, the total CO_2 emission and the time-dependent NOx concentrations can be retrieved, which are used to compare different methods and configurations with each other.

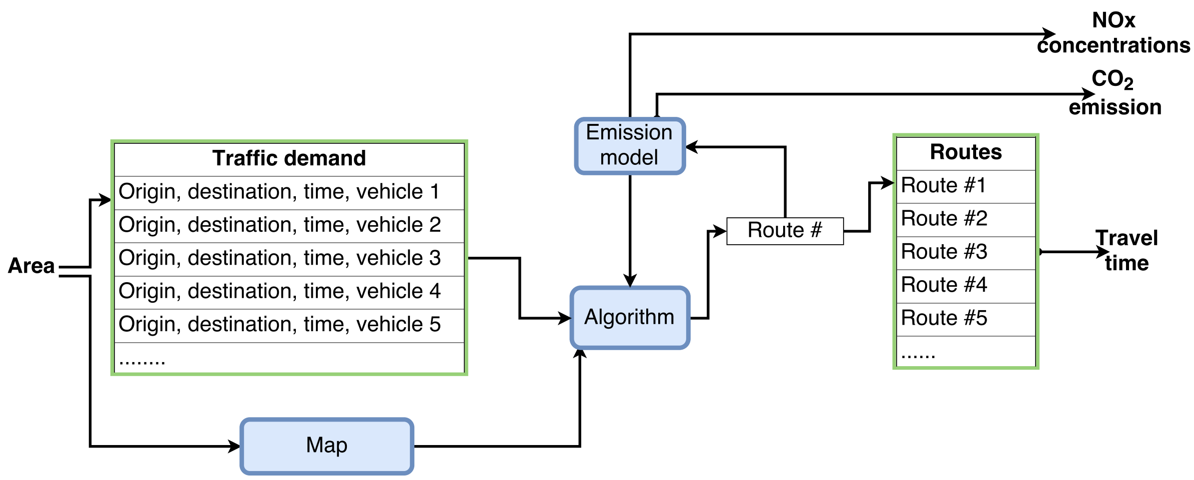


Figure 5.1: The relation between the input and output of the problem.

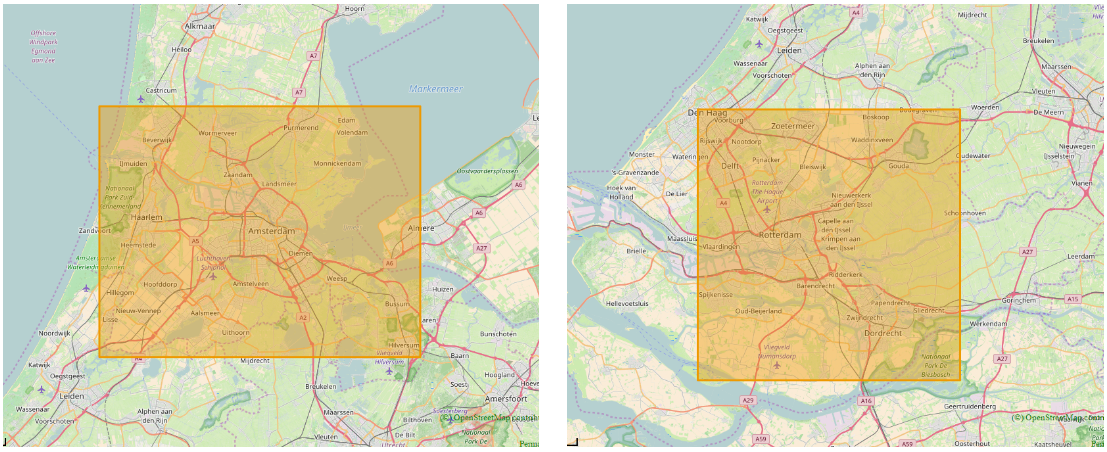
The selection of areas and the acquisition of maps is described in Section 5.1. Based on the selection of the area, a traffic demand is generated, which is described further in Section 5.2. For the constructed traffic demand on an area, a set of routes is sequentially calculated, according to the methods that are described in Section 4.3, using a time-dependent algorithm. The software implementation of these methods is described in Section 5.3. Section 5.4 describes in which way the methods will be evaluated.

5.1. AREA SELECTION

The methods are evaluated in the two largest cities in the Netherlands, which are Amsterdam and Rotterdam. Amsterdam and Rotterdam are situated in the western part of the Netherlands, where the traffic densities are high. Due to the high traffic densities, the air quality is poor in these cities. This makes it an interesting area to explore the effect of traffic management on the air quality. Figure 5.2 shows the areas that are selected for this experiment.

The maps which are used for both cities are generated by the OpenStreetMap (OSM) project[1]. The maps

[1] OSM files can be retrieved from `http://extract.bbbike.org/` by specifying the geographical corners of the area.



(a) Amsterdam, lower left corner: (4.54,52.21), upper right corner: (5.21,52.53)

(b) Rotterdam, lower left corner: (4.29,51.74), upper right corner: (4.84,52.08)

Figure 5.2: The areas which are used for the evaluation of the methods are indicated with orangle rectangles, with the corresponding geographic coordinates of the corners in the description.

that are generated by this project are freely available, and can be downloaded for offline use.

5.2. GENERATION OF THE TRAFFIC DEMAND

The traffic demand consists of multiple individual drivers that want to travel from a certain source to a certain destination, with a vehicle type and a desired time of departure. The traffic demand can therefore be modelled as a list of driver profiles, where every driver profile consists of a point of departure, a destination, a vehicle type and a start time. The list of driver profiles are constructed in such a way that together the fastest routes for each driver profile result in traffic distributions that are similar to real traffic distributions.

The list of driver profiles is constructed using the output of a survey that has been performed by Centraal Bureau Statistiek (CBS), a Dutch office for statistical analysis in 2014. The research is called 'Onderzoek Verplaatsingen in Nederland' (OViN), which is loosely translated to 'Study to displacements in the Netherlands' [26]. In this survey 0.3% of the Dutch inhabitants keep track of their movements for one day in the year. The survey consisted of more than 100 questions, but for the construction of driver profiles only a part of the questionnaire is needed.

GENERATION OF ORIGIN-DESTINATION COUPLES

One of the main components of the OViN survey [26] was the tracking of movements. The participants noted the numerical part of the postal codes of their point of departure and their destination, together with their means of transportation. For the construction of driver profiles only the origin-destination postal codes of participants that drive by car are used.

The participants of the survey are inhabitants originating from all regions of the Netherlands. In this research maps are used that contain only a part of the map of the Netherlands. The participants of the OVIN survey can be categorized into four groups. The first group of driver profiles consists of participants with both their point of departure and their destination postal code within the map that is used. The second group has either its origin, or its destination inside the map. The position outside the map is mapped to the border of the map, by computing the intersection between the border of the map and a straight line between the original point of departure and the destination point. The nearest main road to this intersection point is regarded as the point of departure (or destination) of this driver profile. The third group of driver profiles consists of participants that have their point of departure and their destination outside the map that is used for this research, but the straight line between those points intersects with the map. The fourth group of driver profiles does not have a point of departure or destination within the map, and a straight line from the point of departure to the destination does not cross the borders of the area that is considered. This group of driver profiles is ignored for the creation of a traffic model.

In reality, some drivers from the third group of driver profiles do not drive through the considered area, while some drivers from the fourth group will drive through it. These dissimilarities with the reality will most likely occur on and near the borders of the map. The dissimilarity of the traffic model with the reality could be kept small by choosing the borders of the area at regions with a small road density, for instance at the countryside around cities.

For the privacy of the participants of the OViN survey 4-digit postal codes are used for the points of departure and destinations. In the Netherlands the 4-digit postal codes represent usually large districts of municipalities. Therefore postal services often make use of a 6-token postal code that consists of the 4-digit postal code, enhanced by two characters from the alphabet. These postal codes are usually unique for every street. This precision is needed for the driver profiles to be able to analyse the effects of traffic dispersion within city districts. The list of driver profiles is created by selecting each time the point of departure and destination pair of a random participant of the OViN survey, enhancing the source and destination postal code with two 'random' characters, but the combination must be an existing postal code.

This procedure is repeated until the size of the list is equal to the desired number of driver profiles. The desired number of driver profiles can be estimated by expanding the group of OViN driver profiles according to the participation rate of the OViN survey (0.3% of the Dutch inhabitants took part). The survey has been performed in a broader area than the areas that are concerned in this research. Therefore the number of participants of the OViN survey is too low to select a specific day of all replies. Instead, the movements of a whole year are collected and used for the construction of a list of driver profiles for one day. Merging the movements of week days, with a lot of traffic between homes and work places, and days in the weekend, with more private trips, is the main drawback of this measure. However, a set of driver profiles that is extensively spread out over the map is preferred over the correctness of origin-destination couples.

From the responses to the OViN survey, 3618 movements have one or more endpoints in the area of Amsterdam. We extrapolate from this number that the number of driver profiles in one day should be around 1200000, by considering the fact that 0.3% of the Dutch inhabitants took part in the survey. By applying the same reasoning for Rotterdam, we extrapolate that the number of drivers per day should be around 1750000 in that area.

GENERATION OF START TIMES

Together with the point of departure and destination postal code, participants have also submitted a time of departure, in the remainder of this report called 'start time'. If the driver profile has its point of departure outside the map, the start time of the OViN participant is delayed with the distance [km] of a straight line from source to the border of the map divided by a constant speed of 70 [km/h]. A histogram of the start times of the OViN participants is shown in Figure 5.3. The start times are collected from the responses of every day of the year. It can be observed that most people rounded their start times to the nearest half hour, because these bins have the highest frequency.

Although participants to the OViN survey submitted a start time corresponding to their displacement, this start time should not be adopted directly for the start times of driver profiles in this research, because it is not realistic to assume that a large number of drivers would start from exactly the same time from the same point of departure to the same destination. Therefore, the start times of driver profiles in this model should be constructed using statistical methods. Two methods have been considered for constructing start times for the driver profiles, namely by kernelisation of the OViN entries, or by Gaussian randomization per OViN entry.

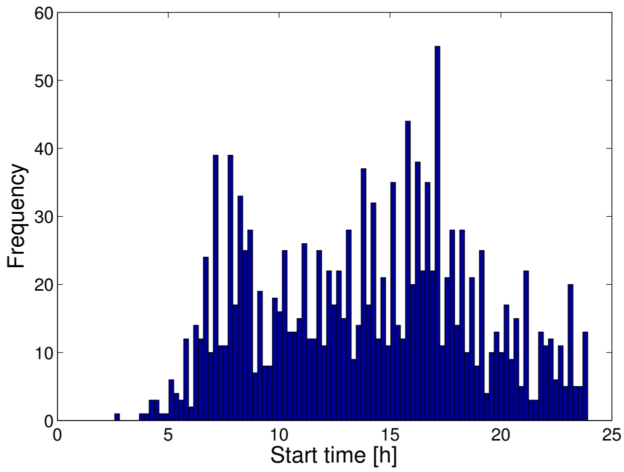


Figure 5.3: Histogram of the original start times, with a bin size of 0:15. The start times are collected from every participant in 2014 in the area of Amsterdam.

Kernelisation of all OViN entries From the OViN start times two peaks can be recognized, corresponding to the morning and afternoon rush hours. Based on the OViN start times an estimation of the probability density function (PDF) can be made, using a mixture of Gaussians procedure. Because we can identify two peaks for the corresponding rush hours, the

PDF of the data can be represented by a mixture of two Gaussians. The resulting estimation of the PDF can be found in Figure 5.4. This PDF can be constructed by two separate normal distributions, with two different means and variances. Each distribution gets a prior that determines the likelihood that a sample is taken from that distribution. In this case the start times of the list of driver profiles is constructed by a normal distribution centered at 9:39, with a variance of 2 hours and 11 minutes, and a normal distribution with its center at 17:22 and a variance of 2 hours and 38 minutes. The probability that a random sample is taken from the first normal distribution is equal to 0.4169, while the probability that it is taken from the second distribution is equal to 0.5831. A histogram of the generated start times is shown at Figure 5.5a.

This method for the generation of start times has two main drawbacks. The driver profiles of the OViN instance are combinations of origin-destination information and a corresponding start time. This method removes the relation between place and time. For instance, assume that in the original set of OViN driver profiles most people would drive from South to North in the morning, and in the afternoon from North to South. Since in this method the start times of the driver profiles are selected randomly from the generated PDF, it is presumably the case that both in the evening and the morning the same number of drivers drive from North to South as there drive from South to North.

Another drawback is that the traffic density appears to be very smooth over different quarters of an hour, as a consequence of the choice for a Gaussian distribution with two means. The behaviour of people is much more complicated than a Gaussian distribution with two or slightly more means. If we were to assume in this research that all traffic is generated as a smooth Gaussian distribution, we would remove the effect of peaks, and we would already assume a better spreading of the traffic for the base case.

Gaussian randomization per OViN entry A better method is to save the connection between the navigation data and the start time. However, it is not realistic to generate more than 100 drivers that drive from the same city districts to the same destination districts at exactly the same second. Instead, a Gaussian noise is added to the original start time of the OViN instance. So, the start times of drivers with the same point of departure and destination are generated based on a Gaussian distribution with the mean equal to the start time of the OViN instance, and a variance of 15 minutes. 95% of the driver profiles based on one OViN entry have a start time with a maximum difference of 30 minutes from the original OViN entry, as a consequence of the Gaussian distribution. Together, the distribution of the start times of the generated list of driver profiles is similar to distribution of the original start times from the OViN survey (Compare Figures 5.3 and 5.5b). Therefore, this method is used for the generation of start times for the list of driver profiles.

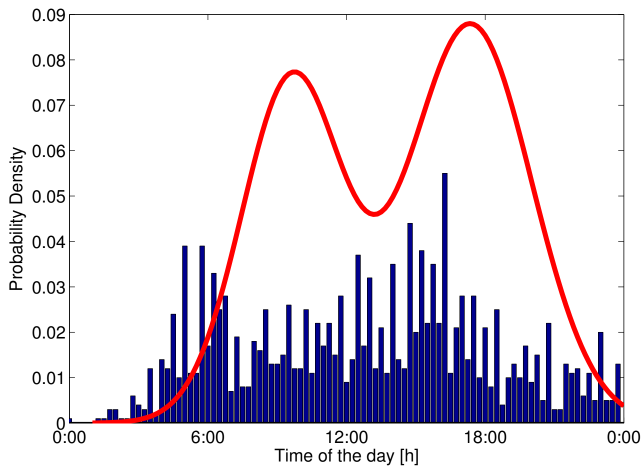
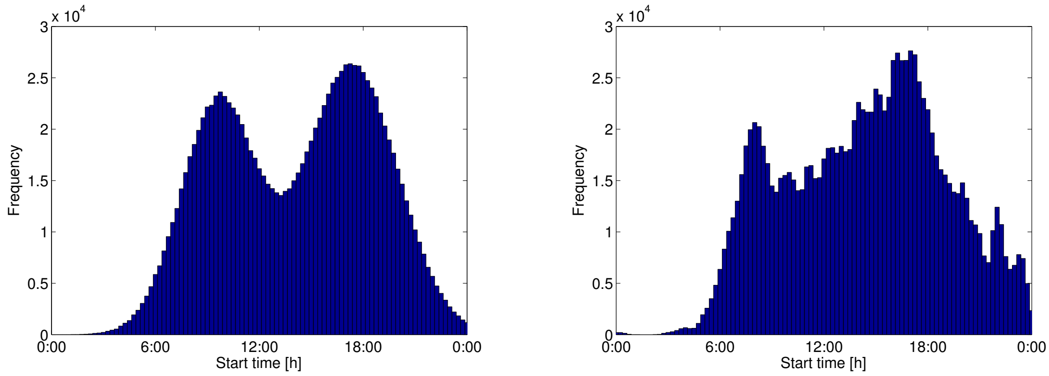


Figure 5.4: In red the Probability Density Function estimation of the OViN start times, with means 9:39 and 17:22, and variances 2:11 and 2:38, and the priors equal to 0.4169 and 0.5831 (time notation [hours:minutes]). The blue bins represent the original start times, which are the same as in Figure 5.3.

DETERMINATION OF VEHICLE TYPES

The emissions vary between different vehicle types. In the Netherlands all vehicles with a license plate are registered at the 'Rijksdienst voor Wegverkeer' (RDW). The RDW shares information about all the registered vehicles through online databases [27]. From this database the vehicle fleet can be divided into three fuel classes: Petrol, Diesel and a remaining group of vehicles that are mostly powered by sources that have little or no influence on the environment. For this model the fleet of vehicles is separated into four groups: Petrol cars, Diesel cars for passengers (<2500kg), Diesel trucks and electric cars. The shares of each vehicle type that is used in this research are shown at Table 5.1. The shares of the fleet are based on the information about the fuel use from the database from the RDW [27]. Approximately 74% of the vehicles uses a Petrol engine as power source. The central statistic agency of the Netherlands (CBS) states that the part of the kilometres that is driven by trucks, is approximately 6% [28]. In this simulation also a part of the vans are classified as 'truck',



(a) With Mixture of Gaussians

(b) With Gaussian Noise on samples

Figure 5.5: Histograms of generated start times per method, with a bin size equal to 15 minutes.

such that we arrive at a part of 8% for the trucks. The assumption is made that almost all trucks are propeled by Diesel engines. From the RDW it is known that 23% of the vehicles uses Diesel as their fuel. This means that 15% of the vehicles can be classified as a passenger car that is propelled by a Diesel engine. The fourth group in this experiment is the group of electric vehicles. Currently only 2% of the vehicles in the Netherlands is actually powered by electricity. However, the remaining vehicle types have in common that they have lower or even zero emissions. Also, a large part of the vehicle fleet is powered by hybrid sources. Therefore this group is represented by an electric car with zero emissions. Although it is known that the number of driven kilometres is larger for vehicles that are propelled by Diesel engines compared to the driven kilometres of Petrol cars, the shares the vehicle fleet of each vehicle type will approximately be equal to the distribution of the registered vehicles.

Table 5.1: The shares for each vehicle type of the total fleet of vehicles that is used in this experiment.

Vehicle type	**Part of fleet** [%]
Petrol	74
Diesel Passenger Car	15
Diesel Truck	8
Electric	3

As stated in Section 2.3, the speed of vehicles at edges is fixed to the maximum speed of that road section. However, some vehicles are not able to drive that speed. On top of that, for some vehicle categories additional constraints apply for the maximum speed. In this model, the speed of drivers is limited to the maximum speed they are allowed to drive. For Petrol, Diesel and Electric cars the speed is limited at 130 km/h. For trucks, the speed limit is set to 90 km/h.

5.3. SOFTWARE IMPLEMENTATION

For the implementation of the routing algorithms and methods we do not have to start from scratch. A lot of software projects exist that have implemented routing algorithms. One of such projects will be used to implement the methods and algorithms that have been constructed in Chapter 4. Section 5.3.1 describes several available routing machines. Once a suitable project is chosen as a basis, the implementation of the methods is described.

5.3.1. DETERMINATION OF BASE ROUTING SOFTWARE PROJECT

In this section a choice will be made on which software the implementation of the simulation model and the methods will be based. The requirements on which the different routing software projects will be compared are as follows.

- The routing software must translate Open Street Maps to graphs of nodes and edges.
- It must be possible to adjust weights of edges dynamically, during the routing process.

- The software project must be well documented, and the code must be easily adaptable.
- Preferred are projects that are used for broader application than only shortest path routing, because it is more likely that these projects provide more adaptability to the needs of this project.

A lot of routers have been published and deployed on the web that can be modified to implement the algorithms described at Chapter 4. This section describes different existing routing machines. One of the main requirements of this project is that the cartography data that will be used is in the OpenStreetMap (OSM) format. Out of many available routing machines[2], three systems make it possible to perform routing around the whole globe, have been intended for multiple vehicle types (cars,bicycles etc.), and have their code published and managed on GitHub.

YOURS YOURS is an Openstreetmap Routing Service of which the actual routing part is performed by the subsystem Gosmore. Although the code of Gosmore is available online via svn and on GitHub.com, it is not actively managed online and little documentation about the code is available. This makes it hard to analyse the software, and most presumably it will be hard to modify the software for the needs of this project.

Open Source Routing Machine (OSRM) A project that is managed much better is the Open Source Routing Machine (OSRM) project[3]. The system is written in C++, and divided in classes, so the code is easily adaptable. The routing server can be deployed locally or online, and is also available as an external server. Before the server is able to handle requests, several steps must be performed. First the OSM data is extracted into OSRM-data that is used by the system to handle route requests. The final step is to prepare for route requests. In order to speed up the response times of routing queries, the routing algorithm is based on Contraction Hierarchies. The weights of the edges are based on a profile that must be fed to the program before extracting the OSM data. In order to be able to handle traffic data, the system can be started differently, by making it possible to change the weights of some edges later on. The weight changes of particular edges can be supplied. However, the server startup procedure must be performed again after the changes have been made.

Because of the use of Contraction Hierarchies the weights of edges can not be adjusted during the routing process. Therefore OSRM seems not a good choice to be used for this project.

GraphHopper GraphHopper is a routing machine that is written in Java. It is actively managed online by means of GitHub[4], where several contributions of external developers have been accepted. The repository contains a local web server that can be queried via a web-based GUI, or directly. The system is designed to handle different routing algorithms. Among others Dijkstra's algorithm, the A* search algorithm and Contraction Hierarchies have been implemented. The weights can be modified during runtime of the server. Individual edges can be found by using geographical coordinates. The software is used by other researchers which design a smart route planning system [29], a multi-modal employee scheduling system [30] or a real-world employee scheduling and routing application [31]. Therefore, GraphHopper has been chosen as the starting point for the development of the traffic simulator.

5.3.2. IMPLEMENTATION OF USER-BASED SYSTEM

Weighting and Time division The calculation of edge weights is done by extending the Weighting class incorporated in the GraphHopper project. An additional function is added which makes it possible to retrieve the weight of an edge on a specific time. The time is divided into periods of 60 seconds. It is possible to vary the parameters α, β, γ, C_{nox} and C_{eco}.

Dijkstra's time-dependent algorithm Dijkstra's time-dependent algorithm is implemented by adjusting the implementation of the original static Dijkstra's algorithm. A time-dependent weight is requested from the weighting class, while also the expected duration in seconds of that edge is returned. The time-dependent weight is used for finding a minimum-weight solution, while the time is stored in the edge-tree, such that the weight of the successive edge can be retrieved at the correct time. Also, once the algorithm has found the optimal path, it must be able to retrieve the expected travel time of a route, because it has to be compared to the base case, where only the travel time is part of the weighting.

[2] An overview of the available routing software for OSM can be found at http://wiki.openstreetmap.org/wiki/Routing/online_routers.

[3] http://project-osrm.org/

[4] https://github.com/graphhopper/graphhopper/

Updating NOx concentrations When Dijkstra's time-dependent algorithm has finished, the NOx concentrations in the grid are increased. Starting from the destination node, the concentrations are increased at the correct times, where the travel time of each edge is used to estimate the time at which the car will drive through that edge. The concentrations of NOx are stored at time periods of 60 seconds. The addition of the concentrations of NOx on grid points on a certain edge at a certain time is given by Equation 5.1, where v represents the expected speed of the vehicle, p_t the expected time at that edge in that time period and NOx(v) the expected emission of NOx, based on the vehicle type and expected speed. The traveled distance is multiplied with the emission characteristic for the right vehicle type (Table 3.1). ϕ(t) is computed by summing up the additions of all drivers. If a driver is expected to drive multiple time periods at the same edge, the NOx concentrations are increased at the corresponding time points, although no distinction is made between grid points on the one end of the edge to the other end.

$$\text{Addition} = v \times t_p \times NOx(v) \tag{5.1}$$

Updating total CO_2 emission In the same way the total CO_2 emission is stored. Each time that a route is found, the emission per edge per timeframe is calculated by Equation 5.2. The traveled distance is multiplied with the emission characteristic for the right vehicle type (Table 3.1). All these additions are summed and stored as the total CO_2 emission.

$$\text{Addition} = v \times t_p \times CO_2(v) \tag{5.2}$$

5.4. CONFIGURATION AND TEST CASES

Because of the many different free parameters in the determination of the weight of edges, the solution space is large. Therefore, in this section different configurations are described that will be used to evaluate the outcome of the methods. In all configurations, the exceedance of the NOx threshold (Equation 2.2) is calculated from 30 minutes after the start of the simulation, because at the first 30 minutes of the simulation the traffic distribution is not a good representation of the real life traffic distribution. These 30 minutes are part of the initialisation phase of the simulation, because the trajectories of drivers that have started before the timeframe of the simulation are not contained in the simulation. In all configurations the timeframe is set to 60 seconds. First, in Section 5.4.1 is described how an optimal parameter setting can be achieved. Then, in Section 5.4.2 some corner cases are constructed, and the expected outcome is described. In Section 5.4.3 is described how the influence of ρ on the outcome of the constructed method is evaluated. Therefore, in Section 5.4.4 is an experiment constructed which will reveal the impact of lower participation rates on the outcomes. Finally, a comparison between method 1 and 2 is constructed in Section 5.4.5. For every configuration is described what the expected outcome of the experiments would be.

5.4.1. PARAMETER TUNING

Many different free parameters determine the weight of edges, which will lead to a lot of different solutions. In Section 5.4.2 is described which cases we will compare with each other. In this section is described how a case is constructed that has a balance between travel times, the avoidance of NOx exceedances and the emission of CO_2. Therefore, we fix the parameters α to 0.4, β to 0.5 and γ to 0.1. By performing route requests for short time periods, a suitable value for the scaling factor C_{eco} is chosen, such that the unweighted CO_2 part is roughly as big as the travel time part.

Once values have been chosen for C_{eco}, α, β and γ, we will explore which value is the most suitable for C_{nox}. The value for C_{nox} that results in the lowest NOx exceedance of the threshold θ will be chosen for the following experiments.

We expect that for large values of C_{nox} the weight is dominated by the travel time, while for small C_{nox} values the weight will be dominated by the NOx part of the equation. If the weight is dominated by the travel time, we expect similar results as for the case that every driver would optimize its travel time, thus a small total travel time, a large exceedance of the NOx threshold and a large total emission of CO_2. If the weight is dominated by the NOx part, a route choice is based with a strong local focus, resulting in longer travel times, but less exceedance of the NOx threshold. If the influence of the NOx part is too large, routes become that long that the total emission of NOx will grow, leading eventually to larger exceedances of the NOx threshold.

The parameter tuning is performed in the area of Amsterdam, on the timeframe from 16:40 to 18:10, when the traffic densities are the highest.

Table 5.2: Parameter configuration of different cases

Configuration name	α	β	γ	C_{nox}	ρ [%]	C_{eco}
Base	1.0	0.0	0.0	1	120	10
Balanced	0.4	0.5	0.1	1	120	10
Eco	0.0	0.0	1.0	1	120	10

5.4.2. COMPARISON OF DIFFERENT CASES

The parameter configuration that leads to the lowest total exceedance of the NOx thresholds in Section 5.4.1 is regarded as the 'balanced' case. This case is compared to two other cases: the extreme configurations of α and γ. We have tuned this case in the area of Amsterdam in such a way that the NOx exceedance is minimal. Therefore, we expect that this case has a minimal NOx exceedance, compared to the other two cases.

Table 5.2 shows the exact configurations per case.

The extreme case where α is equal to 1 and β and γ to 0 is considered as the base case. This case is how we consider that the set of routes would be without the methods in this research. In such a configuration every driver tries to optimize its own travel time. We expect the lowest total travel time, the greatest exceedance of the NOx threshold and the highest total CO_2 emission for this configuration.

The extreme case with $\gamma = 1.0$ corresponds to the situation where every driver tries to minimize the emission of its own route. In some navigation devices such an objective is called 'Eco', therefore in this research this configuration is also labelled by that name. This is somewhat similar to the methods of Patil [11] and others, although in this experiment no congestion is taken into account. We expect that this configuration will lead to longer travel times and the lowest possible total CO_2 emission. Furthermore, because the emission characteristics for CO_2 and NOx have a similar shape, we expect also a reduced exceedance of the NOx threshold, because the total emission of NOx would also be reduced by this method.

The simulations will be performed from 16:40 to 18:10, both for the area of Amsterdam and for Rotterdam. In all cases the maximum detour (ρ) of individual drivers is kept at 20% of the shortest travel time. We expect that the balanced case can be less optimal for the area of Rotterdam, because the configuration of the weights is optimized for the area of Amsterdam.

5.4.3. INFLUENCE OF ρ

Another parameter that is interesting to explore is ρ, which determines the maximum length of the detour of an individual driver. For the balanced case, ρ is set to 10, 20, 50 and 100%, with the same values for α, β, γ and C_{nox} as in Table 5.2. The simulation is performed in the area of Amsterdam from 16:40 to 18:10.

We expect that for small values of ρ the outcome is similar to the base case. For large values of ρ the exceedances of the NOx threshold can grow, because a long detour will lead to larger total NOx emissions. A large ρ will therefore also lead to a higher total emission of CO_2 and a larger total travel time.

5.4.4. INFLUENCE OF THE SHARE OF PARTICIPATING DRIVERS

We expect that not all drivers would be willingly to participate in a collective routing system. Therefore, one of the research questions was to explore the impact of lower participation rates to the project. This is measured by configuring the weighting function for every driver to either the balanced case or the base case, based on the participation rate. The routes of the part of the drivers that participate in the project are calculated first with the parameters for the balanced case. For the remaining part of the drivers the base case is configured.

We expect that the impact of small participation rates on the NOx exceedance is very small. From a certain amount of participants this effect will grow fast.

Because the configuration of the balanced case has a large influence on the travel time of individual drivers, we expect that the total travel time will grow linearly with the participation rates.

5.4.5. COMPARISON BETWEEN METHOD 1 AND 2: INFLUENCE OF ITERATIONS

In this setup we will analyze what the effect is of performing the balanced case with multiple iterations. The simulations will be performed for the area of Amsterdam, from 16:40 to 17:40. Due to the longer run-times for multiple iterations the time duration of the simulation is reduced to one hour.

For method 1 and for method 2, we will analyze the percentual growth of the individual travel times compared to the fastest possible travel time. Is is necessary to apply an averaging filter over this data, because otherwise it would not be possible to analyze a plot of this data.

We expect that for method 1 the average percentual growth will grow with the position in the list, due to the fact that the first drivers in the list have no emission traces from the successive drivers in the list. For method 2 we expect that the percentual detour for all drivers in the list will be the same on average, regardless of the position in the list, because a user equilibrium would be reached.

Furthermore, we will explore the effect of the number of iterations on the NOx exceedance, the total CO_2 emission and the total travel time.

We expect that the NOx exceedance will be reduced for increasing number of iterations, until the user equilibrium would be reached. The travel times will increase with the number of iterations, until the user equilibrium is reached. The total emission of CO_2 will stay roughly the same for larger number of iterations, because the reduction of the own CO_2 emission is not influenced by the order in which the routes are computed for the list of drivers.

6
Results

In this chapter the results are presented for the experiment which is described in Chapter 5.

6.1. Parameter tuning

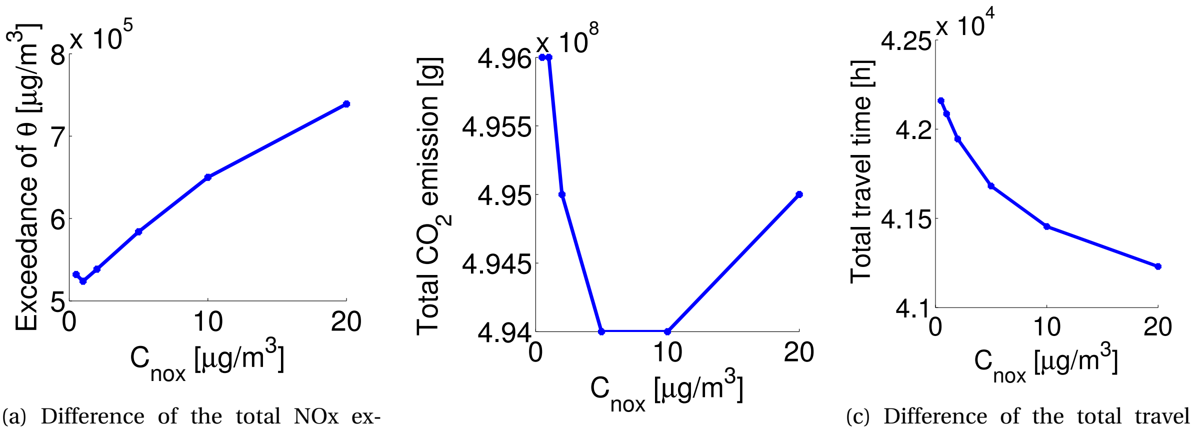
Figure 6.1 shows the effect of C_{nox} on the total exceedance of the NOx threshold that is set to 200 $\mu g/m^3$, measured according to Equation 2.2 from the Problem description at Chapter 2. Like we expected, for large values of C_{nox} the NOx exceedance is large and the travel time short, while for small values of C_{nox} the NOx exceedance is low and the travel time long. For large values of C_{nox} the total CO_2 emission is large, as we expected, but for average values of C_{nox} the reduction of the CO_2 emission is the largest.

For the value of C_{nox} with the lowest exceedance of the NOx threshold, the reduction of the total CO_2 emission is less. Apparantly, if the weight is dominated by the travel time more routes contain more highways at which the emission per km is higher. If the weight is dominated by the NOx part, some routes take such a long detour that the number of driven kilometers is increased that much that the total emission of CO_2 grows. For the reduction of the total emission of CO_2 a balance between travel time and the local NOx focus is optimal.

For the remaining experiments, the optimal configuration for C_{nox} is 1.0, because that configuration has the lowest exceedance of the NOx threshold and a reduced total emission of CO_2.

6.2. Comparison of different cases

Comparing the three cases as described in Section 5.4.2, we see that the base case has indeed the lowest travel time, the greatest exceedance of the NOx threshold, and the greatest total CO_2 emission. We express



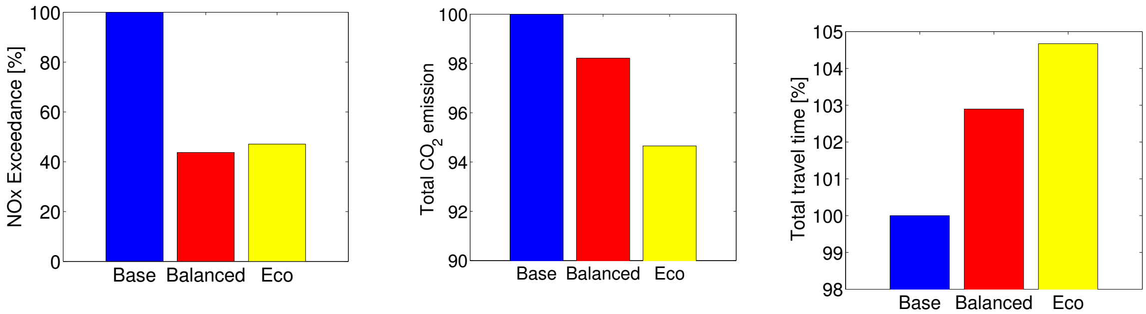
(a) Difference of the total NOx exceedance of θ.

(b) Difference of the total CO_2 emission.

(c) Difference of the total travel time.

Figure 6.1: Determination of the optimal setting for C_{nox}.

the results of the two other cases therefore as percentual differences with this base case in Figure 6.2 for the area of Amsterdam.



(a) Difference of the total NOx exceedance of θ.

(b) Difference of the total CO_2 emission.

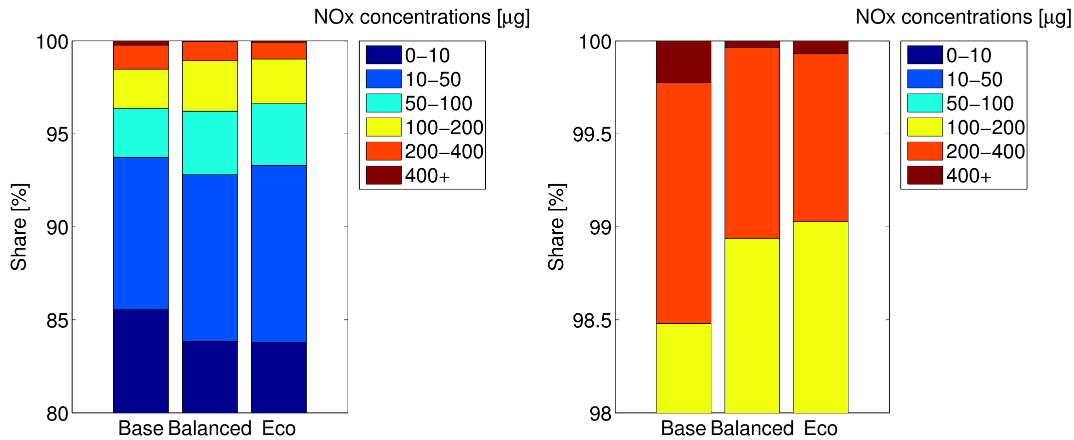
(c) Difference of the total travel time.

Figure 6.2: Results for the different cases on the map of Amsterdam from 16:40 to 18:10. All cases do have an exceedance of the NOx threshold, so the results are presented as percentual differences with the base case.

In the balanced case the exceedance of the NOx threshold can be reduced by 56.3% for the area of Amsterdam, while the total travel time increases by 2.9%. Meanwhile, the total CO_2 emission is reduced by 1.8%. Note that the individual travel times are maximally increased by 20%. We see that the exceedance of the NOx threshold is minimal, compared to the base case and the eco case.

The eco configuration leads also to lower NOx exceedances, the lowest total CO_2 emission, with the cost of increased travel times. This is like we expected.

Figure 6.3 shows the distribution of the NOx concentrations across the whole area of Amsterdam, with a simulation time of an hour. The reduction of the total NOx exceedance of the balanced case compared to the base case leads to an increased number of grid points that have a concentration between 50 and 100 $\mu g/m^3$, which is far below the threshold of 200 $\mu g/m^3$. The greatest reduction is the number of grid points that have a concentration above 400 $\mu g/m^3$. Grid points with such a large exceedance have a large influence on the total exceedance.

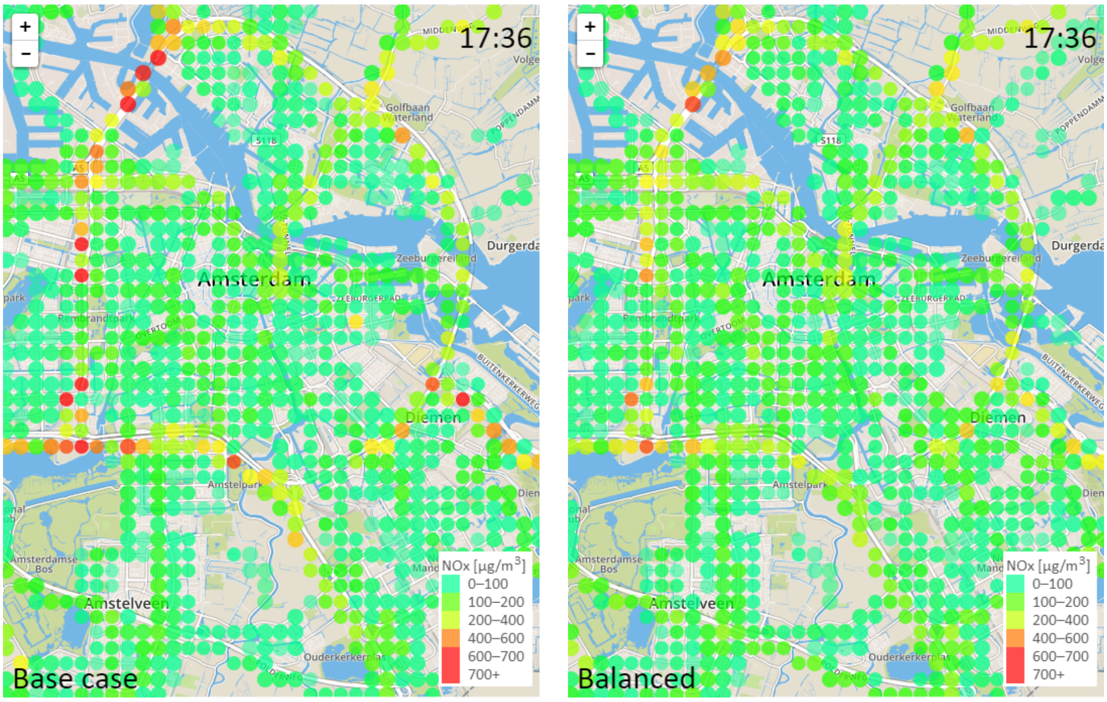


(a) Share of different concentrations levels from 80% to 100% of the number of grid points.

(b) Share of different concentrations levels from 98% to 100% of the number of grid points.

Figure 6.3: Overview of the distribution of the NOx concentrations across the whole area. In the balanced case, the number of grid points that have a concentration between 10 and 50 $\mu g/m^3$ is increased, while the number of grid points that have concentrations above the threshold is reduced, compared to the base case.

Figure 6.4 shows the NOx concentrations of the centre of Amsterdam at 17:36. Corresponding to the legend in the lower right corner, red dots correspond to high NOx concentrations, while green dots represent low NOx concentrations. The NOx concentrations are to a large extent lower than 200 $\mu g/m^3$, especially on the western side of the ring road around Amsterdam.



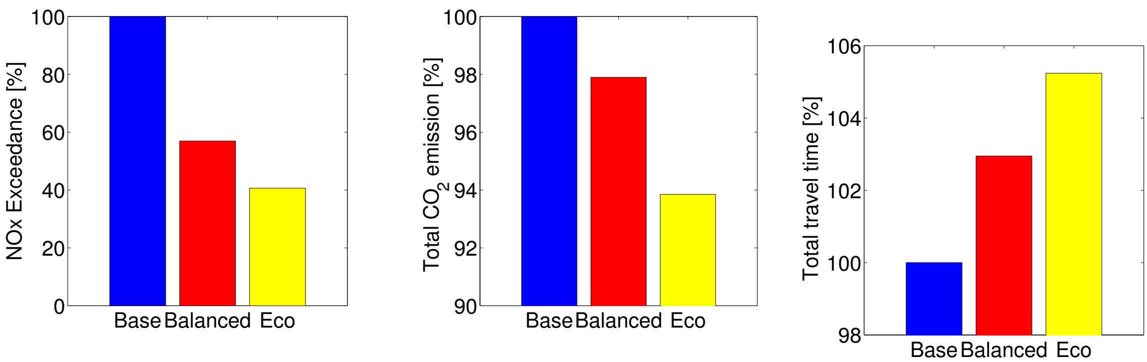
(a) Base case.

(b) Balanced case.

Figure 6.4: Rendition of the NOx concentrations at 17:36 for the area of Amsterdam. The NOx concentrations are to a large extent lower than 200 $\mu g/m^3$, especially on the western side of the ring road around Amsterdam.

The same simulation setup has been applied to the area of Rotterdam. Figure 6.5 shows that the exceedance of the NOx threshold is reduced by 43.0% for the balanced case, compared to the base case. The total CO_2 emission is reduced by 2.1%, while the total travel time is increased by 2.9%. The total travel time in the eco case is even more increased, as well as the reduction of the NOx exceedance. Furthermore, the reduction of the total CO_2 emission is greater.

The balanced case has only a reduction of 43.0% of the exceedance of the NOx threshold for Rotterdam, while this was 56.3% for Amsterdam. An explanation for this behaviour could be that the same configuration from the parameter tuning for Amsterdam has been used for the area of Rotterdam. A different tuning for Rotterdam may result in a greater impact of the NOx part of the weight, leading possibly to better results.



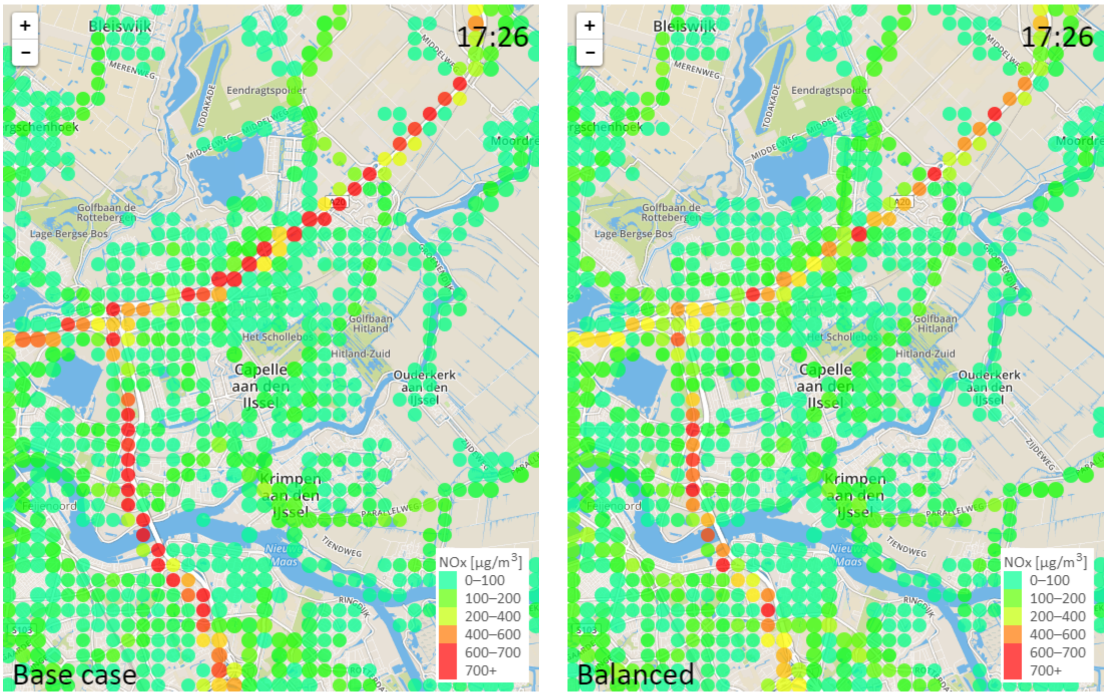
(a) Difference of the total NOx exceedance of θ.

(b) Difference of the total CO_2 emission.

(c) Difference of the total travel time.

Figure 6.5: Results for the different cases on the map of Rotterdam from 16:40 to 18:10. The differences are expressed as percentual differences with the base case.

Figure 6.6 shows the NOx concentrations of the centre of Rotterdam at 17:26. Corresponding to the legend in the lower right corner, red dots correspond to high NOx concentrations, while green dots represent low NOx concentrations. In the balanced case, the number of points that have a concentration above the threshold θ is less. At this timeframe, still at a lot of points the concentrations are above threshold θ, which is due to the large traffic demand at this timeframe, and the requirement to offer short routes for individual drivers.



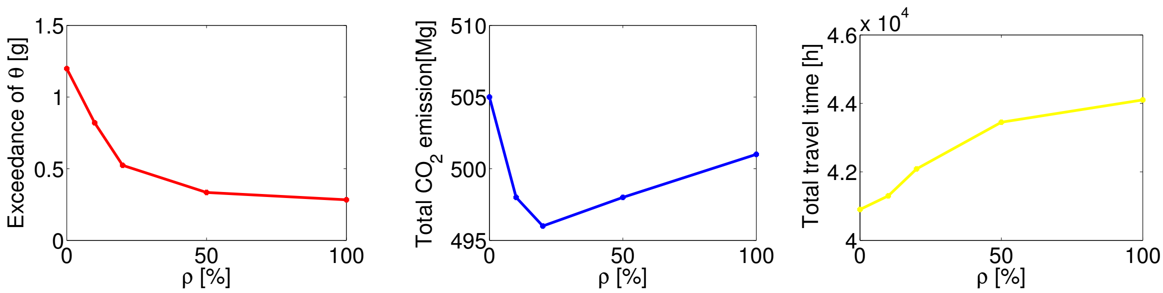
(a) Base case.

(b) Balanced case.

Figure 6.6: Rendition of the NOx concentrations at 17:26 for the area of Rotterdam for the base case and the balanced case.

6.3. INFLUENCE OF ρ

Figure 6.7 shows the impact of maximum length of the detour on the exceedance of the NOx threshold, the total CO_2 emission and the travel time. Like we expected, for small values of ρ the outcome is similar to the base case: short travel times, large exceedances of θ and a large total CO_2 emission. The travel time grows with the length of the detour between 10 and 100%. We see that the exceedance of θ is still reduced if the detour of individual drivers is increased to 100%, which is different from what we expected. Apparently, the tuning process has yielded an optimal balance between reducing the NOx exceedance and the travel time.



(a) The total NOx exceedance of θ.

(b) The total CO_2 emission.

(c) The total travel time.

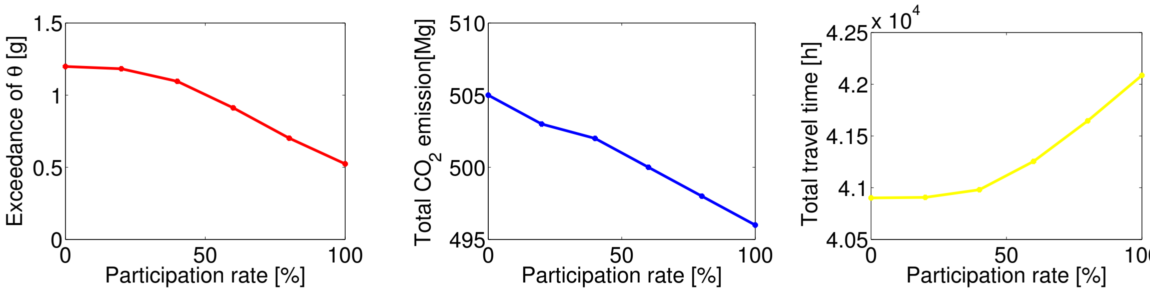
Figure 6.7: The influence of the maximal length of the detour.

We can derive from Figure 6.7 that for the reduction of CO_2 a balance exists between short and longer travel times. Presumably, forcing drivers to roads with a maximum speed close to 80 km/h leads to the lowest total emission of CO_2. Such roads take a slightly longer travel time than highways with a speed limit of 130 km/h. But, if a larger detour is allowed, drivers are forced by the NOx part of the weight to roads with even lower speed limits, which have a larger emission characteristic per driven km. In that case, the total emission of CO_2 is increased.

6.4. INFLUENCE OF THE SHARE OF PARTICIPATING DRIVERS

Figure 6.8 shows the influence of the number of participants on the outcome of the problem. Like we expected, the effect of the system with small participation rates is very small. However, the effect of the method on exceedance of θ grows approximately linear with the number of participants after 40% of the participants takes part.

Other than we expected, the total travel time does not grow linearly with the number of participants. We can derive that there is a direct relation between the exceedance of θ and the travel time. The lower the exceedance of θ, the higher the travel time.

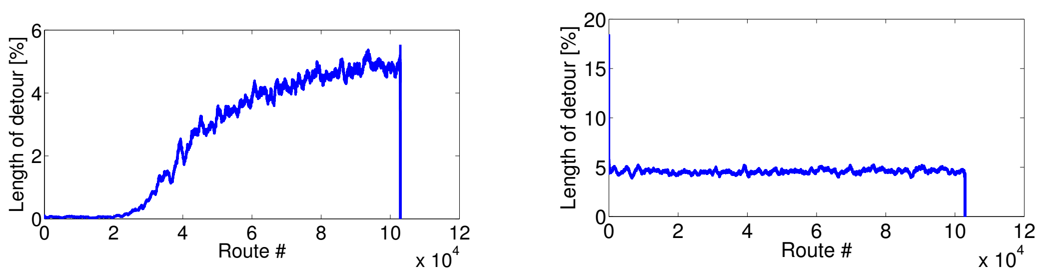


(a) The total NOx exceedance of θ. (b) The total CO_2 emission. (c) The total travel time.

Figure 6.8: The influence of the participation rate of the project.

6.5. COMPARISON BETWEEN METHOD 1 AND 2: INFLUENCE OF ITERATIONS

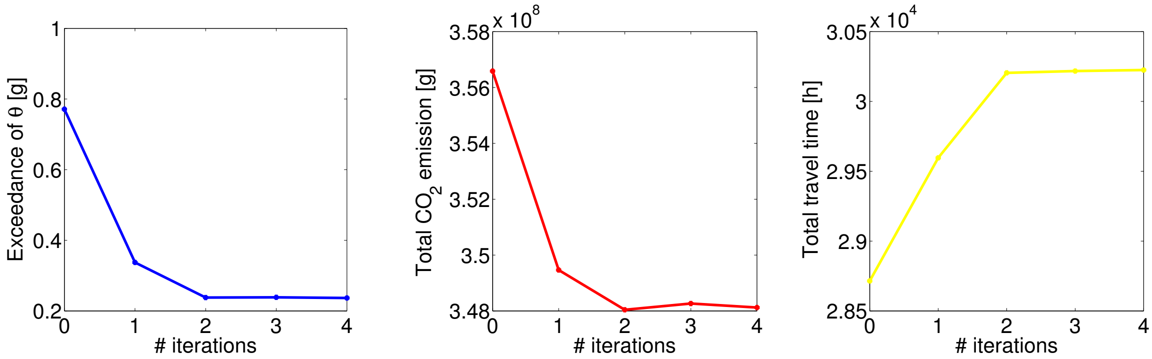
Figure 6.9a shows that for method 1, which is equivalent to method 2 with one iteration, the routes that are calculated last, are longer than the routes that are planned first. The first routes are calculated on an graph with almost no stored emissions, which results in an average detour of 0% for the first 20000 routes. After half of the routes have been planned, the increase of the travel times converges to 5%. This is similar to what we expected, because the first drivers in the list has no emission traces from the successive drivers in the list at his disposal. Figure 6.9b shows that the advantage of drivers that route as first is removed after multiple iterations. Drivers do not get shorter routes if they route earlier, because their route is reconsidered at new iterations.



(a) After one iteration. (b) After four iterations.

Figure 6.9: The increase of the travel times against the number of drivers increases. Drivers who calculate their route later, have generally longer travel times after one iteration. This effect is removed by multiple iterations. In order to be able to view a trend, the travel time increases are shown as a moving average over 1000 samples. These travel times are of routes that are calculated using $\alpha = 0.4$, $\beta = 0.5$, $\gamma = 0.1$ and $C_{nox} = 1$. The peaks at the right side of the figure is caused by the fact that at the end of the list the input of time differences to the moving average filter is too small to remove peaks.

Figure 6.10 shows the effect of multiple iterations on the exceedance of the NOx threshold, the total CO_2 emission and the total travel time, for a simulation for the area of Amsterdam from 16:40 to 17:40. The exceedance of the NOx threshold decreases with the number of iterations, until 2 iterations. After two iterations the NOx exceedance does not further decrease with the number of additional iterations. The total CO_2 emission seems to increase a little with the number of iterations, while the travel time increases. Compared to the base case, after one iteration the NOx exceedance is reduced by 56.3%, while after 4 iterations the NOx exceedance is reduced by 69.3%.



(a) The total NOx exceedance of θ versus the number of iterations.

(b) The total CO_2 emission versus the number of iterations.

(c) The total travel time versus the number of iterations.

Figure 6.10: The effect of the number of iterations.

7

Future work and recommendations

Although the methods that are constructed in this research are able to assign routes to drivers such that the exceedance of NOx concentrations and the emission of CO_2 are reduced while low travel times for drivers are preserved, several enhancements can be made to this research. Furthermore, other methods can be explored that satisfy the requirements of the problem description, which we will show in this section.

7.1. Enhancements to the Simulation

The simulation model consists of two parts, namely the traffic demand and the emission model. Improvements to these models will presumably lead to more accurate simulation results. This section will therefore suggest some possible enhancements.

7.1.1. Enhancements to the Traffic Demand Model

Although the traffic model in this research supplied a diverse traffic demand model to the methods, some enhancements can be made to represent the real-life traffic distributions better.

Vehicle types

In this research only 4 vehicle types are used. In practice the fleet of vehicles is way more diverse. Also, buses are not specifically regarded, while bus routes of regular buses can not be changed dynamically. Furthermore, the routes of trucks are in this research based on displacements of passenger cars. In reality, trucks will have different sources and destinations, like harbours and industrial areas.

Variable speed of drivers

One strong assumption in this research was to assume per edge a fixed speed of the drivers. This would mean that all drivers would keep the maximum allowed speed and have zero acceleration time. Furthermore, waiting times at intersections are removed as well. These factors have a large impact on the travel times of drivers, so for an accurate traffic model these factors must be considered.

Include congestion

In this research for every edge a free-flow speed is assumed. However, in reality many traffic jams occur, which has an adverse effect on travel times and vehicle emissions. In this research methods are explored that reduce the exceedance of NOx thresholds locally by means of a traffic dispersion algorithm. As a consequence of the traffic dispersion, it could be the case that too large a part of the traffic is routed through one-lane edges, which would lead to large traffic jams at these sections. Traffic congestion causes lower speeds which will presumably lead to higher vehicle emissions. The methods that are developed in this research lead to traffic dispersion. Traffic dispersion will generally lead to fewer traffic congestion. It can be the case that the methods of this research lead to higher improvements of the air quality, relative to the base case. It would be interesting to measure this, so that this theory can be proven.

IMPROVE ORIGIN-DESTINATION INFORMATION

The list of driver profiles in this research has been constructed using survey data and assumptions on the generality of this survey. However, the resulting traffic distribution is not compared to vehicle counts in real life. Rijkswaterstaat, a Dutch governmental organisation that is responsible for managing traffic flows, has performed such a research. This resulted in a more realistic and a more accurate Origin-Destination model. This model gives more certainty about the regions where the traffic yields NOx concentrations above the thresholds. For the objectives of this research the OViN survey was sufficient.

7.1.2. ENHANCEMENTS TO THE EMISSION MODEL

The emission model that is used in this research is very basic, so before this system can be introduced in cities, several extensions to this model are needed in order to be able to guarantee that the air quality will be improved by the methods that are explored in this research.

INCLUDE OTHER SOURCES OF NOX EMISSION

In this research only NOx emissions stemming from traffic are used for the construction of a map with NOx concentrations. However, there are other sources of NOx that have a large impact on the local NOx concentrations. For instance, the contribution of the aviation and the shipping industry to the NOx concentrations is substantial. In order to improve the health of citizens in cities, also other contributors to high NOx concentrations must be modelled. In this way, the traffic is further reduced at areas with high NOx concentrations, for instance areas nearby airports, ports, and plants.

INTRODUCE FADING OF EMISSIONS

The emission map in this research does only contain records of vehicle emissions per time and place. Emissions at one timeframe have no influence on the NOx concentrations at the subsequent timeframe. Also, emissions at one grid point are stored independently from surrounding grid points. It would be better to translate the emission model to a concentration model, by assuming spatial and temporal fading. Vehicle emissions at a certain grid point at a certain timeframe would then contribute to the concentration of surrounding grid points, as well as to the concentration of future timeframes.

7.2. INDIVIDUAL APPROACH: ROUTE WITH A PREDICTION OF THE AIR QUALITY

The method in this research is based on the communication of drivers via an emission map. Although it is possible to construct such a system where every driver stores his route in a collective database, a more elementary emission map can be used that is no longer directly dependent on the routes that other drivers store. Such an emission map can be constructed using a prediction of the traffic densities per edge, the emission of other sources of air pollution, like plants. The emission map can even be updated according to the current weather conditions.

In such a method the calculation of routes is almost the same as for the method that is used in this research. However, the cumulative emission function $\phi(t)$ is replaced by a prediction of this value. In this way, drivers can calculate their routes fully distributed, such that sequential routing is not longer needed.

Another advantage of such a system is that the thresholds can be adjusted locally. If a local government requires that the air quality must be improved at a certain area, for instance around a hospital, the thresholds for that area could be lowered, such that the area is avoided as much as possible by traffic.

7.3. ROUTE CHOICES IN A SEQUENTIAL MULTI-OBJECTIVE APPROACH

The methods that are explored in this research construct solution trees with a weighted combination of travel time and emissions. If the resulting paths have a travel time that exceeds the bound on the maximal travel time, the solution is recomputed with a higher weight for the travel time. This leads to larger execution times. Also, the routes that are computed by this method can be less convenient for drivers, because small detours can be part of the route.

It is possible to construct another method that finds routes with low travel times that together reduce the peak NOx concentrations and also the total CO_2 emission, which does not direct drivers through less preferable routes. In this method the route choices are made in a layered fashion. First a set of routes is constructed that have travel times below ρ% of the optimal travel time (Section 7.3.1). For each route in this set the total CO_2 emission weight and the NOx concentration weight can be calculated. By weighted minimization the most suitable routes can be determined per driver.

7.3.1. CALCULATE ALTERNATIVE ROUTES

The Cambridge Vehicle Information Technology Ltd. (CAMVIT) group [32] introduced an algorithm that finds a set of alternative routes in polynomial time. A good set of alternative routes is formed by routes that an expert would also deliver. The shortest path must be contained, as well as routes that are slightly longer, but with a not too large overlapping part of the route. The researchers from CAMVIT constructed the alternative routes by performing a bi-directional Dijkstra search. By looking for so-called 'plateaus', via-points v that are contained in the shortest paths from both the source s and the destination t, alternative routes can be constructed by combining the paths $s-v$ and $v-t$. But unfortunately, this approach is not fully documented and is not fast enough for continental-sized networks. Therefore Abraham et al. [33] refined the problem, and presented a faster solution, by pruning the Dijkstra search tree. Another way of constructing alternative routes is called the 'Penalty' method. The weights of the edges of the optimal route are increased iteratively, where each iteration a new search for the shortest route can be performed. Both methods deliver good alternatives to the shortest route. The methods have been described and improved further by [34], [35] and [36]. In this research use can be made of the implementation of the plateau method by Peter Karich in the GraphHopper project[1].

7.4. CENTRAL APPROACH: FIND SYSTEM OPTIMUM

The methods that are presented in this research work in a distributed way. The emission map is shared, but all route choices can be made in a distributed fashion. This leads to solutions that are not necessarily a System Optimum. This section describes a system that would find a System Optimum for the same problem. A central decision system makes it possible to find a System Wide Optimum. From a set of short routes for each driver a route can be selected such that a global objective is optimized. The problem is formulated as a minimization problem with constraints.

For every car a route must be chosen of the set of alternative routes that is obtained by the way it is described in section 7.3.1. The decision between these routes is made by formulating the routing problem as a Linear Program (LP) (equation 7.1). The objective is to find the minimal sum of the routes $z \in Z_j$ of all cars J. The binary variable X_{jx} is equal to one if route $z \in Z_j$ is chosen for car $j \in J$, otherwise equal to zero. Exactly one route must be chosen for every car $j \in J$, so this is added as a constraint. The optimal solution is bounded by constraints of the air pollution at predetermined nodes. The contribution to the concentration of NOx by a route z at a certain node N at time t is denoted as $I(z,N,t)$. The maximum allowed concentration of NOx at node N at time t is denoted as $B(N,t)$.

$$\begin{aligned} \text{Minimize } & \sum_{j\in J}\sum_{z\in Z_j} l(z)\times X_{jx} & & \\ \text{Subject to } & \sum_{j\in J}\sum_{z\in Z_j} I(z,N,t)\times X_{jz} & \leq B(N,t) \quad & \forall_{(N,t)} \\ & \sum_{z\in Z_j} X_{jz} & =1 \quad & \forall_j \end{aligned} \tag{7.1}$$

The input to the problem must be gathered in advance, so drivers must request a route some time before they start their journey. The system could be implemented such that it calculates every half an hour a set of routes for the drivers that are present at that timeframe, such that this time period can be kept small. However, even with smaller timeframes this method is still not flexible, so the application of this system in practice will not be very straightforward.

The main drawback of this method is that the routes are computed by one central system. It is very costly to deploy a system that is able to construct an optimal set of routes in very short time periods. Moreover, if this system would malfunction, no routes would be available for the vehicles.

However, this sytem finds the optimal set of routes, which can be compared with the sets of routes of other (more flexible) methods. This system could function as an indicator for the quality of other methods.

[1] https://github.com/graphhopper/graphhopper/

8
Conclusion

The approximation of the emissions of a vehicle [20] is used to construct an emission map of all vehicles together. In this research NOx and CO_2 are chosen to represent vehicle emissions. Based on an emission map, an algorithm and corresponding methods are constructed that find routes that reduce the peak concentrations of air pollutants, while the travel times of individual drivers are kept short. The scientific contribution of this project is described at Section 8.1, followed by Section 8.2, which will summarize how this research relates to the application of a load balancing system in practice.

8.1. Scientific contribution

The results of this method are compared to a case were each driver has the incentive to find the fastest route for himself. Compared to this situation, the local exceedance of the thresholds on NOx concentrations is reduced by 56.3%, the total emission of CO_2 is reduced by 1.8%, while the total travel time is increased by 2.9% for the area of Amsterdam. For the area of Rotterdam the total exceedance of the NOx threshold is even reduced by 43.0%, the total CO_2 is reduced by 2.1%, with the cost of an increase of 2.9% of the total travel time. Therefore, it can be concluded that the collective traffic assignment method of this research can improve the air quality in cities, while low travel times for individual drivers are preserved.

In this research, a congestion-free network is assumed. Therefore, individual minimization of CO_2 emissions leads to a lower total CO_2 emission, in contrast to what Patil observed for his environmental user equilibrium (E-UE) [11]. However, for the balanced case it can be assumed that the total congestion in the network will not be increased, because high local NOx concentrations are avoided. Therefore, the total CO_2 emission can be reduced by this method, even if congestion is present in the network.

Sequential calculation of routes makes it advantageous for drivers to request their route as early as possible. Rescheduling removes that unfair effect, and it improves the air quality with approximately 13%. The execution time of this method scales up linearly with the number of iterations, so this method is not preferred.

The simulation model that is constructed in this research can be used for other objectives as well. For instance, this simulation model can also be used to construct methods that keep noise disturbance below acceptable bounds. However, in most cases it would be essential to include congestion in the model, because nowadays congestion is one of the most crucial problems in the field of traffic assignment.

8.2. Application of traffic load balancing in practice

This research has shown that the emission reductions scale up with the number of participants. So, even with lower number of participants, the air quality can be improved by a smart traffic load balancing method.

The research on this subject is not yet far enough to be applied directly in practice, but this research has shown that combining global optimization criteria as travel time and CO_2 emission with local optimization criteria, such as NOx concentrations, can lead to substantial improvements of the air quality.

The constructed simulation model in this research visualizes the impact of traffic distributions on the air quality locally. This simulation model could therefore serve as a basis for research to find solutions for local problems, like air quality and noise disturbance.

BIBLIOGRAPHY

[1] Solomon, S., D. Qin, M. Manning, Z. Chen, M. Marquis, K.B. Averyt, M. Tignor and H.L. Miller (eds.), *Contribution of Working Group I to the Fourth Assessment Report of the Intergovernmental Panel on Climate Change,* https://www.ipcc.ch/publications_and_data/ar4/wg1/en/contents.html (2007).

[2] B. Gurjar, T. Butler, M. Lawrence, and J. Lelieveld, *Evaluation of emissions and air quality in megacities,* Atmospheric Environment **42**, 1593 (2008).

[3] N. Künzli, R. Kaiser, S. Medina, M. Studnicka, O. Chanel, P. Filliger, M. Herry, F. H. Jr, V. Puybonnieux-Texier, P. Quénel, J. Schneider, R. Seethaler, J.-C. Vergnaud, and H. Sommer, *Public-health impact of outdoor and traffic-related air pollution: a european assessment,* The Lancet **356**, 795 (2000).

[4] B. Brunekreef and S. T. Holgate, *Air pollution and health,* The lancet **360**, 1233 (2002).

[5] Rijksinstituut voor Volksgezondheid en Milieu (RIVM), *Actuele luchtkwaliteitsindex (LKI) in Nederland,* http://www.atlasleefomgeving.nl/kijken (2016), accessed at 30 december 2016.

[6] Gemeente Rotterdam, *Milieuzone in rotterdam,* http://gezonderelucht.nl/milieuzone (2015).

[7] R. B. Ellison, S. P. Greaves, and D. A. Hensher, *Five years of london's low emission zone: Effects on vehicle fleet composition and air quality,* Transportation Research Part D: Transport and Environment **23**, 25 (2013).

[8] G. Santos, *Urban congestion charging: a comparison between london and singapore,* Transport Reviews **25**, 511 (2005).

[9] A. May, *Road pricing: an international perspective,* Transportation **19**, 313 (1992).

[10] Y. Zhang, J. Lv, and Q. Ying, *Traffic assignment considering air quality,* Transportation Research Part D: Transport and Environment **15**, 497 (2010).

[11] G. R. Patil, *Emission-based static traffic assignment models,* Environmental Modeling & Assessment , 1 (2015).

[12] J. G. Wardrop, *Some theoretical aspects of road traffic research,* Proceedings of the Institution of Civil Engineers **1**, 325 (1952).

[13] H. Aziz and S. V. Ukkusuri, *Integration of environmental objectives in a system optimal dynamic traffic assignment model,* Computer-Aided Civil and Infrastructure Engineering **27**, 494 (2012).

[14] F. Zhang, Y. Chen, B. Goni-Ros, J. Gao, and V. L. Knoop, *Routing strategy including time and carbon dioxide 2 emissions: Effects on network performance 3,* in *Transportation Research Board 95th Annual Meeting,* 16-1997 (2016).

[15] D. Alves, J. Van Ast, Z. Cong, B. De Schutter, and R. Babuška, *Ant colony optimization for traffic dispersion routing,* in *Intelligent Transportation Systems (ITSC), 2010 13th International IEEE Conference on* (IEEE, 2010) pp. 683–688.

[16] J. Dallmeyer, R. Schumann, A. D. Lattner, and I. J. Timm, *Don't go with the ant flow: Ant-inspired traffic routing in urban environments,* Journal of Intelligent Transportation Systems **19**, 78 (2015).

[17] R. Zhang, Z. Li, C. Feng, and S. Jiang, *Traffic routing guidance algorithm based on backpressure with a trade-off between user satisfaction and traffic load,* in *Vehicular Technology Conference (VTC Fall), 2012 IEEE* (IEEE, 2012) pp. 1–5.

[18] World Health Organization, *Air quality guidelines: global update 2005: particulate matter, ozone, nitrogen dioxide, and sulfur dioxide* (World Health Organization, 2006).

[19] T. T. P. Nguyen, D. Zhu, and N. P. Le, *Factors influencing waste separation intention of residential households in a developing country: evidence from hanoi, vietnam,* Habitat International **48**, 169 (2015).

[20] J. Hickman, D. Hassel, R. Joumard, Z. Samaras, and S. Sorenson, *Methodology for calculating transport emissions and energy consumption,* (1999).

[21] E. W. Dijkstra, *A note on two problems in connexion with graphs,* Numerische mathematik **1**, 269 (1959).

[22] A. Martelli, *On the complexity of admissible search algorithms,* Artificial Intelligence **8**, 1 (1977).

[23] P. E. Hart, N. J. Nilsson, and B. Raphael, *A formal basis for the heuristic determination of minimum cost paths,* Systems Science and Cybernetics, IEEE Transactions on **4**, 100 (1968).

[24] S. J. Russell and P. Norvig, *Artificial intelligence: a modern approach*, Vol. 2 (Prentice hall Upper Saddle River, 2003).

[25] R. Geisberger, P. Sanders, D. Schultes, and D. Delling, *Contraction hierarchies: Faster and simpler hierarchical routing in road networks,* in *Experimental Algorithms* (Springer, 2008) pp. 319–333.

[26] Centraal Bureau voor de Statistiek (CBS), *Onderzoek verplaatsingen in nederland 2014,* (2015).

[27] Rijksdienst voor Wegverkeer (RDW), *Open Data RDW: Gekentekende_voertuigen_brandstof,* https://opendata.rdw.nl/ (2016).

[28] Centraal Bureau voor de Statistiek (CBS), *Transport en mobiliteit,* http://www.verkeerspro.nl/wp-content/uploads/2015/06/CBS-rapport-Transport-Mobiliteit-2015.pdf (2015).

[29] V. Nallur, A. Elgammal, and S. Clarke, *Smart route planning using open data and participatory sensing,* in *IFIP International Conference on Open Source Systems* (Springer, 2015) pp. 91–100.

[30] N. B. Urquhart, E. Hart, and A. Judson, *Multi-modal employee routing with time windows in an urban environment,* in *Proceedings of the Companion Publication of the 2015 Annual Conference on Genetic and Evolutionary Computation* (ACM, 2015) pp. 1503–1504.

[31] E. Hart, K. Sim, and N. Urquhart, *A real-world employee scheduling and routing application,* in *Proceedings of the Companion Publication of the 2014 Annual Conference on Genetic and Evolutionary Computation* (ACM, 2014) pp. 1239–1242.

[32] Cambridge Vehicle Information Technology Ltd. (CAMVIT), *Choice routing,* http://www.camvit.com (2009).

[33] I. Abraham, D. Delling, A. V. Goldberg, and R. F. Werneck, *Alternative routes in road networks,* in *International Symposium on Experimental Algorithms* (Springer, 2010) pp. 23–34.

[34] R. Bader, J. Dees, R. Geisberger, and P. Sanders, *Alternative route graphs in road networks,* in *Theory and Practice of Algorithms in (Computer) Systems* (Springer, 2011) pp. 21–32.

[35] A. Paraskevopoulos and C. Zaroliagis, *Improved alternative route planning,* in *OASIcs-OpenAccess Series in Informatics*, Vol. 33 (Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2013).

[36] D. Luxen and D. Schieferdecker, *Candidate sets for alternative routes in road networks,* Journal of Experimental Algorithmics (JEA) **19**, 2 (2015).