

Dynamic user equilibria in StreamLine

Research on the existence of dynamic user equilibria in StreamLine

Niels Dijkhuis

28-9-2012

Omnitrans International

&

University of Twente
Civil Engineering & Management

Dynamic user equilibria in StreamLine

Research on the existence of dynamic user equilibria in StreamLine

Master thesis report

N.H. Dijkhuis

Supervisors

Dr. J. Bie
Prof. dr. ir. E.C van Berkum
M. Schilpzand MSc.
Dr. ir. E. de Romph

University of Twente
University of Twente
Omnitrans International
Omnitrans International

28-9-2012

Enschede, the Netherlands



UNIVERSITY OF TWENTE.

Preface

With this master thesis, I finish my study Civil Engineering and Management at the University of Twente. One year ago I started my graduation at Omnitrans International in Deventer. The past 12 months were a valuable experience for me. Owing to this research, I got a good impression of the business world and I learnt a lot about the challenging research.

There are some people that I would like to thank. Firstly, I would like to thank my supervisors at Omnitrans International, Maarten Schilpzand and Erik de Romph, for their supervision and support. I also want to thank my colleagues, for being interested in my research and for being open for my questions. Also, thanks for the good atmosphere; I will always remember the discussions during the lunch break and I enjoyed the hours spent on the pinball machine.

I would like to thank my supervisors at the University of Twente for their supervision and time. Finally, thanks for everyone who have contributed to the realisation of this report, in terms of support and feedback on my research progression.

Enschede, 28-9-2012

Niels Dijkhuis

Summary

StreamLine is a dynamic traffic assignment framework in OmniTRANS, a traffic assignment model that is developed by Omnitrans International. StreamLine tries to determine a dynamic user equilibrium (DUE). In this research a DUE is defined as follows: 'An equilibrium arises when for each route choice moment, for each OD pair, the flow unit costs on the utilised routes are equal to the minimum unit path cost at the corresponding point in time'. The goal of the research is to investigate to which extent StreamLine is able to determine a DUE and to look to which extent this equilibrium changes due to parameter changes in StreamLine. This is done by designing test networks in StreamLine, running some simulations with different conditions, and analysing the results of these simulations.

Whether a DUE can be determined, depends on the method how StreamLine determines route fractions during the simulation. When a PCL assignment is used, it is not possible to determine a deterministic equilibrium, since always a small part of the demand is assigned to each route. In theory, this is possible with a series of all-or-nothing assignments.

However, the usage of an all-or-nothing assignment elicits some problems in MaDAM, the propagation model in StreamLine, that makes it impossible (for the designed networks in the research) to reach a realistic DUE. Unrealistic situations arise during the emptying of the network, when large density differences arise near merge nodes, and when the flow (or demand) on a link changes abruptly. These problems can probably be prevented by doing some adaptations to the propagation model.

Hereafter, some parameters are varied. The length of the route choice interval influences the accuracy of the results. The shorter a route choice interval, the more realistic the development of the route costs will be, but there are more iterations needed before the route costs are converged. In the parameter analysis, it became clear that the parameters τ , κ and ν do influence the equilibrium, but the changes in route costs are quite small. κ and ν influence the equilibrium more than τ . It is possible that the effects are different when a larger and more complex network is used. During the analysis, unrealistic results occurred due to the oscillation effect, which means that the route fraction distribution during the simulation time is not stable, and the exceeding of the jam density on some links.

It is recommended to make some changes to the propagation model. To prevent the observed problems, the anticipation term in MaDAM must be changed, StreamLine must calculate route costs based on average speeds instead of densities, and it must not be possible that the density that is calculated by MaDAM exceeds the jam density.

Content

Preface.....	1
Summary.....	3
1 Introduction.....	9
1.1 Research goal	10
1.2 Research questions.....	10
1.3 Report outline	11
2 Theoretical background	13
2.1 Equilibria.....	13
2.1.1 Types of equilibria	13
2.1.2 Discussion and definition	15
2.2 Glossary	17
2.2.1 General terms from transport studies.....	17
2.2.2 Terminology in StreamLine	17
2.2.3 List of abbreviations.....	19
3 The operation of StreamLine.....	20
3.1 Description of StreamLine.....	20
3.1.1 Route generation.....	21
3.1.2 Route costs.....	22
3.1.3 Route fractions	25
3.1.4 Propagation model	26
3.1.5 Stop criterion.....	26
3.2 The operation of MaDAM	28
3.2.1 General description	28
3.3 Traffic propagation in StreamLine	33
3.3.1 Time loop (orange)	33

3.3.2	Link loop (light blue)	34
3.3.3	Segment loop (dark blue)	35
3.3.4	Node loop (green)	35
4	Methodology.....	36
4.1	Requirements of the network	36
4.2	Description of the networks	37
4.2.1	Network 1	37
4.2.2	Network 2	40
4.3	Method.....	42
4.4	Relative standard error	43
4.5	Parameter analysis.....	46
4.6	Elasticity coefficient	47
5	Encountered problems with PCL and MaDAM	49
5.1	Characteristics of PCL	49
5.2	Limitations of MaDAM	51
5.2.1	Emptying of the network.....	51
5.2.2	Links upstream a merge node	58
5.2.3	Flow changes.....	61
5.3	Adaptations to the network.....	63
5.3.1	All-or-nothing assignments.....	63
5.3.2	Urban links.....	63
5.3.3	Background traffic.....	64
6	Equilibrium run	66
7	Iteration and rci analysis	68
7.1	Number of iterations	68
7.2	Route choice interval	70
7.3	Number of iterations and route choice interval combined.....	73

8	Parameter analysis	76
8.1	Network 1	77
8.1.1	Analysis network 1	78
8.2	Network 2	82
8.2.1	Analysis network 2	83
8.3	Conclusions.....	88
9	Conclusions and recommendations.....	90
9.1	Conclusions of the research	90
9.2	Recommendations.....	92
9.3	Further research	93
	References	94
	Appendix I - Variation in traffic propagation parameters.....	i
	Appendix II - Paired Combinatorial Logit (PCL)	iv
	Appendix III - Cross node modelling	vi

1 Introduction

In transport, people make trips to go from an origin to a destination. A common way to illustrate the trip decision process of people is the 'four step model' (figure 1.1). People make four decisions; whether they make a trip (generation), to which destination (distribution), with which mode of transport (modal split) and, finally, the route of the trip (assignment). Normally, when people choose a route they prefer the route with the lowest generalised route costs.

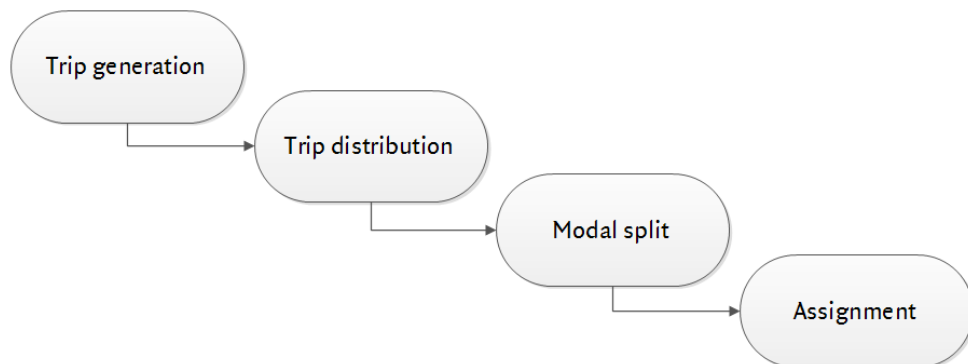


Figure 1.1 - The 'four step model' (Ortúzar & Willumsen, 2001)

For policy makers, it is important that the trips of people in a specific area are mapped and that the traffic distribution in the future can be estimated. For this reason, traffic assignment models were developed in the past decades. OmniTRANS is an example of a traffic assignment model. It calculates how the traffic is distributed in a specific network, given the number of trips between the origins and destinations in the network. OmniTRANS tries to calculate the optimal distribution in the network, which is defined as the situation where the route costs of the used routes in an origin-destination pair (OD pair) are equal. This is called an (user) equilibrium.

Traditional traffic assignment models are static models. In static models, the traffic situation in the network is constant during the whole simulation. In the last few years, dynamic traffic assignment models were developed, which makes it possible to assign traffic over time. As a result, dynamic traffic assignment models can show the consequences of a queue in the network and can handle with capacity constrictions of links in the network. In some models, the departure time of people is variable.

In OmniTRANS, a dynamic traffic assignment framework is built, which is called StreamLine. The goal of StreamLine is to determine a dynamic user equilibrium. However, Omnitrans International, the developer of OmniTRANS and StreamLine, has not researched yet whether a dynamic user equilibrium arises in StreamLine, and to

which extent changes in parameter values and settings in the model influence this equilibrium (the robustness of StreamLine). This research tries to answer these questions.



Figure 1.2 - A screenshot of OmniTRANS. In this network the amount of traffic on the links in the network are displayed.

1.1 Research goal

The goal of the research is to investigate to which extent StreamLine is able to determine a dynamic user equilibrium, to test the robustness of StreamLine, and to advise Omnitrans International for improvements in StreamLine. This is done by designing test networks in StreamLine, running several simulations with different conditions, and analysing the results of these simulations.

1.2 Research questions

This research investigates to which extent several conditions influence whether a dynamic user equilibrium (DUE) arises and to which extent these conditions change the dynamic user equilibrium. There are several conditions that may influence the equilibrium:

- The design of the network, like the flows on the network, the number of routes, the number of OD pairs and the overlap of routes.

- The settings in the route choice model in StreamLine, like the number of route choice moments during the simulation time, the number of iterations and the accuracy.
- Parameters that are used in the traffic propagation model in StreamLine, which can change the user equilibrium.

To meet the research goal, several questions are formulated:

Main question:

Under which conditions can StreamLine determine a dynamic user equilibrium and to which extent changes the dynamic user equilibrium due to changes in these conditions?

To answer this question, some sub-questions are formulated:

1. Which definition of a dynamic user equilibrium (DUE) is used in the research? (Chapter 2)
2. How does StreamLine try to determine a dynamic user equilibrium? (Chapter 3)
3. Does StreamLine determine a DUE when the default settings for the route choice and traffic propagation parameters in the model are applied? (Chapter 5-6)
4. To which extent is a DUE sensitive for changes in these parameters (i.e. does the model always determine the same DUE, regardless of changing values of these parameters)? (Chapter 7-8)

1.3 Report outline

The report begins with a short theoretical background. This chapter (2) gives an overview of the types of equilibria that are available and it explains which definition of an equilibrium is used in the research. Chapter 2 also includes a glossary, which gives an enumeration of the definitions of the most important terms that are used in the report. Subsequently, it is explained in chapter 3 how StreamLine works. All steps that are done in StreamLine and that are relevant for this research are explained. The end of this chapter focuses on the operation of MaDAM, the propagation model in StreamLine.

Chapter 4 describes the methodology that is used in the research. This chapter also gives a detailed description of the two networks that are used in the research. Hereafter, chapter 5 explains why it is not always possible to reach an equilibrium. The limitations of PCL (i.e. a route choice model in StreamLine) and MaDAM are discussed in this chapter. To reach an equilibrium, some adaptations to the designed networks were required,

which are also described in this chapter. Subsequently, chapter 6 shows the results of the equilibrium run.

Hereafter, it is tested to which extent the equilibrium changes due to parameter changes. Chapter 7 shows the results when the number of iterations and the number of route choice moments in the simulation are varied. In chapter 8, several propagation parameters are changed and this chapter describes the results of this test.

Finally, all conclusions of the research are shown in chapter 9. This chapter also gives recommendations for Omnitrans International and tips for further research.

2 Theoretical background

This chapter gives an overview of the most significant theoretical background of this research. Section 2.1 mentions different types of equilibria and explains which formulation of an equilibrium is used in the research, followed by a short discussion. Section 2.2 is a glossary; it gives an overview of general terms from transport studies and terminology in StreamLine. This section ends with a list of abbreviations that are used in the research.

2.1 Equilibria

This research investigates to which extent a dynamic user equilibrium arises in StreamLine. There are different types of equilibria available and there are also different definitions of equilibria available in literature. For the research, it is necessary that a clear definition of an equilibrium is formulated.

Section 2.1.1 illustrates into which main categories equilibria can be subdivided. Subsequently, section 2.1.2 gives a formulation of the equilibrium that is used in this research.

2.1.1 Types of equilibria

User equilibrium vs. system optimum

When people make a trip, they have to choose a specific route. A user will normally choose the route that is most beneficial for him. In other words, a user chooses the route with the lowest generalised cost. When all users choose the cheapest route individually and no user in the network can reduce his costs by unilaterally changing routes, a **user equilibrium** is reached.

In contrast, a **system optimum** is reached when the total travel costs in the network are minimised. In some cases, a system optimum is different from a user equilibrium. For example, when people choose the route with the lowest cost, it is possible that congestion arises somewhere in the network. As a consequence, other users with a different origin or destination may have increased route costs due to this congestion.

When a system optimum is different from a user equilibrium, it is not probable that a system optimum will be reached spontaneously. However, it would prove very helpful to policy makers when they try to reduce congestion or minimise emissions. In this case, policy makers can choose to force a system optimum by, for example, using DTM-

measures¹. Strictly speaking, a system optimum is *not* an equilibrium, since a system optimum can only be reached in a forced situation.

In 1952, Wardrop was the first who made a distinction between these equilibria. He used the following definitions (Wardrop, 1952):

- *Wardrop's first principle*; When an equilibrium is reached, the travel costs on all routes that are used are equal, for each origin-destination pair, and the travel costs are less than the travel costs on the routes that are not used (user equilibrium).
- *Wardrop's second principle*; When an equilibrium is reached, the total travel costs cannot be reduced by assigning alternative routes to users (system optimum).

Deterministic vs. stochastic equilibrium

The definition of a user equilibrium in the previous sub-section is an example of a **deterministic** equilibrium. A deterministic equilibrium is a theoretically optimal solution, where uncertainties are not taken into account during the calculation of the equilibrium.

On the other hand, there is the **stochastic** equilibrium. In this situation, equilibrium is reached when no traveller *believes* that his travel time can be improved by unilaterally changing routes (Sheffi, 1985). At a stochastic equilibrium, the perceived travel costs have to be equal on all routes that are used in the network, not the 'real' cost.

An advantage of this approach is that it takes several disadvantages of Wardrop's user-equilibrium into consideration. Wardrop assumes that motorists have full information about the travel times on every possible route, that they always make the right decisions and that the behaviour of all individuals is identical. In reality, this is not the case.

Static vs. dynamic equilibrium

Wardrop's formulation of an equilibrium (section 3.1.1) is an example of a static user equilibrium. At a **static** equilibrium, the assumption is made that the amount of traffic on a network is constant over time. The traffic situation in the network is constant during the whole simulation. In contrast, a **dynamic** traffic assigning model can handle traffic variations in time. The route choice distribution depends on the busyness on the network and the traffic that was assigned earlier in the simulation. As a result, the traffic situation on the network can be different on several moments of time during the simulation.

¹ DTM stands for Dynamic Traffic Management, which is a set of automatic real-time measures, to regulate the traffic flow. Examples of DTM are traffic signalling, dynamic route information panels (DRIPs) and ramp metering.

Besides, dynamic traffic assigning models can deal with capacity constraints in the network, while static models do not. There are several definitions of dynamic equilibria available in literature, the most important definitions are mentioned below.

The *Boston Traffic Equilibrium* is a straightforward translation of Wardrop's equilibrium. It assumes that travellers know about the traffic situation on the network over a period of time. Travellers choose, besides the route they will take, also the moment they will depart. An equilibrium arises when for each instant of time and for each OD pair, the flow unit costs on utilised paths are equal to the minimum instantaneous unit path cost (Friesz, Bernstein, Smith, Tobin, & Wie, 1993). In some studies, this optimum is also known as the *Dynamic User Optimal assignment* (Ran, Boyce, & Le Blanc, 1993).

A disadvantage of this definition is that it assumes that travellers do not matter about their departure time. In reality this is not true, since for example most people do not like to get up early or arrive at work too late. The *Path integral equilibrium* deals with this problem, by adding penalties to the cost function when people does not depart at the desired time (Friesz, Bernstein, Smith, Tobin, & Wie, 1993).

Bliemer (2001) developed a model where multiple user classes are considered. He formulated the following definition of a dynamic user equilibrium: "For each user-class and for each origin-destination pair, the route travel costs for all users travelling between a specific OD pair and departing at the same time instant are equal, and less than (or equal to) the route travel costs which would be experienced by a single user on any unused feasible route for that user-class" (Bliemer, 2001).

2.1.2 Discussion and definition

The definition of a dynamic user equilibrium that is used in the research is something different from the definitions that are mentioned above. A literal application of the Boston Traffic Equilibrium is not suitable for this research, since departure time plays an important role in this definition. In this research, the travel demand is fixed for a specific time interval, to reduce complexity. This means that people are not allowed to depart earlier or later. As a result, the path integral equilibrium is neither suitable. An advantage of the definition of Bliemer is that departure time is ignored and that for each moment in time a user equilibrium is calculated. In this research, Bliemer's formulation can be broadly applied, but some adjustments need to be made. Firstly, Bliemer requires that an equilibrium is achieved at any moment during the simulation time. To make the research less complex, an equilibrium must be reached only at route choice moments. It is also not needed to involve user-classes in the research.

The following definition of a user equilibrium is used in the research:

An equilibrium arises when for each route choice moment, for each OD pair, the flow unit costs on the utilised routes are equal to the minimum flow unit costs at the corresponding point in time.

This means that at any route choice moment, people cannot improve their situation by taking another route. This definition sounds like the Wardrop's static user equilibrium definition, but the difference is that the route choice is based on the instantaneous flows on the links in the network. Since the demand is time-dependent (i.e. the demand is different for each time step, but fixed²), the flows on the links in the network in the previous step influence the route choice of people.

It is questionable whether a DUE will arise in reality, since the definition assumes that travellers are informed about the conditions on the road, so that they can make well-considered travel choices. In reality, this does not have to be the case, since travellers are sometimes not familiar with the network or travellers do not always know the flows on other roads in the network. However, when travellers make a trip frequently they learn over time. When keeping this in mind, the traffic assignment comes closer to a 'real' equilibrium but it will probably never reach it, since this does not hold true for all travellers. However, travellers who make repeated daily trips, do not always choose the same route every day, even when the situation on the network does not change over time (Horowitz, 1984). So, even if you are an experienced traveller, it is impossible to predict the precise flows in the network, since flows differ from day to day.

Besides, when travellers make a route decision, they do not know what the traffic situation will be in the future. For example, when a traveller makes a trip from Enschede to Amsterdam, he chooses a route via Deventer. However, there are also travellers that depart from Deventer and travel to Amsterdam. At the moment the traveller departs, he does not know how many travellers depart at Deventer when he arrives at Deventer. When keeping this in mind, it is questionable whether a dynamic user equilibrium exists at all.

In other words, StreamLine determines a DUE from a theoretical point of view, while this does not have to be the case in reality. Users of StreamLine should realise that StreamLine does not predict the traffic flows in the network, but tries to estimate them.

² When the demand is 'fixed', the demand does not change during the simulation.

2.2 Glossary

In the research, several terms are used. This chapter gives an overview of the most important terms. Section 2.2.1 shows general terms from transport studies, section 2.2.2 gives an overview of the terminology in StreamLine and, finally, section 2.2.3 gives a list with abbreviations.

2.2.1 General terms from transport studies

Term	Explanation
Demand	The total number of vehicles that travels from a specific origin to a specific destination in a specific time period. Expressed in vehicles per hour.
Density	The number of vehicles that are present on a road with a specific length. Expressed in vehicles per kilometre.
Destination	The location where people end their trip.
Flow	The number of vehicles that drive on a road in a specific time period. The flow is mostly expressed in vehicles per hour. In StreamLine, the term 'load' is used.
Origin	The location where people start their trip.
Origin-destination matrix (OD matrix)	An overview of the number of trips that are made in the network between all origins and destinations.
Origin-destination pair (OD pair)	All set of a specific origin and a specific destination.
Speed	The travelled distance a vehicle covers in a specific time period. In this research the speed is expressed in kilometres per hour.

Table 2.1- Overview of general terms from transport studies.

2.2.2 Terminology in StreamLine

The terminology in StreamLine can be split in two groups; terms that tell something about the visual part of StreamLine (the physical terms concerning the network), and terms concerning the simulation process in StreamLine.

Term	Explanation
Centroid	A centroid is a location inside a zone. In most cases, it is the centre of a zone. All traffic that departs from a specific zone is bundled and departs from the centroid of that zone. Similarly, all traffic that goes to a specific zone arrives in the same centroid.
Connector link	A connector (link) is a link type that connects a centroid with a node in the network.
Destination (node)	A destination node is a centroid with arriving traffic. A centroid can be both an origin node and a destination node.
Entry link	A link (or segment) upstream a node.

Exit link	A link (or segment) downstream a node.
Junction	A node where more than two links come together.
Link	Two nodes can be connected with each other by links. A link represents a road; this can be a motorway or a 30-km road for example.
Link segment	A link is divided in several link segments. The default length of a segment is 300 metres.
Merge (node)	A merge (or merge node) is a node with more than one incoming link and one outgoing link. In this case multiple links merge to one link. An example of a merge node is an on-ramp on a motorway.
Network	A network is the collection of all zones, centroids, nodes and links.
Node	A node makes it possible to connect two or more links with each other.
Origin (node)	An origin node is a centroid with departing traffic. A centroid can be both an origin node and a destination node.
Zone	In StreamLine, the study area is divided in several areas. Such an area is called a zone.

Table 2.2 - Overview of terms concerning the visual part of StreamLine.

There are also some other terms that are used in the research that need some explanation.

Term	Explanation
All-or-nothing assignment	Assignment method where all traffic is assigned to the route with the lowest costs. A series of all-or-nothing assignments can be used to determine a deterministic equilibrium.
MaDAM	The propagation model in StreamLine. MaDAM calculates the speed, the density and the flow for each link segment.
Paired Combinatorial Logit (PCL)	Assignment method, which is used to determine a stochastic equilibrium. This method is explained in appendix II.
Route choice interval (rci)	The period between two route choice moments.
Route choice moment (rcm)	A moment in the simulation time where StreamLine calculates a new route distribution.
Route costs	The costs of a route. In StreamLine, the route costs equal the travel time of a route.
Route fraction	The percentage of the demand of an origin-destination pair that is assigned to a specific route
Simulation time	The period of time that is simulated. In this research, the simulation time is 2 hours (for example from 8:00 AM till 10:00 AM).

Table 2.3 - Overview of terms concerning the simulation in StreamLine.

2.2.3 List of abbreviations

Abbreviation	Explanation
DUE	dynamic user equilibrium
h	hours
km	kilometres
OD matrix	origin-destination matrix
OD pair	origin-destination pair
PCL	paired combinatorial logit
rci	route choice interval
rcm	route choice moment
rse	relative standard error
s	seconds
veh	vehicles

Table 2.4 - List of abbreviations that are used in the research.

3 The operation of StreamLine

This chapter describes the operation of StreamLine. Section 3.1 gives a description of the steps Streamline follows during a simulation. Subsequently, section 3.2 describes the operation of MaDAM, the propagation model in StreamLine. In section 3.3, an overview is given of the propagation process in StreamLine.

3.1 Description of StreamLine

This section explains which steps StreamLine follows to determine a dynamic user equilibrium. Firstly, it is explained which components StreamLine consists of, hereafter the steps that are taken at each component are written down.

Broadly speaking, the operation of StreamLine is shown in figure 3.1 and is split up in the following steps.

1. The generation of routes. For each OD pair the most attractive routes are selected. StreamLine uses these routes during the whole simulation.
2. The calculation of the route costs of the route alternatives. In the first iteration the instantaneous travel times are calculated, which are the free flow travel times.
3. The calculation of the route fractions. StreamLine compares the route costs of the route alternatives that are calculated in step 2. Based on this comparison, StreamLine calculates at each route choice moment the percentage of traffic that chooses a specific route.
4. The propagation model. In StreamLine, the propagation model is called MaDAM. MaDAM calculates to which extent the traffic distribution – which is calculated in step 3 – influences the characteristics of each link segment (i.e. the speed, flow and density on a link segment).
5. The calculation of the route costs, based on the new speeds, intensities and densities.
6. Convergence criterion. The calculated route costs are compared with the previous iteration. If the traffic distribution is not equal to the previous iteration, a new iteration is started and steps 3-6 are repeated. This is called the iteration loop. Otherwise, the simulation will stop and StreamLine has reached the optimal traffic distribution.

In the next few sub-sections, the components of StreamLine (which are shown in the figure above) are explained in more detail.

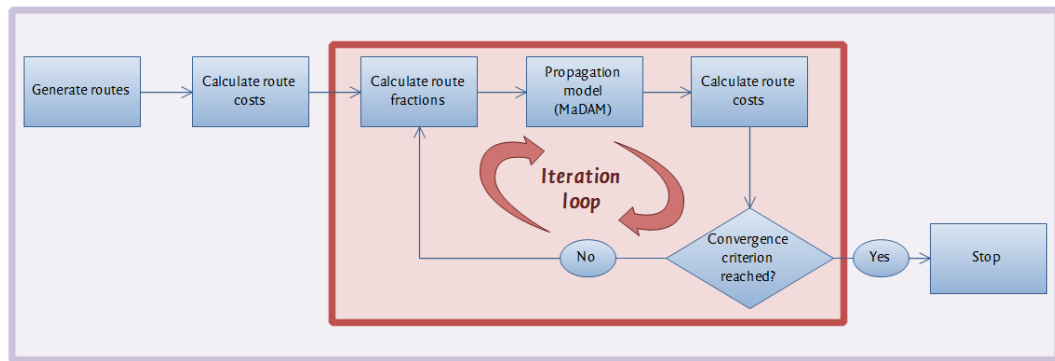


Figure 3.1 - Overview of the operation of StreamLine.

3.1.1 Route generation

In StreamLine, a route set is generated at the beginning of the simulation. This route set is used in the entire simulation. The advantage of this method is that the route set only has to be calculated once, which saves a lot of computation time, especially in large networks with many route alternatives and many OD pairs. However, it is possible that routes that become more important during the simulation are ignored, since it is possible that routes become relatively more attractive when there is much traffic on other routes. When these routes are not selected during the route generation phase, these routes are left aside in the rest of the simulation, which can influence the traffic distribution in the network.

In figure 3.2, a flow chart of the route generation in StreamLine is displayed. Firstly, the shortest route of each OD pair is determined by using the Dijkstra algorithm³. Subsequently, alternative routes are generated. To realise this, StreamLine uses a Monte Carlo algorithm. The costs on all links are randomised, after which the shortest route will be determined again. When a route is not included in the current route set, the new route will be added. When no new routes are found, the variance is increased, which has the consequence that the costs on the links are varied more and more expensive routes are more likely to be selected. This loop repeats until the maximum variance or the maximum number of iterations is reached. These parameters can be set by the user. Finally, routes are filtered from the route set. All generated routes are compared with the initial route. For instance, when the generated route has much overlap with the initial route, or the cost difference between the two routes is too high, the route will be removed. Other filter criteria are mentioned in figure 3.2. When the maximum number of suitable routes is

³ The Dijkstra Algorithm is a graph-search algorithm that determines the shortest path in a network from an origin to a destination.

reached, the filter process stops. The values of these parameters can be set by the user and can have a large influence on the route selection.

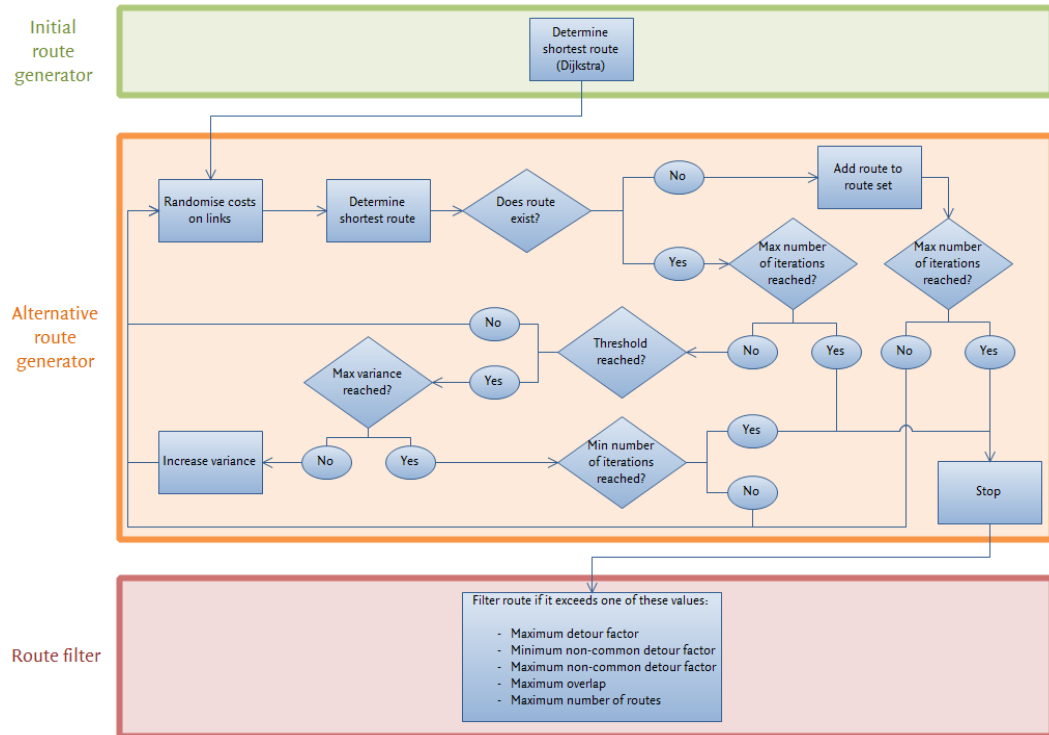


Figure 3.2 - Overview of the route generation process in StreamLine.

3.1.2 Route costs

In a traffic network, the traffic situation in the network changes over time. In reality, the departure pattern of people is continuous. In most dynamic assignment models, it is too complex to model a continuous departure pattern, since the travel demand may change every minute. For this reason, StreamLine models the departure pattern in the discrete way. During a simulation, there are several moments set where people choose their route, the so-called route choice moments. At these moments, StreamLine calculates the route fractions of all route alternatives of all OD pairs. This distribution does not change until the next route choice moment is reached.

To calculate the route fractions of a specific route, the route costs of this route have to be determined. There are two ways in which these route costs can be calculated: the reactive and the predictive approach.

Reactive route costs

In the reactive approach, the current situation on the network is used to determine the route costs, also called *instantaneous travel times*. The travel times are estimated by the

average speeds on the links at that moment in time. These speeds are calculated by MaDAM.

The reactive method is a static approach, since the calculation is based on one single moment within the simulation. Therefore, the results are not very realistic. This method is generally used in the first iteration of the simulation, since no data of a previous iteration is available then.

Predictive route costs

Based on the traffic that is already on the network, StreamLine predicts what the travel time of a route will be. This travel time is called the *trajectory based travel time*. Trajectory based travel times are far more accurate than instantaneous travel times. However, it is more time-expensive to compute these values.

In StreamLine, there are two methods to calculate predictive route costs; the method based on cumulative vehicles and the method based on average link speeds. Currently, the first method is implemented in StreamLine for traffic propagation; for the latter method, a tool is developed that is only available for the user to visualise route costs. This tool is called the TravelTimeCalculator. This method is not used in the algorithm.

Based on cumulative vehicles

In StreamLine, the route cost calculation is based on cumulative vehicles, which means that StreamLine tries to measure the precise travel time of a single vehicle. Since StreamLine is a macroscopic model, it is not possible to simulate single vehicles. For this reason, StreamLine determines route costs by using densities. In each iteration, StreamLine calculates the density of each link segment; which corresponds with the density ρ that is calculated by MaDAM (see section 3.2.2). When the densities of the different time periods are summed, the number of cumulative vehicles that have passed in the simulation is calculated.

To calculate the trajectory based travel time of a specific route, StreamLine determines for each link how long it takes to pass that link. Example 3.1 illustrates how StreamLine determines this travel time.

Example 3.1 – Cumulative vehicles

In this example, one link is considered. The route costs on this link are determined by using cumulative vehicles. Figure 3.3 shows a graph that displays the number of cumulative vehicles as a function of time. There are two curves; one that displays the situation at the end of the link and one at the beginning of the link (which is equal to the curve at the end of the previous link). The difference between these curves is the route cost of this link.

For example, StreamLine wants to determine the route costs of link 1 at $t = 1$ minute. In figure 3.3 the number of cumulative vehicles that has entered link 1 during the simulation can be interpolated. In the graph it is shown that at $t = 3$ minutes the cumulative vehicle numbers of the two curves are equal. Consequently, the route cost of link 1 is $3 - 1 = 2$ minutes.

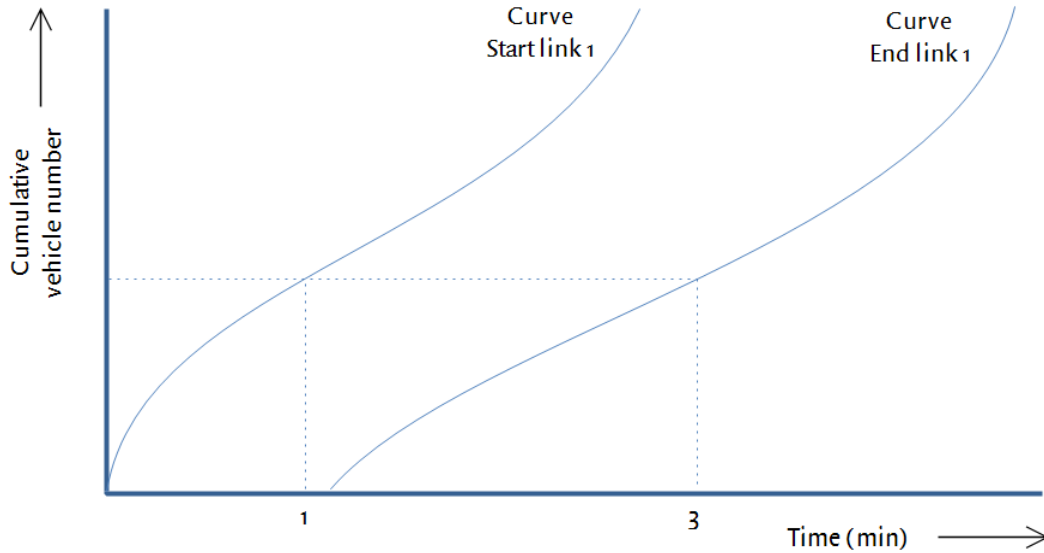


Figure 3.3 – Example of a graph that displays cumulative vehicle numbers as a function of time.

Based on average link speeds

Another way to determine route costs, is to use the average link speeds that are calculated by MaDAM. As mentioned before, this is the method that is implemented in the TravelTimeCalculator; a tool in StreamLine that makes it possible to easily check the route costs of specific routes during a simulation. Example 3.2 gives an example of this method.

Example 3.2 – Average link speeds

In a network there are two links. Link 1 has a length of 500 metres and link 2 has a length of 1 kilometre. A vehicle departs at $t=0$ and at this moment the speeds on link 1 and 2, which are calculated by MaDAM, are respectively 30 and 35 km/h. It takes $0.5/30 \times 3600 = 60$ seconds to pass link 1. At $t=60$ the vehicle enters the second link. Here, the average speeds of link 1 and 2 at $t=60$ are 32 and 40 km/h. The route costs of link 2 are $1/40 \times 3600 = 90$ seconds. The total route costs for the trip are $60 + 90 = 150$ seconds.

Comparing route costs

At each route choice moment, StreamLine compares the route costs of the previous iteration with each other (in case of the predictive approach). StreamLine uses the route

costs belonging to the traffic situation at the middle of the current route choice interval. This is presented with the blue arrows in figure 3.4.

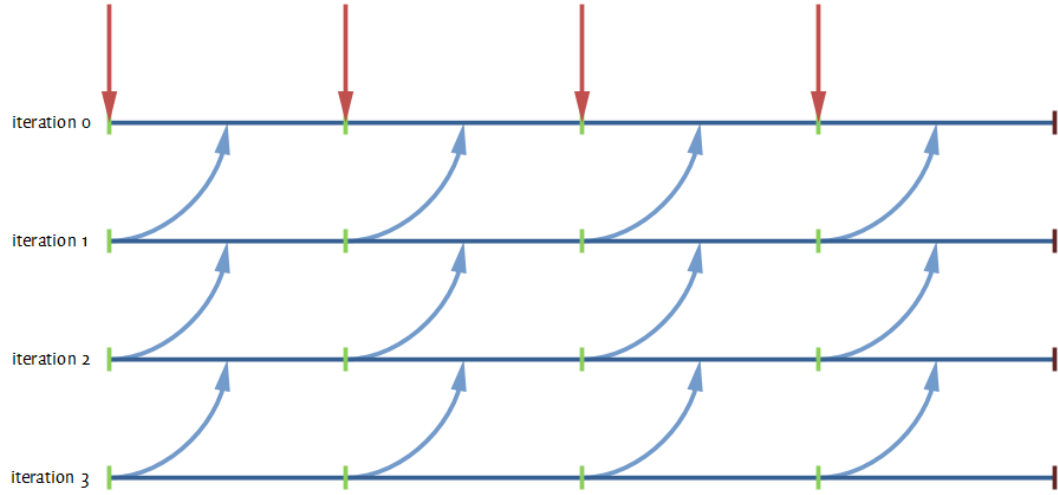


Figure 3.4 - Schematic view of which route costs StreamLine uses at route choice moments. The red arrows represent the reactive approach, the blue arrows the predictive approach. The green lines are the route choice moments.

In the first iteration of the simulation StreamLine uses the reactive approach. The route costs of the traffic that departs at the beginning of each route choice interval in the first iteration are compared with each other. This is displayed in figure 3.4 with red arrows.

3.1.3 Route fractions

When the route costs are determined at every route choice moment, the demand is divided among the route alternatives. There are three methods to do this in StreamLine:

- *Uniform distribution.* All route alternatives of a specific OD pair receive the same proportion of traffic, regardless of the calculated route costs of these routes.
- *All-or-nothing distribution.* The route alternative with the lowest route costs receives all traffic, other route alternatives will receive nothing.
- *Paired Combinatorial Logit.* All route alternatives receive a proportion of traffic, but routes with low route costs receive a relatively higher proportion of traffic than routes with high route costs. The Paired Combinatorial Logit is explained in more detail in appendix II.

Hereafter, the route fractions that are calculated in the current iteration are averaged with the fractions in the previous iterations. The following formula is used.

$$F_{i,r} = \frac{i-1}{i} \cdot F_{i-1,r} + \frac{1}{i} \cdot f_{i,r} \quad (3.1)$$

Variable	Explanation
F	total route fraction, weighted with previous iterations
f	unweighted route fraction of the current iteration
i	iteration
r	route

Table 3.1 - Parameters used to calculate route fractions in StreamLine.

3.1.4 Propagation model

The propagation model of StreamLine is called MaDAM. For every link segment, the speed, flow and density are calculated. This is done in every time step of the simulation, which has a default value of 1 second. Section 3.2 shows a more detailed explanation of MaDAM.

In StreamLine, several variables that influence traffic propagation can be set by the user, like the minimum speed. The user can also enter whether links are urban or rural roads. The variables that are relevant for this research are mentioned in section 4.5.

3.1.5 Stop criterion

StreamLine stops the simulation when a stop criterion is reached. StreamLine applies two stop criteria: the duality gap and the maximum number of iterations.

Duality gap

The duality gap is the extent to which the route costs in the network change in comparison with the previous iteration. The user can set the duality gap.

For each OD pair and each route choice moment, the costs of a specific route are compared with the route with the minimum cost of that OD pair and multiplied with the flow on that route. The values of all OD pairs in the network are summed and are divided by the sum of the minimum cost of an OD pair times the total demand on that OD pair, for all OD pairs and route choice moments.

$$dg_{rcm} = \frac{\sum_{od,rcm,r} q_{od,rcm,r} \cdot (c_{od,rcm,r} - c_{od,rcm}^{\min})}{\sum_{od,rcm} c_{od,rcm}^{\min} \cdot q_{od,rcm}} \quad (3.2)$$

After each iteration, for each route choice moment, the duality gap is compared with the duality gap of the corresponding route choice moment in the previous iteration. When the largest difference of duality gaps (dg_{max}) is lower than the threshold which is set by the user, the simulation will stop.

$$dg_{\max,i} = \max_{rcm} (dg_{rcm,i} - dg_{rcm,i-1}) \quad (3.3)$$

Variable	Explanation
dg	duality gap
dg_{\max}	largest difference of duality gaps
od	OD pair
rcm	route choice moment
r	route
q	flow [veh]
c	route costs [h]
c_{\min}	minimal route costs [h]
i	iteration

Table 3.2 – Parameters used to calculate the duality gap

When StreamLine stops the simulation because of the duality gap, it means that the outcome of StreamLine is close the optimal solution. However, it is not necessary that an equilibrium is reached. The simulation stops when the route costs of a route do not change any more, regardless of whether the route costs of the used route alternatives are equal. In case of a PCL assignment, this situation will occur, see also appendix II. This approach is not correct when StreamLine tries to determine a DUE.

Maximum number of iterations

When a DUE is not reached, the simulation stops when the maximum number of iterations is reached. The user can set this variable. This option is advantageous when large networks are simulated and it takes a long time before a few iterations are finished. A maximum number of iterations saves time. However, the results are less accurate, since the results have not been fully converged to an equilibrium.

3.2 The operation of MaDAM

As mentioned in section 3.1, the macroscopic traffic propagation model StreamLine uses is called MaDAM. The model is based on the METANET model by Papageorgiou and Messmer (Technical University of Crete & Messmer, 1990).

Firstly, a general description is given of MaDAM, followed by a more detailed description with relationships and formulas.

3.2.1 General description

MaDAM is a deterministic macroscopic modelling tool for simulating traffic flows in networks. It can cope with several traffic conditions – like free, dense and congested conditions – and the usage of control actions like ramp metering, route guidance and MTM control (motorway traffic management) is possible.

MaDAM is a *cell based model*, which means that a link is divided in several segments with equal lengths. Each segment holds information about traffic variables like speed, density and flow. The value of these variables is the same along the whole length of each segment. Naturally, a vehicle moves from one segment to another. This idea is shown in figure 3.5. In this figure, N_m is the number of link segments m consists of.

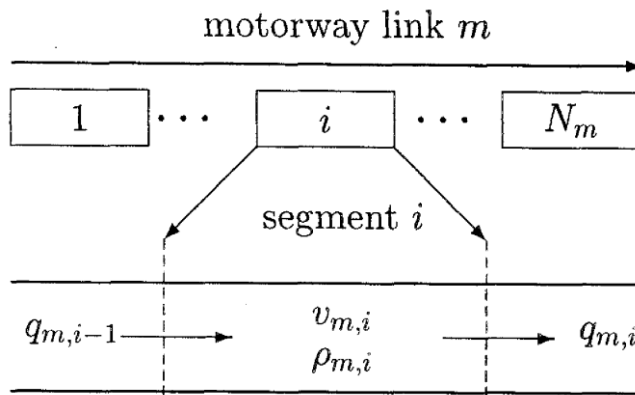


Figure 3.5 - Discretised motorway link (Papageorgiou, et al., 2002)

This figure mentions several variables that are used in the model:

- **Traffic density:** $\rho_{m,i}(t)$ [vehicles/km/lane]. The traffic density is the number of vehicles in segment i of link m at time t , divided by the length of the segment L_m and the number of lanes of link m (λ_m).
- **Mean speed:** $v_{m,i}(t)$ [kilometres/hour]. This is the average speed on a segment i on link m at time t .
- **Traffic flow:** $q_{m,i}(t)$ [vehicles/hour]. The traffic flow is the number of vehicles that leave segment i of link m during the time between time t and $t+1$.

The default length of a link segment is 300 metres, but this length can be changed by the user. However, it is possible that the length of a link is not a multiple of the length of a link segment (e.g. 800 metres and 300 metres). In this case, the last two segments of the link are put together. So, when the length of the link is 800 metres, StreamLine splits up the link into two links of respectively 300 and 500 metres (instead of three links of respectively 300, 300 and 200 metres).

Figure 3.5 shows that in segment i , these variables are based on the traffic flow on the subsequent and preliminary segments. This means that the model reckons with the upstream and downstream traffic, which is called a *second order model*.

By contrast, in a *link based model*, the model uses the characteristics of a whole link to determine the speed, flow and density on that link. Since the length of cells is shorter than the length of a whole link, the results of a cell based model are more accurate than the results of a link based model.

Originally, METANET consists of different models for different links. METANET distinguishes five types of links, which have all different characteristics: *motorway links*, used for homogeneous motorway stretches; *origin* and *destination links*, *store-and-forward links* and *dummy links* (which have zero length, but are used to make some complex nodes more orderly). However, StreamLine only uses the METANET model for motorway links, since StreamLine only uses the macroscopic part of METANET. The model is explained below.

3.2.2 Main formulas

The most important traffic variables are calculated with the equations below.

$$\rho_{m,i}(t+1) = \rho_{m,i}(t) + \frac{T}{L_m \cdot \lambda_m} [q_{m,i-1}(t) - q_{m,i}(t)] \quad (3.4)$$

In this formula, the density of segment i on link m is calculated for the next time step ($t+1$). The difference in flow between two successive segments determines the increase of the density in the next time step (which can also be negative). This flow difference is corrected by multiplying it with a factor, containing the time step size T , the length of a segment L_m and the number of lanes λ_m . Finally, this increase is added to the density of the current time step.

$$q_{m,i}(t) = \rho_{m,i}(t) \cdot v_{m,i}(t) \cdot \lambda_m \quad (3.5)$$

Subsequently, the traffic flow of segment i on link m on time t is calculated by the basic equation above. The density is multiplied by the speed on that link segment and is corrected by multiplying it with the number of lanes on that link.

$$v_{m,i}(t+1) = v_{m,i}(t) + \text{relaxation} + \text{convection} - \text{anticipation} \quad (3.6)$$

$$\text{relaxation} = \frac{T}{\tau} \cdot \left(V(\rho_{m,i}) - v_{m,i}(t) \right) \quad (3.6a)$$

$$\text{convection} = \frac{T}{L_m} \cdot v_{m,i}(t) \cdot \left(v_{m,i-1}(t) - v_{m,i}(t) \right) \quad (3.6b)$$

$$\text{anticipation} = \frac{v \cdot T \cdot \left(\rho_{m,i+1}(t) - \rho_{m,i}(t) \right)}{\tau \cdot L_m \cdot (\rho_{m,i}(t) + \kappa)} \quad (3.6c)$$

The formula for the speed on segment i on link m for the next time step $t+1$ (formula 3.6) is far more complex. Papageorgiou and Messmer state that there are three aspects that influence the speed, which are mentioned below:

- *Relaxation term.* This term describes how the vehicles adapt their speed according to the fundamental diagram (V). Here, the density of the link segment at that time is the input of the fundamental diagram, which corresponds with a specific speed. The difference with the current speed at time t is added to the current speed.
- *Convection term.* The convection term describes how vehicles change speed due to departure and arrival of vehicles. The difference between the average speed on the current and the previous link segment is multiplied by a constant, including the time step size T and the length of the link L_m .
- *Anticipation term.* This term describes to which extent car drivers anticipate on concentration conditions downstream the road. The difference between the density on the current and the next link segment is derived by the density on the current link in the upcoming time step, added by a constant (κ). This change is multiplied by a constant, including the time step size T and the length of the link L_m and two constants (v and τ).

The parameters that are used in the formulas above are summarized in table 3.3.

Parameter	Explanation
i	link segment
m	link
t	time [s]
T	time step size [s]
L_m	length of the link [km]
λ_m	number of lanes of the link
$V(\rho)$	speed according to the fundamental diagram function, with density as input parameter [km/h]
τ	time constant [s]
κ	control variable [veh/km]
v	anticipation constant [km ² /h]

Table 3.3 - Parameters used in MaDAM

3.2.3 Fundamental diagram

As mentioned earlier, to calculate the speed of the vehicles, a *fundamental diagram* is needed. MaDAM uses the so-called *Integration model* by Van Aerde. The model has four input parameters: the roadway free flow speed, the speed-at-capacity, the capacity and the jam density. These input parameters are fixed and are set for every link in the network before the simulation starts. The equations are given below (Rakha & Crowther, 2002).

$$c_1 + c_3 \cdot v + \frac{c_2}{v_f - v} - \frac{1}{\rho} = 0 \quad (3.7)$$

$$c_1 = m \cdot c_2 \quad (3.7a)$$

$$c_2 = \frac{1}{\rho_{cr} \left(m + \frac{1}{v_f} \right)} \quad (3.7b)$$

$$c_3 = \frac{-c_1 + \frac{v_{cr}}{q_{cr}} - \frac{c_2}{v_f - v_{cr}}}{v_{cr}} \quad (3.7c)$$

$$m = \frac{2v_{cr} - v_f}{(v_f - v_{cr})^2} \quad (3.7d)$$

The first two parameters (c_1 and c_2) provide the linear increase in the vehicle speed as a function of the distance headway, while the third parameter (c_3) introduces curvature to the model and ensures that the vehicle speed does not exceed the free flow speed. In this

model, the speed v on the link can be derived by entering the density ρ on the link. To make it easier to calculate v , equation 3.7 can be rewritten as equation 3.8 and 3.9⁴.

$$c_3 \cdot v^2 + \left(c_1 - \frac{1}{\rho} + c_3 \cdot v_f\right) \cdot v - c_2 - v_f \cdot c_1 + \frac{v_f}{\rho} = 0 \quad (3.8)$$

$$v = \frac{-b - \sqrt{b^2 - 4ac}}{2a} \quad (3.9)$$

$$a = c_3 \quad (3.9a)$$

$$b = c_1 - \frac{1}{\rho} + c_3 \cdot v_f \quad (3.9b)$$

$$c = -c_2 - v_f \cdot c_1 + \frac{v_f}{\rho} \quad (3.9c)$$

The following variables are used in the model:

Parameter	Explanation
v_f	free flow speed [km/h]
ρ_{cr}	critical density [veh/km]
q_{cr}	flow at capacity [veh/h]
v_{cr}	speed at capacity [km/h]
c_1	fixed distance headway constant [km]
c_2	first variable distance headway constant [km ² /h]
c_3	second variable distance headway constant [h]
m	constant used to solve the three headway constants [h/km]
a	constant used to determine the zero point of the speed equation
b	constant used to determine the zero point of the speed equation
c	constant used to determine the zero point of the speed equation

Table 3.4 – Parameters of the fundamental diagram by Van Aerde

3.2.4 Urban modelling

MaDAM calculates the average speed on a link, using a convection, relaxation and anticipation term (see section 3.2.2). The outcomes of these speed calculations are quite

⁴ Equation 3.9 is derived from equation 3.8 by using the ‘abc-formula’ or ‘quadratic formula’, which is a general method to determine the zero points in a quadratic equation.

realistic for motorway traffic. However, in urban areas vehicles react quicker on other road users and drive more aggressively, which makes the current anticipation term less suitable for urban areas. For this reason, the user is able to select whether a link in the network is a motorway link or an urban link. In urban areas, a different anticipation term is used (Raadsen, Mein, Schilpzand, & Brandt, 2010):

For motorway networks:

$$anticipation = \frac{v \cdot T \cdot (\rho_{m,i+1}(t) - \rho_{m,i}(t))}{\tau \cdot L_m \cdot (\rho_{m,i}(t) + \kappa)} \quad (3.10)$$

For urban areas:

$$anticipation = T \cdot \frac{\zeta}{2\tau \cdot L_m} \cdot \frac{V(\rho_{m,i+1}(t)) - V(\rho_{m,i}(t))}{\rho_{max} - \rho_{m,i+1}} \quad (3.11)$$

Parameter	Explanation
k_{max}	maximum density
τ	control parameter; the higher the value, the less influence of downstream traffic
ζ	parameter to calibrate the throughput factor

Table 3.5 - Parameters used in the anticipation term for urban areas

3.3 Traffic propagation in StreamLine

This section describes which steps StreamLine takes in the network during the traffic propagation. These steps are visualised in figure 3.6.

3.3.1 Time loop (orange)

The simulation starts at $t=0$. Firstly, StreamLine calculates the traffic characteristics (i.e. density, speed and flow) of the links in the network. When this is done for all the links in the network, StreamLine calculates the traffic characteristics of the link segments around nodes. At a node, there are several entry and exit links with its own traffic characteristics

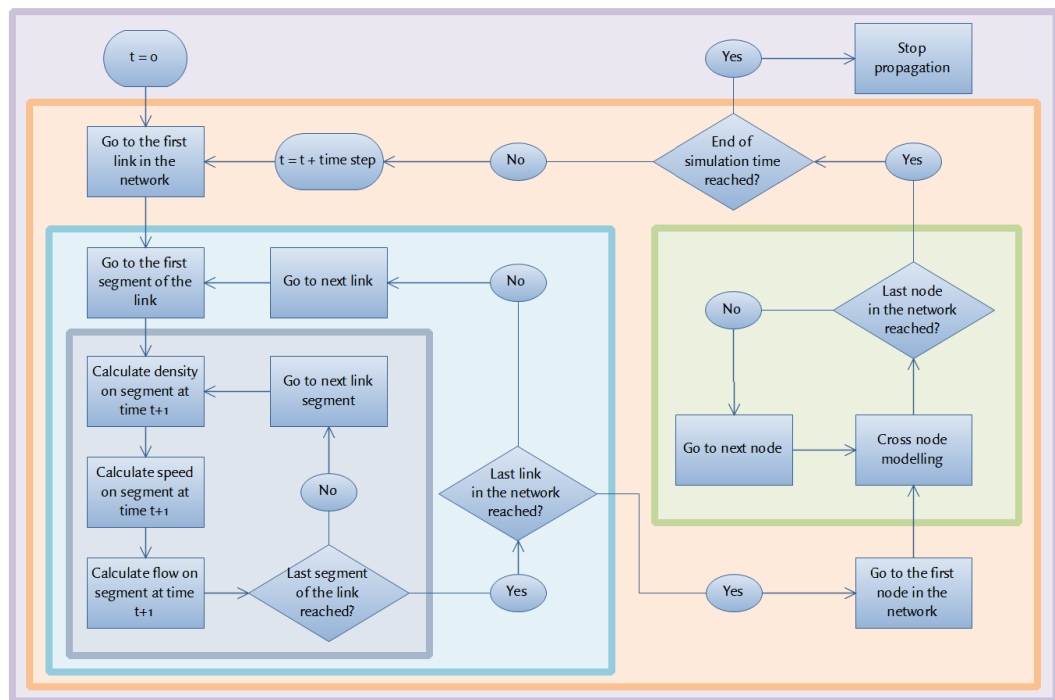


Figure 3.6 - Overview of the steps that are taken in StreamLine during propagation

that influence each other. StreamLine uses Cross Node Modelling to deal with this. This method is explained in appendix III. When the node is a junction, StreamLine uses a junction model (XStream) to calculate the traffic characteristics, but junction modelling is outside the scope of this research.

When the traffic characteristics of all link segments around nodes are calculated, StreamLine checks whether the end of the simulation time is reached. If this is the case, StreamLine will stop the propagation; if not, StreamLine goes to the next time step. The time loop starts again. In StreamLine it is possible to enter the time step. The default value is 1 second, but it is possible to change this value. The smaller the time step, the more accurate the results will be.

3.3.2 Link loop (light blue)

The link loop begins with the first link in the network. Section 3.2.1 mentions that each link exists of several segments. MaDAM calculates the traffic characteristics for each segment of the link. When this is done for all segments of the link, StreamLine questions whether the last link in the network is reached. If this is the case, StreamLine investigates the nodes in the network. Otherwise, StreamLine goes to the next link and the link loop repeats. The sequence of the links does not affect the outcome of the traffic characteristics.

3.3.3 Segment loop (dark blue)

The goal of the segment loop is to calculate the traffic characteristics for all segments of the link. The loop starts with the first segment of the link. StreamLine uses the formulas of MaDAM (section 3.2.2) to calculate the density, the speed and the flow of this segment for the next time step. To make this possible, StreamLine uses the traffic characteristics that are calculated in the previous time step. After this, the traffic characteristics of the next link segment are calculated, until the last segment of the link is reached.

3.3.4 Node loop (green)

When the traffic characteristics of all links are calculated, StreamLine uses Cross Node Modelling to transpose the traffic from the end to the ingoing link to the beginning of the outgoing link. After all, StreamLine needs information about the upstream and downstream traffic to calculate the speed and the density. Cross Node Modelling combines the information of the ingoing and outgoing links, to spread the traffic from the ingoing links over the outgoing links. Appendix III gives a more detailed description of Cross Node Modelling in StreamLine.

Hereafter, StreamLine goes to the next node in the network, until the last node in the network is reached. Again, the sequence of the nodes does not influence the results, since StreamLine uses the traffic characteristics that are calculated in the previous iteration.

4 Methodology

This chapter describes the methodology of this research. It describes the steps that are taken to answer the research questions that are mentioned in chapter 1.

Firstly, it is necessary that a suitable definition of a DUE is formulated. A literature study gives a good impression of the different definitions of a DUE that are available in literature. In combination with a research to the characteristics of MaDAM and StreamLine, a definition of a DUE is formulated that can be used in this research. This step is described in chapter 2 and 3.

To find out whether a DUE arises in StreamLine and to investigate the sensitivity of this DUE to parameter changes, a situation must be created where a DUE can be determined. This is done by creating fictive networks in StreamLine. To answer the main question of this research comprehensively, several networks must be created, so that the research investigates all different traffic situations that occur in reality. Since it is too time-expensive for this research to investigate all these networks, a selection of two networks is made. When choosing suitable networks, it must be taken into account that the demand pattern in StreamLine is discrete and fixed.

Section 4.1 lists the requirements that the networks in StreamLine must meet. Subsequently, in section 4.2, the used networks are described. Section 4.3 describes the research method, followed by an explanation of the ‘relative standard error’ and the ‘elasticity coefficient’ in section 4.4 and 4.5.

4.1 Requirements of the network

To determine whether StreamLine determines a DUE, a suitable test network needs to be set up in OmniTRANS. To answer the research questions, the network must meet a number of requirements:

- The network has to be **small**, so that the duration of a simulation is not very long and it is easy to keep overview of what happens in the network. With a small network, it is also possible to recalculate the model by hand.
- A DUE is reached when the travel times of all route alternatives of an OD pair are equal. Therefore, travellers must have **several route alternatives** to go from origin to destination. If this is not the case, it is not possible to see route changes, which is quite important in this research.
- Initially, each OD pair must have a **limited number of route alternatives**, to reduce complexity and to keep a good overview. For the same reason, the overlap between the routes must be minimised. When this simple network works well, it

- must be possible to increase the number of routes of each OD pair. In the latter case, overlap is allowed to add an extra dimension of complexity to the network.
- The **free flow travel times** of the generated routes may not differ too much from each other. The situation must be created that travellers change their route when congestion arises on a specific link.
- In the research some parameters are varied. The network has to be set up in such a way that **changes in parameters influence route costs**. For example, when the minimum speed is varied in the network, there must be some situations in the network where the speed of a links drops to the minimum speed. The parameters that are varied are mentioned in section 4.5.
- StreamLine must be able to **generate routes** and distribute traffic in the network.

According to these requirements, two networks are designed. The first network is a simple network without overlap of routes and with one OD pair. The second network is slightly more complex than the first network. The routes of this network overlap each other and this network contains a congestion situation. The congestion situation is needed to ensure that the traffic in the network reaches the minimum speed. The two networks are described in the following section.

4.2 Description of the networks

4.2.1 Network 1

The first network that is used in the research is a simple network with only three routes and one OD pair. The routes do not overlap. The simplicity of this network makes it easy to reproduce the results of StreamLine and to recalculate the results by hand.

Figure 4.1 shows the first network. In this network, the traffic goes from the left centroid to the right centroid. On the two shortest routes (route 1 and 2) the maximum allowed speed is 50 kilometres per hour; the maximum allowed speed on the upper route (route 3) is 60 kilometres per hour. The reason for this difference in speed is that the free flow travel times of the three routes are close to each other. As a result, when it gets busier on a specific route, a vehicle will change his route. This will hardly happen when the difference in free flow travel times of routes is quite large.

The links around junctions and merge nodes have a maximum allowed speed of 30 kilometres per hour. This makes the behaviour of the traffic more realistic, since when a vehicle approaches a junction, its speed decreases. In the middle of each route there are also a few links where the maximum allowed speed is 30 km/h. This is done to create speed differences within a route. When in a later stage of the research several parameters are varied, differences can be observed in terms of acceleration and deceleration.

The three routes do not have overlap with each other, except on the connector links. The connector link is the link that makes it possible to connect traffic from the centroid to the network. However, the length of the connector links is minimal and the capacity of the connector links is very large. Therefore, these points of overlap will not influence route choice much. After all, all vehicles are obliged to use the connector links, irrespective of the route they choose. So, this will not lead to differences of the results of the different routes.

All links in the network have a length of 300 metres. This is because StreamLine divides each link automatically in segments of 300 metres. Thus, each link in the network has the same length as a link segment. When all links in the network are 300 metres long, it is possible to display the results (i.e. flow, density or speed) very accurately in StreamLine.

Figure 4.2 shows the length of the different routes. Route 1 is the shortest route and has a length of 4.5 kilometres. Route 2 is slightly longer with 4.8 kilometres. The longest route is route 3 with a total length of 6.0 kilometres. When these distances are combined with the maximum allowed speeds from figure 4.1, the free flow travel times for the three routes are respectively 0.106 hours (382 seconds), 0.112 hours (403 seconds) and 0.12 hours (432 seconds).

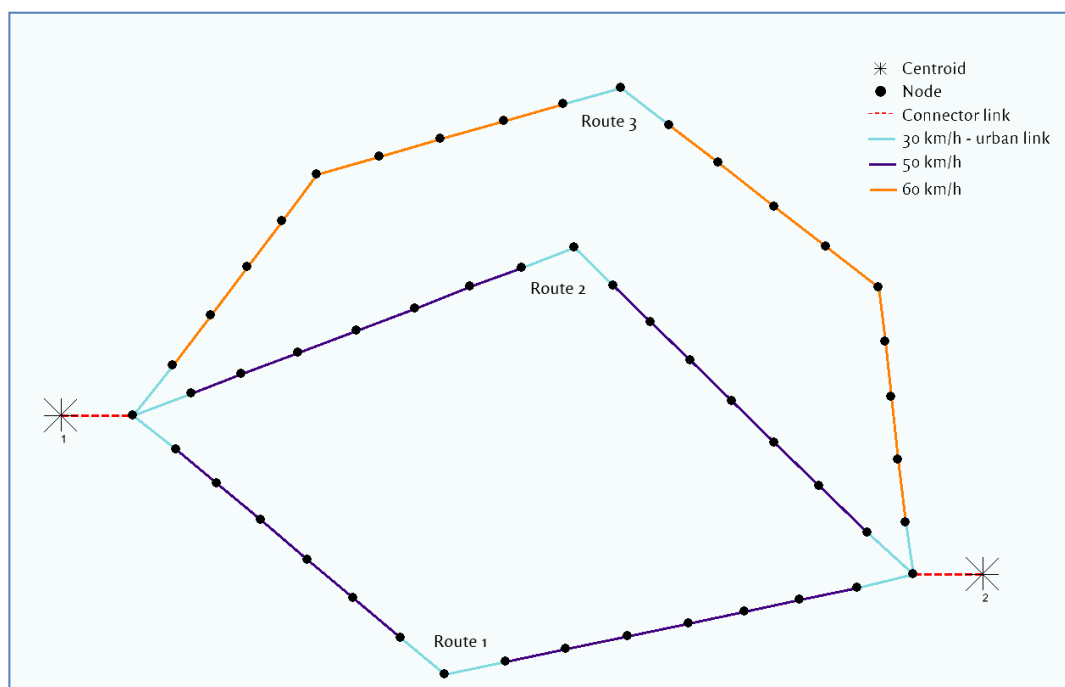


Figure 4.1 - Overview of the allowed speeds in network 1.

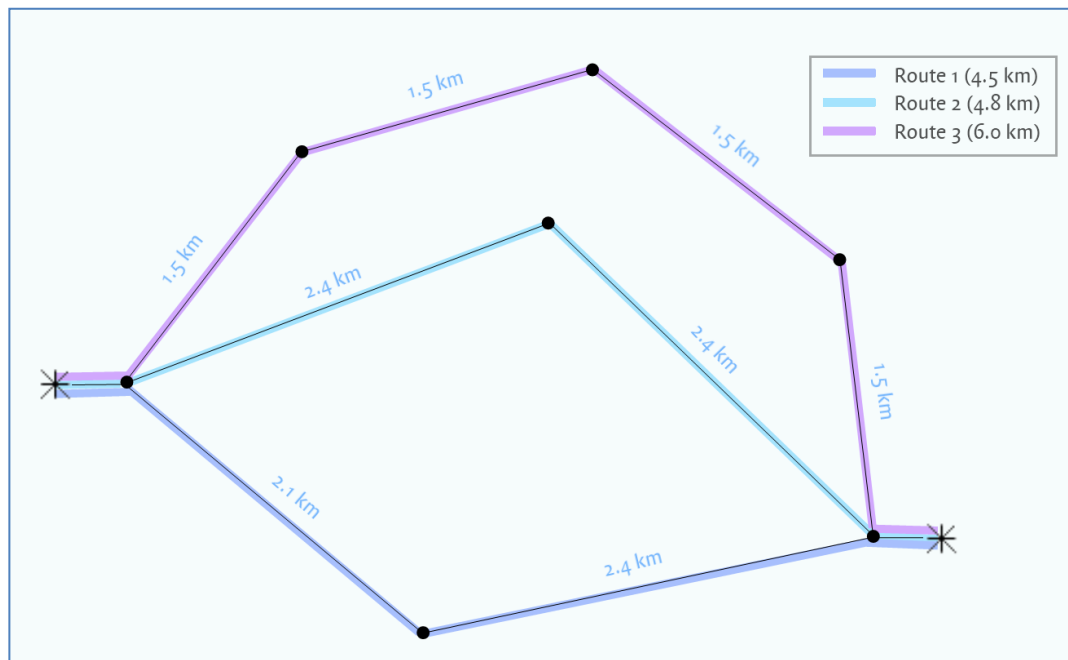


Figure 4.2 - Overview of the routes and distances in network 1.

Figure 4.3 shows the demand on the network during the simulation. The simulation time is divided in four equal segments of 1800 seconds (30 minutes). In the first 1800 seconds, the demand is set to 300 veh/h, in the second segment, the demand is increased to 1500 veh/h. From 1800 to 3600 seconds, the demand is increased again to 3000 veh/h and at the fourth part of the simulation no traffic is departing. The demands are chosen in such a way that in the first 1800 seconds only the shortest route (route 1) will be chosen, that in the second 1800 seconds the traffic will be distributed between route 1 and 2. After 3600

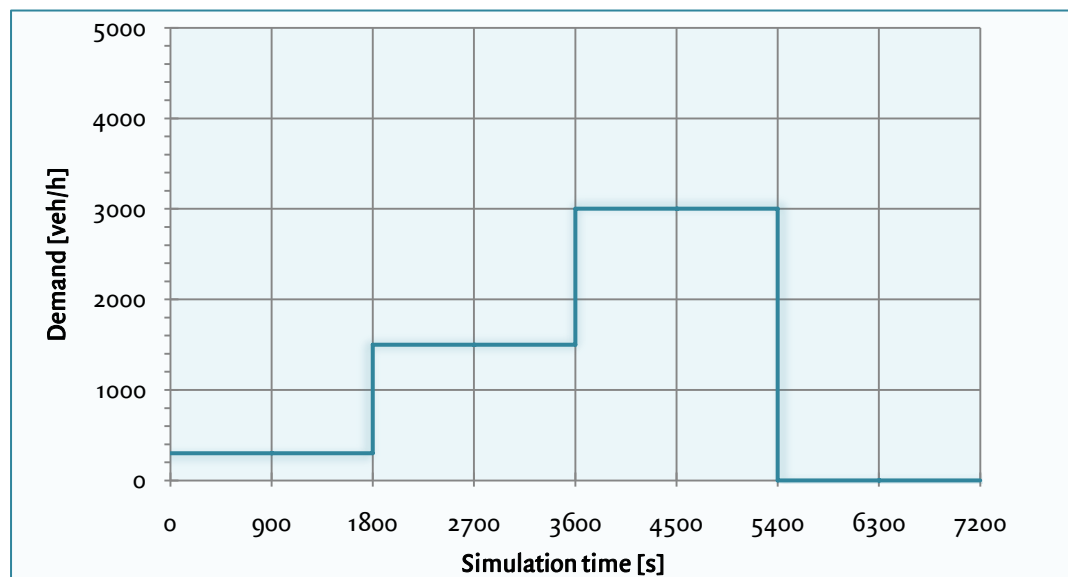


Figure 4.3 - Departure pattern of network 1.

seconds (60 minutes), route 1 and 2 are quite busy, so that also traffic will choose the longest route (route 3). In the fourth part of the simulation no traffic departs, to make it possible for the traffic on the network to reach their destination before the simulation ends.

The capacity of all links is 1800 vehicles per hour. When all three routes are used, the total capacity of the network is $1800 \times 3 = 5400$ vehicles per hour. Since the maximal demand is only 3000 vehicles per hour in this simulation, the network can process the demand and it is very unlikely that congestion will occur.

4.2.2 Network 2

The second network that is used in this research is slightly bigger. Similarly to the previous network, the second network only has one OD pair and three route alternatives. However, in this network the routes overlap and there are two locations where several links merge into a single link. The latter characteristic makes it possible to analyse δ .

Figure 4.4 shows the second network. In this network, people travel from the centroid on the left to the centroid on the right. The fictive network can be compared with a situation of a city. To pass a city, people have the opportunity to drive through the city centre (route 1), use the ring road (route 3) or take a road around the city (route 2). The maximum speeds and the number of lanes on the roads are different, which is shown in figure 4.5.

The links upstream a junction and the links in the 30-km/h-zone are classified as urban links. This prevents that the speeds on links upstream merge nodes drop to zero km/h and makes the simulated behaviour in 30-km/h-zones more realistic.

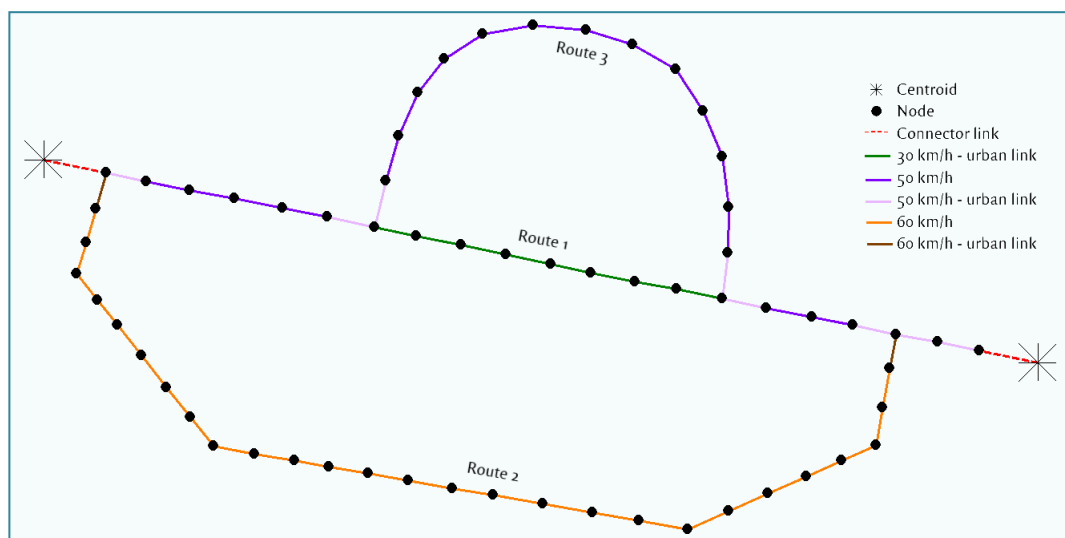


Figure 4.4 - Overview of the allowed maximum speeds in network 2.

Figure 4.5 shows the route alternatives people have. The routes have different lengths, but the free flow travel times are almost similar. Route 1 and route 2 have a free flow travel time of 0.152 hours (547 seconds), the travel time of route 3 takes 0.156 hours (562 seconds). The length of the connector links is minimal.

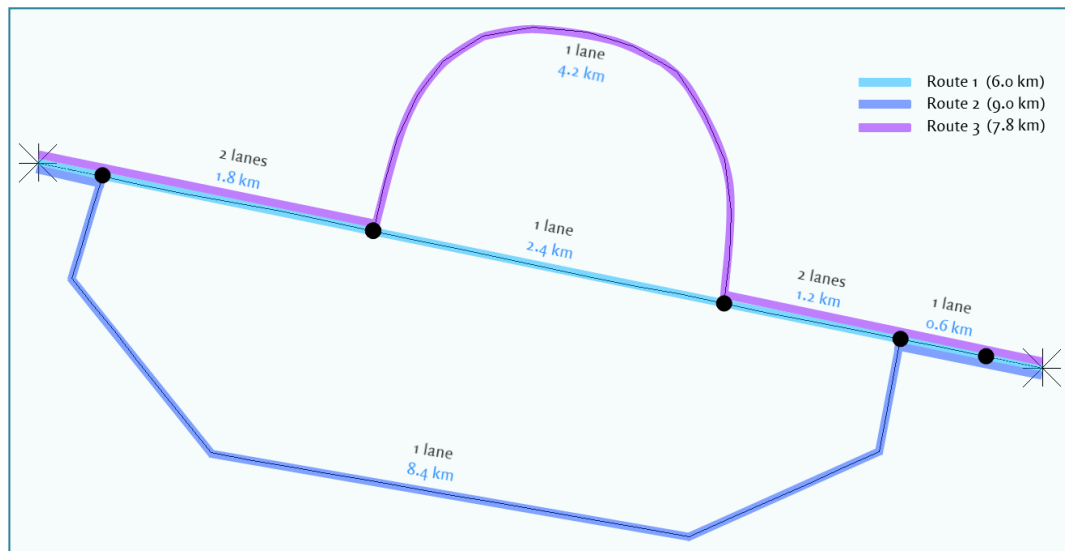


Figure 4.5 - Overview of the routes, the length of the links and the number of lanes in network 2.

To make it possible to investigate the value of the minimum speed, a congestion situation is created. The capacity of a link is 1800 vehicles per hour (3600 veh/h in case of a two-lane road). Near the destination, three lanes merge to one single lane, which is the bottleneck of the network. At this point the capacity of the bottleneck of the network is 1800 veh/h, while the capacity of the rest of the network is 5400 veh/h. This can cause congestion, dependent on the demand.

The departure pattern is shown in figure 4.6. The simulation time contains two hours and is divided in eight equal segments. In the first segment (0-900 seconds) the demand is set to 1800 veh/h, which is the capacity of the bottleneck. No congestion should occur. In the second period (900-1800 seconds) the demand on the network increases, so that the flow exceeds the capacity of the bottleneck and a congestion situation occurs. In the third segment (1800-2700 seconds) the demand on the network decreases to 1000 veh/h. This demand is lower than the capacity of the bottleneck, which makes that the congestion in the network slowly disappears. This demand remains constant till the end of the simulation. In figure 4.6, the dotted line is the capacity of the bottleneck.

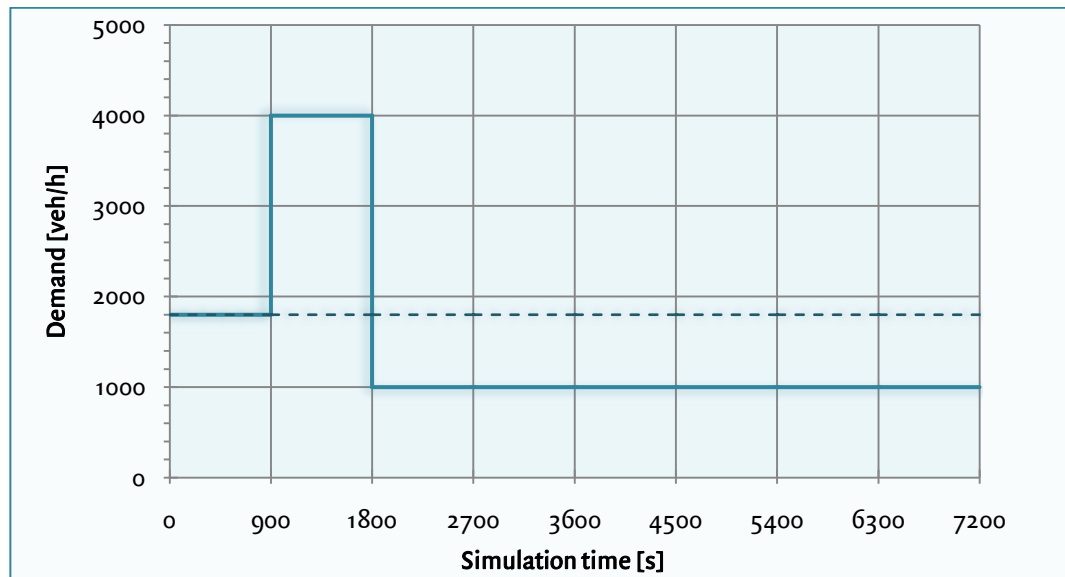


Figure 4.6 - Departure pattern in network 2. The dotted line displays the capacity of the bottleneck (1800 veh/h).

4.3 Method

To test whether StreamLine reaches a dynamic user equilibrium, the traffic flow from zone 1 (left centroid) to zone 2 (right centroid) is taken into account. This means that an OD matrix with only traffic from zone 1 to 2 is assigned to the network. It is chosen to begin with this situation, since this is a very simple network and it is easy to recalculate the results by hand. There are several possible routes for this OD pair. StreamLine chooses the number of routes that is used in the beginning of the simulation. By doing this, the least attractive routes are filtered from the route set. It is also possible to enter the maximum number of routes that are used for each OD pair.

Then it is checked whether StreamLine determines a DUE, according to the definition formulated in chapter 2. In StreamLine, route costs are subdivided into travel time and travel distance. It is assumed that people mostly choose the route with the shortest travel time, so only travel time is taken into account. In StreamLine, it is possible to collect the route costs of each route at each route choice moment. The route costs are displayed in a graph with the development of the route costs of the different routes during the simulation. The graphs of the different routes are compared.

To check whether a DUE is reached, a measure is designed that indicates the closeness to equilibrium for every simulation. This measure is called the relative standard error (RSE) and is explained in more detail in section 4.4. The RSE makes it possible to compare the results of different simulations quite easily. The closer the RSE comes to zero, the closer the simulation comes to an equilibrium.

When the simulation comes close to equilibrium, the number of iterations and the length of the route choice interval are varied. For each simulation, the RSE is calculated to find a relationship between the number of iterations, the length of the route choice interval and the closeness to equilibrium. When all combinations are tested, it can be determined which combination is suitable for the parameter analysis later in this research. Accuracy and running time are important criteria for this decision.

Subsequently, a parameter analysis will be executed. In this parameter analysis the robustness of StreamLine is investigated. The analysis starts with an equilibrium run for both network 1 and network 2. This equilibrium run uses a large number of iterations (10,000) and a large number of route choice moments (the route choice interval is 1 minute). This is done to create an accurate and detailed equilibrium. This equilibrium will be the benchmark for the parameter analysis.

Hereafter, some parameters in StreamLine are varied. This is done to investigate which influence parameter changes have on a DUE. Several parameters are used that influence route choice or traffic propagation. These parameters are described in section 3.2. As a result, these parameters can influence the outcome of the model. To look to which extent the results of the model are sensitive for parameter changes, it is investigated whether the model always determines the same DUE, regardless of changing the values of these parameters. To check this, the parameters are varied and the results are compared with each other. Section 4.5 explains which parameters are varied and which measures are used to test the robustness of StreamLine.

4.4 Relative standard error

To make it possible to compare dynamic user equilibria, a measure is defined. On every route choice moment, the relative deviation of the travel time on utilised routes is calculated, in comparison with the average travel time among the utilised routes. A route is utilised when at a route choice moment the route is chosen more than once during the whole simulation.

It is chosen to calculate the relative deviation instead of the absolute deviation, since in reality, travellers make choices that are based on relative differences in route costs. Example 4.1 illustrates this.

Example 4.1 – Absolute and relative travel time

Imagine that in the one situation the travel times of two routes are respectively 1.0 and 3.0 minutes and in the other situation respectively 100.0 and 102.0 minutes. In the first situation almost every traveller chooses the shortest route, while in the second situation the travellers are probably (almost) equally divided over the routes. While the absolute

difference is the same (2.0 minutes), the relative difference is quite different (200% versus 2%).

In this research, the relative deviation is determined by calculating the relative standard error (Massachusetts Institute of Technology, 2011), which is written down in formula 4.1 and 4.2.

$$RSE = \frac{s}{\bar{x}} \cdot 100\% \quad (4.1)$$

With;

$$s = \sqrt{\frac{\sum_{i=1}^N (x_i - \bar{x})^2}{N - 1}} \quad (4.2)$$

Parameter	Explanation
s	standard deviation of a measurement
\bar{x}	mean value of the measurements
x_i	value of a measurement
N	number of measurements

Table 4.1 - Used parameters in RSE calculation.

To compare dynamic user equilibria, for every route choice moment the relative standard error is calculated. To achieve this, some adaptations are made to formula 4.2. Firstly, in formula 4.2 the measurements (which are the travel times at a route choice moment) are not compared with the mean of these measurements, but with the travel time at equilibrium. Since it is difficult to determine whether a ‘true’ equilibrium is reached, a simulation is run with many route choice moments and a high number of iterations. The more route choice moments StreamLine uses, the more the departure pattern in StreamLine corresponds with a realistic continuous departure pattern.

Furthermore, in the RSE calculation the sum of the squared differences is divided by the degrees of freedom. When the measurements are compared with the mean, the degrees of freedom are similar to N-1, since the Nth measurement is pointed when all other measurements and the mean are known. In this research this is not the case, since the measurements are compared with an independent value. Therefore, the degrees of freedom are set to N instead of N-1.

Subsequently, the average value of the calculated RSEs of one OD pair is calculated and this is multiplied by the demand of this OD pair. This is done to increase the share of OD

pairs with a large demand to the overall measure in comparison with OD pairs with a smaller demand. When these values of all OD pairs in the network are summed, the value is divided by the total demand in the network and the total number of OD pairs in the network. The result of this equation is μ , which is the measure that describes the average relative standard error during the whole simulation. The lower the value of μ is, the closer the results of StreamLine are to an equilibrium. With this measure it is easy to compare two different simulations.

$$\mu = \frac{\sum_{od=1}^{OD} \left(\frac{\sum_{m_{od}=1}^{M_{od}} (RSE_{od,m})}{M_{od}} \cdot Q_{od} \right)}{OD \cdot \sum_{od=1}^{OD} Q_{od}} \quad (4.3)$$

With;

$$RSE_{od,m} = \frac{\sqrt{\frac{\sum_{r_{od,m}=1}^{R_{od,m}} (T_{od,m,r} - T_{eq_{od,m}})^2}{R_{od,m}}}}{T_{eq_{od,m}}} \cdot 100\% \quad (4.4)$$

Parameter	Explanation
μ	Average relative standard error
m	Route choice moment
r	Route
od	Origin-destination pair
OD	Total number of origin-destination pairs in the network
m_{od}	Route choice moment at OD pair od
M_{od}	Number of route choice moments at OD pair od
Q_{od}	Demand at OD pair od
$RSE_{od,m}$	Relative standard error at OD pair od at route choice moment m
$r_{od,m}$	Utilised route at OD pair od at route choice moment m
$R_{od,m}$	Total number of utilised routes at OD pair od at route choice moment m
$T_{od,m,r}$	Travel time of route r of OD pair od at route choice moment m
$T_{eq_{od,m}}$	Travel time at the equilibrium run for OD pair od at route choice moment m

Table 4.2 - Parameters that are used to calculate the relative standard error.

4.5 Parameter analysis

The main parameters are mentioned in the table 4.3. The parameters are varied between the values that are written in the last column. The varied parameters are parameters that influence traffic propagation. The chosen boundaries in these tables are underpinned in appendix I.

Not all parameters that are used in the model are varied. Some parameters are kept constant to reduce complexity. The following parameters are not changed in this research:

- Free speed on links. The influence of the free speed on a DUE is assumed to be minimal, since the speed ratio between the links remains the same when the free speed on the links changes. This is only effective when all links in the network have the same speed.
- Jam density. The current default value in OmniTRANS is 180 vehicles per hour and is based on literature studies (Kerner & Konhäuser, 1994) and is the same on all links, regardless of the speed of that link. It is not realistic when this parameter changes.
- Number of iterations. This parameter does change the user equilibrium, since the accuracy of the results is dependent on the number of iterations StreamLine executes.
- Urban traffic. In StreamLine, an option is available that the traffic in the network behaves like in an urban environment. To reduce complexity, this parameter is kept constant. On the other hand, the parameters tau, kappa, nue and delta are MaDAM variables, which cannot be varied when there are only urban links.

Traffic propagation parameters

Variable	Explanation	Default value	Variation
τ (tau)	Variable for non-urban roads to calculate the speed. The higher the value, the less influence of partial speed calculation [s].	20	5 – 10 – 20 – 40 – 80
ν (nue)	The higher the value, the more influence of anticipation of next link segment [km ² /h]	35	0 – 17.5 – 35 – 70 – 140
κ (kappa)	The higher the value, the less influence of anticipation on next link segment [veh/km].	13	1 – 7 – 13 – 19 – 25

δ (delta)	The higher the value, the less influence of merging effects [s^{-1}].	0.8	0 – 0.4 – 0.8 – 1.2 – 1.6
minimum speed	The minimum speed on a link. In case of congestion, the calculated speed will be replaced by this value, if it drops below this minimum speed [km/h].	7	1 – 4 – 7 – 10 – 13

Table 4.3 - Traffic propagation parameters in StreamLine that are varied.

In the parameter analysis three things are checked:

- Check whether an equilibrium is reached.
- Check whether the equilibrium changes in relation to the equilibrium situation with default parameters.
- Check to which extent the sensitivity of the parameter changes with respect to the sensitivity of other parameters.

To check to which extent a DUE is reached, for each simulation the average route cost difference is calculated. At each route choice moment the difference between the maximal and the minimal route costs are calculated. The average route cost difference is the average difference of all route choice moments. The lower the average route cost difference, the more the simulation comes close to a DUE.

The elasticity coefficient is a measure that indicates to which extent an equilibrium changes as a result of parameter changes (point 2 and 3 in the enumeration above). Section 4.6 gives an explanation about elasticity coefficient.

4.6 Elasticity coefficient

To test the robustness of StreamLine, several parameters are varied in StreamLine (which is explained in section 4.5). When the value of a parameter changes, it is possible that the dynamic user equilibrium changes. The elasticity coefficient is used to indicate the sensitivity of a parameter. The elasticity coefficient originates from economics and calculates the ratio between the relative change of the parameter and the relative change of the depending variable. The lower the value of the elasticity coefficient, the lower the sensitivity of the changed parameter is. When the elasticity coefficient is lower than 0.001, the sensitivity of the parameter is assumed to be negligible. In this research, the dependent variable is the average RSE, μ . The elasticity coefficient is written down in formula 4.5 (Missouri State, 2011).

$$\varepsilon = \frac{\% \Delta V}{\% \Delta P} \quad (4.5)$$

Parameter	Explanation
ε	Elasticity coefficient
$\% \Delta V$	Relative change in depending variable
$\% \Delta P$	Relative change in parameter

Table 4.4 - Parameters that are used to calculate the elasticity coefficient.

Example 4.2 – Elasticity coefficient

StreamLine varies the minimum speed in the network. Two runs are done; one simulation with default parameters (minimum speed is 7 kilometres per hour) and one simulation with a minimum speed of 14 kilometres per hour. The average RSE, μ , increases from 0.04 to 0.06. So, the elasticity coefficient is:

$$\varepsilon = \frac{\frac{0.06 - 0.04}{0.04} \cdot 100\%}{\frac{14 - 7}{7} \cdot 100\%} = \frac{50\%}{100\%} = 0.5$$

In this example a change in the minimum speed in StreamLine of 1% results in an increase of μ of 0.5%. Since this is a quite large value, it can be said that the sensitivity of the minimum speed on μ is large.

5 Encountered problems with PCL and MaDAM

In the previous chapter, the methodology of the research is described. The first phase of the research, tries to reach a DUE in a simple network. During this step, several problems emerged that prevent that StreamLine determines a DUE. This chapter describes these problems. Section 5.1 describes why the Paired Combinatorial Logit model (a route choice model in StreamLine) does not lead to a deterministic equilibrium in most cases. Section 5.2 mentions several characteristics of MaDAM that are unrealistic and that influence route choice in StreamLine. In section 5.3, several adaptations are made to the current network, to deal with the encountered problems that are described in section 5.1 and 5.2.

5.1 Characteristics of PCL

Section 4.3 describes that there are three methods to calculate the route fractions during the simulation in StreamLine. The Paired Combinatorial Logit (PCL) model is one of these route choice models. In StreamLine, PCL is the default route choice model. PCL is a logit model, which means that it calculates the route distribution based on the cross-elasticity between pairs of route alternatives. An advantage of this method is that it converges quite fast in comparison with other route choice models, like a successive all-or-nothing assignment. Appendix II gives a more detailed description of PCL.

Since PCL is a logit model, the probability that a route alternative is chosen is based on the relative route costs in comparison with other route alternatives. Consequently, at least a small percentage of traffic will be assigned to all generated routes, even when the travel costs of a specific route are larger than other routes. Referring to the definition of a DUE that is used in this research, the route costs of all used routes have to be equal to each other at every route choice moment. Especially when the demand on the network is low and there are large differences between route alternatives, a DUE will never occur when PCL is used. This situation is illustrated in example 5.1. However, when the spread parameter is set to a value close to zero, this effect is smaller.

Example 5.1 – Dynamic User Equilibrium with a PCL-assignment

In a small network with one OD pair there are three routes with different lengths and no overlap with each other. The flow on the network increases every 1800 seconds (30 minutes) and is in the first four time periods respectively 1800, 3000, 4200 and 0 vehicles per hour. The capacity of a route is 1800 vehicles per hour and the route choice moment interval is 600 seconds (10 minutes). StreamLine uses the PCL algorithm to determine an equilibrium, with $\mu=0.14$ (default). The route fractions and the route costs of each route are shown in figure 5.1, when StreamLine states that an equilibrium is reached (duality gap = 0.001).

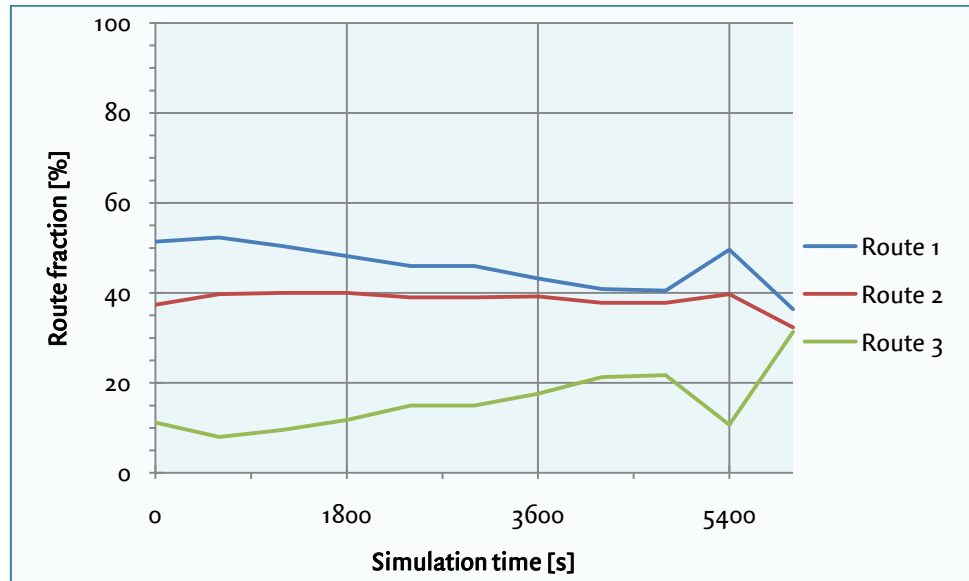


Figure 5.1 - Results of a PCL assignment: evolution of the route fractions over time.

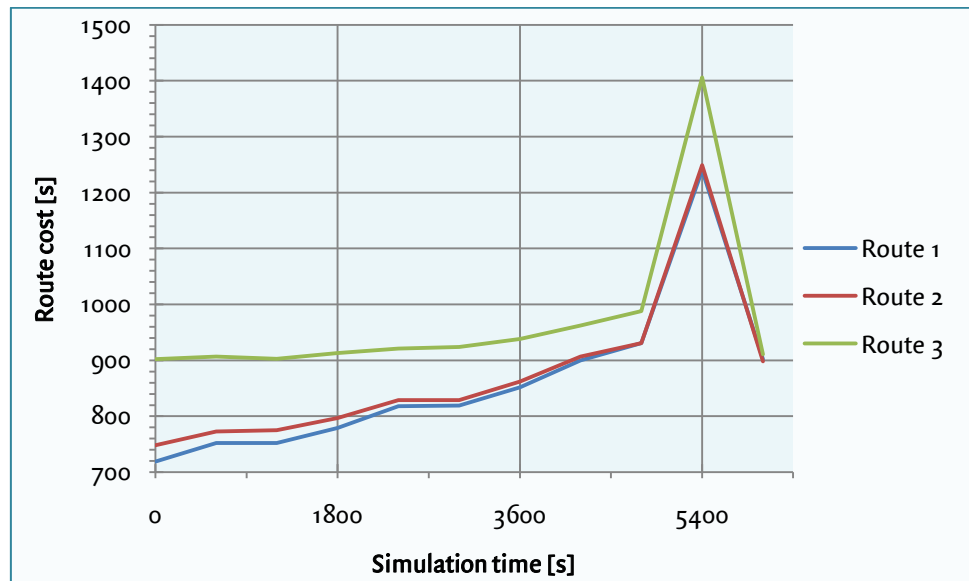


Figure 5.2 - Results of a PCL assignment: evolution of the route costs over time (travel time).

Figure 5.1 and 5.2 show that a DUE is not reached, according to the definition of a DUE that is used in this research (section 2.1.2). In the first time period (from 0 till 1800 seconds) 10% of the vehicles choose the route 3 (the green route), while the route costs of route 1 and 2 (the blue and the red route) are quite smaller than the route costs of the red route (750/770 seconds, versus 900 seconds). In this time period the route costs deviate from each other, while all three routes are used. According to the definition of a DUE, a DUE is not reached in this case. Also in the rest of the simulation the route costs of the three routes are not identical.

Section 2.1.1 shows the difference between a deterministic and a stochastic equilibrium. The definition of a DUE that is used in this research is a deterministic equilibrium, while PCL calculates a stochastic equilibrium⁵. For this reason, a series of all-or-nothing assignments is used in the research to answer the research question.

As mentioned earlier, an advantage of PCL is that it converges fast. In practice, it is possible to combine PCL and all-or-nothing assignments. When the user chooses PCL in the first couple of iterations, the distribution of traffic among the routes is relatively close to equilibrium. In the following iterations, a series of all-or-nothing assignments can be used to fine-tune the results. An advantage of this approach is that the fraction of traffic that is assigned to a too expensive route will be smaller than the situation where only PCL assignments are used. In spite of the better results, StreamLine does not reach a deterministic equilibrium, since a small part of traffic is still assigned to the most expensive route. However, it is expected that the results approach a deterministic equilibrium and the running time of the simulation will decrease. Since it is not possible to combine different distribution methods in StreamLine currently, this is not included in this research.

5.2 Limitations of MaDAM

The speed calculation in StreamLine is based on MaDAM, as mentioned in section 3.3. In most situations, this speed calculation is quite realistic; however, there are some situations where MaDAM gives unrealistic results. As a consequence of these situations, route choice can be influenced. These situations are discussed in this section.

5.2.1 Emptying of the network

In StreamLine, it is possible that the flow on a link in the network drops to zero. This is the case when in the end of the simulation, no traffic is assigned to the network so that the network can be emptied and all vehicles can reach their destination within the simulation time. Also, in the first iterations of a successive all-or-nothing assignment, those big flow changes can happen. For example, when at the first route choice moment in a simulation, all traffic is assigned to one route; it is likely that in the next route choice moment all traffic will be assigned to another route.

Since StreamLine is a cell-based model, StreamLine does not simulate individual vehicles, but calculates the characteristics (i.e. speed, density and flow) of every cell in the network. As a result, the way that StreamLine simulates the emptying of the network is not realistic. This phenomenon is shown in figure 5.3.

⁵ Theoretically, this should be the case. It is not tested whether PCL does determine a stochastic equilibrium in reality.

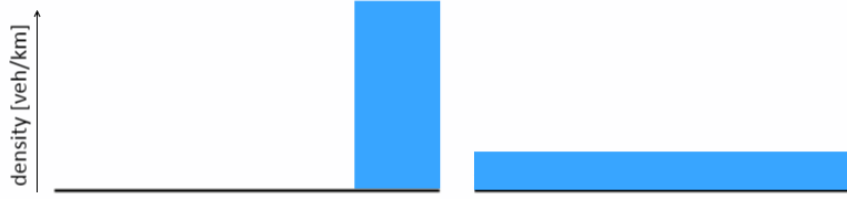


Figure 5.3 - A simplification of a cell with its demand. A situation is displayed where the largest part of the cell is empty, but there is some traffic on the end of the link. Left: density pattern of a cell in reality, right: density pattern in StreamLine.

When there are only a few vehicles on the end of the link, MaDAM spreads those vehicles equally over the entire link. Subsequently, the density of the link in the next time step is calculated by formula 5.1.

$$\rho_{m,i}(t+1) = \rho_{m,i}(t) + \frac{T}{L_m \cdot \lambda_m} [q_{m,i-1}(t) - q_{m,i}(t)] \quad (5.1)$$

When this density formula is used, the density decreases slowly in every time step. Consequently, it takes long before the network is empty. This becomes clear in example 7.2. In example 7.3 the consequences of the emptying of the network on travel times are shown.

Example 5.2 – Density change

At $t=0$, the flow in the network drops from 1000 vehicles per hour to zero vehicles per hour. Before $t=0$, the density in the network is 25 vehicles per kilometre and the speed is 40 kilometres per hour.

StreamLine calculates the link characteristics every second, so the time step T equals $1/3600$ hour. The length of a cell is 0.3 kilometre and the number of lanes on the link is 1. At $t=1$, the new density of the first cell in the network becomes:

$$\begin{aligned} \rho_{m,i}(1) &= \rho_{m,i}(0) + \frac{T}{L_m \cdot \lambda_m} [q_{m,i-1}(0) - q_{m,i}(0)] \\ \rho_{m,i}(1) &= 25 + \frac{1}{0.3 \cdot 3600} [0 - 1000] = 24.07 \text{ veh/km} \end{aligned}$$

So, the density at $t=1$ drops, which leads to a smaller flow on that link (962.9 veh/h). As a result, in the next time step the flow difference is smaller (926.9 instead of 1000 veh/h), so that the density decrease becomes lower and the density goes to zero more slowly. This is also the case in the following time steps.

Example 5.3 – Emptying of the network

In a simple network with only one route, there is a constant flow of 1500 vehicles per hour from origin 1 to destination 2 and the average travel time is 364 seconds. At $t=0$ the flow drops to zero vehicles per hour. The route exists of 15 links with a length of 300 meters. Figure 5.4 shows the development of the flows on the links in time. Every two minutes (120 seconds) an overview is given.

Figure 5.4 shows that it takes a long time before all traffic arrives at the destination. In reality, the last vehicle that departs from the origin has the same travel time as all vehicles in front of him. This means that this vehicle should arrive at $t=364$. In StreamLine it takes approximately 8 minutes (476 seconds) longer until the network is empty. This phenomenon can have big consequences on route choice.

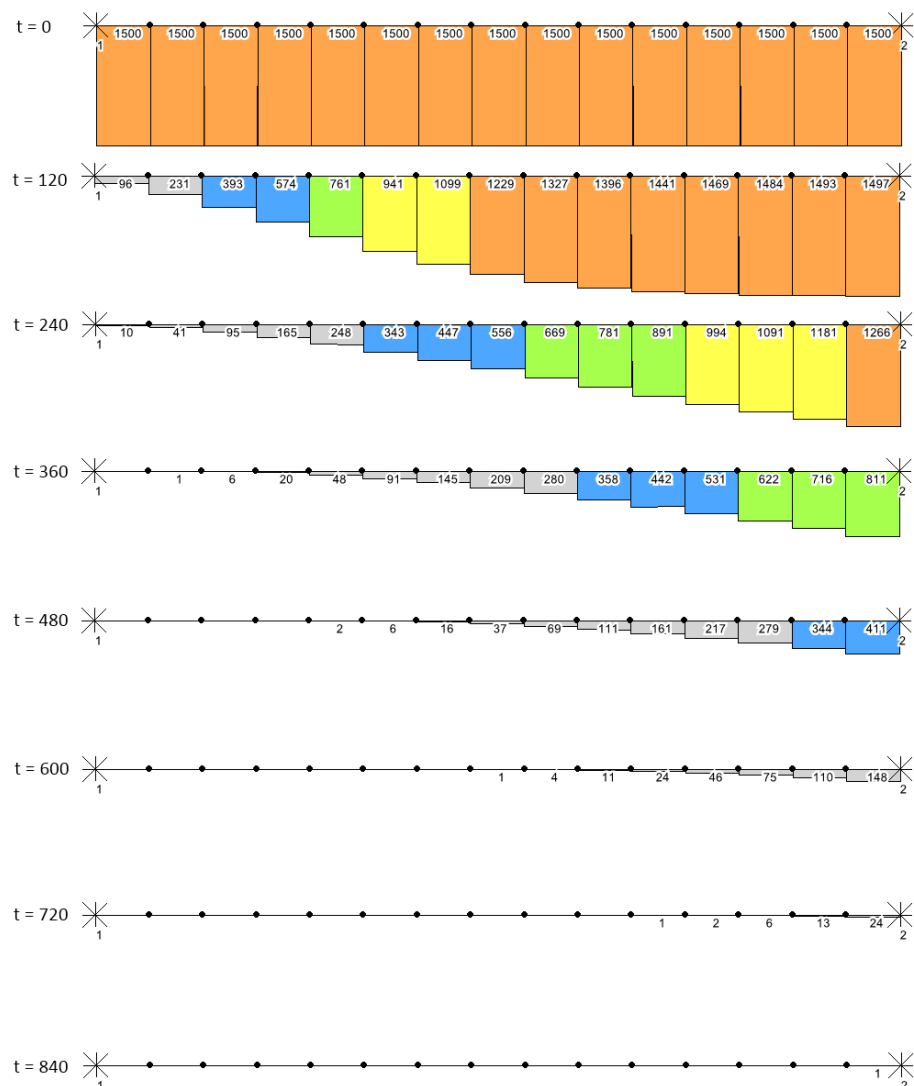


Figure 5.4 - Overview of a network that becomes empty. The flow on every link is shown every two minutes.

Streamline calculates the travel times of routes based on the real arrival and departure times of vehicles. The densities of the cells in the network are used to estimate the arrival and departure time of every single vehicle. At every route choice moment, the route costs of the previous iteration in the middle of the same time period are compared and, as a result of this comparison, a route distribution is calculated. If at this moment one of the last vehicles of a peloton departs, the travel time that is used for the comparison is not realistic and can influence route choice. Example 5.4 illustrates this problem.

Example 5.4 – Influences of the emptying of the network on route choice

In a small network with one OD pair and two routes with almost the same lengths, a successive all-or-nothing assignment is used to determine an equilibrium. The demand on the network is constant (2000 vehicles per hour), the length of a link is 600 metres and every 10 minutes (600 seconds) a new route choice distribution is calculated.

The route costs of the iterations are calculated by using densities (which is the way that StreamLine calculates route costs) and by using the average speeds (which is used by the TravelTimeCalculator). In the second iteration of the simulation, both route costs are compared and are shown in figures 5.5 and 5.6.

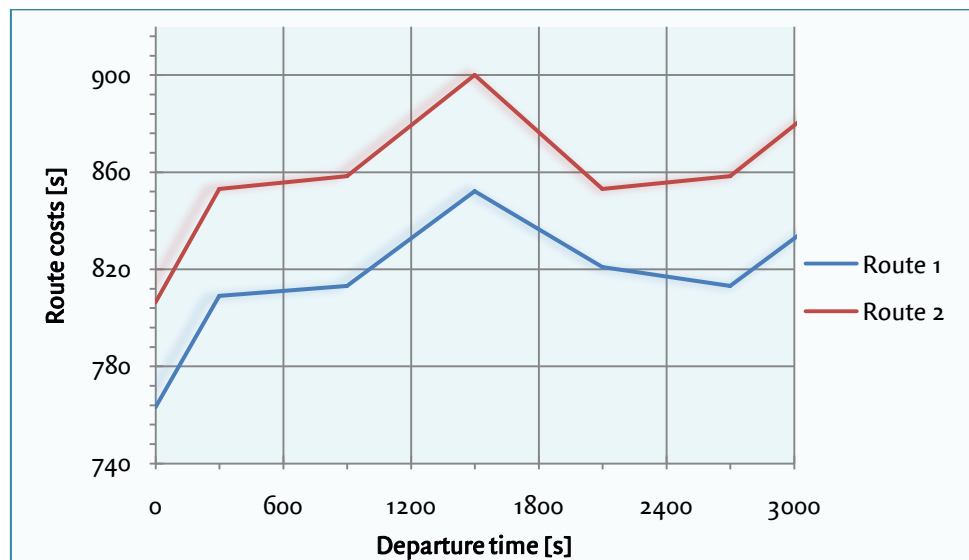


Figure 5.5 –Route costs based on cumulative vehicles (used by StreamLine to calculate route fractions). All-or-nothing assignment, second iteration.

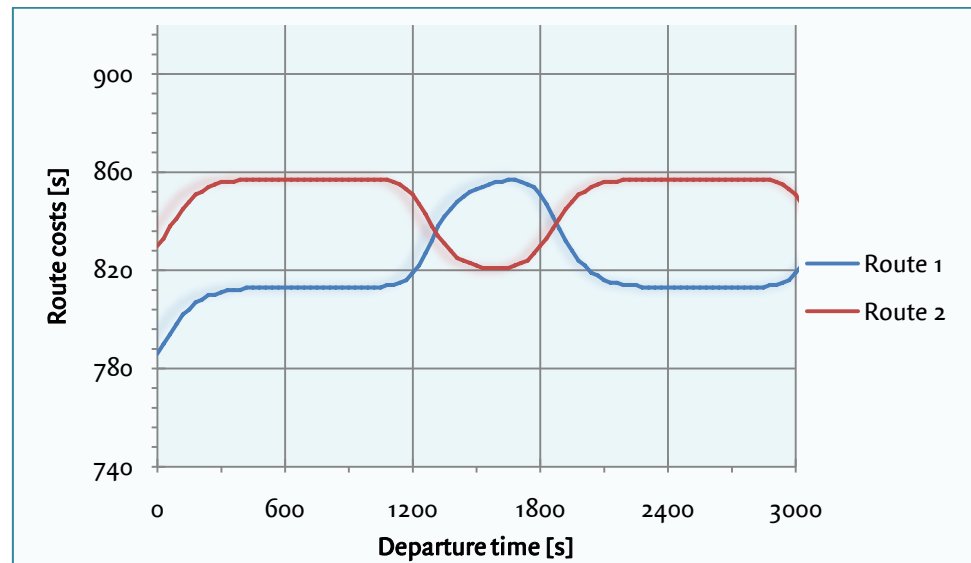


Figure 5.6 – Route costs that are calculated by the TravelTimeCalculator. Successive all-or-nothing assignment, second iteration.

The results of both methods are quite similar from $t=0$ till $t=1000$ and from $t=2000$ till $t=2700$, but at $t=1500$ the costs of the blue route are quite different. The route costs that are used by StreamLine go up, while the route costs based on average speeds decrease. As a result, in the next iteration at $t=1200$ the first method will choose the red route, while the second method will choose the blue route. This pattern repeats every 1800 seconds, but this is not shown in the graphs.

This problem can be explained by evaluating the flows at the second iteration at $t=1500$, which are shown in figure 5.7. In this figure the upper route is the blue route in the graphs and the lower route is the red route. People travel from origin 1 to destination 2.

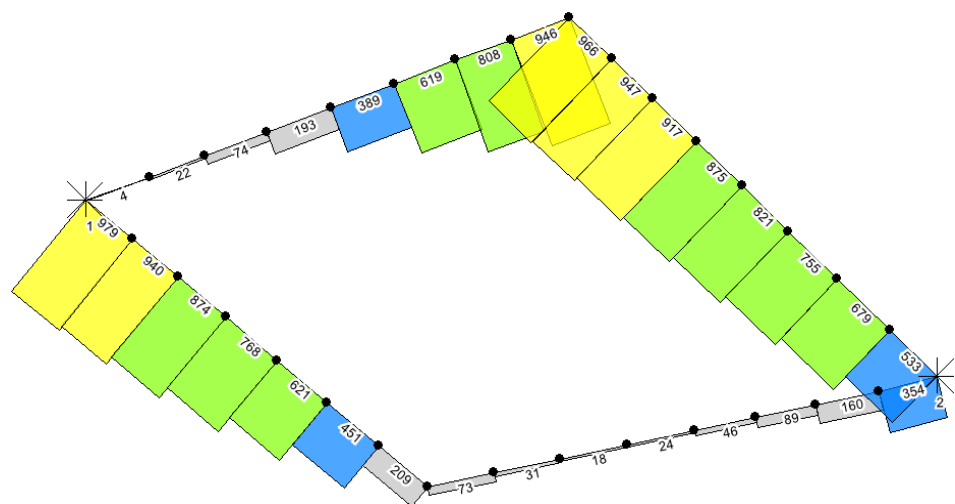


Figure 5.7 - Flows in the network at $t=1500$ in the second iteration. The upper route is the 'blue' route and the lower route is the 'red' route.

In the first iteration, at $t=600$ the upper route is chosen and at $t=1200$ the lower route is chosen. However, in figure 5.7 it is shown that there are still departing vehicles at $t=1500$. These are one of the last vehicles of the peloton that should depart at $t=1200$. Referring to example 5.3, it takes a long time before those vehicles reach their destination, which results in higher travel times. This explains why there are big differences in travel times between the two methods.

After 19 iterations an equilibrium is reached (when the duality gap is 0.001). At $t=1200$ the blue route has never been chosen. The travel times of the blue and the red route still differ much, but since the route fraction of the blue route is zero, the blue route is ignored. So, according to the definition of a DUE, an equilibrium is reached. However, the equilibrium would be different if the route costs were based on average speeds.

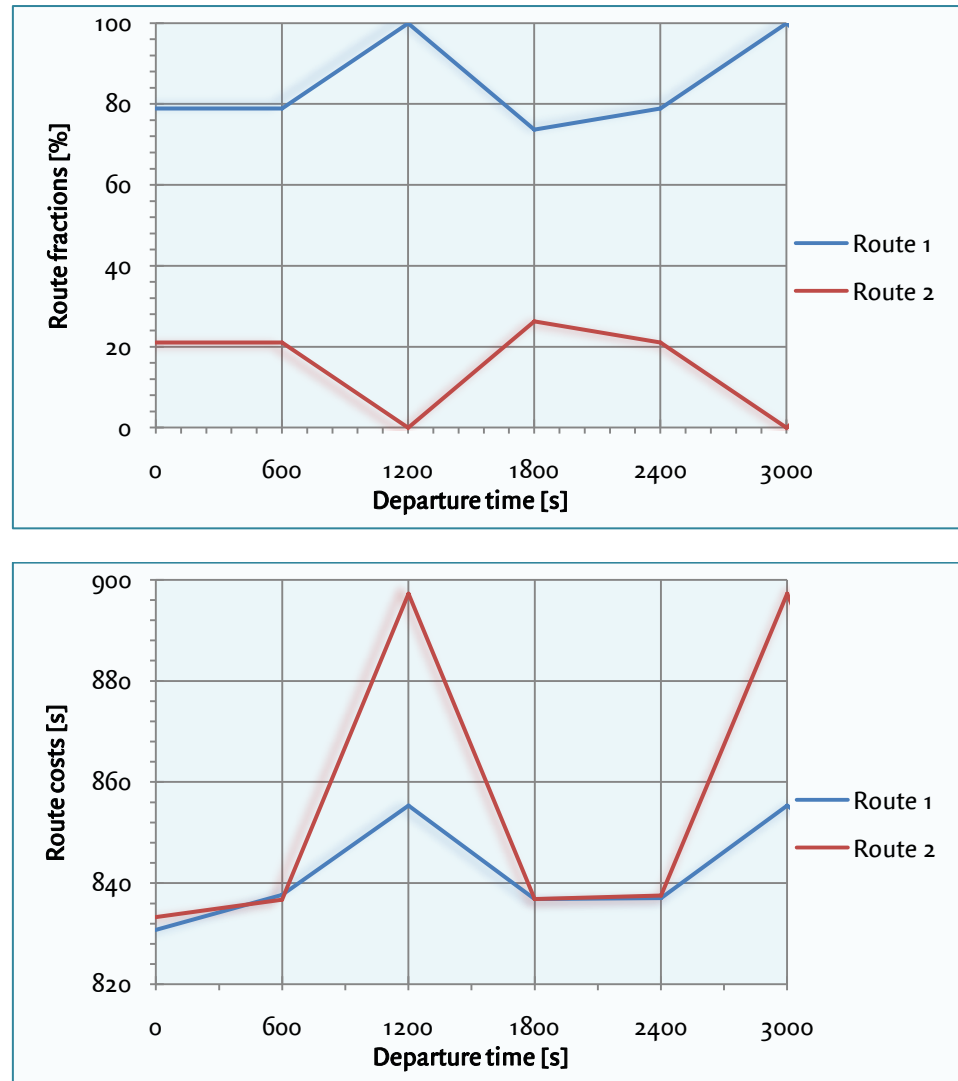


Figure 5.8 - Route fractions and route costs at equilibrium, iteration 19. The route costs are based on cumulative vehicles.

TravelTimeCalculator

In example 5.4 it becomes clear that the route costs that are calculated by StreamLine (using cumulative vehicles) can be different from the route costs that are calculated by the TravelTimeCalculator (using average speeds). Problems concerning the emptying of the network can be solved when StreamLine uses the route costs of the TravelTimeCalculator for determining the route fractions, since the speed of a link goes to the maximum speed when the density of a link is close to zero. However, using average speeds has disadvantages. When a connector link⁶ is congested, sometimes a queue arises inside the origin node. The TravelTimeCalculator ignores the time between the departure time and the time that a vehicle enters the network, which leads to lower route costs in comparison with using cumulative vehicles. Referring to figure 5.5 and 5.6, the route costs of both methods differ slightly (when the extreme results are ignored), but are close to each other. Research is needed to check whether the differences in route costs between the two methods are acceptable.

Recommendations

The problems that are mentioned above about the emptying of the network can also occur in larger and 'real' networks. When a successive all-or-nothing assignment is used, this problem can occur in the first iterations of the simulation, irrespective of the existence of multiple OD pairs. However, when multiple OD pairs are used, there is more overlap between routes, which makes the chance that no traffic is assigned to a link in the first iterations a bit lower.

In practice, most simulations will be done with a PCL assignment, because of the desirability of lower running times in large networks. PCL always assigns a small part of the traffic to each route, which makes it almost impossible that the flow on a link drops to zero during the simulation. This is only possible when, during the simulation, the demand of an OD pair changes to zero in the departure pattern of the traffic. This must be set by the user. For PCL assignments no problems are expected concerning the emptying of the network.

In other words, this problem will mainly arise with successive all-or-nothing assignments, to determine a deterministic equilibrium in a real network or for research motives. This is undesirable. To prevent this, it must be avoided that StreamLine uses low speeds when the density converges to zero. For example, when the situation occurs that the calculated density is lower than a specific value (say, 0.5 veh/km) *and* when the flow that is calculated by MaDAM is zero, the speed StreamLine uses must go to the free-flow speed. If this is implemented, it is possible that the calculated route costs by StreamLine are still

⁶ A connector link is a link that connects an origin node to the network.

some higher than the ‘real’ route costs (since the density still does not go to zero abruptly, according to the density formula in MaDAM), but the differences with reality will be much smaller than in the current case. However, this method has not already been tested. So, it is recommended to investigate this in a later research.

Besides, it is recommended to change the way StreamLine calculates route costs. Where StreamLine currently uses cumulative vehicles to determine route fractions, it is better to use route costs based on the average speeds that are calculated by MaDAM. Note that this leads to problems when a queue appears inside an origin node due to congestion. In StreamLine it is possible to show the length of the queue inside the origin nodes. With these data it is possible to calculate the time that a vehicle is queued before it can enter the network. This time needs to be added to the travel time that the TravelTimeCalculator calculates. Further research is needed to check to which extent the calculated travel times are close to the current travel times.

One may argue that the decreasing of the link segment length (L_m in formula 5.1) will result in a solution to the problem concerning the emptying of the network. When the segment length decreases, the anticipation term increases. This means that the speed on the link decreases faster and the density on the link reaches zero quicker. This is a good way to diminish the problem, but it *does not solve* the problem. Another disadvantage is that a decrease in segment link length results in longer simulation times. Besides, when the segment lengths in the network are decreased, it is possible that MaDAM overshoots a segment when the segment length is too short. It is wise to keep this in mind. When the results are updated every second, *and* when the maximum speed on a link is 130 km/h, this problem may arise when the segment length is shorter than 37 metres.

5.2.2 Links upstream a merge node

In some cases the speed calculation in MaDAM at links upstream a merge node is quite unrealistic. For example, when the flow on an entry link is low and the density on the exit link is high, the speed on the entry link may drop to the minimum speed. This situation is likely to happen in the first iterations of a series of all-or-nothing assignments, which is pointed out in example 5.5. Because of this, a series of all-or-nothing assignments converge to an equilibrium more slowly. The speed term in MaDAM is shown in formula 5.2 and 5.3

$$v_{m,i}(t+1) = v_{m,i}(t) + \text{relaxation} + \text{convection} - \text{anticipation} \quad (5.2)$$

$$\text{anticipation} = \frac{v \cdot T \cdot (\rho_{m,i+1}(t) - \rho_{m,i}(t))}{\tau \cdot L_m \cdot (\rho_{m,i}(t) + \kappa)} \quad (5.3)$$

Upstream a junction, it is possible that the density on the exit link is higher than the density on the entry link (which is also the case in example 7.5). As a result, the anticipation term in formula 5.3 will increase extremely, so that the total speed on that link (formula 5.2) decreases quite fast.

This phenomenon can have big consequences for the route choice StreamLine calculates. Due to low speeds in the network, it may take longer before all vehicles on that route have left the network. Otherwise, the travel costs of that route go up, which means that it influences route choice in StreamLine.

Example 5.5 – Speed drops on links near junctions

In a small network with one OD pair, there are two routes that only have overlap in the link downstream the destination origin. The maximum speed on the links is 50 km/h, on the last two links of each route the maximum speed is 30 km/h. From $t=0$ till $t=3000$ the demand on the lower route is 1000 vehicles per hour, from $t=3000$ the demand on the upper route is 1000 vehicles per hour. The results are shown in figure 5.9.

It this situation, the speed of the last link upstream the junction of the lower route drops to the minimum speed of 7 kilometres per hour. The flow on this link decreases while the flow on the outgoing link is relatively high, so that there is a large density difference between these links. Since the density difference does not differ much in time, the low speed on this link does not increase.

Since the speeds on the links upstream the junction are low, it takes a long time for a vehicle to pass that link. For instance, when the length of a link is 600 metres the travel time of that link is 309 seconds, while the travel time is 72 seconds when vehicles drive with the maximum speed of 30 km/h.

Recommendations

The problem that is described in this section, can lead to unrealistic results in larger and more complex networks. When a series of all-or-nothing assignments is used, large density differences can occur, mainly in the first iterations. Also in case of a PCL assignment, this situation is supposable. For example, a situation may occur that an on-ramp meets a highway. Here it is possible that the flow on the on-ramp is low and the flow on the highway is large. However, the speeds on a highway are in general much higher than the speeds in the current test network. As a result, it is less likely that the speed on the on-ramp decreases to the minimum speed.

To avoid problems which are caused by large density differences on consecutive links, the anticipation term in the speed calculation of MaDAM must be adapted. In the current anticipation term, the density difference influences the value of the anticipation term

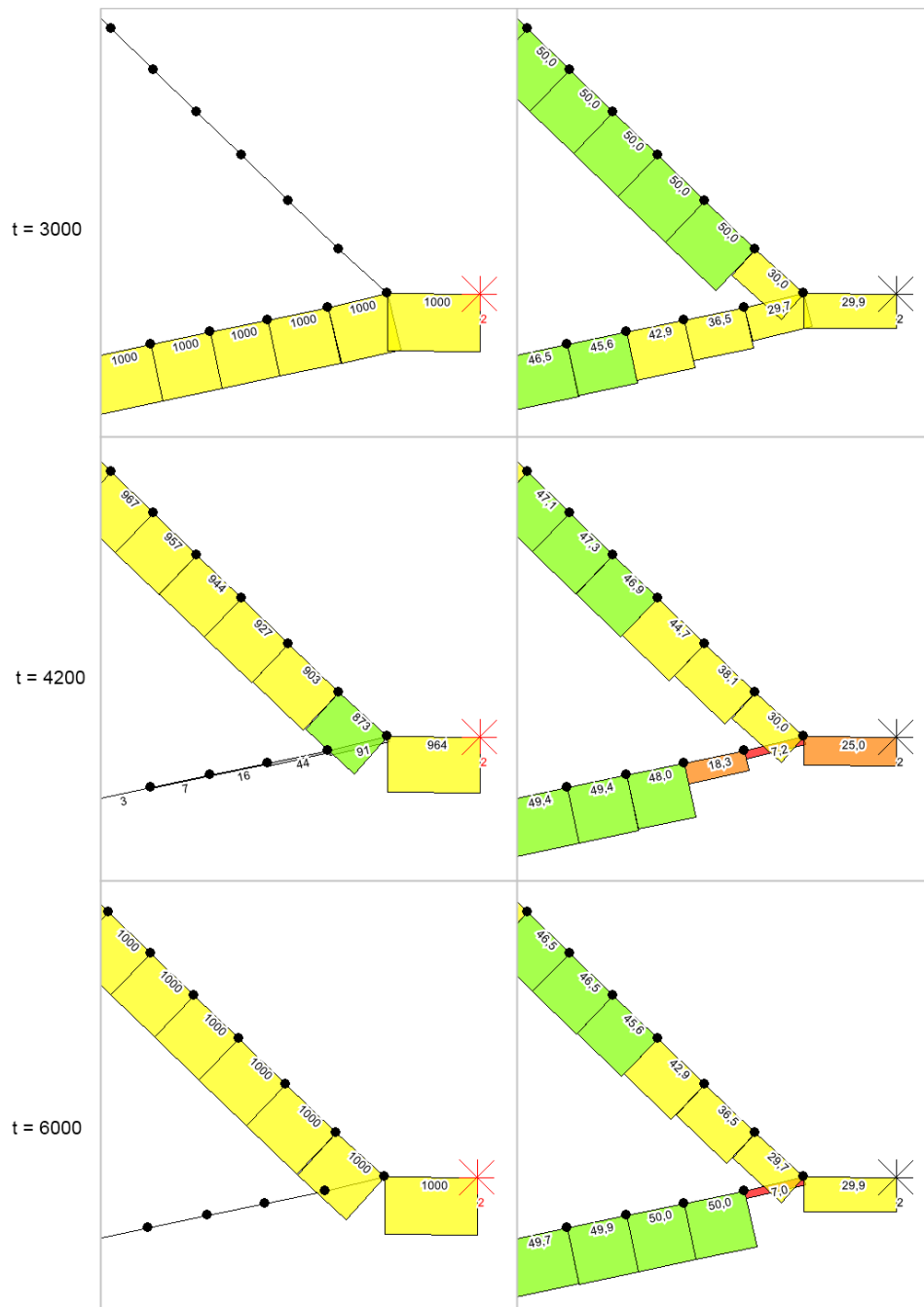


Figure 5.9 - Situation where two links merge to one link. Left: the flow on each link, right: the speed on each link.

much. However, when there are no large density differences in the network, the anticipation term seem to work well. It is recommended to add an extra factor to the anticipation term, α for example. When the density difference of two consecutive links is smaller than a specific value, the value is 1, when the density difference exceeds this value, α will be set to a smaller value (close to zero). An example of a suitable equation is shown in equation 5.4. This equation is not tested yet, it is advised to do this in a later research.

$$\text{anticipation} = \frac{v \cdot T \cdot (\rho_{m,i+1}(t) - \rho_{m,i}(t)) \cdot \alpha}{\tau \cdot L_m \cdot (\rho_{m,i}(t) + \kappa)} \quad (5.4)$$

It is possible that the implementation of this new anticipation term leads to instable behaviour, which is undesirable. For example, when the value of α is 1 when the density difference of two consecutive links is larger than 10 veh/km and α is 0.1 when the density difference is smaller than 10 veh/km, it is possible that problems arise with densities around 10 veh/km. Then, it is possible that in consecutive iterations the value of α continually flips between 1 and 0.2. In the research, there was not enough time available to check whether this is the case. It is advised to check this in a later research.

Another possibility is to create a new anticipation term, in case that the density difference is larger than a specific value. This is not included in the research.

On first sight, the problem that is described in this section should be explained by the way how the node is modelled, which is described in appendix III (Cross Node Modelling). This appendix describes how the traffic from the entry links is distributed over the exit links of a node. However, the problem concerning the low speeds upstream a merge node cannot be explained by Cross Node Modelling. In example 5.5, the speed on the link upstream the node on the lower route has reached the minimum speed, even when there is no traffic on this link any more. This proves that this problem is not caused by Cross Node Modelling.

5.2.3 Flow changes

In StreamLine, the demand of an OD pair is not continuous; it is only possible to enter demands for discrete time periods. As a result, it often happens that the flow on a link changes drastically, which results in unrealistic speed calculations. Referring to formula 5.2 and 5.3, big density changes have the consequence that the anticipation term in the speed formula will be either extremely positive or negative. When the flow on a link increases, the anticipation term becomes negative and the speed on that link goes up. In case of a flow decrease the same argumentation holds, but then the other way around.

Example 5.6 – Flow changes

On a single link with a maximum speed of 50 kilometres per hour, the demand during the simulation time is variable. In the first 30 minutes (1800 seconds) the flow from origin 1 to origin 2 is 500 vehicles per hour, in the second 30 minutes the flow goes up to 1000 vehicles per hour, in the third 30 minutes to 1500 vehicles per hour and in the last 30

minutes no traffic is departing. The speed on the cells after the flow change at $t=1800$ is shown in figure 5.10.

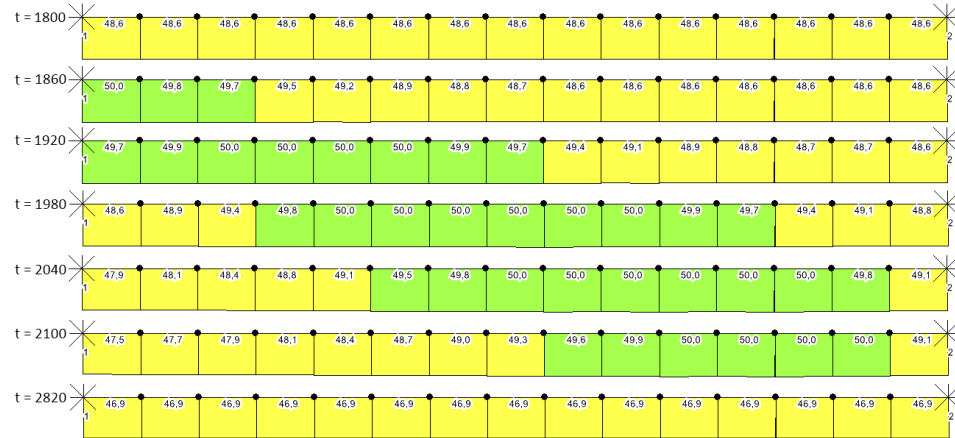


Figure 5.10 - Speeds in the network after a flow increase at $t=1800$.

Figure 5.10 shows that the speed on the cells goes up when the flow on the link changes. At $t=2820$ the speeds on the link does not change any more. Figure 5.11 shows to which extent these speed changes influence travel time. The results are calculated by the TravelTimeCalculator, which uses the average speeds on the links to calculate the travel time.

Figure 5.11 shows that travel time decreases when the flow on the network increases ($t=1800$ and $t=3600$) and increases in case of a decreasing flow ($t=5400$). The peak at $t=5400$ is larger than the peaks at $t=1800$ and $t=3600$, since the flow difference is three times higher at $t=5400$.

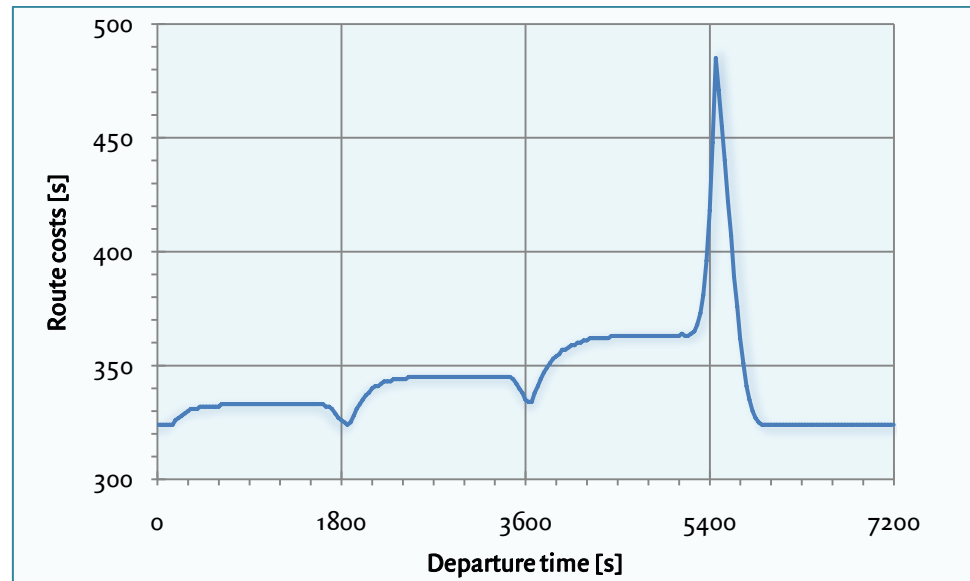


Figure 5.11 - Travel times during a simulation time with flow changes.

Example 5.6 shows that flow changes influence travel time and, therefore, it influences route choice. Therefore, it is possible that the equilibrium that is calculated in StreamLine deviates more from the equilibrium that arises in reality.

This effect arises with both PCL and all-or-nothing assignments. However, when a series of all-or-nothing assignments is used, it will lead to more problems than when PCL is used. With all-or-nothing, there are many flow changes within an iteration, especially in the first iterations of a series of all-or-nothing assignments. The same applies when the spread parameter in PCL has a low value. As a result, a successive all-or-nothing assignment converges slower to an optimal solution than a PCL assignment.

Recommendations

The flow changes that are discussed in this section do have consequences for ‘real’ networks, both for all-or-nothing and PCL. In every simulation, the user must enter a departure pattern in StreamLine, which results in flow changes. Relatively, this problem has fewer consequences for speed, and so on route choice, than the problems that are mentioned in section 5.1 and 5.2.

Equally to the situation with large density differences around junctions (section 5.2), this problem is caused by the anticipation term of the speed term of MaDAM. Again, the problem can be solved by changing the anticipation term. The same solution as in section 5.2.1 can be used.

5.3 Adaptations to the network

The previous sections describe the limitations of MaDAM and PCL. These factors prevent that StreamLine determines a realistic equilibrium in the current network. To solve these problems, several adaptations are made to the current network.

5.3.1 All-or-nothing assignments

The default route choice model in StreamLine is PCL. In this research, a series of all-or-nothing assignments is used, to prevent that at least a small fraction of traffic will be assigned to every route alternative. As a result, it is necessary to increase the number of iterations, since it takes longer before the route costs of route alternatives converge.

5.3.2 Urban links

To prevent that the speed on links upstream junctions drop to the minimum speed, the characteristics of these links are changed. In StreamLine, a distinction is made between motorway links (default) and urban links. In the speed calculation of MaDAM, a different anticipation term is used for urban areas, since the traffic behaviour of people is quite different in comparison with the traffic behaviour on motorways in terms of acceleration

and deceleration. In urban areas, the links in succession have a lower weight in the speed calculation. The links in the network that are changed to urban links are marked in blue in figure 5.12⁷.

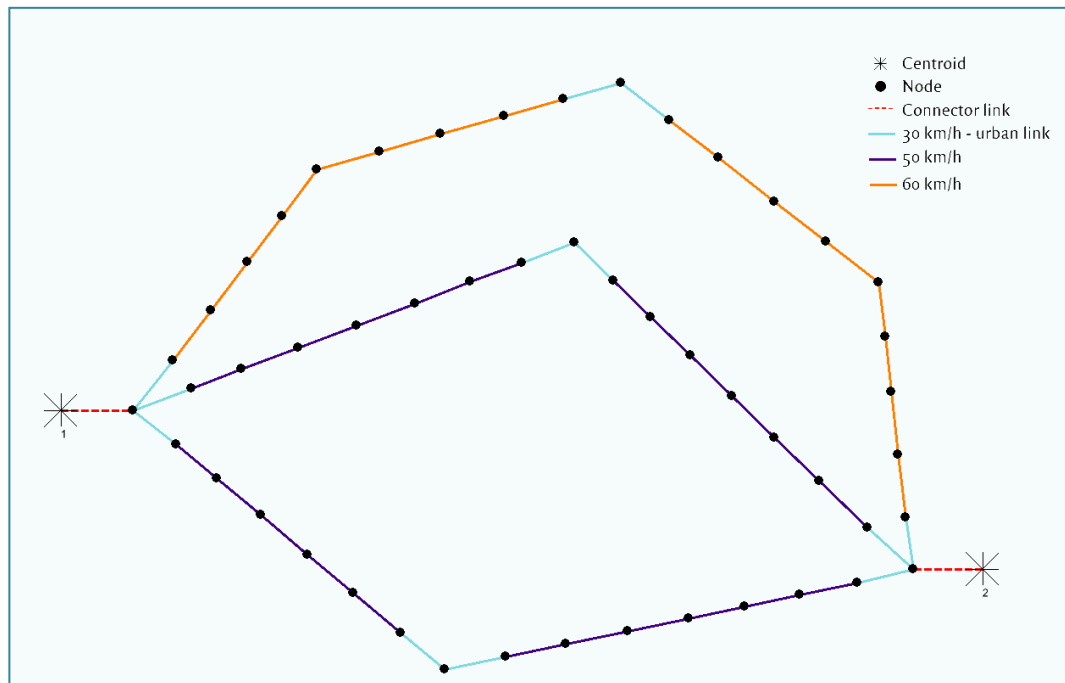


Figure 5.12 - Overview of the speeds in network 1. The blue links are urban links.

5.3.3 Background traffic

In section 5.2.1, it is stated that the emptying of the network can cause problems in determining a DUE. Since the slow emptying of the network only occurs when the flow on a link drops to zero, it is chosen to put a small amount of traffic of 5 vehicles per hour on each route during the whole simulation. As a result, the total flow on a link will not drop to zero, which makes it possible that StreamLine can handle with this problem in this research.

The new network that is used, is shown in figure 5.13. Here, three origins and three destinations are added to the network to provide all routes with background traffic. Each OD pair (i.e. 5-8, 6-9 and 7-10) has a different route alternative (which is set beforehand) with a constant flow of 5 vehicles per hour. The OD pair 1-2 is used for the original assignment.

⁷ In the middle of the routes, there are also two urban links, but these links are not close to a junction or merge node. This is done to make it possible to expand the network with an extra OD pair to make the network slightly more complex. There was not enough time available in the research to investigate this.

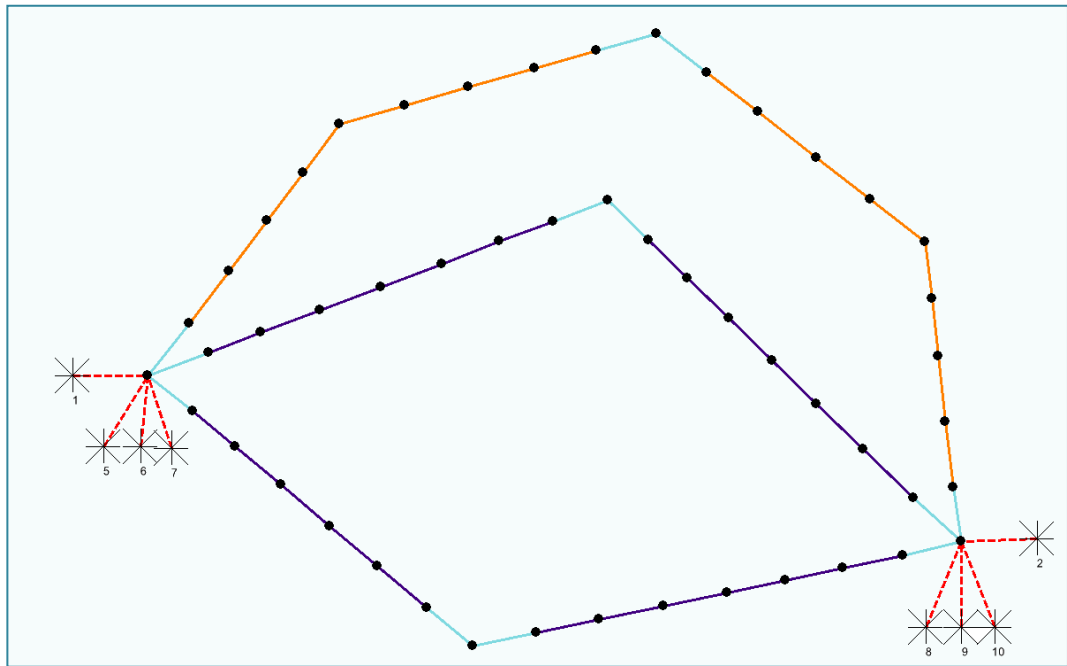


Figure 5.13 - To make it possible to assign background traffic, three extra OD pairs are added to the network; one OD pair for each route.

6 Equilibrium run

When the adaptations to the network are made, a successive all-or-nothing assignment is done with a route choice interval of 60 seconds and 10,000 iterations. For the accuracy it would be better to use a route choice interval of 1 second, since the route fractions are updated every second then. However, for this research it is too time-consuming to do this (regardless of the small size of the network). The current equilibrium run took 2 days running time, which means that a run with a route choice interval of 1 second will take 120 days (which is 4 months). It is assumed that a route choice interval of 60 seconds will give results that are accurate enough for this research.

This equilibrium run is the benchmark for the last part of the research. The results after the simulation are displayed in figure 6.1 and 6.2.

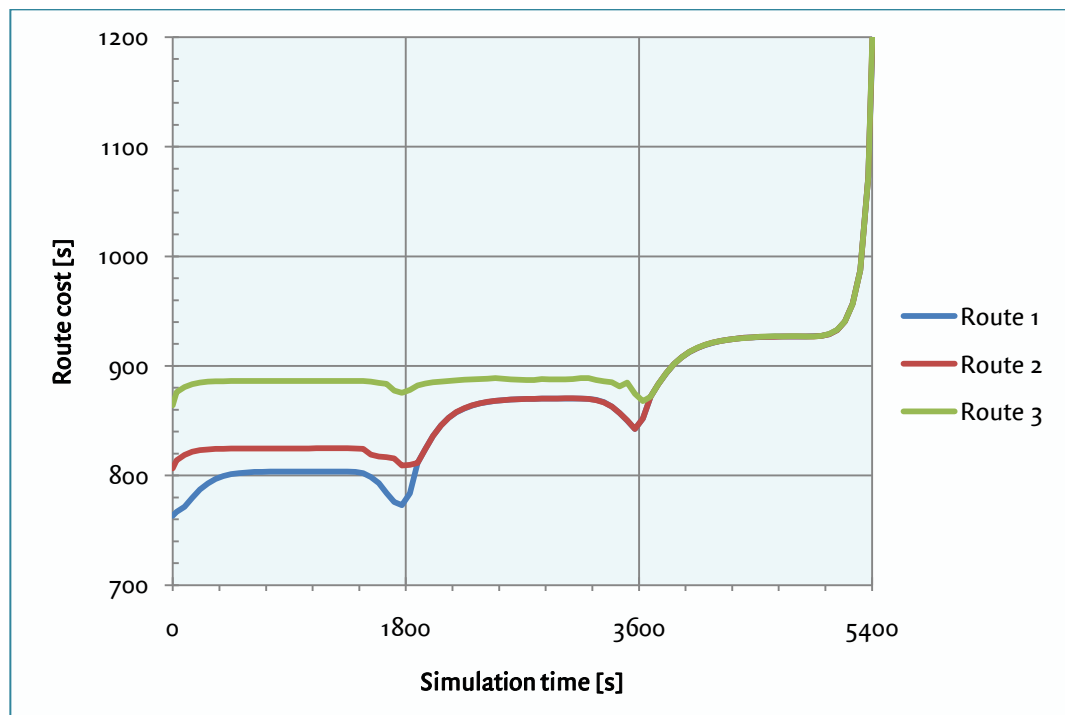


Figure 6.1 - Route costs at equilibrium

The results of the simulation are quite close to equilibrium. At each route choice moment, the difference in route costs of the used routes is negligible. In the first time period (from 0 till 1800 seconds) only the blue route is chosen, in the second period (from 1800 till 3600 seconds) the blue and the red route are chosen, while in the third period all three routes are used.

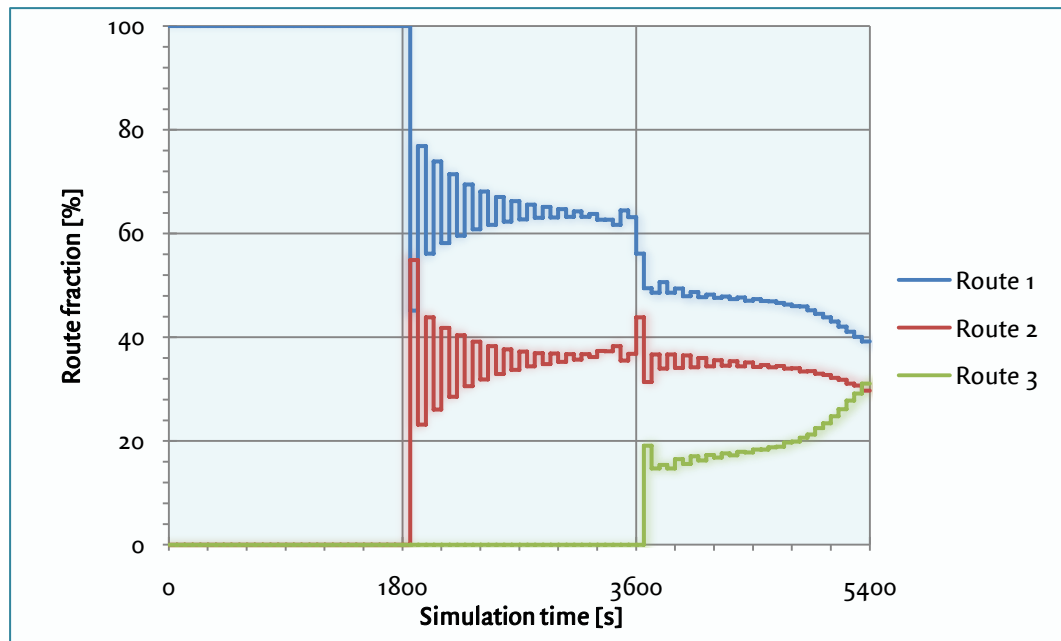


Figure 6.2 - Route fractions at equilibrium

To check how close these results are to equilibrium, the difference between the maximal and minimal route costs is calculated at each route choice moment between $t=0$ and $t=5400$. The average difference is 0.092 seconds, while the maximal difference is 0.4919 seconds. Relatively spoken, the average and maximal deviation are respectively 0.00010% and 0.00048% of the average route cost at the corresponding route choice moment. Based on these results it is assumed that a DUE is reached. This equilibrium is used as a benchmark in the rest of the research. When the number of iterations is increased, these deviations will probably be lower, but in this research it is not necessary to investigate.

Despite that an equilibrium is reached, there are still unrealistic peaks in the route cost graph (figure 6.1) when the total flow in the network changes (at $t=1800$, $t=3600$ and $t=5400$). To prevent this problem, adaptations have to be made in the speed calculation of MaDAM and this adaptation has to be changed in the code of StreamLine. Since this is quite time-consuming, this subject is not included in the research. In the current network, this phenomenon does not influence route choice extremely.

The graphs in figure 6.2 show some oscillation. From $t=1800$, the route fractions of two successive differ strongly from each other. This is caused by the small route choice intervals that are used in the simulation. Since the route choice interval is small, extreme route fractions hardly influence the continuation of the simulation. After all, a new route choice distribution is calculated by StreamLine 60 seconds later. Example 8.1 in section 8.1 gives more information about the flip-flop effect or oscillation effect.

7 Iteration and rci analysis

This chapter investigates to which extent the number of iterations in the simulation and the length of the route choice interval (rci) influence the equilibrium. In section 7.1, the number of iterations is varied, while the length of the route choice interval is varied in section 7.2. Section 7.3 combines these results and gives a short conclusion about these variables.

7.1 Number of iterations

The number of iterations in a simulation influences the extent to which route cost differences converge. To look to which extent number of iterations influence the convergence rate, the same simulation as in the previous chapter is run, and the results after a specific number of iterations are written down in the table below. This gives insight into the distance that the results are away from equilibrium. In this table, the average route cost difference is the average absolute difference between the maximal route cost and the minimal route costs of used routes at each route choice moment. At the average difference with the equilibrium, at each route choice moment, the route costs are compared with the route costs after 10,000 iterations (which is assumed to be an equilibrium). Both the absolute difference and the relative difference are shown in table 7.1 and figures 7.1 and 7.2.

Number of iterations	Average route cost difference [s]	Average difference with equilibrium [s]	Average difference with equilibrium [%]
20	25.564	11.248	1.253
50	16.133	6.793	0.752
100	11.255	4.875	0.540
200	5.182	2.125	0.233
500	1.833	0.778	0.085
1,000	1.047	0.445	0.049
2,000	0.540	0.223	0.024
5,000	0.246	0.105	0.011
10,000	0.092	0.039	0.004

Table 7.1 - Relationship between the number of iterations of a simulation and the average route cost difference and the average difference with the equilibrium run.

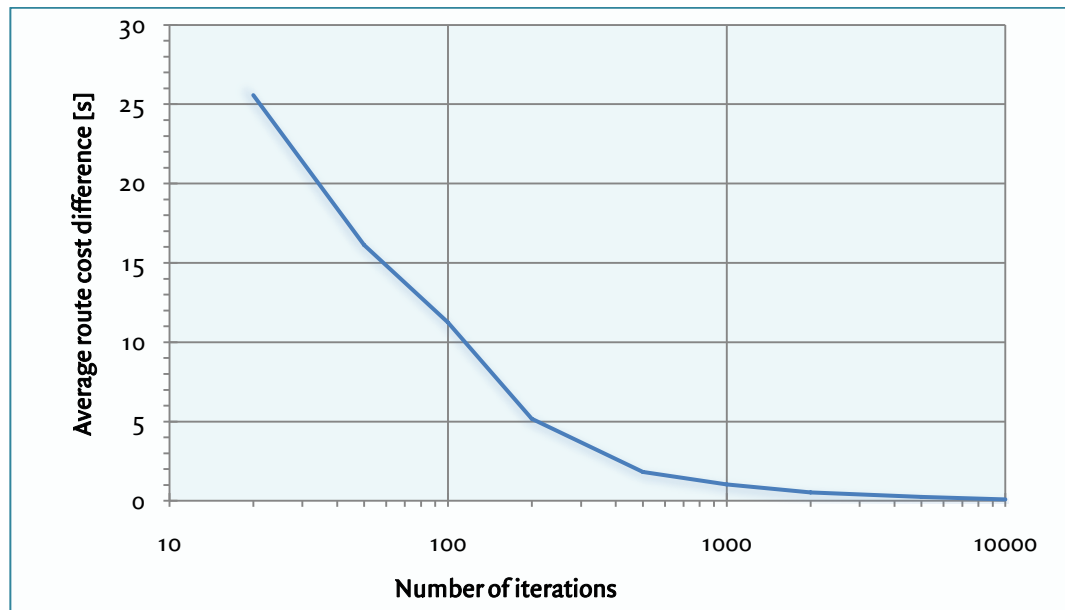


Figure 7.1 - Relationship between the number of iterations and the average route cost difference. Network 1, route choice interval = 60 seconds.

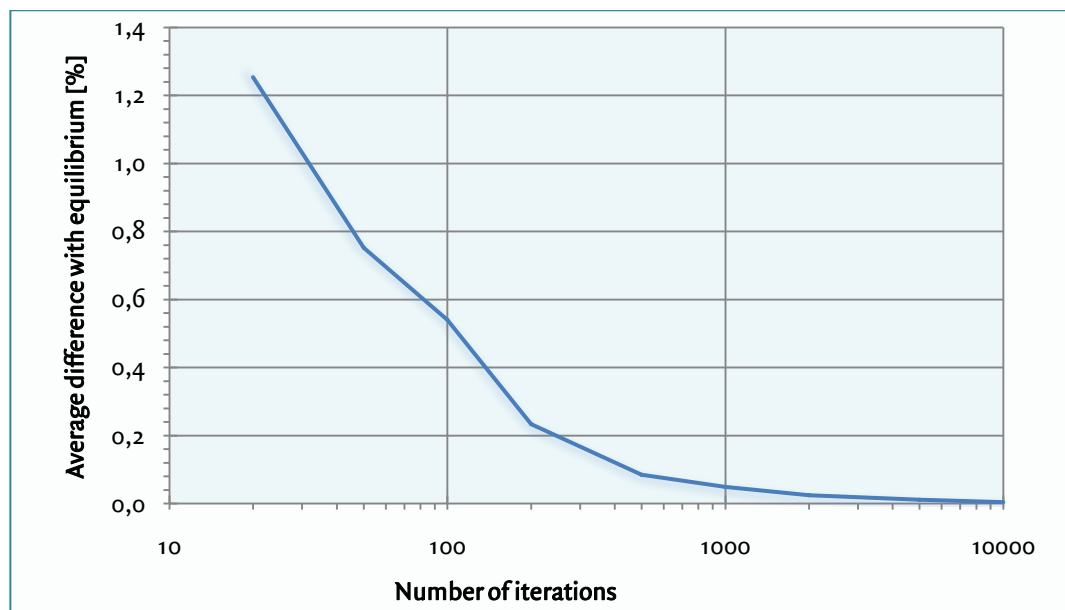


Figure 7.2 - Relationship between the number of iterations and the average difference with the equilibrium. Network 1, route choice interval = 60 seconds.

As expected, the average route cost difference decreases when the number of iterations goes up. After 1,000 iterations, the route costs of the different routes deviate slightly more than 1 second from each other. After 10,000 iterations, the route costs still converge and the route cost difference is quite close to zero. When figure 7.1 is considered, it becomes clear that the average route cost difference decrease more slowly when more

iterations are run. In addition, on the x-axis of figures 7.1 and 7.2, a logarithmic scale is used, which makes it more difficult to observe this effect.

When considering the average difference with the equilibrium run, the relationship with the number of iterations is for a large extent similar to the relationship that is described above. The average difference with the equilibrium decreases when the number of iterations increase and it increases more slowly when more iterations are done. In this situation, this decrease is quite explainable, since the results are compared with the same simulation at 10,000 iterations. As a result, the average difference with the equilibrium at 10,000 iterations is almost zero.

The results in this section depend strongly on the complexity of the network that is used. The more complex a network is – in terms of number of OD pairs, overlap of routes, amount of congestion – the longer it should take until StreamLine reaches an equilibrium⁸. In the current network, only three routes without overlap are used, which means that the route costs converge quite fast. As a result, in case of a more complex network, the graph in the figures 7.1 and 7.2 are less steep than in the current network.

In section 7.3, the results of the number of iterations and the different route choice intervals are combined.

7.2 Route choice interval

As mentioned in section 2.2, a route choice interval is the period between two route choice moments. During a route choice interval the route choice distribution among routes does not change. To check the influence of the number of route choice moments in a simulation, the length of the route choice interval is varied. The results of the simulation are shown in table 7.3.

Since the route choice interval is varied, the number of route choice moments of each simulation is different. To make it possible to compare two equilibria with different route choice intervals, all the route choice moments of a simulation have to be corresponding with some of the route choice moments in the equilibrium situation. Therefore, the used route choice intervals can be divided by 60.

In section 3.1.2, it is explained that the route choice at a route choice moment is based on the comparison of the route costs in the middle of the route choice interval. To make a fair comparison between different equilibria, the route costs at the same time moment

⁸ This is an assumption; it is not tested whether this is the case. In case of many OD pairs, there may be much overlap of routes, so that it is possible that StreamLine reaches an equilibrium faster when the network is more complex. However, this is not likely.

are compared. This means, for example, that when the route choice interval is 300 seconds, the used costs at route choice moment $t=0$ are compared with the used costs at route choice moment $t=120$ in the equilibrium run. In both cases, StreamLine uses the route costs at $t=150$ ⁹.

When the route choice interval is either 600 or 1,800 seconds, the midpoints of the route choice intervals do not correspond with the midpoints of the route choice interval of the equilibrium situation. Therefore, in simulations with these route choice intervals, the first route choice moment is at $t=30$ instead of $t=0$. Table 7.2 shows the used route choice moments and midpoints of route choice intervals.

Route choice interval [s]	Route choice moments [s]	Midpoints of route choice intervals [s]
60	0, 60, 120, ... , 5340, 5400	30, 90, 150, ... , 5310, 5370
180	0, 180, 360, ... , 5220, 5400	90, 270, 450, ... , 5130, 5310
300	0, 300, 600, ... , 5100, 5400	150, 450, 750, ... , 4950, 5250
600	30, 630, 1230, ... , 4830, 5430	330, 930, 1530, ... , 4530, 5130
1,800	30, 1830, 3630, 5430	930, 2730, 4530

Table 7.2 - Overview of used route choice moments and midpoints of route choice intervals at different route choice intervals.

The results of the simulation after 10,000 iterations are shown in figures 7.3 and 7.4.

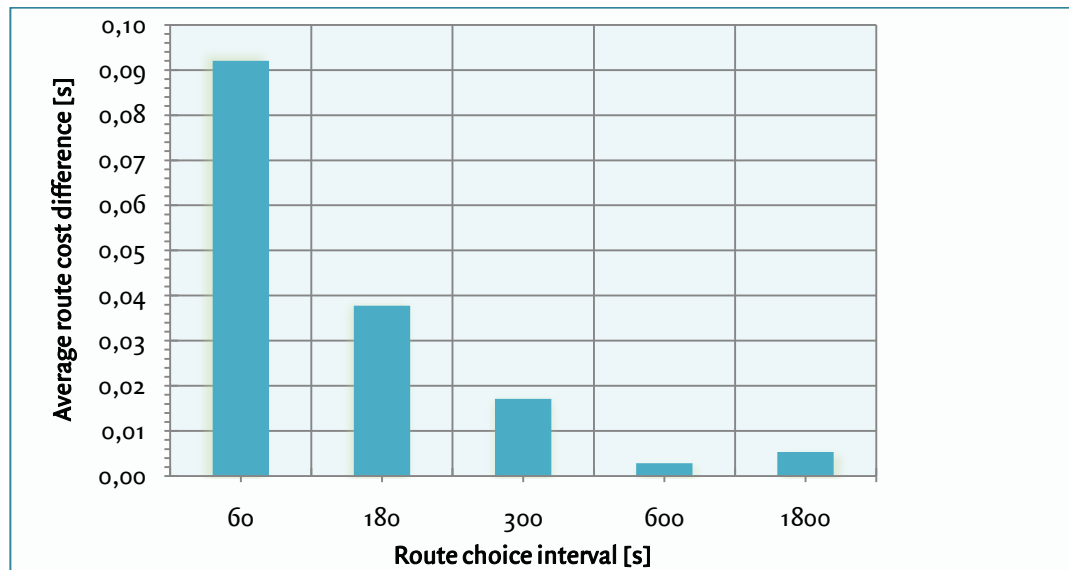


Figure 7.3 - Relationship between the average route cost difference and the route choice interval. Simulation with 10,000 iterations.

⁹ It does, since $0+(300/2)=150$ and $120+(60/2)=150$.

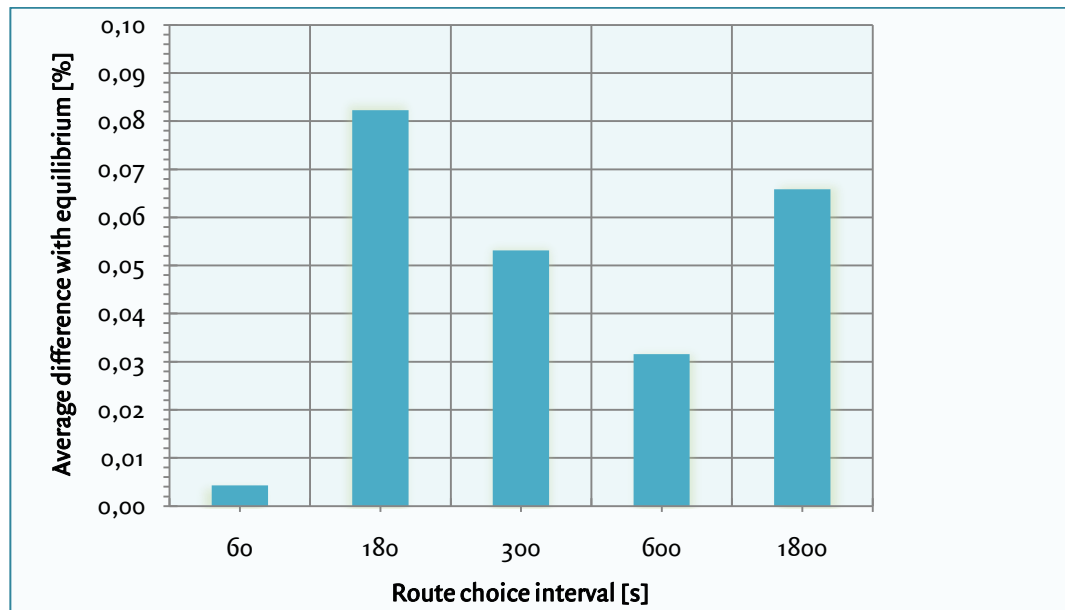


Figure 7.4 - Relationship between the average difference with the equilibrium ($rci=60$, $it=10,000$) and the route choice interval. Simulation with 10,000 iterations.

Looking at the average route cost difference in figure 7.3, it becomes clear that the route costs are closer to each other when larger route choice intervals are chosen. There are two explanations for this. Firstly, when the route choice intervals are small, there are many route choice moments where the route costs of the used routes have to be optimised. For comparison, at $rci=60$, 90 route choice moments are optimised, while the route costs at $rci=1800$, only 4 route choice moments have to be optimised. As a result, the larger the route choice interval, the faster the route costs converge.

Besides, in case of small route choice intervals, the distribution at a specific route choice moment does not only influence the route choice in the current route choice interval, but also the route choice in the next route choice intervals. In other words, the traffic that departs at a specific route choice moment has not reached its destination at the end of the route choice interval. As a result, it takes longer before route costs converge. In contrast, when a route choice interval of 600 or 1800 seconds is chosen, the traffic that departs at the beginning of the route choice interval has completely or almost left the network at the end of that route choice interval. In this case, the calculated route distribution is only used to optimise the route costs in the current route choice interval. This leads to smaller differences in route costs.

The graph that displays the difference with the equilibrium (figure 7.4) is more variable. In this example, at $rci=60$, the difference with the equilibrium is quite small. This is logical, since this is the run that is assumed to be the equilibrium, so the run is compared with itself.

It is expected that the results are more accurate when the route choice interval is smaller, since there are more route choice moments where the route distribution can change. However, this graph does not show this relationship. This is caused by the fact that the route costs are only compared with the equilibrium at the route choice moments. It is not possible to easily compare the route costs at other moments during the simulation in StreamLine. This means that it is possible that there is a small difference with the equilibrium run at the route choice moments, but that there is a larger difference in the period between two route choice moments. For example, when there are route choice moments at $t=60$ and $t=120$, it is not possible to compare the route costs at $t=90$.

According to the graph, the route costs are the closest to the equilibrium run when $rci=600$. It seems that the route choice interval is too large at $rci=1800$, which makes that the results are less accurate.

In section 7.3, the results of the number of iterations and the different route choice intervals are combined.

7.3 Number of iterations and route choice interval combined

Section 7.1 and 7.2 varies both the number of iterations and the route choice interval. In this section, all results are combined and are put in a 3-dimensional graph. Figures 7.5 and 7.6 show the results.

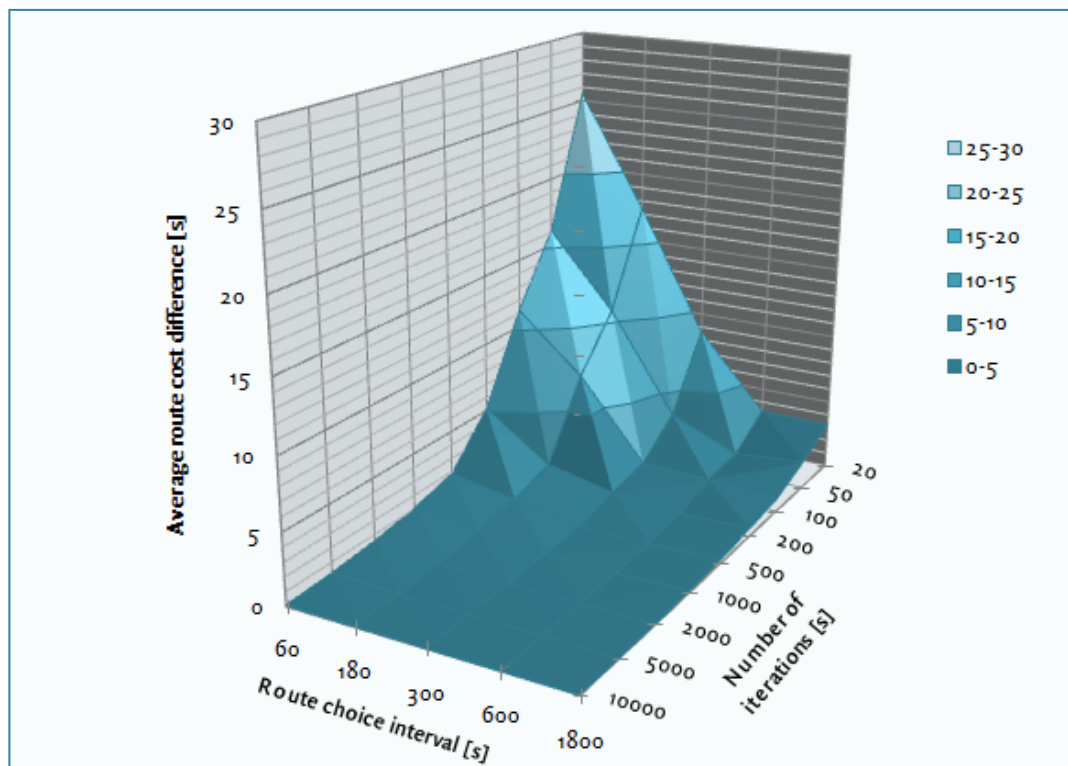


Figure 7.5 - Relationship between the average route cost difference, the route choice interval and the number of iterations. Compared with the equilibrium run with $rci=60$ and $it=10,000$.

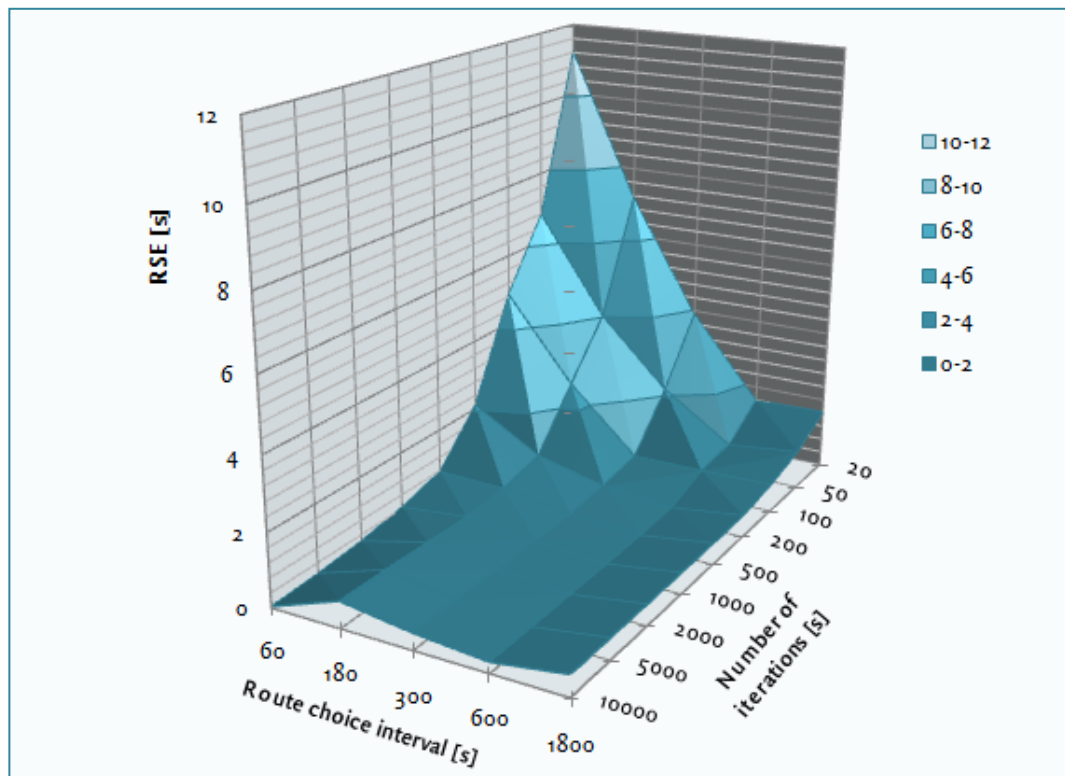


Figure 7.6 - Relationship between the average difference with the equilibrium (RSE), the route choice interval and the number of iterations. Compared with the equilibrium run with rci=60 and it=10,000.

Figure 7.5 shows the average route cost difference, which is the degree of convergence. In this graph, it becomes clear that irrespective of the chosen route choice interval, the average route cost difference decreases when the number of iterations goes up. However, the larger the route choice interval, the less iterations are needed to reach the same accuracy. As already discussed in section 7.2, this is because the number of route choice moments where the route costs have to be optimised is smaller when a larger route choice interval is chosen; and the influence of a route distribution on other route choice intervals is minimal.

Figure 7.6 shows a clear relationship between the RSE, the length of the route choice interval and the number of iterations. The RSE decreases when the number of iterations increases or when the route choice interval increases. In other words, the route costs deviate more from the equilibrium run when a small route choice interval is chosen. The results of rci=60 are somewhat deviant, especially when the number of iterations are more than 1000. As already mentioned in section 7.2, this occurs because the route costs are compared with the same simulation run.

Referring to the results above, different route choice intervals and number of iterations have advantages and disadvantages. Small route choice intervals give better results; after

all, there are more route choice moments and so, the route distribution is updated more often during the simulation to give a more optimal result. As a result, the results will correspond better with route choice in reality. However, a big disadvantage of small route choice intervals is that StreamLine needs more iterations to reach an equilibrium. Besides, the running time of a single iteration is quite higher, since there are more route choice moments for which a route distribution must be calculated. In case of an all-or-nothing assignment, the flip-flop effect increases. Example 8.1 at section 8.1 explains this phenomenon in more detail.

On the other side, an advantage of a large number of iterations is that the route costs of the used routes are more converged than when a small number of iterations is used. The results are closer to an equilibrium. A big disadvantage is that the running time of the total simulation increases, and particularly in large and complex networks it can lead to extreme running times.

Before running a simulation, a good consideration must be made about the number of route choice moments and the number of iterations. The network that is used in this research is quite simple, which makes it possible to run with many iterations and short route choice intervals. In large networks with many OD pairs, many routes and much overlap between routes, this is a hopeless task in consideration of long simulation times.

In chapter 9, different parameters are varied. Since it is needed to run several simulations, it is wise to avoid that the total running time of the simulations will be too large. Based on figure 8.5 and 8.6, it is chosen to use a route choice interval of 600 seconds and to use 500 iterations. In this combination, the average route cost difference is 0.17 seconds and the RSE is 0.32 seconds. These values are very small and are quite close to equilibrium. As a result, one simulation will be 200 times faster than the equilibrium run with 10,000 iterations and a route choice interval of 60 seconds.

8 Parameter analysis

In the first network, three parameters are varied. Section 4.5 describes the parameters that are chosen in this research. The following enumeration repeats these parameters.

- *Tau*; this parameter influences the relaxation term and the anticipation term in the speed calculation in MaDAM. Since a higher value of tau leads to a lower relaxation and anticipation term, and these effects counterbalance each other partially, it is expected that changes in tau will not fundamentally change the dynamic user equilibrium.
- *Nue*; this parameter influences the anticipation term in the speed calculation. An increase in nue should result in a decrease of speed and an increase of travel time.
- *Kappa*; like nue, this parameter influences the anticipation term in the speed calculation. An increase of kappa should lead to an increasing travel time.

The parameters *delta* and *minimum speed* are also mentioned in section 4.5, but are not analysed in this network. Delta influences the speed term at merges. Since there are no merges and junctions in the first network, delta should have no influence on travel time, and so, on the equilibrium. The minimum speed is also not analysed here, since no congestion occurs in this network. The demand on the network is lower than the total capacity of the network (5400 veh/h), so that it is very unlikely that the speed on a link in the network drops to the minimum speed. As a result, other values of the minimum speed will not influence the equilibrium.

For tau, nue and kappa, five measures are calculated. The abbreviations are shortly repeated and explained.

- *Avgdiff*; the average route cost difference over all route cost moments. This measure gives a good impression of the degree of convergence in the simulation. The closer *avgdiff* comes to zero, the more the average routes costs are equal and the closer the simulation is to an equilibrium.
The convergence criterion that is used in StreamLine is called the ‘duality gap’ (section 3.1.5). It is chosen to use *avgdiff*, because its formula is far less complex than the formula of the duality gap. *Avgdiff* is easier to use and simple to understand.
- *RSE*; the relative standard error (section 4.4). This is the relative difference between the travel times at the current equilibrium and the reference equilibrium with default parameter settings. The closer *RSE* comes to zero, the lower the difference between the two equilibria.

- *RseEL*; the RSE elasticity of the parameter. The higher the value, the more sensitive the equilibrium is to parameter changes. A positive elasticity means that an increase of a parameter leads to an increase of the RSE; in case of a negative elasticity it is the other way around.
- *AvgCost*; the average route costs of all route choice moments. This measure gives an indication of the influence of a specific parameter setting on the average route cost.
- *AvgCostEL*; the average cost elasticity of the parameter. The higher the value, the more the average cost changes due to parameter changes.

8.1 Network 1

The results of the variation of tau, nue and kappa are shown in table 8.1. The first column of the table shows the values of the varied parameters. In the last two columns the elasticities are shown.

Tau	avgdif	RSE	rseEL	avgCost	avgCostEL
5	0.232	0.2278	-6.5942	860.57	0.0013
10	0.191	0.1759	-7.1793	860.49	0.0022
20	0.171	0.0383		861.44	
40	0.192	0.6861	16.9058	867.47	0.0070
80	0.037	3.2275	27.7450	889.28	0.0108

Nue	avgdif	RSE	rseEL	avgCost	avgCostEL
0	0.117	1.9079	-48.7956	844.79	0.0193
17.5	0.164	1.2784	-64.7333	850.39	0.0257
35	0.171	0.0383		861.44	
70	0.247	3.0020	77.3520	888.30	0.0312
140	17.354	9.3089	80.6524	943.33	0.0317

Kappa	avgdif	RSE	rseEL	avgCost	avgCostEL
1	2.557	3.0857	-86.1633	888.10	-0.0335
7	0.163	1.3207	-72.5149	873.50	-0.0304
13	0.171	0.0383		861.44	
19	0.136	0.5280	27.6896	857.08	-0.0110
25	0.225	0.8332	22.4750	854.38	-0.0089

Table 8.1 - Results of the variation of tau, nue and kappa.

8.1.1 Analysis network 1

Convergence

The degree of convergence gives information about the distance that a simulation is removed from an equilibrium. When the values of *avgdiff* in table 8.1 are considered, it becomes clear that in the largest part of the simulations the route costs converge to an equilibrium. After 500 iterations, the average difference between the maximal and minimal route cost at a route choice moment is smaller than 0.25 seconds at 13 of the 15 simulations. In two simulations the *avgdiff* is quite larger; at $nue=140$ the average difference is more than 17 seconds after 500 iterations, and at $kappa=1$ the average difference is 2.6 seconds.

The outlier at $nue=140$ can be explained and strongly depends on the network that is used. In the first 30 minutes of the simulation, the demand on the network is quite low, so that in most of the simulations all travellers choose the shortest route. However, when nue is set to 140, the route costs of the shortest route increase faster, which the result that also the second shortest route is chosen in the first 30 minutes of the simulation. In the second 30 minutes of the simulation, the same situation comes forward; here, the most expensive route is chosen, while in other simulations this is not the case. See example 8.1.

Relative standard error

For all three parameters, the relative standard error becomes higher when the value of the parameter diverges more from the default value of the parameter. This is explicable, since the relative standard error with default parameters is quite low (0.0383), which means that this simulation does not deviate much from the equilibrium run. When the average route costs change due to parameter changes, it is logical that the value of the relative standard error becomes larger. However, this does not mean that a simulation is worse or less realistic than the simulation with default settings. A big change in the relative standard error says more about the degree in which the equilibrium changes in relation to the equilibrium with default settings.

In case of τ , the elasticity of the relative standard error becomes larger when τ becomes larger. This means that the equilibrium changes relatively more when larger values are awarded to τ . The opposite conclusion can be drawn for $kappa$; the lower the value of $kappa$, the faster the equilibrium changes. The elasticity of a larger $kappa$ is approximately three times higher than when a smaller $kappa$ is used. For τ the same relation exists, but vice versa. The elasticities of the RSEs of nue differentiate in a less degree. The equilibria deviate slightly more when the values of nue increase.

Route costs

When τ is considered, the average route costs in the network go up when the value of τ increases. However, the average cost elasticity is quite low, especially when these values are compared to the average cost elasticities of nue and κ . As mentioned earlier, τ influences both the relaxation term and the anticipation term of the speed calculation in MaDAM. Since the average cost elasticity of τ is positive, it is likely that the influence of τ on the anticipation term is bigger than the influence of τ on the relaxation term.

There is also a positive relationship between nue and the average route costs, since the elasticities of the average costs are positive for all values of nue . This finding is conformable to the expectation that the average route costs go up when nue decreases. The differences in route costs are relatively larger when nue has a higher value, but these differences are relatively small.

Besides, the hypothesis that the average route costs decrease when the value of κ increases, can be confirmed. However, the sensitivity of the route costs is larger when κ decreases. This means that small values of κ influence the route cost, and so the equilibrium, more than large values of κ .

Example 8.1 - $nue=140$, flip-flop effect

The route fractions at $nue=140$ are shown in figure 8.1. In this graph, the flip-flop effect becomes noticeable. Since the route choice interval in this simulation is quite high (i.e. 600 seconds), a route fraction change at a specific route choice moment influences the

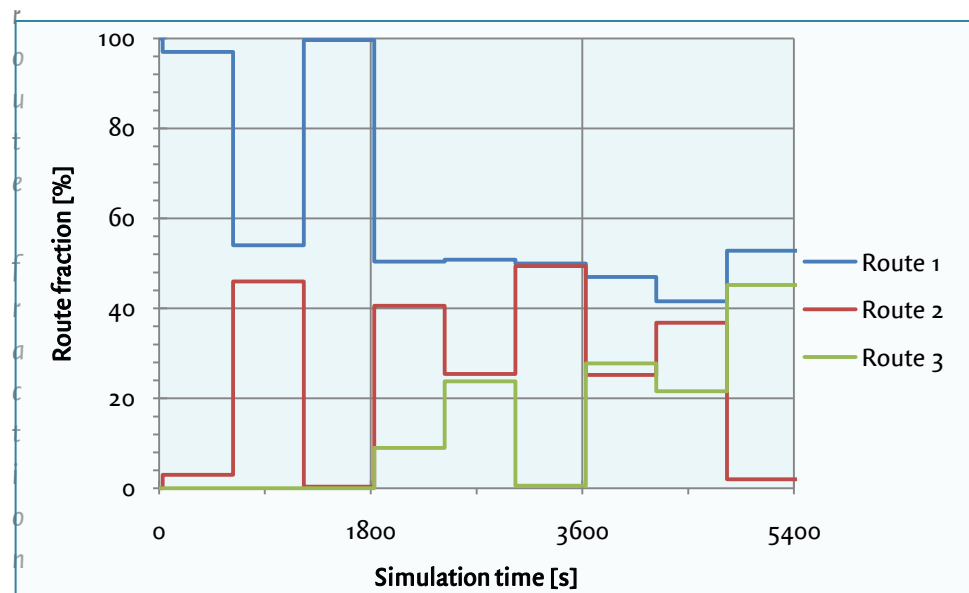


Figure 8.1 - Simulation run in network 1, $nue=140$, route fractions.

in the rest of the simulation relatively more than when a smaller route choice interval is chosen. At the first route choice moment, a large route fraction is assigned to the blue route (97%). As a consequence, at the second route choice moment the blue route becomes more expensive and more than 40% of the traffic is assigned to the red route. This behaviour affects the whole simulation.

By contrast, figure 8.2 shows the route fractions at the default value of nue . Here, the route fractions change more gradually.

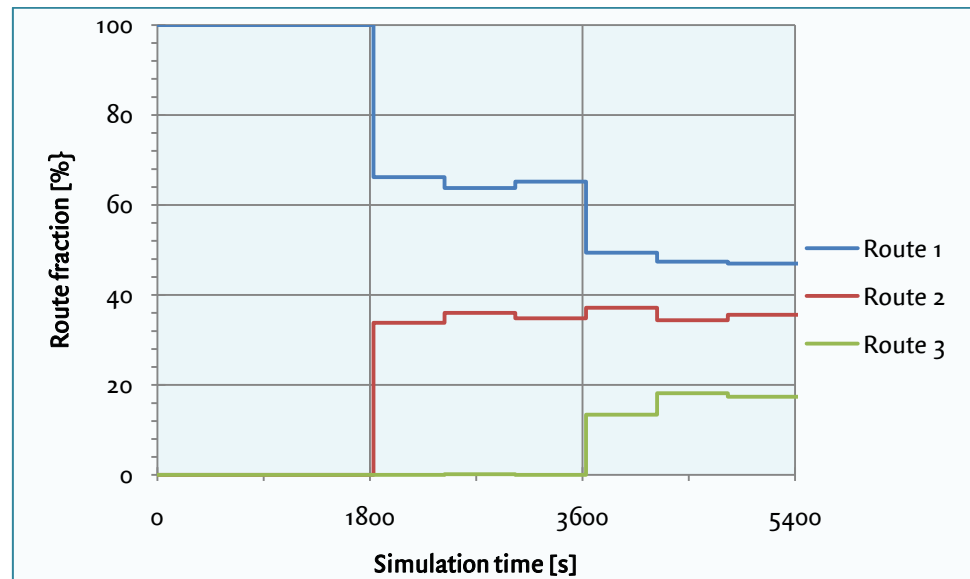


Figure 8.2 - Simulation run in network 1, $nue=35$, route fractions.

The reason for the large value of the convergence measure is a combination of these two phenomena, but is mainly caused by the calculation of the route fractions in the first iterations.

The route costs and route fractions at the 500th iteration are shown in table 8.2. In the last column of the table, the route cost gap is shown, which is the difference between the highest and lowest route costs of the utilised routes at a route choice moment. The largest part of these gaps lies between 0 and 1, however, there are three outliers; at $t=1530$, 3330 and 5130. Due to these outliers, the average route cost gap is higher than at the other simulation runs. At these three departure times, there is the similarity that there is one route with a very low route fraction. At $t=1530$, this route fraction is 0.4%, at $t=3330$ and 5130, this route fraction is respectively 0.6% and 2.0%. In the first iterations of the simulation this route is chosen a few times, but this is not the case in the rest of the rest of the simulation, because the route has become relatively more expensive. Since a (small) fraction of traffic is assigned to these routes, the routes are used according to the definition of a DUE. Since it is quite unlikely that these routes will be used in later iterations, in this situation a DUE will not exist.

Depart. time	Costs Route 1	Costs Route 2	Costs Route 3	Fraction Route 1	Fraction Route 2	Fraction Route 3	Route cost gap
15	765.40	812.45	877.77	100.00	0.00	0.00	0.00
330	823.96	824.02	885.87	97.00	3.00	0.00	0.07
930	841.43	841.41	886.09	54.00	46.00	0.00	0.02
1530	821.94	860.21	885.50	99.60	0.40	0.00	38.27
2130	899.04	899.19	899.12	50.40	40.60	9.00	0.15
2730	946.53	946.63	946.26	50.80	25.40	23.80	0.36
3330	937.63	937.47	946.11	50.00	49.40	0.60	8.63
3930	990.88	991.45	990.61	47.00	25.20	27.80	0.85
4530	1047.02	1046.39	1046.44	41.60	36.80	21.60	0.63
5130	1124.20	1231.33	1124.13	52.80	2.00	45.20	107.20

Table 8.2 - Data of network 1, nue=140, iteration 500. Route costs and route fractions

As mentioned earlier, the cause of this problem is the assignment of traffic in the first iterations of the simulation. In the first iteration, instantaneous route costs are used to determine the route fractions, which mean that the average speeds of the link are used at a specific point in time (see section 3.1.2).

Departure time	Costs Route 1	Costs Route 2	Costs Route 3	Fraction Route 1	Fraction Route 2	Fraction Route 3
0-30	765.43	814.97	877.79	100	0	0
30-630	823.07	824.22	885.89	100	0	0
630-1230	859.93	817.52	886.32	100	0	0
1230-1830	987.19	851.15	885.85	0	100	0

Table 8.3 - Data of network 1, nue=140, iteration 1. Route costs and route fractions.

In the beginning of the simulation, route 1 has the lowest route costs and is chosen at the first route choice moments. At $t=630$, the vehicles that departed at $t=0$ and $t=30$ have not arrived at the destination (the free flow travel time of route 1 is 765 seconds). This means that there are no vehicles on the last links of route 1 and the speeds on these links have not decreased yet. Route 1 is still the cheapest route and is chosen. However, at $t=1230$ there is traffic on all links of route 1, while the links of route 2 are empty. Route 2 is chosen here.

In the second iteration, route 2 is chosen at $t=630$, since route 1 becomes more expensive than route 2. At $t=1230$, route 2 is chosen again, since the route costs of the first iteration are used and, here, all traffic is assigned to route 1 from $t=0$ till $t=1230$. However, in later iterations the traffic at $t=630$ is more evenly distributed among the two routes, so that route 2 becomes less attractive at $t=1230$. Consequently, this route is not chosen any more during the whole simulation. The same applies for the outliers at $t=3330$ and $t=5130$.

It is difficult to avoid this problem when using a successive all-or-nothing assignment, since no trajectory-based route costs are available during the first iteration. This phenomenon is not likely to occur when a PCL-assignment is used; here, to all routes a fraction of traffic is assigned, relatively to each other. Whether this problem arises, depends strongly on the combination of the type of network, the settings and the flows in the network. In most networks this problem will not happen.

8.2 Network 2

In the second network, five parameters are varied. Besides tau, nue and kappa, delta and the minimum speed are also varied. The results are shown in table 8.4.

Tau	avgdiff	rse	rseEL	avgCost	avgCostEL
5	18.45	2.462	0.0687	911.56	0.0016
10	17.84	2.396	0.1534	911.13	0.0033
20	23.37	2.596		912.65	
40	23.15	3.194	0.2306	915.79	0.0034
80	17.52	2.158	-0.0562	910.12	-0.0009

Nue	avgdiff	rse	rseEL	avgCost	avgCostEL
0	26.02	2.537	0.0224	910.30	0.0026
17.5	221.68	9.817	-5.5648	895.19	0.0383
35	23.37	2.596		912.65	
70	14.17	2.206	-0.1501	912.83	0.0002
140	18.81	2.564	-0.0040	915.12	0.0009

Kappa	avgdiff	rse	rseEL	avgCost	avgCostEL
1	13.52	2.218	0.1576	913.25	-0.0007
7	10.41	1.695	0.7520	909.19	0.0082
13	23.37	2.596		912.65	
19	19.86	1.936	-0.5504	908.42	-0.0100
25	13.69	1.591	-0.4195	906.22	-0.0076

minSpeed	avgdiff	rse	rseEL	avgCost	avgCostEL
1	7.91	1.257	0.6016	906.19	0.0083
4	7.91	1.257	1.2032	906.19	0.0165
7	23.37	2.596		912.65	
10	135.93	6.649	3.6440	907.00	-0.0145
13	25.04	2.108	-0.2190	908.74	-0.0050

Table 8.4 - Results of the variation of tau, nue, kappa and the minimum speed. Network 2.

Note: the results of the variation of delta are not included in table 8.4, because all simulations gave the same results. Apparently, the used network is not suitable for delta. It was assumed that delta plays a role at merge nodes, but this is not the case. To investigate the influence of delta on an equilibrium, a different network with junctions must be used. Since there is not enough time available to include this in the current research, it is recommended to investigate the influence of delta in a later research.

8.2.1 Analysis network 2

Convergence

In network 2, there is no clear trend between the degree of convergence and the values of the different parameters. For both tau, kappa and nue, the degree of convergence fluctuates between 10 and 23 seconds, independent of the value of these parameters. It is difficult to explain these differences. However, in this network, slight variations in demand can have large consequences for the traffic distribution in the rest of the simulation, which becomes clear in example 8.2. Since the differences in degree of convergence seem quite random, it is assumed that parameter changes do not influence the degree of convergence much in this network.

Relative standard error

When the relative standard errors of the second network are compared with the relative standard errors of network 1, the values are smaller. This does not automatically mean that the parameters influence a DUE less than in the first network. In network 1, the default run is more converged than the default run in the second network – the average differences (*AvgDiff*) of the default runs are respectively 0.17 and 23.37 seconds. Since the degree of convergence is used to calculate the RSEs, the RSEs of network 1 are higher than the RSEs of network 2.

The RSEs that are calculated does not differentiate much when tau, nue and kappa are varied. Since the network is more complex, parameter changes have less influence on a DUE. Because of the occurrence of several outliers, there is no clear relationship between the parameters and the RSEs. There are similarities with the relationships in the first network, but the relationships are weaker. The RSEs of kappa are larger than the RSEs of tau and nue, which means that variations in kappa influence the equilibrium more than tau and nue.

Route costs

In the second network, changes in parameters have less influence on route costs than in network 1. Equally to the first network, there is a positive relationship between the route costs and tau. However, the route costs at tau=80 deviate from this trend; the average

route costs decrease, while it is expected that the average route costs would increase. The relations between the average route costs and the parameters ν and κ are consistent with the results in the first network, but the relations are weaker. Again, there are several outliers in the results. At $\nu=17.5$, the average route costs are remarkably lower than the other results. The reason of this deviation is set out in example 8.2. Furthermore, the average route costs at $\kappa=7$ are also deviant.

When the minimum speed is considered, the results are unexpected. Beforehand, it is expected that the average costs decrease when the minimum speed increases, since it takes longer before the speed on congested links reaches the original speed when a low minimum speed is used. However, in this network the average route costs at a minimum speed of 1 and 4 kilometres per hour are lower than the route costs at higher minimum speeds. Additionally, the average route costs at $\text{minSpeed}=13$ are higher than the average route costs at $\text{minSpeed}=10$, which is unexpected as well. It is difficult to find an interrelation between the minimum speed and the route costs. According to these results, the average route costs are weakly positively correlated with the minimum speed. If this is true, the default run is an outlier. Since these results are at odds with the hypothesis, it is likely that the relationship between the minimum speed and the average route costs will be different in another network. Further research is needed.

Example 8.2 - $\nu=17.5$; Exceeding of the jam density

Table 8.4 shows that the results at $\nu=17.5$ strongly deviate from the results at other values for ν . After 500 iterations the average route cost difference (AvgDiff) is more than 200 seconds - the average route costs are about 15 seconds lower than other results - and the RSE is about four times higher than the results at other values of ν . These divergent results can be explained by the exceeding of the jam density on some links in the network.

Figures 8.3 and 8.4 show the route costs in the network at iteration 500 and 499. There is a big difference between these graphs. At iteration 500, routes 1 and 3 have almost the same route costs, while the maximal route costs of route 2 (the red route in the graph) are 700 seconds higher than the maximal route costs of route 1 and 3. By contrast, the route costs of the three routes at iteration 499 are almost equal. The big route cost difference between the two iterations is remarkable, since the traffic

distribution among the routes does not differ much. In the 500th iteration, only 0.2% of the total amount of traffic is distributed. Converted to the amount of vehicles, the flow on route 2 does not change more than vehicles in the 500th iteration. This small change has large consequences for the route costs in the network.

This route cost difference can be explained by investigating the densities on the links. In this network, a congestion situation is created. Consequently, the density on the links increases, mainly on the links upstream the bottleneck. This is natural, since the amount of traffic that enters those links exceeds the capacity of the link. As a result, a queue arises and the density comes close to the jam density. The jam density is the density that appears when the speed on a link is 0 km/h and it has a default value of 180 vehicles per kilometre. In reality, it is impossible that the density on a link exceeds the jam density, since the average length of a car is around 5 metres and there must be some space between two vehicles. In other words, there is not enough space to locate more than 180 vehicles on a single lane with a length of one kilometre.

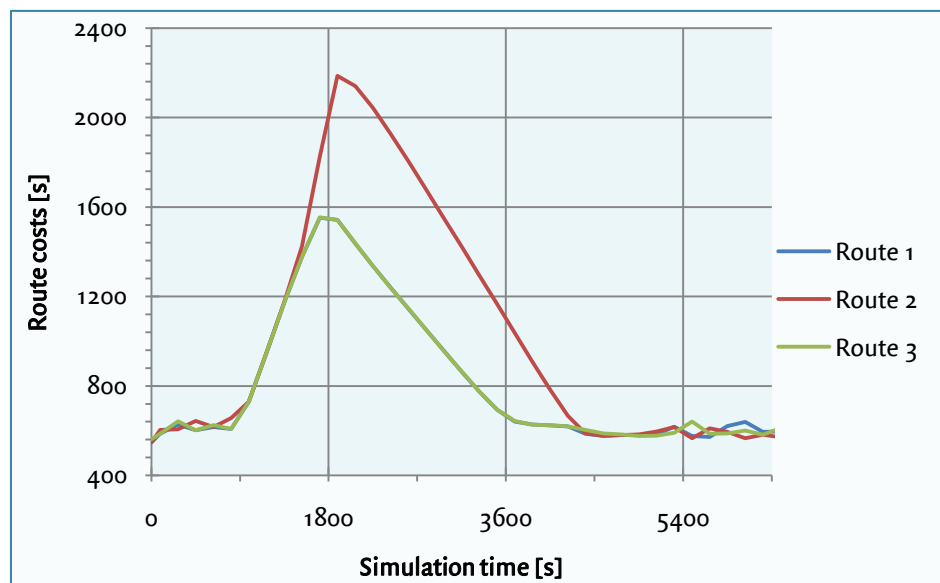


Figure 8.3 - Route costs, nue=17.5, iteration 500

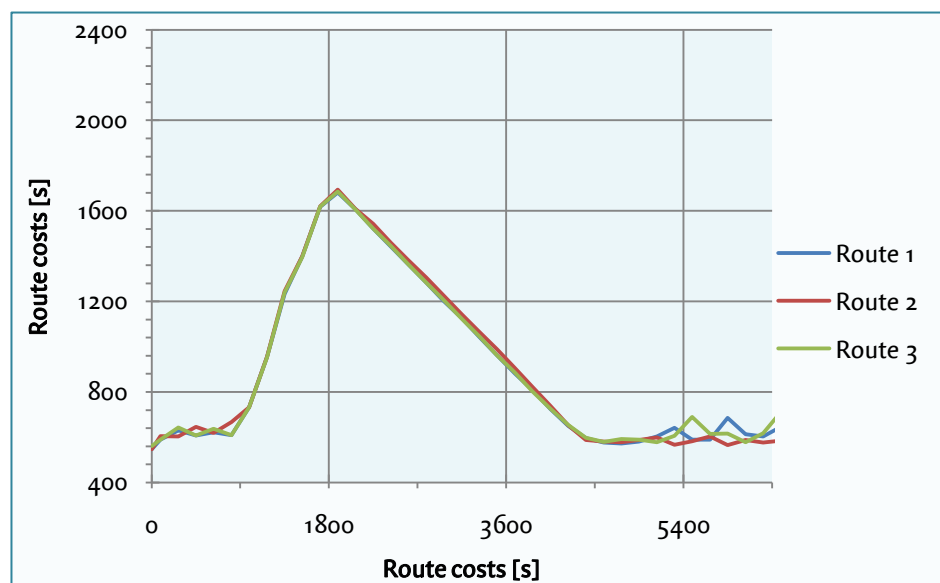


Figure 8.4 - Route costs, nue=17.5, iteration 500

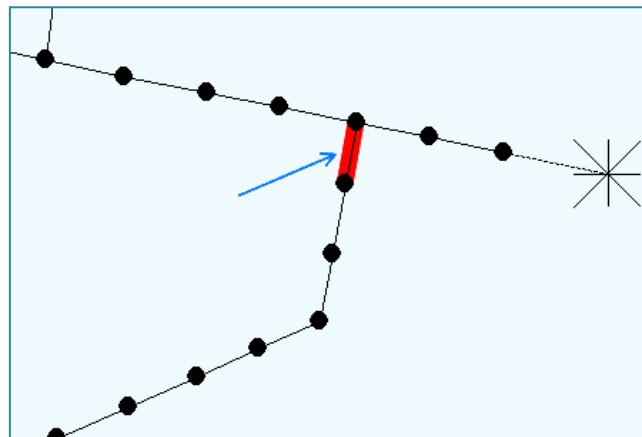


Figure 8.4 - Overview of the last part of the network. The red link is the link that exceeds the jam density.

However, on the last link of the second route (the route underneath) upstream the bottleneck, the density exceeds the jam density. In figure 8.5 this link is highlighted in red. Figure 8.6 shows the density development on this link during the simulation time. In the 500th iteration, the density exceeds the jam density of 180 veh/km, while in the 499th iteration, the density goes up to a maximum of 179.6 veh/km.

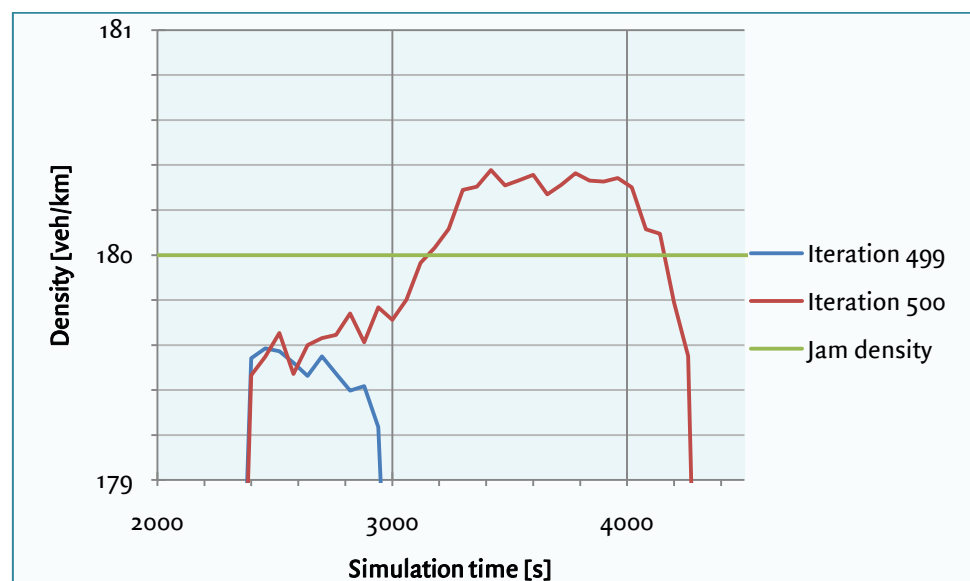


Figure 8.5 - Development of the density on the 'red' link at iteration 499 and 500.

In this situation, the following problem arises. When MaDAM calculates a density that exceeds the jam density, StreamLine sets the density of the link to the jam density. As a result, there is a difference between the density that is calculated by MaDAM and the density that is used in StreamLine, and upstream the link, a small queue appears. In the next time step, these extra vehicles are put on that link, which leads to an increasing

density. Consequently, the density upstream this link will also increase and the speed on those links increase too.

Besides, StreamLine sets the outflow of a link to zero when the jam density is exceeded. After all, when the jam density is reached, there is a queue on the link and the vehicles do not move. Since StreamLine only uses the outflows of a link to determine the route costs, the route costs increase quickly. This explains the large difference in route costs between the 499th and the 500th iteration in this example.

The difference between the two iterations is smaller when the route costs are calculated by the TravelTimeCalculator (i.e. based on average link speeds, see section 3.1.2). Since the link speeds are always larger than or equal to the minimum speed, the vehicles on the network keep moving, regardless of congestion. However, this does not solve the problem, it only reduces the problem. Because of the small queues upstream the links, there is still a large difference in route costs between the iterations.

There are similarities with example 5.5 in section 5.2. In this example, the speed on the link upstream a merge node drops to the minimum speed as a result of large density differences of two successive links. When this example is compared with the example that is described in this section, in both examples the speed upstream a junction drops to the minimum speed. However, the cause of this speed drop is different. In example 5.5, a large density difference causes the speed drop, while in the current example the speed drop is caused by the exceeding of the jam density. Actually, it is true that there is also a large density difference in the current example, but because of the urban anticipation term, the effect on speed is quite small here. After all, the problem that is described in this section only occurs with small values of n_{ue} , which means that the anticipation term is low and the density difference plays a small role at the determination of the speed on a link.

There are also some similarities with example 5.4 in section 5.2. Here, the emptying of the network goes slowly, so that the route costs that are calculated by StreamLine increase. In both examples, the route costs are unrealistically high, since StreamLine uses cumulative vehicles to calculate route costs. However, in example 7.4 the high route costs are caused by the emptying of the network; in this example it is caused by the exceeding of the jam density.

8.3 Conclusions

This chapter describes to which extent parameter changes influence the dynamic user equilibrium. In two different networks, the parameters τ , ν , κ and the minimum speed are varied. Network 1 is a simple network without overlap of routes or congestion; in network 2, a congestion situation is simulated with overlap of routes.

In both networks, changes in parameters do influence the dynamic user equilibrium. Changes in average travel time are observed, but the changes are quite small. In the congested network (2) the changes in travel time are smaller than in the first network. When the parameters are compared to each other, κ and ν influence the equilibrium more than τ .

The values that are given to the parameters are far removed from each other. The advantage of this approach is that changes to the DUE become more obvious. A disadvantage is that the chosen values are probably unrealistic and will never be used in practice. Since only slight differences to the DUE are observed with these extreme values for the parameters, it is likely that changes in parameters do not have large consequences for the DUE.

However, only two different simplified networks are used to answer the question. It is possible that the influence of parameters on a DUE is more evident when a larger and more complex network is used, with multiple OD pairs and much overlap between the routes. Because of the complexity and the long running times of these networks, this is not included in the research. Further research is needed to answer this question.

In the tests, there are several results that are deviant from the other results. Especially in the congested network (2), there is a large difference between the results. One cause of these differences is the *exceeding of the jam density* at some links (see example 11.2). When the density that is calculated by MaDAM exceeds the jam density (180 veh/km), the speed on that link strongly decreases and it takes an unrealistic long time before the speed on that link increases. When the jam density is reached, StreamLine sets the outflow of the link to zero, and since StreamLine uses the outflows of the links to determine route costs, the route costs increase. The effects of this phenomenon on route choice strongly depend on the network that is used.

Another cause of the deviance of the results is the *oscillation effect* or flip flop effect (see also example 8.1). When a successive all-or-nothing assignment is used in StreamLine, the route distribution does not change within a route choice interval. As a result, it is possible that the route fractions of two successive route choice intervals strongly differ from each other. In the used networks, small changes in route fractions in the beginning of the simulation have large consequences for the rest of the simulation. This explains

why the convergence of the results differs from each other in network 2. This phenomenon strongly depends on the network that is used.

Problems with the oscillation effect can also occur in real and larger networks. However, the effects on route choice would be smaller, since there are more OD pairs in the network with overlapping routes. It is possible to reduce oscillation problems by increasing the *number of route choice moments*. As a result, the route choice intervals are shorter. Since the route choice distribution does not change within a route choice interval, the effect of a route choice distribution at a specific route choice moment on the next route choice interval is smaller when more route choice moments are used. A disadvantage of this solution is the longer running time. On the other hand, there are fewer problems with oscillation when StreamLine runs a *PCL-assignment* instead of a series of all-or-nothing assignments. With PCL, the differences between the route fractions at a specific route choice moment are smaller in the first iterations of the simulation, so that fewer problems will arise in the rest of the simulation.

9 Conclusions and recommendations

9.1 Conclusions of the research

This research investigates whether StreamLine is able to determine a dynamic user equilibrium (DUE), and to which extent this equilibrium changes due to parameter changes in StreamLine.

Whether a DUE can be determined in StreamLine, depends on the method how StreamLine assigns route fractions during the simulation. When StreamLine uses a PCL assignment (which is default in StreamLine), it is not possible to determine a deterministic equilibrium. In case of PCL, StreamLine always assigns a (small) part of the traffic to each route, even when the route costs of a specific route are that high that the route should not be chosen. As a result, it is possible that the route costs of the used routes are not equal. When a series of all-or-nothing assignments is used, it is theoretically possible to determine a deterministic equilibrium. Since StreamLine assigns all traffic to the shortest route in each iteration, very expensive routes will not be chosen during the simulation.

In the research, three limitations of MaDAM - the propagation model in StreamLine - were elicited. These limitations basically lead to problems when a successive all-or-nothing assignment is used. However, these problems can also occur at PCL assignments, but the consequences are smaller on route choice.

- When the flow on a link drops to zero during the simulation, the density on this link decreases (according to the formulas of MaDAM) very gradually. As a result, it takes a long time before the density goes to zero. This situation occurs very often in the first iterations of a successive all-or-nothing assignment. Since StreamLine uses cumulative vehicles to determine the route costs of routes, the situation can occur that the route costs that are determined by StreamLine are unrealistic, so that it influences route choice. Whether this phenomenon occurs, depends on the network that is used. When PCL is used, it is very unlikely that it will happen.
- When there is a large density difference between an ingoing and the outgoing link at a merge node, it is possible that the speed on the link upstream the merge node decrease unrealistically. This speed drop is caused by the anticipation term in the speed formula of MaDAM. This situation is likely to happen in the first iterations of a successive all-or-nothing assignment, but also on on-ramps on highways.

- When the flow on a link changes quite fast – for example when a demand pattern is set in StreamLine – the speed on this link changes unexpectedly. When the flow increases, the speed increase on the link increases too, and vice versa. This can have an influence on route choice, however, the effect is minimal. This problem is caused by the anticipation term in MaDAM. The problem occurs with both successive all-or-nothing assignments and PCL-assignments. However, at a successive all-or-nothing assignment, this will occur more often, since there are more flow changes in the network during the simulation.

In the research, two adaptations are done to the network to avoid the problems that are mentioned above. With these adaptations, StreamLine determines a deterministic user equilibrium. The adaptations are not suitable for use in practice.

- On all links in the network some background traffic is assigned, to avoid that the density in the network drops to zero.
- The links upstream and downstream merge nodes and junctions are replaced by urban links. Urban links have a different anticipation term, so that it avoids unrealistic speed drops.

The length of a route choice interval influences the accuracy of the results. The shorter a route choice interval, the more realistic the development of the route costs will be, but there are more iterations needed before the route costs are converged. Besides, the flip-flop effect increases when the route choice interval becomes shorter and the running time becomes larger.

In the parameter analysis, it became clear that the parameters τ , κ and ν do influence the equilibrium, but the changes in route costs are quite small. κ and ν influence the equilibrium more than τ . These conclusions are based on tests with two small networks. Theoretically, it is possible that the effects are different when a larger and more complex network is used.

Some results of the parameter analysis were unrealistic due to the exceeding of the jam density. When MaDAM calculates a density that is higher than the jam density, StreamLine sets the outflow of a link to zero. Since StreamLine uses the outflows to determine route costs, this can lead to an extreme increase in route costs.

Due to the oscillation effect or flip-flop effect, unrealistic results may arise. Since the route distribution does not change within a route choice interval in StreamLine, it is possible that large differences in route fractions arise at subsequent route choice moments. Increasing the number of route choice moments will reduce this problem. This

phenomenon strongly depends on the network that is used and will principally occur at all-or-nothing assignments.

9.2 Recommendations

- The stop criterion that is used in StreamLine is incorrect (section 3.1.5). StreamLine stops the simulation when the duality gap has not changed more than a specific value in comparison with the previous iteration. When the simulation stops, StreamLine shows a message that an equilibrium is reached. This does not have to be the case. For example, when a PCL assignment is used, the simulation stops when the route costs does not change any more. In this case the route costs of the utilised routes are not equal, so a deterministic equilibrium is not reached. It is advised to stop the simulation when the route costs of each OD pair are equal, in case of a deterministic equilibrium. In case of a stochastic equilibrium the simulation should stop when the perceived route costs are equal.
- To avoid problems due to the emptying of the network, it is necessary that the density in MaDAM goes to zero very quickly. It is recommended that the speed on a link goes to the free speed when the calculated density is lower than a specific value (say, 0.5 veh/km) *and* when the flow on that link is zero. If this combination occurs in the network, it is needed to change the speed to avoid problems with route choice. This solution is not tested in StreamLine yet.
- The current anticipation term of the speed formula that is used in StreamLine can cause problems, like speed drops near merge nodes and on-ramps on highways, unexpected speed changes when the flow on a link changes, but also the slow emptying of the network. All those problems are caused by a large density difference of two subsequent links. However, when there are no large density differences, the anticipation term seems to work well. It is advised to use a different anticipation term when the difference in density of two subsequent links exceeds a specific value. In this new anticipation term, the density difference should have a smaller impact on speed. Another possibility is to keep the current anticipation term and multiply the density difference in the formula with a constant, say α . In this case α can have two possible values; 1 if the density difference is small, and a lower value if the density difference is larger than a specific value. However, this formula change has not been tested yet. The results may be instable when the density fluctuates around α . It must be tested whether this will be the case.
- The more route choice intervals are used in the simulation, the more accurate the results are, but the longer the simulation time and the lower the convergence speed. When StreamLine will be used in practice, a trade-off between these factors is needed for each simulation.

- To avoid problems due to the exceeding of the jam density in StreamLine, it should not be possible that MaDAM calculates a density that is higher than the jam density. It is recommended to replace the calculated density by the jam density.
- It is recommended to change the way StreamLine calculates route costs. The route calculation that is based on cumulative vehicles (the method StreamLine currently uses) leads to inaccurate results in case of the emptying of the network (section 5.2.1, example 5.3) and in cases when the speed drops below the minimum speed (section 8.2, example 8.2). It is recommended to use the average link speeds to determine the route costs, which is the method the TravelTimeCalculator uses. Currently, the TravelTimeCalculator ignores queuing time at origin nodes; when StreamLine will use average link speeds for the route cost calculation, this queuing time should be added to the calculated route costs.

9.3 Further research

- In the current research, several problems occurred that are caused by the anticipation term (emptying of the network, speed drops near merge nodes etc.). Only small networks were used in this research. To which extent do these problems occur in larger and more complex networks? Do they influence route choice?
- In section 9.2, it is recommended to change the current anticipation term. How can the current anticipation term be improved, so that the problems that are mentioned earlier can be solved?
- In this research, several parameters are varied. What are the effects of parameter changes in a larger network?
- The current research focuses on a deterministic user equilibrium. However, when PCL is used, a stochastic user equilibrium should be determined. To which extent can StreamLine determine a stochastic user equilibrium?
- To which extent does junction modelling affect the dynamic user equilibrium?
- To which extent do DTM-measures affect the dynamic user equilibrium?
- At the moment, departure time choice is not involved in StreamLine. Is it possible to implement departure time choice in StreamLine, and if so, to which extent does it affect the dynamic user equilibrium?
- Currently, StreamLine calculates a route set once, in the beginning of the simulation. In this research, the used networks were quite small, and the calculated route set was comprehensive. In larger networks, there are more routes possible. To which extent does the number of routes that are included in the route set influence the dynamic user equilibrium?

References

- Bliemer, M. (2001). *Analytical dynamic traffic assignment with interacting user-classes*. Delft: DUP Science.
- Bliemer, M. (2007). Dynamic Queuing and Spillback in an Analytical Multiclass Dynamic Network Loading Model. *Transportation Research Record* 2029 , 14-21.
- Friesz, T., Bernstein, D., Smith, T., Tobin, R., & Wie, B. (1993). A Variational Inequality Formulation of the Dynamic Network user Equilibrium Problem. *Operations Research, Volume 43* , 1120-1136.
- Horowitz, J. (1984). The Stability of Stochastic Equilibrium in a Two-Link Transportation Network. *Transportation Research Part B Vol.18* , 13-28.
- Kerner, B., & Konhäuser, P. (1994). Structure and parameters of clusters in traffic flow. *Physical Review E* 50 , 54-83.
- Koppelman, F., & Wen, C.-H. (2000). The paired combinatorial logit model: properties, estimation and application. *Transportation Research Part B* 34 , 75-89.
- Maarseveen, M. v., & Zuilekom, K. v. (2006). *Verkeer; Civiele Techniek: dictaat Verkeer*. Enschede.
- Massachusetts Institute of Technology. (2011). *Definition of standard deviation*. Retrieved August 26, 2011, from Chemicool: http://www.chemicool.com/definition/standard_deviation.html
- Missouri State. (2011). *Elasticity*. Retrieved August 12, 2011, from <http://courses.missouristate.edu/ReedOlsen/courses/ec0165/Notes/elasticity.PDF>
- Ortúzar, J., & Willumsen, L. (2001). *Modelling transport*. Chichester, England: John Wiley and Sons.
- Papageorgiou, M., Kotsialos, A., Fellow, IEEE, Diakaki, C., Pavlis, Y., et al. (2002). Traffic Flow Modeling of Large-Scale Motorway Networks Using the Macroscopic Modeling Tool METANET. *IEEE Transactions on Intelligent Transportation Systems, Vol 3 (4)* , 282-292.
- Peeta, S., & Ziliaskopoulos, A. (2001). Foundations of Dynamic Traffic Assignment: The Past, the Present and the Future. *Networks and Spatial Economics, 1* , 233-265.
- Raadsen, M., Mein, H., Schilpzand, M., & Brandt, F. (2010). Implementation of a single dynamic traffic assignment model on mixed urban and highway transport networks including junction modelling.
- Rakha, H., & Crowther, B. (2002). A Comparison of the Greenshields, Pipes, and Van Arde Car-Following and Traffic Stream Models.
- Ran, B., Boyce, D., & Le Blanc, L. (1993). A New Class of Instantaneous Dynamic User-Optimal Traffic Assignment Models. *Operations Research, Vol. 41, No. 2* , 192-202.

Schilpzand, M. (2008). *X-Stream in MaDAM, New junction modelling in macroscopic dynamic traffic assignment models*.

Sheffi, Y. (1985). *Urban transportation networks: Equilibrium analysis with mathematical programming methods*. New Jersey: Prentice Hall.

Technical University of Crete & Messmer. (1990). *METANET, a Simulation Program for Motorway Networks, Documentation*.

Wardrop, J. (1952). Some theoretical aspects of road traffic research. *Proceedings of the Institute of Civil Engineers*, 325-378.

Appendix I - Variation in traffic propagation parameters

In the research, several traffic propagation parameters are varied. The variation of these parameters is argued in this section. In chapter 3, the propagation model of StreamLine, MaDAM, is described. Many of the traffic propagation parameters influence the speed calculation. The formula to calculate the speed of each cell is formulated below.

$$v_{m,i}(t+1) = v_{m,i}(t) + relaxation + convection - anticipation \quad (I.1)$$

$$relaxation = \frac{T}{\tau} \cdot (v(\rho_{m,i}) - v_{m,i}(t)) \quad (I.1a)$$

$$convection = \frac{T}{L_m} \cdot v_{m,i}(t) \cdot (v_{m,i-1}(t) - v_{m,i}(t)) \quad (I.1b)$$

$$anticipation = \frac{v \cdot T \cdot (\rho_{m,i+1}(t) - \rho_{m,i}(t))}{\tau \cdot L_m \cdot (\rho_{m,i}(t) + \kappa)} \quad (I.1c)$$

Tau

The tau parameter τ influences both the relaxation and anticipation term in the speed formula. Tau must be higher than zero, otherwise the relaxation or anticipation term has to be divided by zero, which is impossible. When tau is increased with 100%, both the relaxation and anticipation term decrease with 100%.

The default value in StreamLine is set to 20 seconds. To find out to which degree tau influences the DUE, the tau is varied between 5 and 80 seconds. The default value is doubled and halved twice. This means that the relaxation and anticipation term are quadrupled or four times lower when these extreme values are applied.

Nue

The nue parameter v only influences the anticipation term. When nue is doubled, the anticipation term also increases with 100%. By varying the nue, the influence of the anticipation term is determined. When nue is set to 0 km²/h, anticipation is neglected in the speed calculation. The maximum value of nue is set to four times the default value, as is done with the tau parameter. Since the default value of nue is 35 km²/h, nue is varied between 0 and 140 km²/h.

Kappa

Kappa (κ) influences the anticipation term. The formula of the anticipation term shows that the influence of kappa on the speed depends on the density and the density changes on the link. When kappa increases, the anticipation term will be relatively smaller since the denominator in the formula increases faster. The following graph shows this proposition. In this test, the kappa is varied from 0 to 50 veh/km, with a density difference of 0.1 veh/km and a current density of 50 veh/km. The other variables are kept constant.

The default value of kappa is 13 veh/km. It is hard to choose suitable boundaries for kappa since the relation between kappa and the anticipation term is not linear and depends on several factors. The values 1 and 25 are chosen.

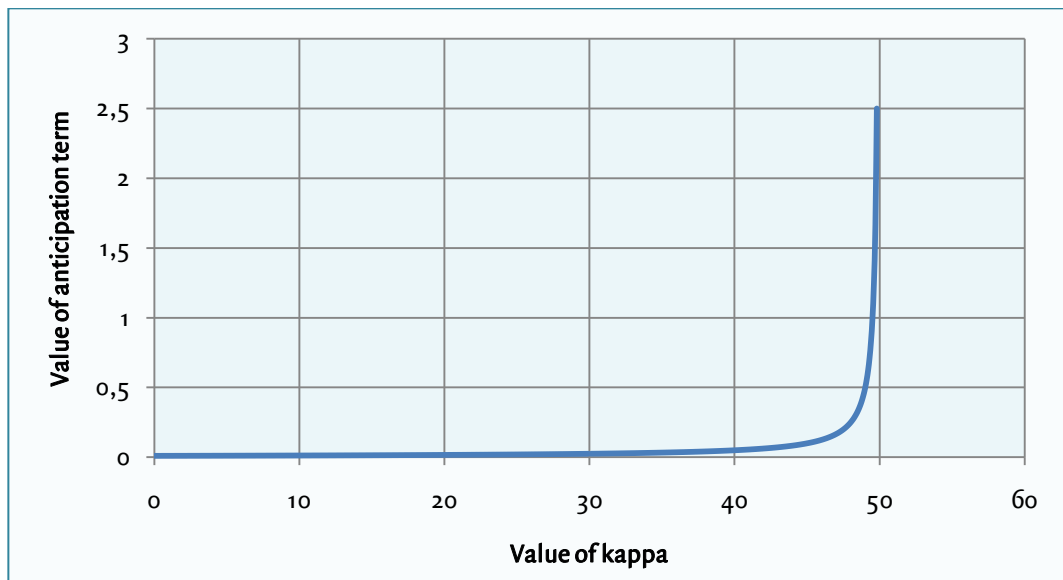


Figure I.1 - Relation between the value of kappa and the value of the anticipation term, with a density difference of 0.1 veh/km and a current density of 50 veh/km.

Delta

Delta (δ) does not influence the three terms that are described in the introduction of this section, but it effects the merging term, which is subtracted from the speed formula. This merging term is applied when the number of lanes changes. In the research, this is the case near the destination node, when three roads merge to one road with two lanes. When merging takes place the delta is multiplied with the merging term, which means that an increase of delta with 100% leads to an increase of 100% of the merging term (Technical University of Crete & Messmer, 1990).

The default value of delta is set to 0.8. The merging term is ignored when delta is zero, which is interesting to investigate. On the other hand, it will be investigated which

influence the merging term has on the speed on a link when the merging term is doubled. Therefore, delta is varied between 0 and 1.6.

Minimum speed

The minimum speed is not anchored in a formula, since StreamLine uses the minimum speed when the speed calculation drops below this minimum speed. So, in case of congestion vehicles do have a speed in StreamLine. The lower the minimum speed, the longer it takes before vehicles accelerate to a specific speed. The default value is 7 km/h. Differences in speed can be seen more easily when the varied minimum speeds have extreme values. For the minimum value a value of 1 km/h is chosen. In this situation, it is expected that it takes long before the desired speed is reached after congestion. The maximum value of the minimum speed is 13 km/h, which is a duplication of the default value.

Appendix II - Paired Combinatorial Logit (PCL)

The main route choice model that is used in StreamLine is the Paired Combinatorial Logit model, also called PCL. PCL is a logit model, which means that it calculates the route distribution based on the cross-elasticity between pairs of route alternatives. An advantage of this method is that it converges quite fast to an optimal solution in comparison with other route choice models, like a series of all-or-nothing assignments.

In StreamLine, a spread parameter (μ) is defined. A high value of μ means that the route distribution is close to uniform; when μ reaches a value of (almost) 0, almost all traffic is assigned to the cheapest route (all-or-nothing) in every iteration. The default value of μ in StreamLine is 0.14.

The probability that a route in the network is chosen, can be calculated with the following formula (Koppelman & Wen, 2000).

$$P_i = \sum_{j \neq i} P_{ij} \cdot P_{ij} \quad (II.1)$$

Where

$$P_{ij} = \frac{e^{\frac{V_i/\mu}{1-\sigma_{ij}}}}{e^{\frac{V_i/\mu}{1-\sigma_{ij}}} + e^{\frac{V_j/\mu}{1-\sigma_{ij}}}} \quad (II.2)$$

$$P_{ij} = \frac{\left(e^{\frac{V_i/\mu}{1-\sigma_{ij}}} + e^{\frac{V_j/\mu}{1-\sigma_{ij}}} \right)^{1-\sigma_{ij}}}{\sum_{k=1}^{n-1} \sum_{m=k+1}^n \left(e^{\frac{V_k/\mu}{1-\sigma_{km}}} + e^{\frac{V_m/\mu}{1-\sigma_{km}}} \right)^{1-\sigma_{km}}} \quad (II.3)$$

In these formulas, the following variables are used.

Parameter	Explanation
i	route alternative
j	route alternative
k	route alternative
m	route alternative
P_i	the probability that route alternative i is chosen
P_{ij}	the conditional probability of choosing alternative i , given the chosen

	binary pair (i,j)
P_{ij}	the unobserved probability for the pair (i,j)
σ_{ij}	overlap between the route alternatives i and j
V_i	the utility of route alternative i
μ	spread parameter

Table II.1 - Parameters used in the PCL-algorithm.

Appendix III - Cross node modelling

A network in StreamLine exists of a collection of nodes and links. At a node, there are several links that enter the node (entry links) and several links that leave the node (exit links). Chapter 3 describes that MaDAM calculates the density, flow and speed for each link segment. At a node, these characteristics must be transported from the entry links to the exit links. Since there are several entry and exit links at a node, this is a complex task. To solve this problem, StreamLine uses Cross Node Modelling (CNM). Cross Node Modelling calculates which distribution of the flow on the entry links will be transferred to an exit link. The cross node model that is used in StreamLine is derived from a cross node model that is developed by Bliemer (2007).

An advantage of Bliemer's model is that it takes oversaturation of links into account. This can be explained by figure III.1. Here, a simplification of a node is shown, with two entry links (B^{in}) and three exit links (B^{out}). The entry links have outflow rates (\bar{v}_{am}^{out}), that are equal to the flows that enter the exit links. If the exit link reaches its capacity (C_b^{in}), MaDAM does not transfer the total outflow rate to the outgoing links. Otherwise, the flow on the exit link exceeds the capacity, which is unrealistic.

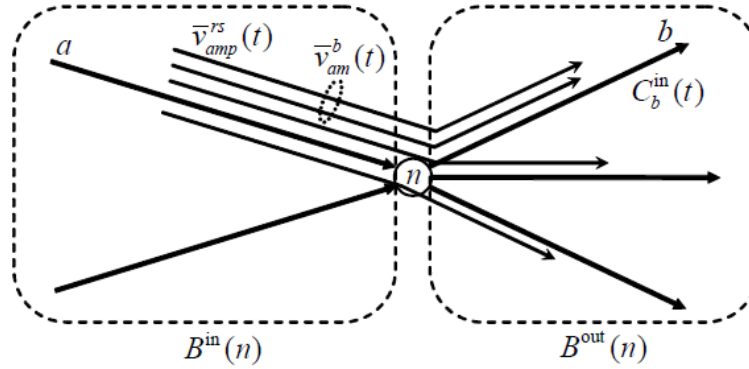


Figure III.1 - Simplification of a node, and its entry and exit links (Bliemer, 2007)

Bliemer's model first calculates the potential outflow rate at time t . This potential outflow rate is compared with the capacity of the exit link. If the potential outflow rate exceeds this capacity, the outgoing flows decrease with a specific percentage, which is equal for all incoming flows. The analytical solution is as follows:

$$v_{am}^b(t) = \min_{b' \in B^{out}(n) \setminus \{b \mid \bar{v}_{am}^{b'}(t) \geq 0\}} \left\{ \bar{v}_{am}^b(t), \frac{\bar{v}_{am}^b(t)}{\sum_{a'} \sum_m \rho_m \cdot \bar{v}_{am}^{b'}(t)} \cdot C_b^{in}(t) \right\} \quad (III.1)$$

$$v_{amp}^{rs}(t) = \frac{\bar{v}_{amp}^{rs}(t)}{\bar{v}_{am}^b(t)} \cdot v_{am}^b(t) \quad (III.2)$$

In table III.1, the variables which are used in the formulas are explained:

Parameter	Explanation
a	link
m	vehicle type
t	time
r	origin
s	destination
p	route
b	in the direction of link b
\bar{v}	potential outflow rate [veh/h]
v	outflow rate [veh/h]
ρ	pcu-value ¹⁰
C^{in}	flow capacity of the link [veh/h]

Table III.1 - Variables used in Bliemer's model

StreamLine uses the same model, but the model is adapted and the notation is simplified. Some variables are left out, like the pcu-value (ρ) and the vehicle type (m). The modified flow calculation is as follows (Raadsen, Mein, Schilpzand, & Brandt, 2010):

$$\hat{f}_{ij} = f_{ij} \cdot \min(\alpha_{ij}) \quad (\text{III.3})$$

$$\alpha_{ij} = \begin{cases} 1 & \Rightarrow f_{ij} = 0 \\ \min\left(1, \frac{c_{\text{exit},j}}{f_{\text{exit},j}}\right) & \Rightarrow \text{else} \end{cases} \quad (\text{III.4})$$

Parameter	Explanation
α_{ij}	scale factor for incoming link i to outgoing link j
$c_{\text{exit},j}$	capacity of outgoing link j [veh/h]
$f_{\text{exit},j}$	total offered flow to outgoing link j [veh/h]
f_{ij}	offered flow from incoming link i to outgoing link j [veh/h]
\hat{f}_{ij}	allowed flow from incoming link i to outgoing link j [veh/h]

Table III.2 - Parameters used in the cross node model, as used in StreamLine

¹⁰ PCU stands for Passenger Car Unit, which is a vehicle unit that is used for expressing highway capacities. A single car has a value of 1, while common values for other modes like motor cycles and buses are respectively 0.4 and 2.0 (Maarseveen & Zuilekom, 2006).