

Multi-Agent Model for the Urban Distribution Centre

*Scenario search and dynamic urban distribution centre pricing
to find a positive business case*

MoT Master Thesis

Author: Antal van Kolck

Studentnumber: 1220659

Delft, October 2010



Delft University of Technology
Faculty Technology, Policy and Management
Transport Policy and Logistics' Organization

Master thesis professor: Prof. Dr. Ir. L. Tavasszy
First attendant: Dr. R van Duin
Second attendant: Dr. I. Nikolic



Kyoto University graduate School of Engineering
Department of Urban Management
Logistics Management Systems

Master thesis professor: Prof. Dr. Taniguchi
First attendant: Dr. A. G. Qureshi

The important thing is not to stop questioning. Curiosity has its own reason for existing. One cannot help but be in awe when he contemplates the mysteries of eternity, of life, of the marvelous structure of reality. - Albert Einstein

Preface

This report contains the results from master thesis research by Antal van Kolck. It was carried out at the section Transport Policy and Logistics' Organization in the Faculty of Technology, Policy and Management of the Delft University of Technology in collaboration with the Department for Urban Management in the Faculty of Engineering at the Kyoto University Graduate School of Engineering.

The research aims to provide insight in the urban distribution centre (UDC) success by investigating dynamic price settings with a multi-agent model. This is based on cost-valued choices by individual agents and presents results in financial and environmental impact. The generation of this model in Netlogo will serve as a starting point for further research on UDCs and other City Logistics multi-agent models.

For this research I want to express my gratitude to Dr. Ron van Duin and Prof. Dr. Taniguchi for the opportunity to go to Japan and their great support and supervision. Dr. Ir. Igor Nikolic I want to thank for his hands-on mentality and high research standards. Dr. Ali Gul Qureshi and Prof. Dr. Ir. Lori Tavasszy I want to thank for their great input for the research and report. Furthermore I want to thank Sander for his patience with my work style and support on reviewing the grammar and spelling of this report.

Antal van Kolck

Oktober 2010

Abstract

In a 6 month research at Kyoto University and Delft University of Technology, the urban distribution centre (UDC) was investigated with a multi-agent model to find situations where the UDC can be financial viable. From a literature study the main modeling components were found: commodity estimation, vehicle routing, traffic environment, agent interaction and various policy measurements. Policies to test the impact on the business case of the UDC are: different congestion and toll rates, subsidy and a dynamic fee.

Hereby a gap in literature is addressed by dynamic modeling with a multi-agent model with vehicle routing and incorporating dynamic usage. Dynamics are included with variable traffic conditions (resulting in dynamic travel times) and dynamic demand locations. Dynamic UDC usage was tested by various delivery schemes and a dynamic fee, which changes hourly.

After generating the multi-agent model with a genetic algorithm for vehicle routing, it has to be checked that an increasing UDC usage rate corresponds to decreasing both the NOx emission and km count (NOx emission reduction of $19.0\% \pm 13.7\%$ and km count reduction of $18.8\% \pm 12.9\%$ over 2318 runs). The model tested different UDC delivery schemes (early delivery, fixed time delivery or full truck delivery), congestion rates (low, normal or high), toll rates (none, low or high) and subsidy levels (none, 25%, 33% or 50% subsidized). The only positive business case for the UDC is with a subsidy rate of 50% (rate 4), which shows the complexity of generating a positive business case for an UDC. Per setting 360 runs were performed.

To enhance the UDC financial performance the dynamic UDC fee was tested. This fee varies during the different hours of the day. Five cases were tested by increasing the general dynamic fee curve with an additional global shift of -20, -10, 0, 10 or 20 Yen per parcel. The impact of NOx emission or km count reduction corresponds to the fixed fee scenario. Per setting 50 runs were performed. Significant differences between various dynamic fee settings are present. The fact that the UDC income is close to being significantly higher than the -300.000 Yen per day, compared to similar fixed fee conditions is highly encouraging. However further and full evaluation of all possible dynamic fee settings is desired to find potential better dynamic fee settings than the currently used settings.

At the end of this research three cases are compared for dynamic fee usage: a reference case, highly congested city case and the involved municipality case. Comparing the three scenarios, there is no significant difference between the NOx emission or km count reduction and the freight carriers costs. For the UDC income there is a significant income increase in the third scenario with respect to the other scenarios. Also for the municipality there is a

significant increase in costs in the third scenario. Since there is no significant decrease in NO_x emission or km count between the second and third scenario, the municipality will need other social or financial incentives, before the municipality will participate in providing such high and long term subsidies.

From this research it is clear that generating a positive business for a UDC is highly challenging. For a UDC to be profitable, it seems to be necessary to receive subsidy permanently. This analysis of the UDC analysis relates to the low UDC success rate of the research of Browne, Sweet et al (2005). The conditions for generating a successful business case for the UDC are hard and limited. Further research of dynamic fee settings and investigation of the possibility to implement this in a real-life case can contribute to future prospective and the success rate for urban distribution centres.

Summary

In this report the conclusion of a short literature review on Urban Distribution Centres (UDCs) and the generation of a multi-agent model about an UDC are described. With the multi-agent model of an UDC, the research questions are answered to find a positive business case for the UDC.

Problem identification and research objective

In recent decades cities are getting larger and becoming denser, which put tremendous demand on business deliveries and passenger transport. The increased importance of environmental and social impact of such issues adds additional dimensions to the city's problem to improve the traffic flow inside the city. Freight transport contributes an estimated 10% - 30% of the traffic flow in various cities (Ministry of Land Infrastructure and Transport Japan) (Crainic and Ricciardi, 2009), but an estimated 40% in pollution and noise (COST 321, 1998). One of the proposed policy measures, besides toll ways and time windows for trucks in the inner city, is the urban distribution centre (Dablanc, 2007). The urban distribution centre is a promising concept, where the load of trucks entering the city is transferred to new (and fewer) trucks to increase the load factor. This allows for easier time-windowed operation to avoid traffic jams (Quak & de Koster, 2009) and a higher load factor in the city can decrease harmful effects associated with city logistics.

In practice the UDC concept has failed due to lack of participating stakeholders and unsatisfactory financial performance (van Duin and Quak, 2008). In many modelling attempts the UDC appeared to be successful (Marcucci and Danielis, 2008), which is in sharp contrast with the fact that only 15 out of 200 centres were running after 5 years (Browne and Sweet, 2005). In many models with static conditions (constant cost/demand/traffic flow) a global optimum is sought without individual agent perspective and decision making. Therefore a multi-agent model is desired to find out if (and under what conditions) this promising UDC concept can contribute to enhanced city logistics. The main objective of this research is to investigate the usage of urban distribution centres with dynamic multi-agent modelling.

Research Questions

The main question to be addressed is:

What is the impact of dynamic urban distribution centre usage on city logistic parameters investigated with multi-agent modelling?

In order to answer this question the following sub questions have to be answered:

- 1) Which models, needed for the UDC description, exist to describe city logistics processes?
- 2) What are critical success factors for an UDC and how does an UDC contribute to city logistics?
- 3) Which model of an UDC can include most critical success factors?
- 4) What policy measures contribute to the successful functioning of the UDC?
- 5) How do various scenarios impact the business case of an UDC?

Methodology

A brief literature review will answer the first two sub-research questions and provide practical knowledge on modelling approaches as well as a theoretical framework for the stakeholders. In this research the main effort is towards generating a multi-agent model to describe a city with an UDC and a framework is formed to answer question three. A multi-agent model is chosen to capture the complexity of individual behavior towards a societal goal. Various modelling components like vehicle routing are present at Kyoto University and were adapted for this model. This model is used to answer the fourth question and while the business case behind the UDC is used for answering the last sub question. The combination of this information answers the main research question.

Results literature study: City Logistics Models (Sub question 1)

From the literature review it is found that the field of city logistics is a fast developing field as can be seen by the increase in articles over the last years with much effort focused on modelling. Most of these models use the approach, where first vehicle/commodity quantities are estimated followed by vehicle routing and impact calculations. Different policies are examined with these models. Multi-agent models exist in city logistics to model interaction as well as the decision making and with the goal to capture highly complex nature of city logistics with many stakeholders. The information from the literature review was used for generating the new multi-agent model.

Results literature study: Urban Distribution Centres (Sub question 2)

The UDC is a promising concept to reduce congestion and emission inside the city centre, but main concerns are the financial viability of the centre. In a large study results show only 15 out of 200 centres are still running after 5 years (Browne and Sweet, 2005). Additional value added services can boost the value of the UDC, but collaboration with involved actors is necessary to determine the need. Main critical success factors are presented below. Main policy measures, which contribute to the successful functioning of the urban distribution centre, are toll requirements, subsidy providing and time windows inside the city. The last two policies will be studied with the multi-agent model. The business case behind the UDC was investigated based on the Cost Benefit Analysis by Dennis Kuiper (van Duin and Quak, 2008).

Table 1: Critical success factors for an UDC (Marcucci and Danielis, 2007), (Browne, Sweet et al 2005).

Critical success factors for the UDC
Location in/near the city
Subsidy collection
Collaboration with shipper and freight carriers
Financial viable
Service cost of the UDC
Access permit cost
Delay in delivery time
Distance of the parking bay from the shop

Multi-Agent Model for an Urban Distribution Centre

For the model generation it is important to address as much critical success factors as possible. A reasonable assumption for a good location in/near the city was made, but this will not be tested in the model. In the model the collaboration between shippers and freight carriers is assumed to be perfect as they act as one agent. The service cost of the UDC and subsidy are under investigation with the model. Access costs in the form of toll to the city centre will be a policy measure to investigate. Delay in delivery time is currently not measured, but is a recommendation for a later stage. Distance from the parking bay to the shop will not be included in this model.

The agents in the model behave purely rational to financial and/or added value motives determined per specific stakeholder. There are 5 types of agents, namely freight carriers, an UDC, trucks, retailers and roads. The municipality is considered an agent from the theoretical point of view, but in the practical model is it part of the environment. There are also two types of dummy agents, namely the streets and nodes, which are needed to perform calculations at this model scale/size. The multi-agent model is described schematically the figure below, where agents and flows are represented (see Figure 22). The trucks, roads and retailers are purely reactive agents. The UDC and freight carriers are objection function agents.

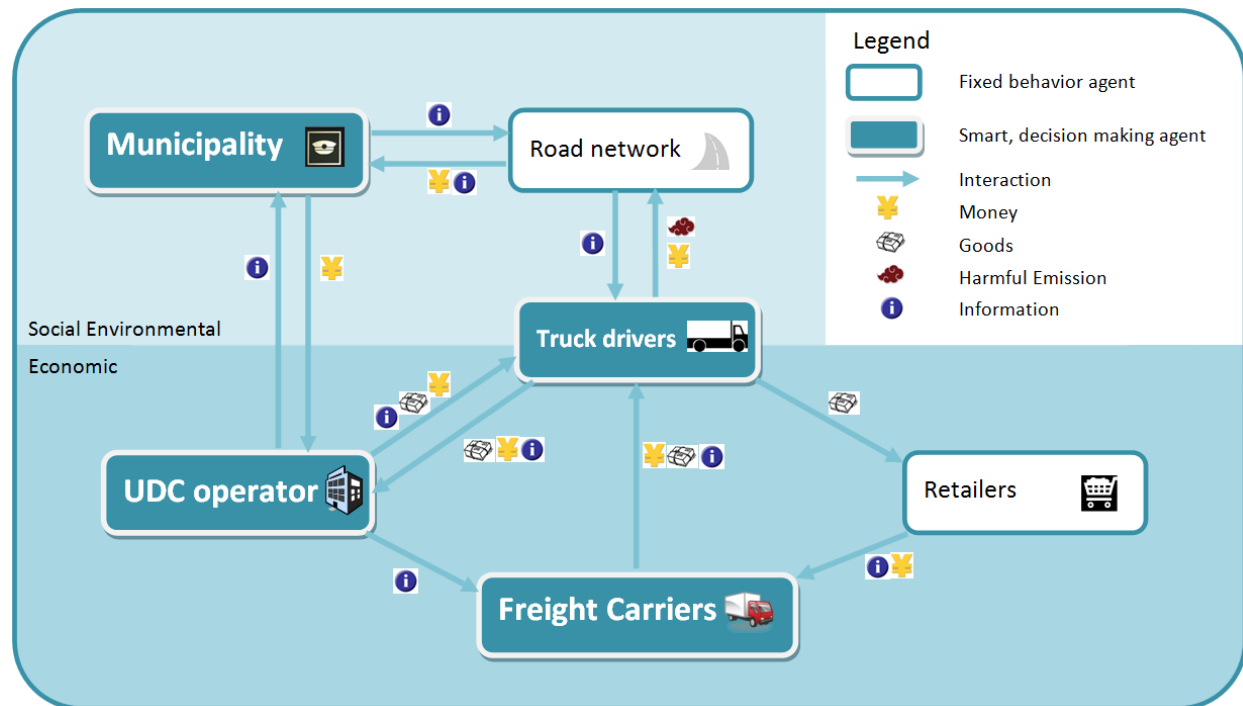


Figure 1: Schematic representation of the multi-agent model

In Figure 2 the input and output parameters are presented. The fee setting for the UDC can be dynamic or fixed at certain levels. The congestion rate can have three levels (low, medium or high (corresponding with an average speed of 13 km/hour, 15 km/hour and 18 km/hour respectively). The toll can be set to 0 Yen, 400 Yen and 800 Yen. The UDC delivery scheme can be early delivery scheme, fixed time delivery or full truck delivery scheme. Subsidy can ranges from 0% subsidy on the fee price to 50%.

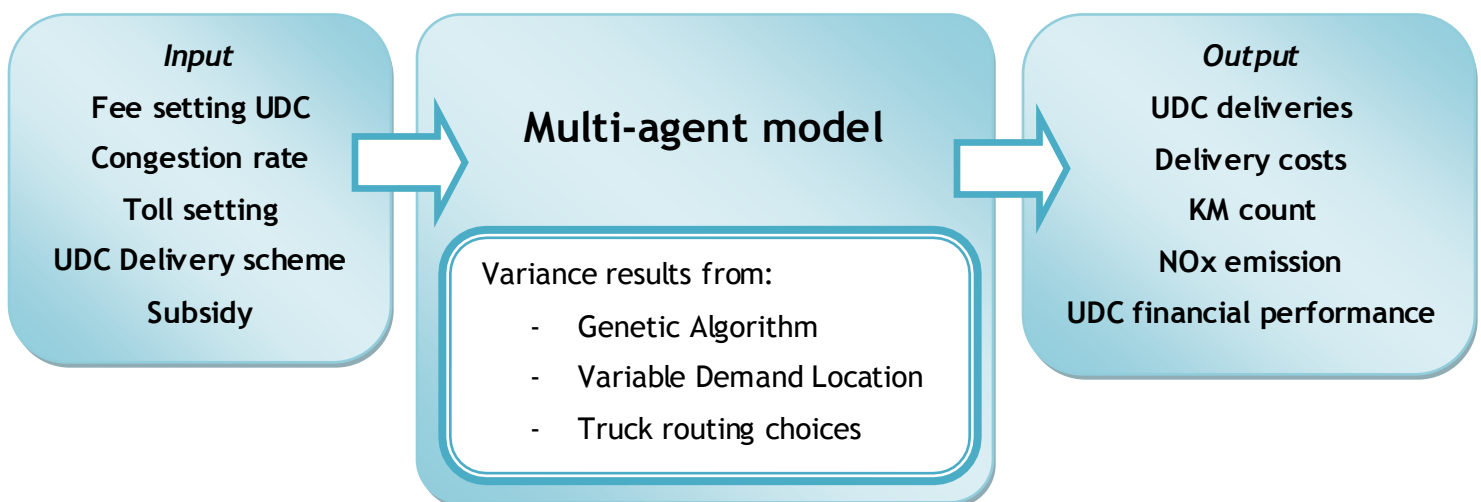


Figure 2: Schematic representation of input, output and variance account for the UDC multi-agent model

Variance is caused by internal Netlogo programming, input parameters and the multi-agent model. Inside the multi-agent model the genetic algorithm, the variable demand location and truck routing choices add variability. The output is recorded in the number of UDC deliveries, delivery costs for freight carriers and the UDC, KM count, NOx emission and the UDC financial performance.

Results: Policy measures and their impact on the business case

First it has to be checked if an increasing UDC usage percentage decreases both NOx emission and km count. The NOx emission decreases with 19.0% with standard deviation of 13.7 % and the km count decreases with 18.8% with standard deviation of 12.9% in reference with no UDC reference case. This was comparing the 0% UDC usage case with the cases in the interval between 70% and 90% UDC usage. Next the impact on the NOx emission and km count decrease for the several input settings are presented. Each presented input settings consists of 360 model runs for the 36 UDC fee settings.

Table 2: NOx emission and km count information for various input settings

Delivery scheme	Mean difference NOx emission	Standard deviation NOx emission	Mean difference km count	Standard deviation km count
Various Delivery Schemes				
Scheme 1	21.2%	10.6%	17.0%	9.1%
Scheme 2	24.8%	11.3%	19.8%	10.5%
Scheme 3	24.2%	8.2%	18.5%	7.5%
Various Congestion Rates				
Rate 1	12.0%	15.7%	15.4%	13.4%
Rate 2	25.2%	15.0%	19.8%	14.4%
Rate 3	27.5%	9.5%	21.2%	9.2%
Various Toll Rates				
Rate 1	25.2%	15.0%	19.8%	14.4%
Rate 2	24.3%	11.7%	19.8%	10.9%
Rate 3	22.3%	12.5%	17.7%	12.3%

With the information from this table, it is visible that different delivery schemes or toll rates do not have a significant impact on NOx emission decrease or km count reduction. There is however a significant different between the low and high congestion rate of one standard deviation for NOx emission.

Next the impact on four UDC financial parameters is investigated. These parameters present the fee with the highest UDC income per day, the highest UDC income and the fee at the 80% and 20% UDC usage boundary.

Table 3: Financial impact of various input parameters on 4 UDC parameters

	Optimal fee (Yen)	Maximum income per day (Yen)	80% usage boundary (Yen)	20% usage boundary (Yen)
Various Delivery Schemes				
Scheme 1	70	-370.000	50	100
Scheme 2	60	-300.000	55	105
Scheme 3	70	-320.000	40	100
Various Congestion Rates				
Rate 1	50	-430.000	40	80
Rate 2	60	-300.000	50	105
Rate 3	70	-200.000	80	110
Various Toll Rates				
Rate 1	60	-300.000	55	105
Rate 2	70	-300.000	55	105
Rate 3	70	-260.000	55	105
Various Subsidy Rates				
Rate 1	60	-300.000	50	105
Rate 2	75	-150.000	65	130
Rate 3	90	-3.400	80	160
Rate 4	120	300.000	105	210

The only positive business case for the UDC is with a subsidy rate of 50% (rate 4), which indicates the complexity of generating a positive business case for an UDC. For the optimal fee the fee with the highest average income per day was chosen. The fees are lower than prices indicated in literature (leda, 2005).

Results: Dynamic usage of Urban Distribution Centre

For the dynamic UDC fee five cases were tested with varying fee per hour. These are displayed in the figure below.

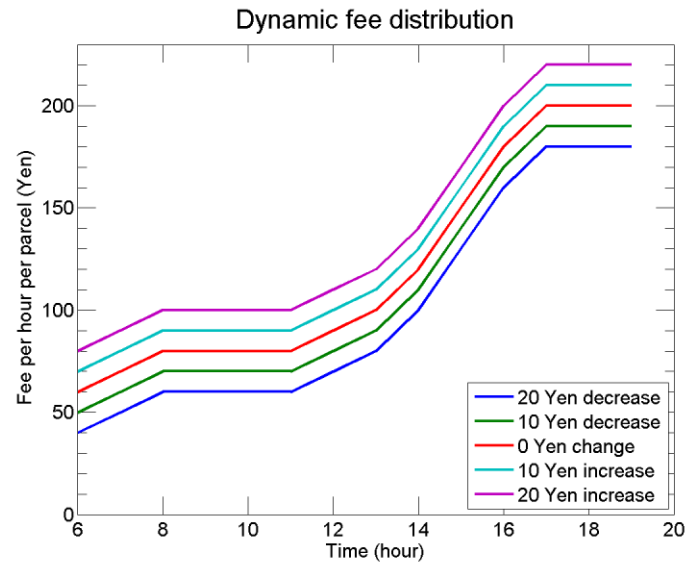


Figure 3: Dynamic UDC fee profile

For this scenario the impact of NOx emission and km count reduction correspond to the scenario with no toll, no subsidy, fixed time UDC delivery scheme and normal congestion. The average UDC usage percentage is presented per fee setting. The mean and corresponding standard deviation for the UDC financial parameters are presented below per fee setting.

Table 4: Mean and standard deviation of the UDC parameters per dynamic fee setting

Fee setting	Mean of UDC usage (%)	Standard deviation of UDC usage (%)	Mean of UDC Income (Yen)	Standard deviation of Income (Yen)
-20	90	11	-368.000	43.700
-10	83	13	-284.000	67.400
0	79	12	-239.000	86.100
10	58	14	-258.000	102.000
20	46	7.6	-342.000	144.000

The differences in mean of the UDC usage percentage are around or more than one standard deviation for the -20, 0, 10 and 20 dynamic fee settings. The 20 setting is not significantly different from any other fee setting due to the high standard deviation for the UDC income. Besides the 20 setting, the -20 setting is different from -10, 0 as well as 10. In total 50 runs were performed.

The fact that the UDC income is almost significantly higher than the -300.000 Yen per day in the similar situation for fixed fee, is highly encouraging. However further and full evaluation of all possible dynamic fee settings is desired to find potential better dynamic fee settings than the currently used settings.

Results: Scenario comparison

At the end of this research three cases are compared for dynamic fee usage: a reference case, highly congested city case and the involved municipality case. In all scenarios the toll rate is zero and the delivery scheme is fixed time delivery. In the reference case the congestion rate is normal, but in the other scenarios the congestion rate is high. In the involved municipality scenario a subsidy of 33% of the fee is provided by the municipality.

Comparing the three scenarios there is no significant difference between the NOx emission or km count reduction and the freight carriers costs. For the UDC income there is a significant increase in the third scenario with respect to the scenarios. Also for the municipality there is a significant increase in costs in the third scenario.

Since there is no significant decrease in NOx emission or km count between the second and third scenario, the municipality will need other incentives like social or corporate well-being associated with the UDC, before the municipality will participate in providing such high and long term subsidies. Other benefits like reduction in road maintenance are possible.

From this comparison it is clear that other incentives need to be present for the municipality to participate. Therefore for a UDC to be profitable, it seems to be necessary to receive subsidy permanently or/and to find other activities like for example pre-retail activities (See Figure 12) to increase profitability. This less optimistic analysis of the UDC analysis relates to the low UDC success rate of the research of Browne, Sweet et al (2005). The conditions for generating a successful business case for the UDC are hard and limited.

Further research of dynamic fee settings and investigation of the possibility to implement this in a real-life case can enhance the future prospective and success rate for urban distribution centres.

Table of Contents

Abstract	ii
Summary.....	iv
Table of Contents	1
1. Research Proposal.....	4
1.1. Problem Identification	4
1.2. Research objective and questions	5
1.3. Research Methodology.....	6
1.4. Scientific and Management Relevance.....	8
1.5. Thesis Outline	8
2. City Logistics Modelling	9
2.1. City logistics research field	9
2.2. General Model Generation Framework	11
2.3. City Logistics Modelling Framework	12
2.4. Well-known Large Iterative Models	14
2.5. Data Availability and Reliability	15
2.6. Conclusion	16
3. Urban distribution centres.....	18
3.1. Urban distribution centre concept.....	18
3.2. SWOT analysis	20
3.3. Critical Success factors for an UDC	23
3.4. Basic business model behind UDC.....	24
3.5. Conclusion	26
4. Modelling an UDC.....	28
4.1. Existing modelling efforts on UDC	28
4.2. Other models	30
4.3. Model for this research.....	32
4.4. Conclusion	33
5. Multi-Agent Model for a dynamic Urban distribution Centre	34

5.1. Concept of multi-agent model	34
5.2. Problem formulation and Actor identification	35
5.3. System Identification and decomposition	37
5.4. Behavior Identification.....	42
5.5. Formalization and Implementation	48
5.6. Model verification	51
5.7. Model representation in Netlogo.....	54
5.8. Experimental Set-up.....	55
5.9. Model validation and use.....	55
5.10. Conclusion	55
6. Results of Dynamic usage of Urban Distribution Centre	57
6.1. Effect UDC on NOx and km count.....	57
6.2. Effect fee on UDC usage for various delivery schemes.....	58
6.3. Effect of congestion on fee vs. usage.....	59
6.4. Effect of toll on fee vs. usage.....	61
6.5. Impact of subsidy on the UDC business case	63
6.6. Impact of dynamic fee on the UDC business case	65
6.7. Conclusion	66
7. Policy measures and their impact on the business case	68
7.1. Impact of scenarios on the UDC.....	69
7.2. Impact on environmental parameters	69
7.3. Impact on freight carriers.....	70
7.4. Impact on municipality	71
7.5. Conclusion	71
Conclusion	73
Reflection and Recommendation.....	76
References	79
Appendices	84
A Additional results	85
A.1 Various delivery schemes	85

A.2 Various congestion rates.....	89
A.3 Various toll rates.....	92
A.4 Subsidy for the UDC	95
A.5 Scenarios	96
B Multi-agent model	100
B.1 Problem formulation and Actor identification	101
B.2 Model assumptions and choices.....	101
B.3 Model verification	117
B.4 Experiment setup.....	125
C Freight carrier prices recalculation with inflation.....	126
D Road network traffic conditions.....	128
E VRPTW with the Genetic algorithm	130
E.1 Algorithms for the NP-hard travelling sales man problem	130
E.2 Main components of the Genetic Algorithm.....	132
E.3 Initialization	134
E.4 Body of the Genetic Algorithm.....	136
E.5 Genetic algorithm Validation in Netlogo	142
E.6 Conclusion	143
F GA verification FC.....	144
G GA verification UDC	147
H Q learning	163
I Netlogo code.....	165

1. Research Proposal

1.1. Problem Identification

New York, Moscow, Paris, Tokyo, Delhi, Chicago and London are examples of large cities in today's world. Cities in the developed world have advanced systems for housing, sanitation, transportation and various other utilities. The high density of people and utilities greatly facilitates business as a relatively large portion of GDP is earned inside highly urbanized areas (OECD, 2009). In recent decades cities are getting larger and becoming denser, which put tremendous demand on business deliveries and passenger transport. As a result traffic jams increase in number and severity. The increased importance of environmental and social impacts of such issues adds additional dimensions to the city's problem. Therefore efforts are made to improve the traffic flow inside the city. Passenger transport is investigated relatively well inside the city (Bestufs website), but a gap in knowledge is present on the freight transport inside the city. Freight transport contributes an estimated 10% - 30% of the traffic flow in various cities (Ministry of Land Infrastructure and Transport Japan)(Crainic and Ricciardi, 2009), but an estimated 40% in pollution and noise (COST321, 1998). This trend of urbanization and increased transport has lead to the research field of City Logistics. In 1999 the definition of City Logistics was given by prof. Taniguchi in the first International City Logistics conference (Taniguchi, Thompson, 1999):

City Logistics: The process for totally optimizing the logistics and transport activities by private companies in urban areas while considering the traffic environment, traffic congestion and energy consumption within the framework of a free market economy.

In this field efforts are focused on data collection, traffic modelling, vehicle routing, multi-agent modelling and the investigation of the potential impact of policy measures. One of the proposed policy measures, besides toll ways and time windows for trucks in the inner city, is the urban distribution centre (Dabanc, 2007). The urban distribution centre is a promising concept, where the load of entering trucks is transferred to new trucks to increase the load factor and to allow for easier time-windowed operation (to avoid traffic jams) (Quak & de Koster, 2009). A higher load factor in the city can decrease harmful effects associated with city logistics. In practice the concept has failed due to stakeholder behavior as freight transport itself is under high pressure with strong competition and the Just-In-Time (JIT) delivery system (Germain & Droge, 1996). The unsuccessful business case behind many urban distribution centres lead to an unhealthy financial status of the urban distribution centres itself (van Duin and Quak, 2008). In previous modelling attempts the urban distribution centre

appeared to be successful in many cases (Marcucci and Danielis, 2008), which is in sharp contrast with the fact that only 15 out of 200 centres were running after 5 years (Browne and Sweet, 2005). This difference could be caused by the fact that most models use average values for estimations (for traffic flows and demand for example). In most of these models the individual stakeholders' perspectives were not taken into account and a general society perspective was adapted (in contrast to Wisetjindawat and Sano (2007) and Zargayouna and Balbo (2008)). In these static situations (constant cost/demand/traffic flow) a global optimum is sought, but not the actual impact on routing is estimated for various agents. Therefore there is a desire to use a multi-agent model and find out if (and under what conditions) this promising concept of the urban distribution centre can contribute to enhanced city logistics.

1.2. Research objective and questions

1.2.1. Research objective

This research project addresses the issue of freight transport in cities and more precise the mixed predictions and results of urban distribution centres (UDCs). The main objective of this research is to investigate the dynamic usage of urban distribution centres with multi-agent modelling in a dynamic situation. The discrepancy between actual and expected UDC performance could be caused by the bird view representation of reality in current modelling. Using more dynamic multi-agent model with individual actor perspective can potentially represent reality more closely. By modelling a virtual city with an urban distribution centre in a dynamic multi-agent model, an improved business case can be developed for urban distribution centres and a gap in literature can be addressed.

Collaboration in this research is made between Kyoto University and Delft University of Technology as excellent knowledge on this topic is present at both universities and the model will be based on building blocks from modelling research performed in earlier studies at Kyoto University.

1.2.2. Research Questions

Therefore the main question to be addressed is:

What is the impact of dynamic urban distribution center usage on city logistic parameters investigated with multi-agent modelling?

In order to answer this question the following sub questions have to be answered.

- 1) Which models can describe city logistics processes with an UDC concept?
- 2) What are critical success factors for an urban distribution centre and how does an urban distribution centre contribute to the city logistics?
- 3) Which model of an urban distribution centre can include most critical success factors?
- 4) What policy measures contribute to the successful functioning of the urban distribution centre?
- 5) How do various scenarios impact the business case of an urban distribution centre?

In the first question various important models in city logistics will be discussed in order to illustrate the contours of the general research field and find relevant models for this project. In the second question the general concept of an urban distribution centre will be explained and various cases will be discussed. With the third question a multi-agent model with vehicle routing and traffic simulation will be developed and combined with a multi-agent model. With the model in place, the impact of various policy measures on the urban distribution centre and other stakeholders can be investigated to answer the fourth question. In the last question the impact on a simple business case for the urban distribution is given as well as policy recommendations to increase the success rate for urban distribution centres.

1.3. Research Methodology

In this research the focus is on obtaining information from modelling efforts to answer the research question. The main effort is towards generating a multi-agent model to describe the situation for the UDC. A multi-agent model is tool to incorporate individual actor choices and their impact on the general performance of the system. Data collection is highly important to the research field, but due to time limits this will not be performed in this research.

Before modelling a brief literature review will be conducted to investigate the important parameters/characteristics for modelling with respect to multi-agent modelling and urban distribution centres. The literature review will provide practical knowledge on modelling approaches as well as a theoretical framework for stakeholders. The total model will consist of various layers with top-down policy measures, a multi-agent model, vehicle routing, dynamic traffic conditions and output parameters for impact as well as the UDC performance (see Chapter 2).

Most important here is the general stakeholder framework for the multi-agent model as presented in Chapter 5. A multi-agent model is chosen to capture the complexity of individual behaviour towards a societal goal. With cost benefit analyses it is hard to capture this complexity and emergent behaviour. The framework determines stakeholder behaviour and

therefore model output. A large part of the research will focus on how to incorporate interaction and decision making into this multi-agent model. A simplification of reality is inevitable and needed to reduce computation time and complexity. Various modelling components like vehicle routing are present at Kyoto University. Efforts were centred at combining the available information and adding missing parts to generate a model for dynamic usage of UDCs and to answer the research question.

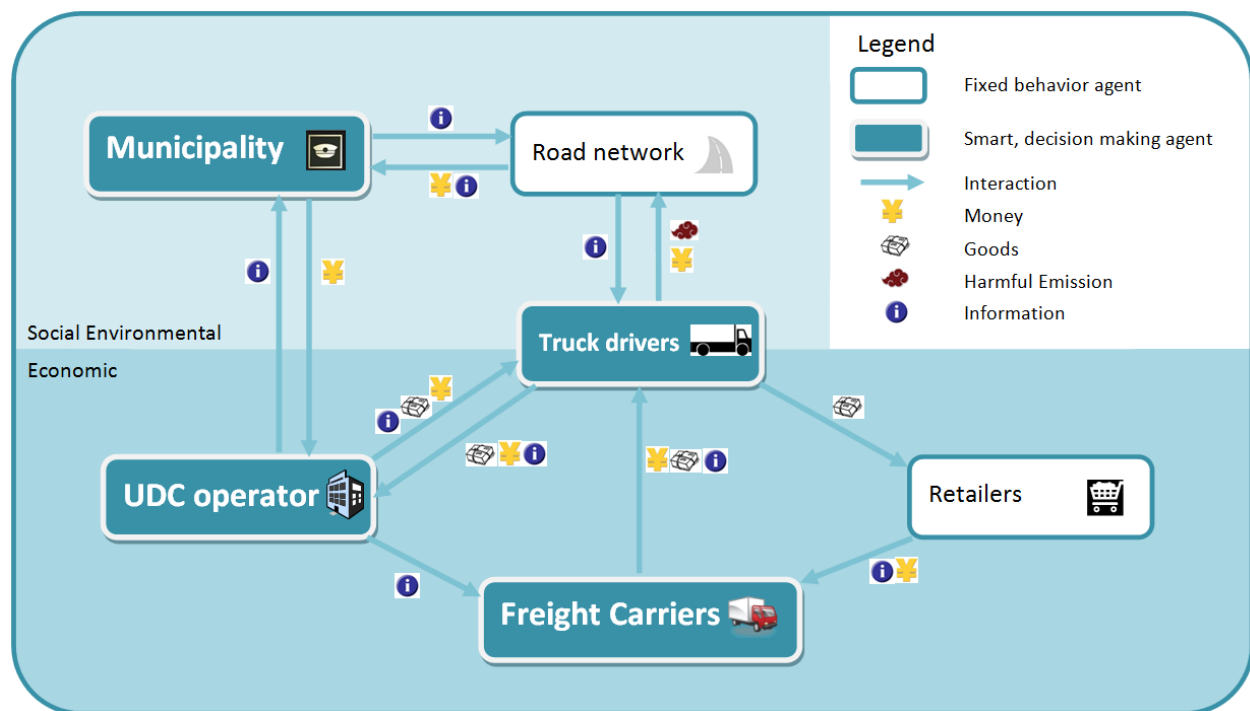


Figure 4: Schematic representation of the multi-agent model

In all modelling approaches the way to generate a model is iterating the following steps: construct (and add to) a framework, convert this to a computer model and validate the output (preferable with comparison to reality). This process is described in Figure 5 and more in-depth in Chapter 2. This approach is in agreement with Banks (1998) and Shannon (1998).

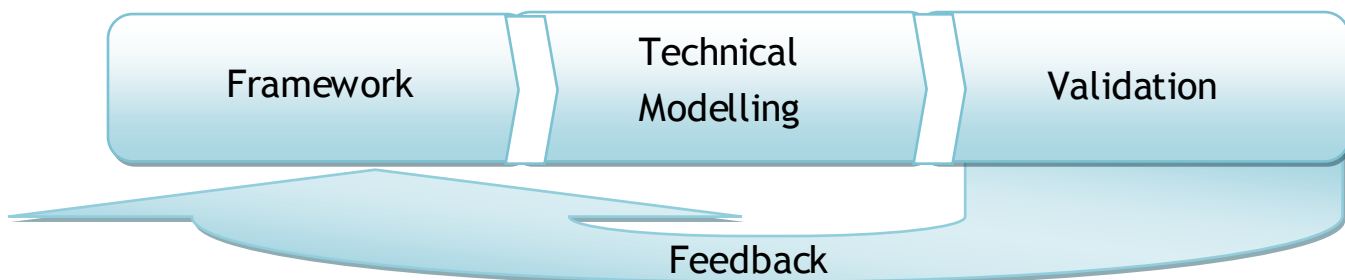


Figure 5: Schematic representation of the approach to generate a model

1.4. Scientific and Management Relevance

1.4.1. Scientific relevance

Urban distribution centres are promising concepts to contribute to improved city logistics, but in practise they often fail in contradiction to theoretic predictions (van Duin & Quak, 2008). UDCs have been modeled in static situations, thereby lacking dynamics of traffic and actor behavior. A modelling approach, enabling dynamic usage and incorporating actor behaviour, could shine light on the explanation why UDCs fail and how their performance can be improved. The dynamical component is expected to add serious complexity and to bring the model closer to reality in combination with the multi-agent system. Thereby this research hopes to contribute to fill this gap in literature on UDCs as well as to contribute towards improved practical usage of UDCs.

1.4.2. Managerial implications

The managerial implications are large as this will help determining if and how to use UDCs. The multi-agent modelling will contribute to the understanding of how to deal with the various stakeholders. Policy makers can benefit with guidance to decide how and when to use UDCs. It has impact on freight companies as they will change their routings and their supply chain, which has a serious impact. Also for management of UDCs themselves, this research is highly relevant and beneficial as it can contribute to a successful business case.

1.5. Thesis Outline

In the first part a literature is presented on City logistics modelling (Chapter 2) with more in-depth information on the approach towards generating models. In Chapter 3 the UDC is investigated to understand the behavior characteristics and performance. Further more in Chapter 4 some previous model approaches for the UDC and other interesting models are presented. The second part is Chapter 5. In Chapter 5 the created multi-agent model is presented and validated. The required vehicle routing for this model is presented in Appendix E: VRPTW with the Genetic Algorithm. In the third part, results from the model are presented in Chapter 6, while in Chapter 7 three main scenarios are compared with each other. This is followed by a conclusion and additional information in more appendices.

2. City Logistics Modelling

A short literature review is presented here as an introduction into the field of City Logistics with the focus on modelling. This fast growing field deals with freight traffic inside urban areas and investigates this from a social, environmental and economic perspective. The research area is very heterogenic from technical traffic modelling and algorithms to models and surveys to support policy making. This review focuses strongly on the modelling aspects in City Logistics as the core of this thesis is constructing a model. Therefore the general approach to produce a model is discussed as well as the lay-out for vehicle routing based modelling. The chapter finishes with describing current pressing issues in City Logistics modelling and a conclusion.

2.1. City logistics research field

City logistics field is a fast growing research field. The definition for City Logistics was given by prof. Taniguchi in the first International City Logistics conference as *'The process for totally optimizing the logistics and transport activities by private companies in urban areas while considering the traffic environment, traffic congestion and energy consumption within the framework of a free market economy.'* (Taniguchi & Thompson, 1999)

Current efforts can be divided in 1) policy oriented/quantitative research, 2) empirical research and 3) modelling/simulation-oriented research (Vleugel, 2004). Data collection, vehicle routing and the potential impact of policies measures are the main issues. A large simplification of reality is required due to the complex nature of City Logistics with many involved actors and reticence. Serious effort is put towards data collection to verify the models and increase reliability. Policy measure models contribute to practical applications and solutions.

It is interesting to see how the field of City logistics has developed since 1999. Therefore the number of articles on the field of city logistics versus time is presented in Figure 6 (ISI Web of Knowledge) and Figure 7 (Scopus). The number of articles here is a large underestimation of the actual number for three reasons. Many articles are not published in English (prof. Taniguchi heads a Japanese group). Many important articles presented at the yearly City Logistics conferences are not in Scopus or ISI Web of Knowledge. Additionally various logistics magazines are not listed in the ISI Web of Knowledge or Scopus database. Therefore the actual number is dismissed, but the global trend is clearly increasing, which is in agreement with Quak and van Duin (2008), where submissions to the International City Logistics Conferences were analyzed. In 2010 the total amount of articles is lower as this

analysis was made in April. When investigating the content of the field after 2008 an expansion is visible as research on risk incorporation by natural and manmade hazards (Taniguchi and Thompson, 2009).

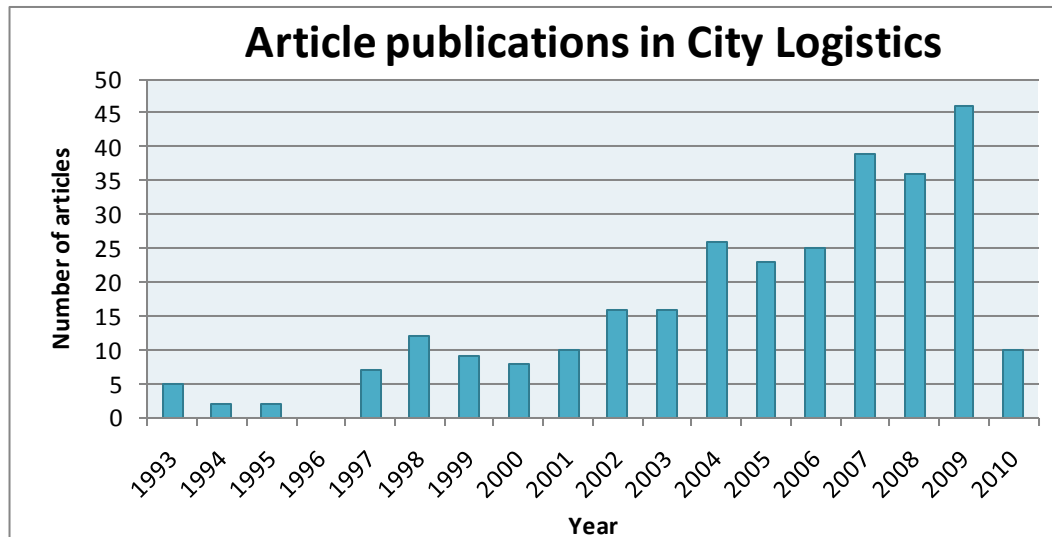


Figure 6: Amount of articles on ISI Web of Knowledge with search terms City Logistics, Urban Logistics and Urban Freight.

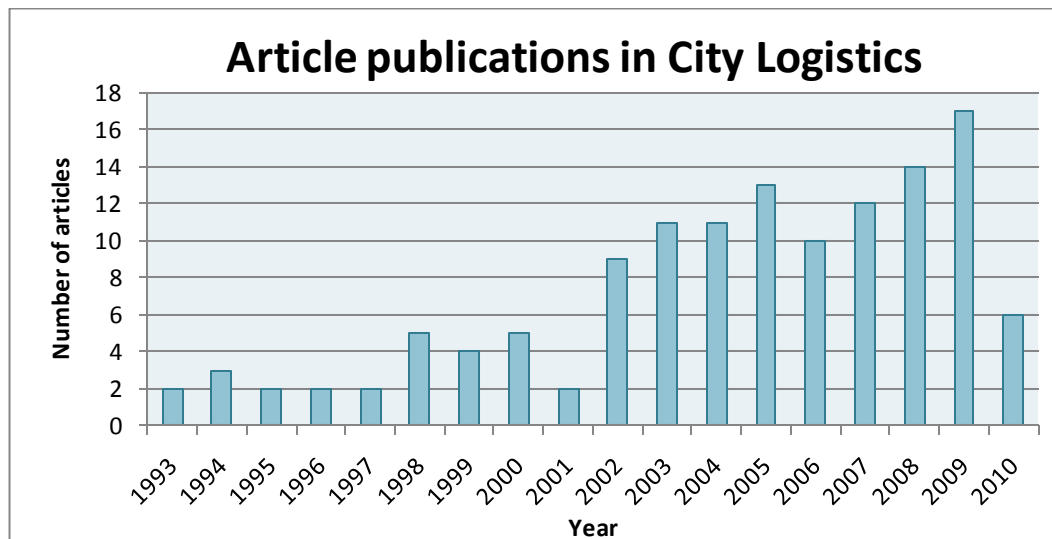


Figure 7: Amount of articles on Scopus with search terms City Logistics, Urban Logistics and Urban Freight.

For general review articles on various aspects of city logistics the following articles are recommended. In Russo and Comi (2009) main aspects of the city logistics field are presented and a general classification of policy measures is proposed. In Muñuzuri and Cortés (2009) the

concepts behind modelling are explained, followed by a very interesting model with entropy maximization technique applied to Seville, Spain. The application of French models to UK cities is presented by Patier and Browne (2009) in order to contribute to a consistent methodology. In Benjelloun and Crainic, (2009) an overview of city logistics projects activity per country is presented. A good overview of the entire research activities on city logistics is presented by Quak and van Duin (2008).

Before presenting the regular approach in City Logistics for generating large vehicle routing models, the general layout to create any model is explained to clarify the taken approach for building a model.

2.2. General Model Generation Framework

In this research a new model was build and therefore it is useful to investigate what approach is taken in generating a model. For any model the following steps have to be taken: construct a framework, generate the model and validation of the model (Figure 8). This approach is in agreement with Banks (1998) and Shannon (1998).

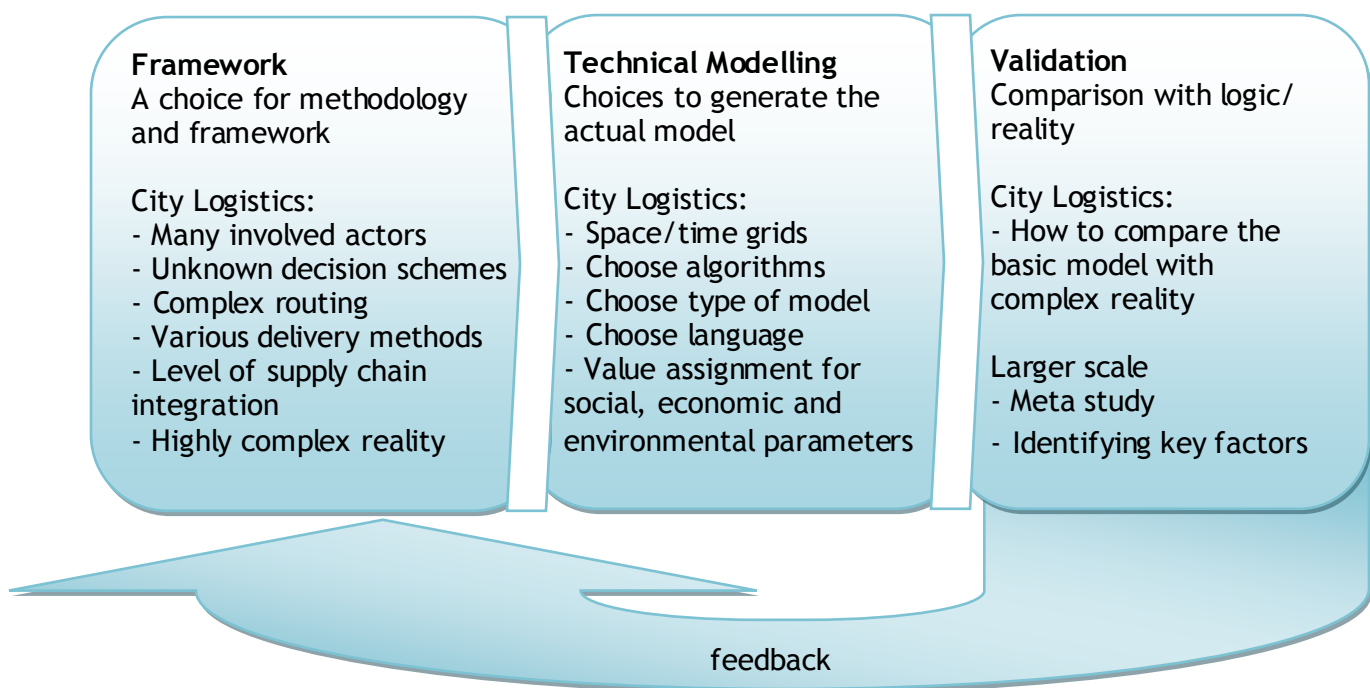


Figure 8: General modelling approach with specific information for city logistics

Initially a framework is constructed by choosing a limited topic, finding relevant actors and identifying key interactions. Choices are made upon what part of reality to model and how to represent it. In the next step (technical modelling) the translation from a framework into a

model is made by choosing solving algorithms and computer representations. These choices can actually have severe impact on the output. After this phase the model is ready and requires validation to check if it accurately represents reality. Validation in City Logistics is complex due to a lack of data. Validation without data consists of comparison with common sense cases and extreme cases, while regular validation compares model results to reality.

When making an actual model, normally first a basic model is developed (and validated by passing all stages) before adding more complex features to resemble reality. When the total model is finished, the output can be used to draw conclusions or answer research questions. In City Logistics the framework is highly complex and validation with reality is highly challenging due to a lack of available data. This will be discussed further in Data Availability and Reliability, but first the steps for generating a City Logistics vehicle routing model are described.

2.3. City Logistics Modelling Framework

In city logistics much effort is towards vehicle routing modelling, but also game theoretic models and cost benefit analysis exist for example. Before the 1990's urban goods movement, modelling was generally based on costs (transport and facility costs), but later a shift towards policy oriented modelling took place (Routhier, 2006). Policy oriented modelling tries to understand urban truck movement under the influence of various policies. In the vehicle routing models the general approach is to first estimate commodities or trucks origins and destinations. Next, vehicle routing calculates the optimal route for the required goods or destinations. In the next step further influence of traffic conditions can be taken into account in traffic calculation and finally output parameters are generated. Possible agent interaction (in the form of a multi-agent model for example) can influence everything or potential policies are simulated to see effects on sustainability and economic/social aspects. This is displayed in Figure 9.

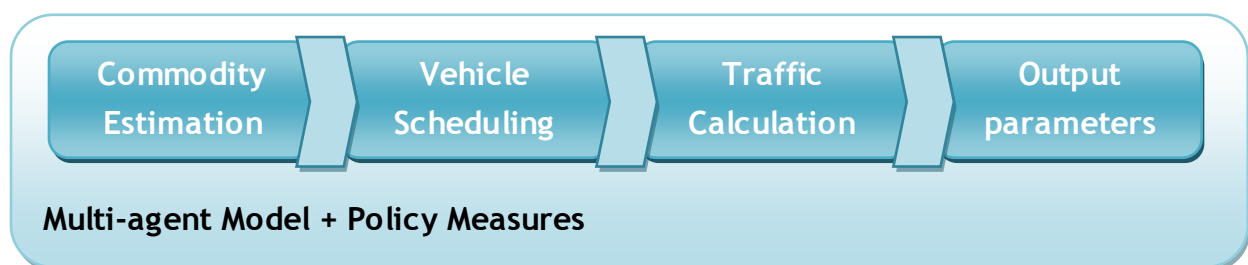


Figure 9: Framework of Model layout in City Logistics modelling

2.3.1. Traffic flow

In order to estimate the impact on traffic flow, first commodity or vehicle estimations need to be made as input for the model (Muñuzuri and Cortés, 2009). Here the transported commodities can be estimated on socio-economic demographic parameters of various sectors, as 81% of total trucks trips inside the city are for distribution and purchases (Russo and Comi, 2004). With the vehicle calculation the amount of trucks is estimated based on the delivery information of the various store types present in the sector as for example a pharmacy has different delivery characteristics as a supermarket with fresh products (Muñuzuri and Cortés, 2009).

With this information vehicle routing is performed to estimate truck movements. This can be done with various algorithms (Wang and Holguín-Veras, 2008). The grid and time windows have to be taken into account to include effects of rush hour (Woudsma, 2001). Here a representation of an actual city can serve as the basis of the grid or a virtual grid can be used to investigate effects and modelling behavior (Nakamura and Taniguchi, 2008).

The results from vehicle routing can directly serve as the impact on traffic flow or as input for large traffic simulators. Currently, Intelligent Transport Systems to contribute to real-time enhanced trip planning are investigated with these simulators (Taylor, 2004), (Tseng and Lin, 2008), (Taniguchi and Shimamoto, 2004). Within various groups different methods are used for these steps.

2.3.2. Output parameters

The output parameters of the model are generally represented with social, environmental and economic scale by most authors (Taniguchi and Thompson, 2001), (Patier and Brown 2009). Social output is the reaction of the urban inhabitants in complaints about noise or traffic jams for example. For environmental output harmful emissions like CO₂ or noise are taken as indicators. For economic output the profit or turnover of freight carriers are used or financial output of other actors. This information can then be used for cost/benefit analysis as in van Duin and Quak (2008). With these output parameters the impact of various policies can be investigated.

Policy measures

Policy measures to reduce environmental impact or other purposes can be in the model by comparing various scenarios (including a reference case). Examples are the presence of an urban distribution centre and toll ways (Russo and Comi, 2009), (Browne and Piotrowska, 2008). These measures influence the total model and are therefore represented as overlaying box. An evaluation of various policies is given by Russo and Comi (2009).

Multi-agent modelling

Another layer of complexity is to implement a multi-agent model. With a multi-agent model individual stakeholder behavior and actions are modeled, which in turn has impact on the overall outcome of the model. With these models for example the effect of price settings by freight carriers on the global outcome can be estimated (Tamagawa and Taniguchi, 2009), (Roorda and Cavalrcante, 2010). Another example is the implementation of residents complaining after experiencing high noise levels. With a multi-agent modelling these individual choices and reactions are included in the model (Taniguchi and Thompson, 2008). Also learning behavior and reaction time can be included (Watkins and Dayan, 1992). Further development of specific stakeholder modelling lead to the auction concept in (van Duin and Tavasszy 2008), but also game theory and other negotiation techniques can be included to imitate human behavior (Anand and van Duin, 2010).

Validation

Here the interesting case is the comparison of the model results of a city with the behavior of the actual city as was done by Muñuzuri and Cortés (2009), Patier and Browne (2009) and Christian and Meimbresse (2008). However a lack of data and the high complexity of the model representation of a city make this a less common approach. Most often virtual grids are used to reduce calculation time, to clearly image impact of measures and to test various modelling options. Then conclusions are extended to reality carefully by comparing two cases.

2.4. Well-known Large Iterative Models

Currently there are three main models (Freturb, Goodtrip and Wiver) in City Logistics and they will be described here briefly as much literature is present on this. These models are iterative models of traffic flow combined with external effects or policy measures.

Freturb as described by Routhier (2006) is a French model, which uses survey based urban goods estimations and socio-economic zoning to describe traffic flow and estimate policy measures. With the demand accessibility of various zones is determined and an Origin/destination matrix is assigned. Now traffic calculations are made, so energy consumption and environmental parameters can be established. Also various policy measures can be incorporated in the model. The model uses data from among others Marseille and Dijon.

Goodtrip is a Dutch model (Boerkamps and van Binsbergen, 1999). In this model estimated goods flows, urban freight traffic and its impacts. From the goods demand and supply the flow of goods is estimated. Via transport services the traffic flow is generated. The traffic

flows generates congestion feedback and from these results environmental and societal impacts are calculated. This is compared to the City of Groningen.

Wiver, a German model (Meimbresse and Sonntag, 2000), makes the complexity of city logistics explicit by using multiple vehicle classes and trip chains. By taking several sectors in account the purpose of the trips is explicit and considerations on efficiency can be made.

Main issues for all models in City Logistics however remain model reliability and validation.

2.5. Data Availability and Reliability

Data availability and reliability are pressing issues for the City Logistics field and complicate the model validation, which is needed for feedback on the models. This is discussed below in data collection. Data availability and reliability is difficult due to the complex nature of city logistics, which is explained in model complexity. In the last part the idea to generate simple business cases is mentioned in order to achieve larger stakeholder and policy maker involvement to obtain more data.

2.5.1. Data collection

Data collection is highly important to verify the models and increase understanding on the relevant parameters (Bestuufs website). Often no data of needed variables is present. The highly competitive freight market complicates the data collection as companies are not willing to provide inside information about their freight transport for competitive reasons. Also different data sets are acquired with the different methodologies, thereby complicating comparison. Conducting meta-studies and generalizing approaches towards data collection would contribute to characterization of the generated models (Patier and Browne, 2008).

2.5.2. Model complexity

The comparison of city logistics models to reality for verification is highly complex due to the large amount of actors, unknown goods flows and a large batch of traffic parameters (Patier and Browne, 2009). Also delivery systems are changing and now goods can be delivered to the final customer in various ways like direct home delivery and distribution centre pick-up. A shift from pick-up to home delivery after internet ordering by individual consumers increases uncertainty in models (Chopra, 2003).

Also steps are taken towards dynamic modelling. The solution to a static problem with average parameters can vary significantly from dynamic and variable parameters. This results in complex models, which require high computational power and smart algorithms like the genetic algorithm (Reeves, 1993) and entropy maximization method (Wang and Holguín-Veras,

2009). Different approaches need to be developed further and compared to reality for verification. The next step is comparison in performance between the models (with for example Solomon benchmarks, so appropriate techniques can crystallize out.

2.5.3. Business case development

In various literature it is stated that the acknowledgement of policy makers about the importance and impact of city logistics is still not established properly (Bestufs website) even though the number of city logistics projects in collaboration with the government are increasing (Benjelloun and Crainic, 2009). A potential cause could be that clear factual information about the expected importance or impact of the improved city logistics is not present on main collaboration platforms like BESTUFS. A business case could be a desirable way for communication. The business case is difficult to generate due to lack of information and comparisons, but several initial attempts have been made by van Duin and Quak (2008) and Ruesch and Hegi (2008). The effects can be communicated more clearly with scenarios and estimations. This should motivate policy makers and other stakeholders to enhance collaboration, which would result in more possibilities for data collection and thus model verification.

2.6. Conclusion

The field of city logistics is a fast developing field as can be seen by the increase in articles over the last years. Much effort inside city logistics is focused on modelling. Most efforts are based on the approach, where first vehicle/commodity quantities are estimated followed by vehicle routing and impact calculations. Now different policies can be compared on basis of the output of various scenarios. Due to the highly complex nature of city logistics with many stakeholders some multi-agent models are generated to include their interactions explicitly and include individual benefits.

Model validation remains highly complex due to low data availability and reliability, but examples are present. A start has been made to work towards a more global methodology by various authors. However a step further has to be taken towards dynamic and complex modelling as this will increase resemblance with reality. By incorporating multi-agent modelling, emergent behavior can be modeled. Together with goods flow generation and vehicle routing this will then represent more non-linearity as present in reality.

To improve data availability and reliability more data is required. A businesses case could highlight the benefits of city logistics to policy makers to increase their interest. Business cases can enhance communication between involved stakeholders to gain acceptance and collaboration among the many stakeholders.

Answering sub-question 1

The sub question "Which models can describe city logistics processes with an UDC concept?" provides information for the general layout of a City Logistics model. At the origin of the model estimation for truck flow with origin-destination-matrices or commodity flow is needed. Subsequently vehicle routing has to be performed. The routing affects social, environmental and economic parameters. With this model impacts from policy measures or individual actor behavior (in a multi-agent model) can be calculated. This information provided a guideline and examples for constructing the UDC model.

3. Urban distribution centres

This chapter contains information about urban distribution centres. In the first section the concept behind the urban distribution centre is explained to indicate why benefits are expected. Net success factors, identified in literature, are presented to identify key factors to include in the model. Next, current modelling efforts on urban distribution centres are presented.

3.1. Urban distribution centre concept

The urban distribution centre is schematically represented in Figure 10.

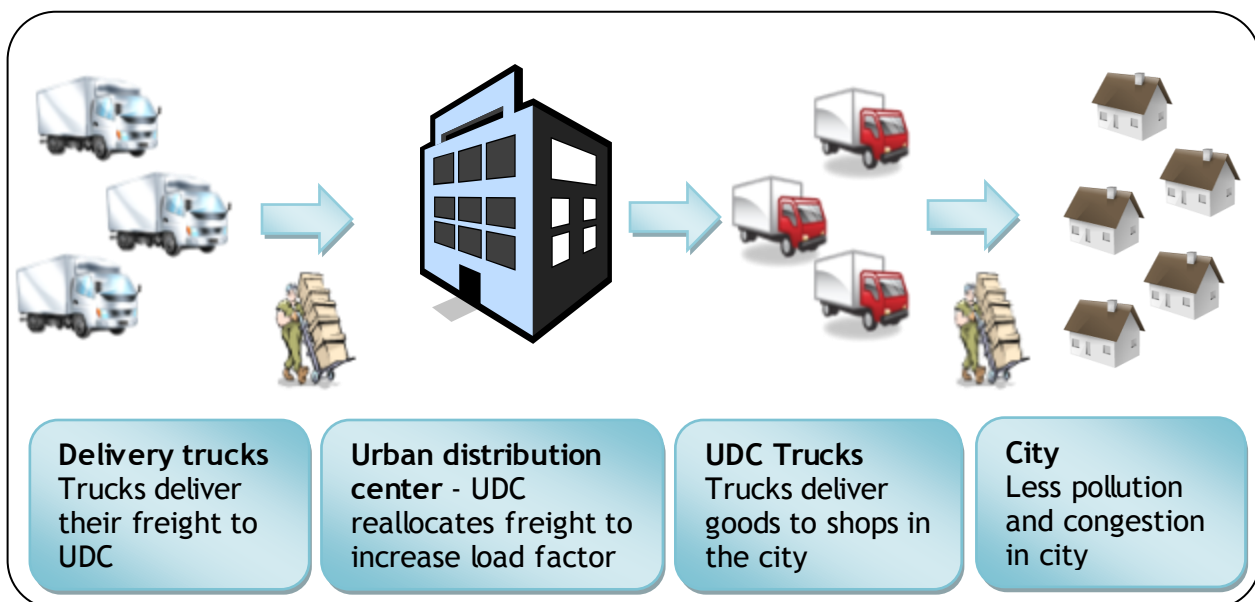


Figure 10: Schematic representation of the concept behind Urban Distribution Center

Currently many (large) trucks enter the city centre to deliver parcels (estimated 10-30 % of traffic by Crainic and Ricciardi (2009)). These trucks cause inconvenience at loading and unloading sites, increase congestion and emit large amounts of undesired gasses (Delaître, 2009) (Quak and de Koster, 2009). Currently there are already many congestion problems in the city, but in the future deliveries are expected to increase (Marcurri and Danielis, 2008). Thereby increasing harm to the business and residents of the city. However many of these trucks have a low load factor or even drive empty (Freight Best Practice Benchmarking guide, 2006). This is most likely caused by the high importance of JIT deliveries and the geographic spread of parcel deliveries.

A suggestion to increase load factor, decrease congestion and pollution and improve social living inside the city is the urban distribution centre. The idea is that (large) trucks deliver their freight to the urban distribution centre, which is strategically located preferable at the outskirts of the city close to a high way or large road. Trucks deliver their goods here and do not need to enter the city. The UDC has its own fleet of UDC trucks, which deliver the parcels to the shops inside the city. The UDC trucks can be zero emission vehicles or have other special characteristics like size or man-powered.

Now goods can be bundled more effective for specific locations/areas, thereby trucks drive less vehicle kilometers inside the city with an increased load factor. This reduces emissions and traffic congestion as fewer truck trips are required. The freight carrier avoids getting stuck in traffic jams inside the city. This optimization is depicted in Figure 11.

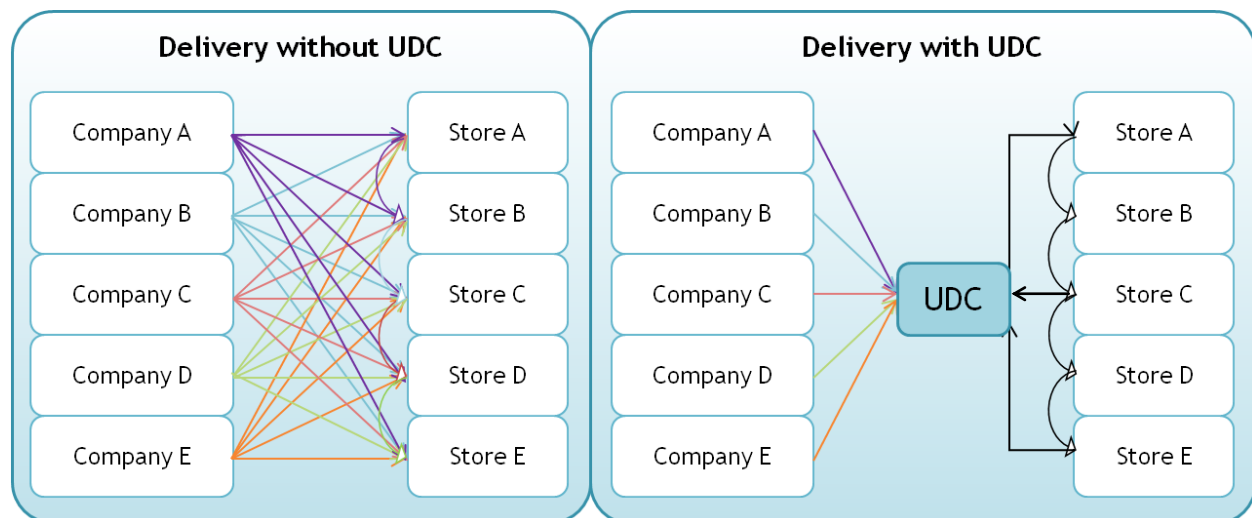


Figure 11: Schematic representation of deliveries in the city with and without a UDC. Note that here the returns trips of the trucks are not taken into account and trucks are assumed to make single deliveries except for the UDC truck.

Often UDCs are implemented in areas with policy measures like time windows to enter the city or tollways. Another potential benefit of the UDC lies in the fact that UDC trucks can enter the city in convenient time windows (so at night or by avoiding rush hour). This increases delivery certainty and reduces inconvenience for freight carriers. One of the big disadvantages of an UDC is an additional transfer of goods, requiring time and a location. With the pressure of JIT deliveries this is a critical factor for carriers. However, the total sum of consolidated and route-optimized deliveries by the UDC is expected to be positive (Marcucci and Danielis, 2008).

3.1.1. Definition for this research

In literature various definitions (and names) exist for an urban distribution centre as shown here with some examples (Browne, Sweet et al, 2005): public distribution depot, central goods sorting point, urban transshipment centre, shared-user urban transshipment depot, freight platforms, cooperative delivery system, consolidation centre, urban distribution centre, city logistics schemes and urban freight consolidation centre (UFCC).

Most of these names use slightly different definitions. This indicates that no consensus has been reached on this topic yet. Secondly it also indicates that specific location might have specific demands on such a centre (Browne, Sweet et al, 2005).

In this research the name Urban Distribution Center (UDC) aims to address wish for the avoidance of the need for vehicles to deliver part loads into urban centers or other large developments. This objective can be achieved by providing facilities whereby deliveries can be consolidated for subsequent delivery into the area in an appropriate vehicle with a high level of load utilisation (Browne, Sweet et al, 2005).

According to the classification by Browne, Sweet et al. (2005) the investigated UDC is a Urban Consolidation Centre (UCC) serving a town/city.

3.2. SWOT analysis

Here the most important advantages and disadvantages are presented as found in various literature sources. Main (and less disputed) results are environmental and social benefits from more efficient and less intrusive transport operations within urban areas. Main disadvantage is the financial viability and additional step in the value chain (requiring additional time and reduction of market) (Marcucci and Danielis, 2008), (Browne, Sweet et al, 2005), (Browne and Piotrowska, 2008), (van Rooijen and Quak, 2009).

Table 5: SWOT analysis for the UDC based on (Marcucci and Danielis, 2008) (Browne, Sweet et al, 2005) (Browne and Piotrowska, 2008) (van Rooijen and Quak, 2009))

	Positive	Negative
Internal	Strengths • Reduction in restocking time in store • Improve load factor • Reduction of large trucks inside city • Reduction of fuel consumption • Reduction of pollution • Reduction of noise • Fewer vehicle kilometers • Reduction of unit cost of urban transport (by larger volume) • Better planning and implementation of logistics operation • Public relations benefits for participants • Theoretical cost benefits from contracting out “last mile”	Weaknesses • Large deliveries (insufficient storage at store) • Additional stage in the supply chain • Security • High start-up costs • Complex to handle various types of goods. (Difficult for a single centre to be able to handle the wide range of goods moving in and out of an urban area, for example due to different handling and storage requirements) • Time penalty • Organizational and contractual problems often limit effectiveness • Loss of the direct interface between suppliers and customers
External	Opportunities • Better inventory control, product availability & customer service (storage service) • Switch from push to pull logistics through better control/visibility of supply chain • Link in with wider policy and regulatory initiatives • Opportunity to introduce new information systems at the same time • Allow better use of resources at delivery locations (pre-retail service) • Opportunity for carrying out value-added activities (storage/ pre-retail service) • Addition of city services (Municipal services, like part of the waste collection by the UDC) • Use of environmental friendly vehicles	Threats • Job reduction • Limited potential scope (Much urban freight is already consolidated at the intra-company level or by parcels carriers, so limited benefits (or even negative consequences) for trying to channel these flows through a consolidation centre.) • Limited effectiveness due to lack of enforcement of regulations • Create monopolistic situation, eliminating competition (and perhaps legal issues)

3.2.1. Detailed Range of potential logistics and pre-retail activities

Browne, Sweet et al. (2005) made a broad analysis on the potential benefits of a urban consolidation centre (UCC) with transport service, storage space service and pre-retail services to indicate the full potential of the UCC concept (see Figure 12). An UCC has transport services, but can expand to storage services and pre-tail services (like pricing of products) to increase its added value. Storage services improve JIT performance as the UCC also acts as a warehouse (depending on the type of deliveries).

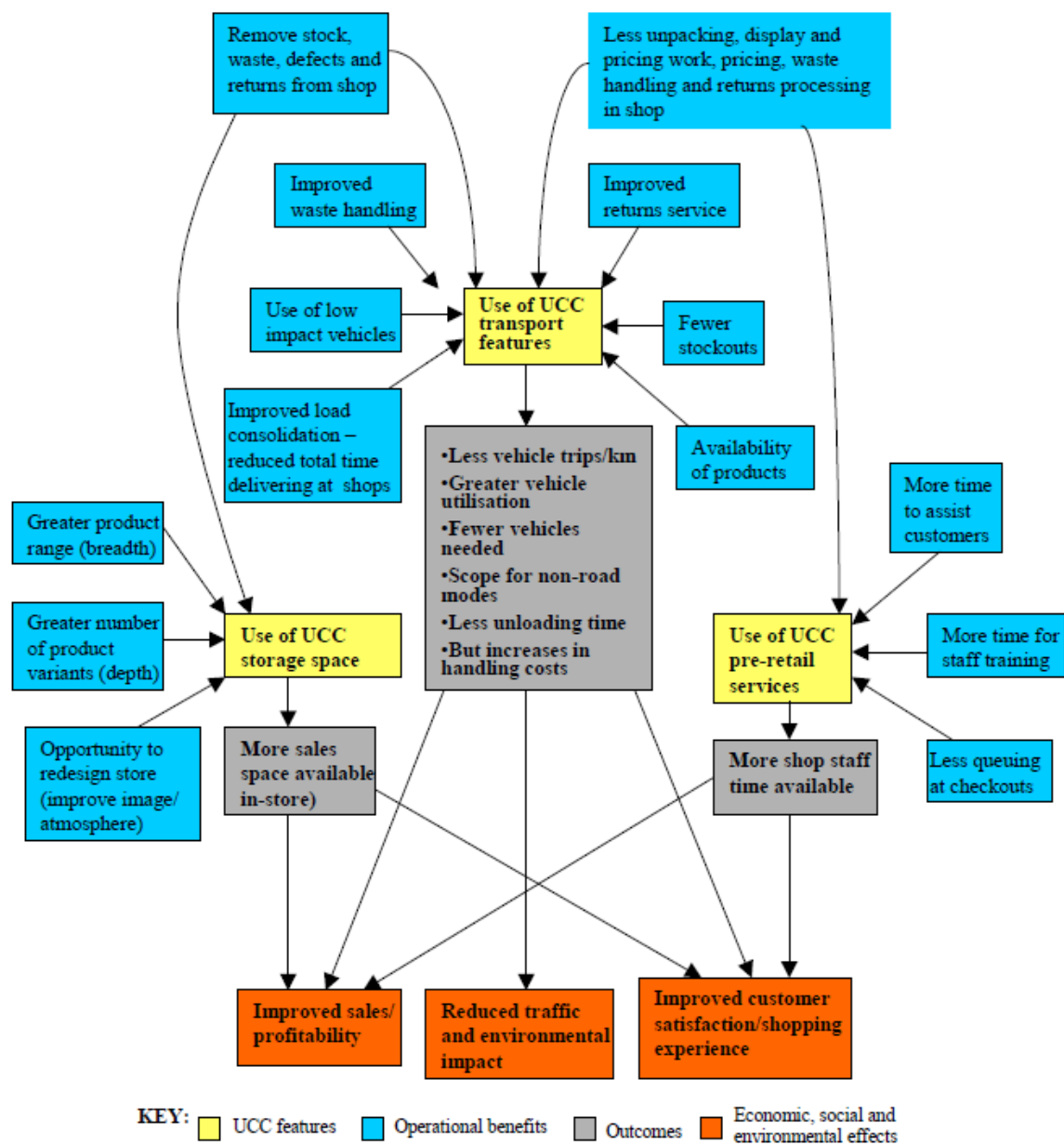


Figure 12: Range of potential logistics and pre-retail activities at UCC and possible benefits (Figure from Browne, Sweet et al. 2005).

3.2.2. Concretization of three additional business extension ideas for the UDC

There are several potential initiative takers for the UDC: Freight Carriers, Shops and Municipalities. From each side business concepts can be brought to the UDC to add value.¹ Further options exists, here some basic business extension possibilities are presented:

- | | |
|--------------------------------|---|
| 1 "Waste Collection" | <ul style="list-style-type: none"> -> Less waste collection trucks from the city
(Note: only waste like paper or plastic) -> Collection from shops and other points -> Reallocation of trucks from FC -> Most heavily impacted player: city -> Higher load factor and reduction of trucks |
| 2 "Post Collaboration" | <ul style="list-style-type: none"> -> Commitment of a big player -> Upon going back, taking packages from post offices and shops -> Government agreement with post company like DHL |
| 3 "Building activities" | <ul style="list-style-type: none"> -> Storage of building materials in UDC -> Delivery by UDC -> Smaller building sites -> Optimize truck load easier |

3.3. Critical Success factors for an UDC

Critical success factors are dependent on the location of the city and the type of urban distribution centre. The main critical success factors have been identified in Table 6 based on review articles of Marcucci and Danielis (2007) and Browne, Sweet et al. (2005).

Table 6: Critical success factors for the UDC based on Marcucci and Danielis (2007) and Browne, Sweet et al. (2005).

Critical success factors for the UDC
Location in/near the city
Subsidy collection
Collaboration with shipper and freight carriers
Financial viable
Service cost of the UDC
Access permit cost
Delay in delivery time
Distance of the parking bay from the shop

¹ Information from brainstorm and (Browne and Sweet et al, 2005).

Here a notion should also be taken that a single UDC is unlikely to be attractive for many suppliers' flows due to the degree of diversion required from normal route and may therefore negate transport savings for onward distribution (Taniguchi E and Noritake M, 1999).

Also it is highly important that there is a clear need for an UDC like with an (old historic) centre with much congestion without UDC. For succesfull implementation the UDC has to add value for total case and preferable for specific stakeholders. In order to get all stakeholders on board a clear UDC objective (like less congestion or environmental effects) is needed to succeed.

3.4. Basic business model behind UDC

In this research the business case for the UDC will be estimated based on parameters from the vehicle flow model. A complete technical description of the vehicle routing and multi-agent model will be presented in Chapter 5 and Appendix E: VRPTW with Genetic Algorithm.

3.4.1. Business finance for the UDC

Assumptions here are based on the Cost Benefit Analysis by Dennis Kuiper (van Duin and Quak, 2008). Here the green areas require input from the model later on.

In this model some adaptations were made to the model by Dennis Kuiper and Ron van Duin.

Table 7: Budget for the UDC for one year (original data from Kuiper via van Duin)

BUDGET Year 1			
	No. of units	Price per unit	Total
INCOME			
customer deliveries	14425	0,24	346.188
sub-total			346.188
COSTS			
Road toll	1	0	0
Fuel costs	1	0	0
Personnel	60	1.600	96.000
Insurance	1	53.400	53.400
sub-total			149.400
Vehicles maintenance	1	0	48.000
Building(s) (maintenance)	1	400.000	400.000
Other Additional Costs	0	2.000.000	0
sub-total			597.400
Total Revenues			346.188
Total Costs			597.400
Total (revenues MINUS costs)			-251.212

Table 8: Assets acquired by the UDC

Assets	Life time (years)	1
Vehicles	10	1,400,000
Infrastructure	35	600,000
Building(s)	35	9,000,000
Remaining investment costs	20	2,000,000
Land	99	5,800,000

Table 9: Cash Flow statement for the UDC

CASH FLOW				
Year	1	2	3	4
Beginning Cash Position	0	-629.426	-342.955	-56.485
Revenues	0	346.188	346.188	346.188
Cash IN	0	346.188	346.188	346.188
Costs	0	597.400	597.400	597.400
Fixed asset purchases	18.800.000	0	0	0
Value added taxes (VAT)	-2.820.000	-37.682	-37.682	-37.682
Corporate Profit Taxes	449.426	0	0	0
Cash OUT	16.429.426	559.718	559.718	559.718
Operational cash flow:	-16.429.426	-21.3530	-213.530	-213.530
Subsidy	15.800.000	500.000	500.000	500.000
Finance IN:	15.800.000	500.000	500.000	500.000
Cash flow per year	-629.426	286.470	286.470	286.470
Ending cash position	-629.426	-342.955	-56.485	229.985

3.5. Conclusion

The urban distribution centre is a promising concept to reduce congestion and emission inside the city centre, but main concerns are the financial viability of the centre. Additional services can boost the value of the urban distribution centre, but this has to be done in collaboration with involved actors to determine the need.

For the model generation it is important to address as much critical success factors as possible. The location in/near the city will not be tested with the model, but a reasonable assumption will be made for a good location. The critical success factor subsidy will be investigated with the model. In the model the collaboration between shippers and freight carriers is assumed to be perfect as they act as one agent. The service cost of the UDC is under investigation with the model. Access costs to the city centre will be a policy measure to investigate with the model. Delay in delivery time can be measured with the model in a later stage. Only the distance from the parking bay to the shop will not be included in this model.

Sub question 2

The sub question "What are critical success factors for an urban distribution centre and how does an urban distribution centre contribute to the city logistics?" provides inside into the working and effects of the UDC. Critical success factors are identified as: Location in/near the city, Subsidy, Collaboration between shippers and freight carriers, Service cost of the UDC, Delay in delivery time and Financial viability.

The UDC tributes to City Logistics by reducing the amount of vehicle kilometers inside the city. By smart rerouting of separate delivery trucks, goods can be combined more efficiently. This increases the load factor, while decreasing noise, pollution and congestion, which exerts negative effects on the city.

The information from this question will be used to answer question 3; the generation of a framework and model. In this model the impact of policy measures will also be investigated, therefore sub-question 4 will also be answered with results from the literature study.

Sub question 4

The answer to the sub question "What policy measures contribute to the successful functioning of the urban distribution centre?" is additional logistics measures like toll requirements, time windows inside the city and subsidy providing. The impact of subsidy, toll settings and congestion rate settings will be tested later in the multi-agent model.

Future policies like NO_x emission tax can definitely exert positive influence as well, but were not considered in this analysis as they are not yet commonly used.

4. Modelling an UDC

In order to find out why there is such a large discrepancy between expected and actual results of UDC implementation, this chapter will first describe the current modelling efforts on UDCs. This will be followed by the model, which served as the basis for this master thesis.

4.1. Existing modelling efforts on UDC

In most UDC modelling efforts a global analysis is made. In this section first information is presented on the cost-benefit analyses performed. Next the detailed traffic modelling of the Binnenstadservice.nl is presented.

4.1.1. Cost benefit analysis

An example on global cost benefit analysis on urban distribution centres is presented by van Duin, Quak et al 2008). Here the total benefit exceeds to the total costs, but for individual stakeholders this does not necessarily means positive results. In this cost benefit analysis the environmental impact is expressed in monetary terms, but in reality (currently) stakeholders do not receive direct financial benefits/punishments due to environmental impact without policy measures. In the future it could be possible to charge emissions directly. In (van Rooijen and Quak, 2009) it also becomes evident that changes in emission from UDC implementation do not provide significant results compared to the total traffic. This gives an indication that the relation of a UDC towards environmental aspects is not the main solution for environmental issues, but more for congestion related problems.

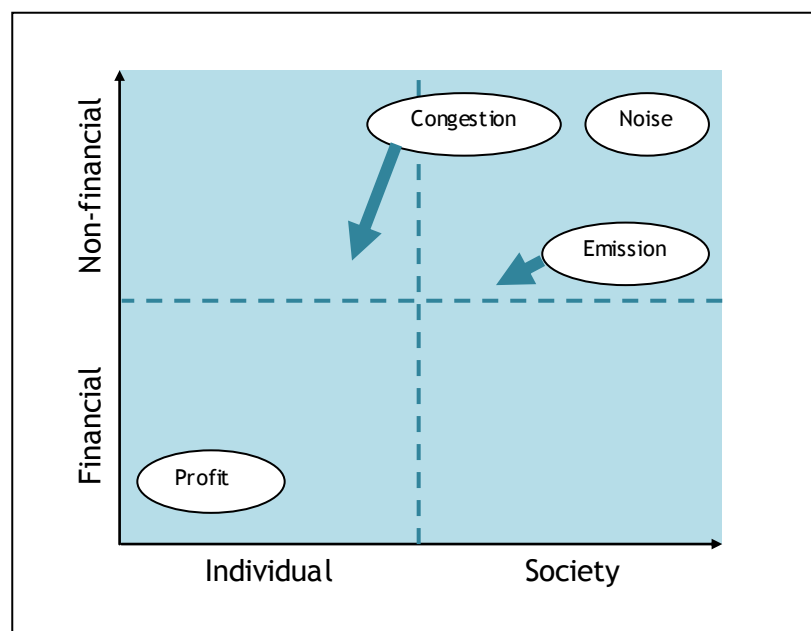


Figure 13: Important factors in City logistics issues with indication for individual or society relevance and (non-)financial relevance.

Arrows indicate a shift in perceived importance.

For various cities with implemented UDCs a cost benefit analysis were also made to estimate the impact from the UDC. Examples can be found on Bestuufs (website) and in Browne, Sweet et al (2005), Browne, Piotrowska et al (2008), Meimbresse and Sonntag (2000), van Rooijen and Quak (2009), Routhier (2006) and van Duin, Quak et al (2008).

4.1.2. Binnenstadservice.nl

The impact of an UDC in the Netherlands (Binnenstadservice.nl) on the air quality, hindrance for residents and noise hindrance is presented visually in van Rooijen and Quak (2009). However vehicle routing and individual stakeholder business cases for all involved stakeholders are less common. The last article uses two models to generate the information: first a vehicle routing model to obtain fleet size, travelling time and distance and a second model to calculate the local impact and to visualize this. Even though clearly a thorough model is generated here, only the results are presented. As example the difference in air quality is presented below in Figure 14.



Figure 14: Limited difference in air quality between two scenarios with and without Binnenstadservice.nl (van Rooijen and Quak, 2009)

4.2. Other models

Tamagawa and Taniguchi (2009) used a multi-agent approach and a learning model is incorporated as well as addressing vehicle routing and scheduling problems with time-window-forecast (VRP-TW-F). By using a multi-agent approach the issue of individual stakeholder behavior can be addressed. Here first the multi-agent model from this article will be discussed, followed by a general model set-up.

4.2.1. Multi-agent model interaction

The stakeholder interaction is given in Figure 15. In a multi-agent model, stakeholders have their own objectives and act accordingly. Freight carriers want to maximize their profit. They offer transport charges to shippers and deliver goods of shippers. The shippers want to minimize transportation cost (paid to the freight carriers) and therefore they select freight carriers and order them to deliver the goods. Transportation cost consisted of a transport charge paid to freight carriers and penalties for delivering outside the desired time window. The resident objective is to keep NO_x emissions low and they can complain to administrators or accept it when NO_x emissions in their local area exceeded the environmental limit. The administrators want to minimize complaints from residents about NO_x emissions and they have the ability to implement city logistics measures. The motorway operators want to maximize their toll revenue and they can change the motorway toll.

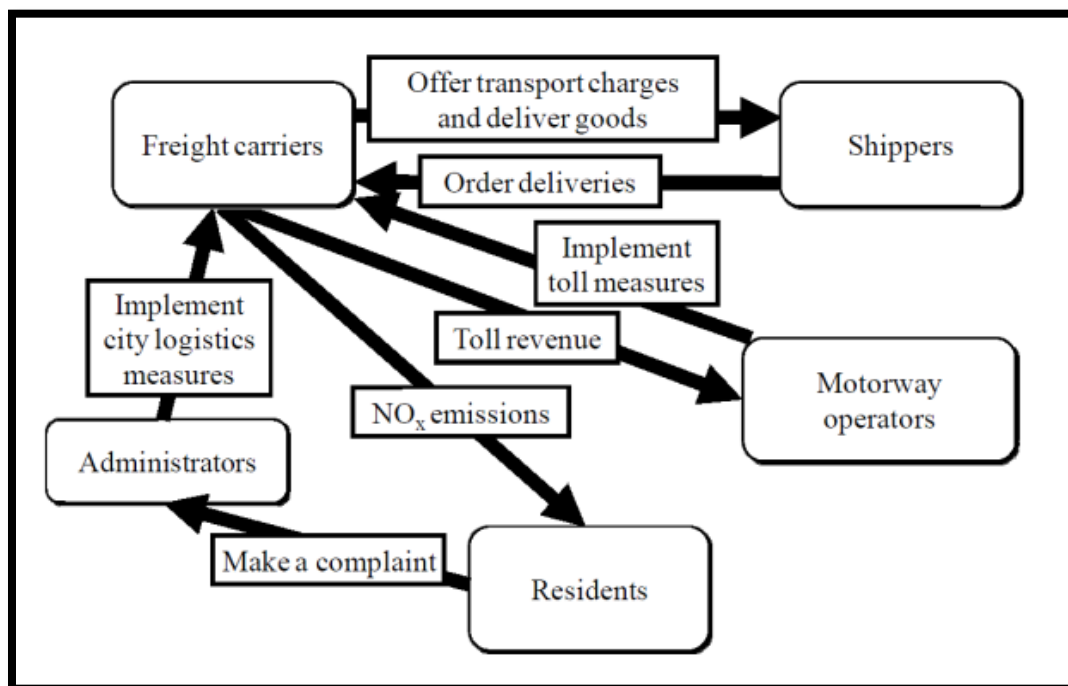


Figure 15: Interactions among stakeholders (Tamagawa and Taniguchi, 2009).

4.2.2. General Model setup

This multi-agent model is combined with vehicle routing, a learning model and scheduling problem with time-window-forecast (VRP-TW-F) as presented in Figure 16. Here the Learning model contains the multi-agent aspects of the model.

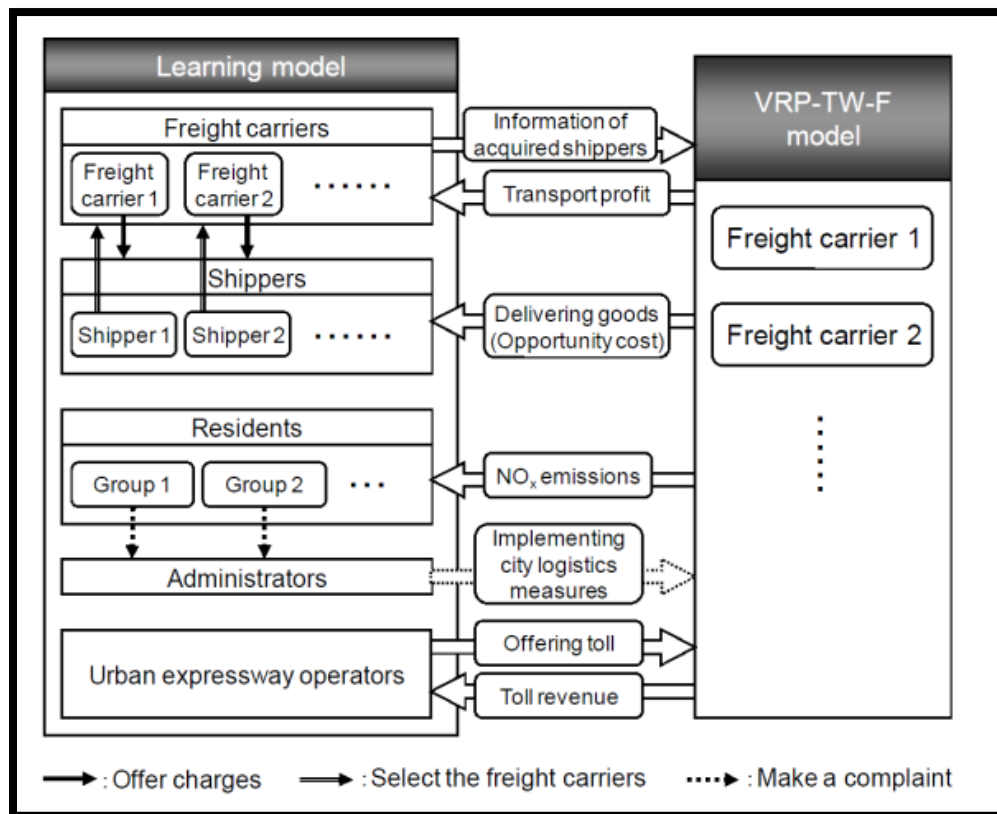


Figure 16: Framework of the multi-agent model (Tamagawa and Taniguchi, 2009).

The two models are run alternately. In the learning model stakeholder behavior is evaluated and assigned a value. On the basis of this value a behavior is chosen. The VRP-TW-F model plans and implements truck delivery schedules for each freight carrier. By iterating these two models stakeholders learn their preferred actions considering interaction with other stakeholders. The calculation steps are as follows:

1. The learning model determines and implements the behavior of stakeholders. Freight carriers offer charges to shippers and the shippers select the freight carriers.
2. The VRP-TW-F model, plans and implements delivery schedules of trucks for each freight carrier according to the orders from shippers.
3. The amount of NO_x emissions and toll revenue are calculated after delivery by the freight carriers. The network environment is updated.
4. Information about the updated environment is returned to the learning model.

5. The learning model evaluates stakeholder behavior with respect to the updated environment and the experiences of the stakeholders are updated.
6. The model repeats above steps.

In the Multi-agent model the agents will change their behavior to adapt to the changing environment, but they have to learn which behavior is favorable. Assuming that the agents do not know what is the optimal behavior they have to find it by trial and error.

Q-learning is a reinforcement learning technique based on the value of action-value functions for agents assuming a fixed policy afterwards. Upon deciding the next behavior Q-learning maximizes the action-value function for the future. A major strength of the Q-learning is that no model of the environment is required by the agent as the agent will compare outcomes of behavior himself. A full description of Q-learning is presented in the Appendix Q-Learning.

The complete approach for the vehicle routing from this article based on the Genetic Algorithm will also be used in this master thesis. The full description of the version for this research can be found in Appendix E: VRPTW with the Genetic Algorithm.

4.3. Model for this research

Various calculations on Cost-benefit analysis on the UDC have been presented in literature, but concrete impact and individual choices for stakeholders are not displayed/calculated. Normally a certain percentage of stakeholders is expected to get involved. In order to investigate why the UDC fails often, it is interesting to take the perspective from the individual stakeholders related to this issue. Impact of their objectives and individual choices on the total result of the UDC implementation can provide insight on the high failure rate of UDCs. An approach based on individual agent behavior is a Multi-agent model as was present by Tamagawa and Taniguchi (2009). The complete analysis of the use of a multi-agent model in logistics is presented by Davidson and Henesey (2005).

Therefore the approach with a multi-agent model for freight carriers and an UDC is required. To incorporate city policies a municipality agent has to be included and shops are required for the demand generation. This basic form of the multi-agent model is displayed in Figure 17. The freight carriers and the UDC will use vehicle routing as internal resource to plan deliveries on a cost basis. Dynamic interaction will be obtained by varying the actual deliveries with individual trucks and changing network conditions, which adds valuable information compared to a cost-benefit analysis to the current research on UDC. By this type of modelling various price settings schemes as well as impact of toll or general congestion can be tested.

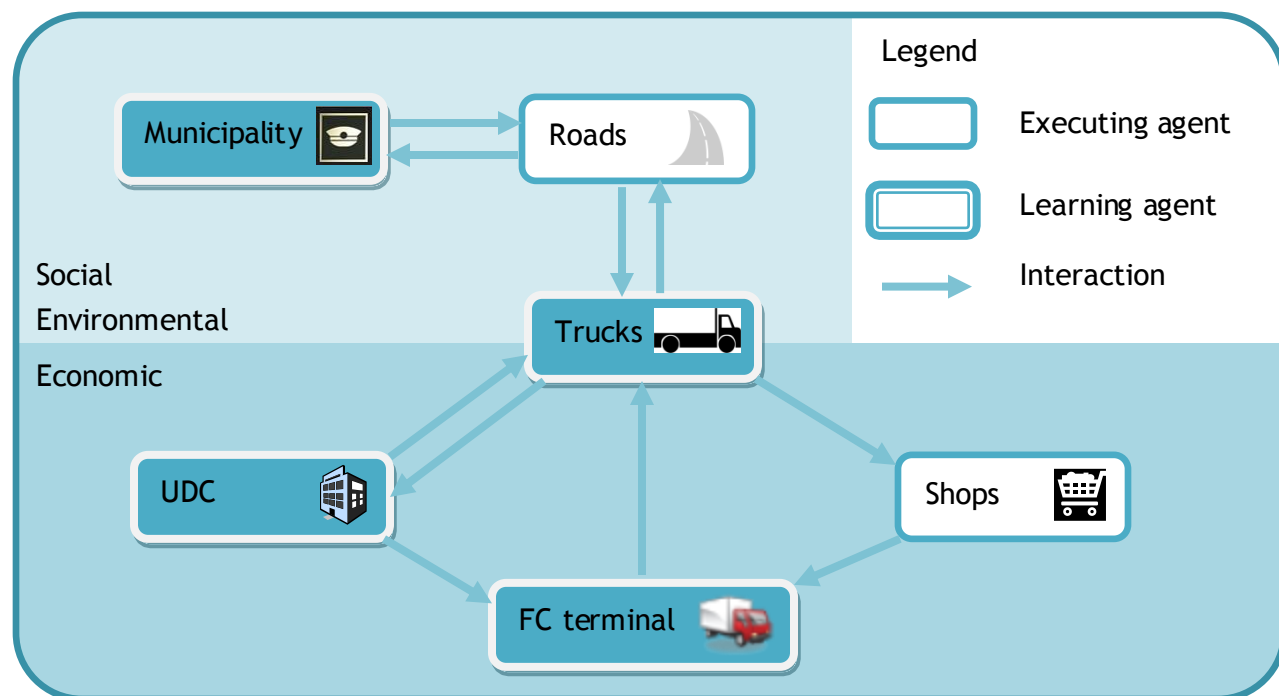


Figure 17: Schematic representation of Multi-agent Model

4.4. Conclusion

Current model efforts on UDCs are often based on cost-benefit analysis or flow models. Estimations of the need for an urban distribution centre often show positive outcomes in these analysis, which is in sharp contrast with the large amount of negative turnover real-world cases. A reason for this difference could be the fact that the UDC might have a positive overall outcome, but it might present negative results for individual stakeholders. Individual stakeholders might not collaborate with the UDC for financial or strategic reasons and thereby block the total positive outcome.

In order to capture the individual choices made by actors to collaborate with the UDC, the choice is to generate a multi-agent model. An example of this in a city logistics content is presented by Tamagawa and Taniguchi (2009). This will generate new input information for a simple cost-benefit analysis on the UDC.

In the next chapter the full multi-agent model is discussed and in Chapter 6 the required vehicle routing for this is presented.

5. Multi-Agent Model for a dynamic Urban distribution Centre

Since multi-agent modelling is not highly common in city logistics (Davidson, Henesy et al, 2005), a general approach towards generating a multi-agent model and the steps taken are described in detail in Appendix B: Multi-agent Model. The main steps of the approach are: Problem formulation and Actor identification, System identification and decomposition, Concept formalization, Behavior Identification, Model Narrative and formalization, Model verification, Experiment set-up, Data Analysis, Model validation and Model use. These will be explored after the basic concept of a multi-agent model is explained. The model will be explained by providing the behavior per agent.

5.1. Concept of multi-agent model

In the general conceptual form of the multi-agent model each agent is considered to have interaction with its environment and based on that it will act (see below).

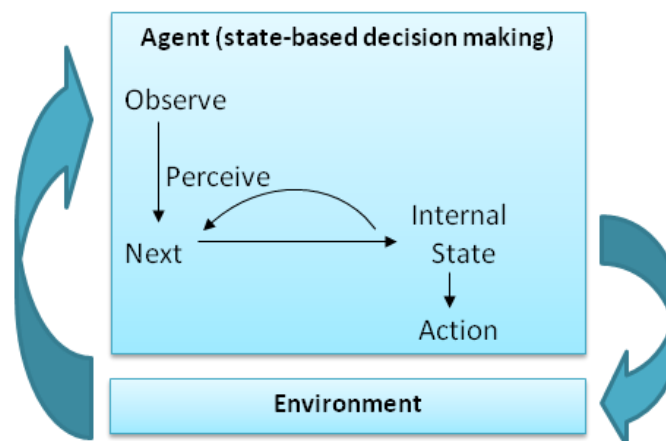


Figure 18: Schematic representation of an agent in a multi-agent model (Wooldridge, 2009)

The agent can observe its environment. This environment holds parameters with information for the agent. This observation of the environment can change the internal state, in which the agent is. Possibly the agent can reflect on the next environment due to this internal state. With this internal state a behavior is associated. With its behavior the agent also impacts the environment. In a multi-agent model several agents are put together to see and understand how they together influence the situation.

In our model there are 5 types of agents, namely the freight carriers, the UDC, the trucks, the retailers and the roads. The municipality is considered as an agent from the theoretical point of view, however in the practical model they are from the environment. There are also two types of dummy agents, namely the streets and nodes. In the model the trucks, roads and retailers are purely reactive agents. The UDC and freight carriers are objection function

agents. The dummy agents streets and nodes are purely reactive agents, but they exist only to provide other agents with information about the road network and therefore only indirectly influence the environment through these agents.

5.2. Problem formulation and Actor identification

The problem identification is described in the Chapter Research Proposal. Below the problem owner Identification and Actor Identification are performed.

5.2.1. Problem Owner Identification

In order to decrease congestion, various solutions have been proposed among which the UDC. Congestion has impact on all involved actors in the city like residents, freight carriers and administrators by direct means, such as longer travel time or indirect means like environmental issues. The question addressed by Multi Agent System (MAS) is: under which conditions will involved actors start using the UDC and under which conditions can the UDC be sustainable (from financial and environmental point of view)?

Even though all actors experience negative efforts from congestion, the administrator is appointed as the problem owner for his central role in policies. Note however that examples exist where other actors initialized UDC efforts (like Binnenstadservice in Nijmegen where the shop owners are the initiators).

5.2.2. Actor Identification

Actors involved in this problem are the city government, shippers, residents, freight carriers, local shops and the UDC. The city government has to deal with the negative effects of congestion, pollution and decreased business opportunities. Secondly the freight carriers are involved in the problem as the scenario with the UDC can endanger their business as the UDC can be seen as additional competition. However they also experience direct negative efforts from congestion. Local shops are involved in this problem as congestion decreases access to their shops and increases delivery cost and time. The UDC is involved by distributing the goods of the FCs to the city. Residents are affected by congestion by longer travel time and negative social & environmental efforts, but this will be monitored by the City municipality as this is where residents make their complaints.



Figure 19: Supply chain in City Logistics

The shippers, who also form part of this supply chain, are not included, because only freight carrier trucks enter the city and these are expected to make this choice. In reality shippers are also needed in the choice for an UDC as this increases delivery time in this JIT business. Also the shippers hold substantial decision making power in this choice as the freight carrier business is highly competitive. In a first schematization of this problem the focus is on the individual (freight carrier) trucks entering the city and their choice to use the UDC. This is described below.

Two types of supply chain for delivery to retailers

Interaction with an UDC and other agents can take place in several ways. The potential demand and payment of the UDC can come from shippers, carriers or retailers as the initiative can come from these various actors. Also there are several supply chain types to deliver goods, where here one distinction is made upon who delivers. Large companies like Seven Eleven or Albert Hein often have their own organized fleet to deliver goods to several locations, so the shipper delivers the goods themselves. Normally consolidation takes place before these trucks go to a city. This is indicated by path 1 in Figure 13. The second option is that a company hires a freight carrier to deliver goods, which is shown by path 2. This is the mentioned highly competitive freight carrier market.

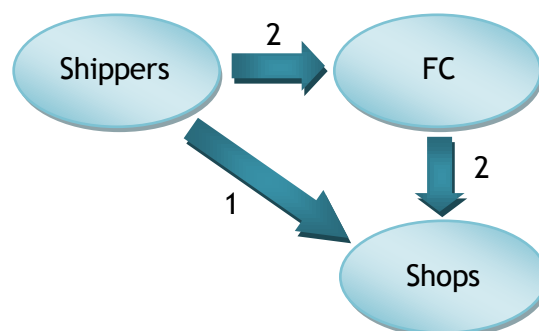


Figure 20: Path for good delivery to a store.

Initiator for UDC demand

This leads to several potential ways to get demand for the urban distribution centre. In the first case the shippers or large companies see a financial benefit to ship products to the UDC or by letting others use their consolidation services. This would lead to path 1. In the second case the freight carrier will deliver goods to the UDC (path 2). In the last case the shops have a desire to have either the shipper or FC use the UDC. Reasons for this are related to improving their bargaining position on the price or use the UDC as storage location. This has a similar path as 1 or 2, but the initiative comes from the shops.

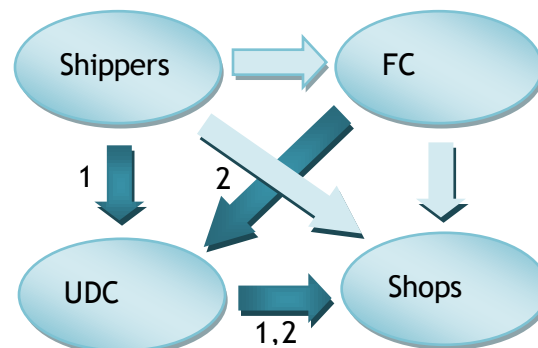


Figure 21: Path for UDC demand

The shops can benefit from the UDC to improve inventory parameters and a FC can benefit by spending less time in traffic jams. This also depends on the type of truck and how easily this truck can access the city centre. The city municipality can initiate the UDC in an effort to decrease congestion effects. However financing is always a critical point for the initiator. In this scenario it is assumed that the city municipality is the initiator to value cost advantages for the freight carriers to use the UDC and subsidy is required normally to finance the UDC anyway.

5.3. System Identification and decomposition

Next a list with basic assumptions is presented, which characterize the system. In the next part the individual agents are described in system decomposition.

In this table the agents with their identified properties and abilities are presented. Below the identified relationships between these agents are presented.

Table 10: Agents and their properties

Agents and their properties	
Freight Carriers	
Have:	- Terminal from which trucks leave and return - Possibility to produce trucks with a given route - Vehicle routing skills and information parameters to determine routes - KM counter and a toll counter - Demand information - Best route and best route cost information
Can:	- Collect good demands from retailers in order to do the vehicle routing - Perform vehicle routing with the generation of a best route and its costs in order to minimize their expenses - Produce trucks in order to deliver the goods to the retailers - Add km counts and toll payments from individual trucks to calculate their total price
UDC	
Have:	- Terminal from which trucks leave and return - Trucks are given a route - Vehicle routing skills and information parameters to determine routes - KM counter and a toll counter - Demand information - Best route and best route cost information - UDC fee per parcel
Can:	- Collect good from freight carriers trucks in order to do the vehicle routing - Perform vehicle routing with the generation of a best route and its costs in order to minimize their expenses - Produce trucks in order to deliver the goods to the retailers - Add km counts and toll payments from individual trucks to calculate their total expenditure - Change the UDC fee per parcel in order to see the impact on the total system
Municipality	
Have:	- NOx counter per time phase for the total city - Average congestion rate
Can:	- Set the average speed to three different levels in order to estimate the impact of this by reducing the speed of the roads - Collect the emitted NOx emission per time phase for calculating the total NOx emission

Trucks	
Have:	Direction for next destination in a x and y coordinate Travelling speed Certain amount of goods that need to be distributed to the shops Kilometer counter and toll counter An owner, either one of the freight carriers or the UDC
Can:	Emit NOx emission in order to estimate environmental impact Drive to a destination in order to deliver goods Deliver goods to retailers and determine the next direction in order to fulfill all their deliveries
Retailers	
Have:	Maximum speed and real speed per time phase (which is set to zero in order to identify problems with trucks driving off road). Number of goods
Can:	Request goods by freight carriers by reducing the number of goods by the ordered number. Receive goods from trucks in order to get their ordered goods
Roads	
Have:	Maximum speed and real speed per time phase Toll count NOx counter Different colors to indicate crossings
Can:	Collect NOx emission in order to estimate environmental impact Change the speed to account for traffic conditions Request toll from trucks
Dummy agent: street	
Have:	Distance parameter Link travel time and cost
Can:	Calculate the travel time and costs per link per time phase with the roads below the link in order to reduce calculation costs for the nodes during the Dijkstra algorithm
Dummy Agent: node	
Have:	Travel time and costs from this node to all other nodes per time phase
Can:	Calculate the travel time and costs from the current node to all other nodes per time phase in order to reduce calculation costs for the freight carriers during their vehicle routing with the Dijkstra algorithm

Relationships between agents are in the following forms:

Goods, which have

- A place of demand (x and y coordinate)
- An amount in the unit of the size of a 100 parcels

Information, which can be

- Travel speed information from roads (or other patches) to trucks
- UDC fee in Yen
- Travel costs and time between two nodes in order to calculate the optimal route of the Freight Carriers and the UDC.

Money, which has

- An amount in units of Yen

NOx emission, which has

- An amount in arbitrary units

Goods are transferred between trucks and FC terminals, UDC terminal or retailers. Information is transferred between roads and the following agents: streets, FC terminals, UDC terminal and trucks. Nodes transfer information to or from FC terminals, UDC terminal and streets. Trucks exchange information with FC terminals, UDC terminal and roads. Lastly roads present information to the FC terminals. Money travels between trucks and FC terminals or UDC terminal and between trucks and roads. Also money flows between FC terminals and the UDC. NOx emission is emitted by trucks and collected by the roads.

In this model description, the vehicle routing of the freight carriers and the direction/driving behavior of the trucks is not further specified yet. A full description can be found in Appendix E: VRPTW with the Genetic Algorithm.

The environment consists of the two outside black banners on the left and right side of the modelling grid to provide a toll for the trucks to identify the end of the model/city. The environment sets the UDC fee and the municipality congestion rate.

This leads to the following model structure as displayed in the next figure.

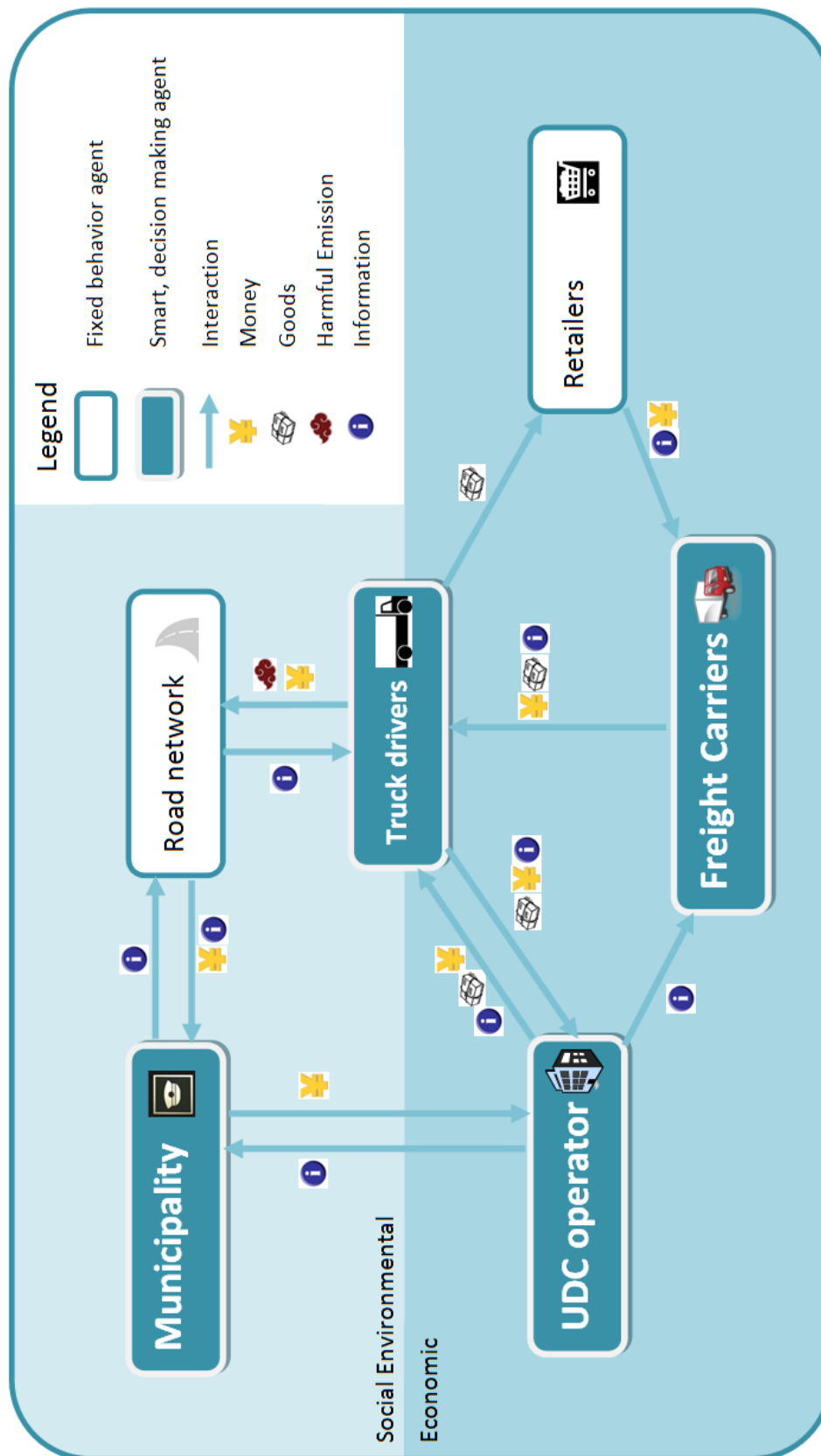


Figure 22: Schematic representation of Multi-agent Model

5.4. Behavior Identification

The behavior identification is represented in the model narrative per agent. Here it is important to realize that one day consists of multiple ticks in Netlogo. The behavior is described per agent.

5.4.1. Freight Carriers Terminal agent

At the start of each day (first tick of the day) the freight carriers collect the demand from 10 (or any other number of) retailers as show in the figure below.

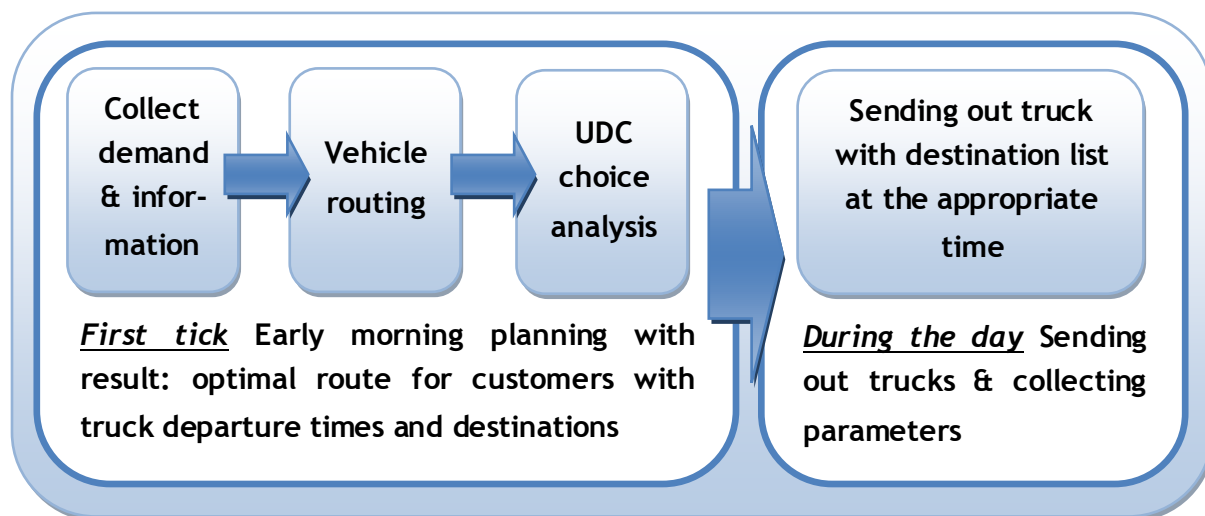


Figure 23: Schematic behavior of the freight carrier terminal

Besides demand information, which contains a number of goods and a location, the freight carrier also collect the UDC fee information and network travel conditions. With this demand information the freight carriers perform the vehicle routing. The vehicle routing is a VRPTW-F solved with the genetic algorithm based on the objection function (cost minimization). After performing the vehicle routing the freight carriers have generated a best route for all their trucks. In this route the departure times of the trucks with the order of the destinations are contained. At the time of departure, the truck is generated and send out by the freight carrier. During the day empty trucks return to the freight carrier terminal reporting their toll costs and km count.

5.4.2. UDC Terminal agent

At the start of the program the UDC fee is set by the user. During the day trucks with goods and destinations arrive at the UDC (demand). Vehicle routing on the current demand is performed in an early delivery scheme, fixed time delivery scheme or when a truck is full.

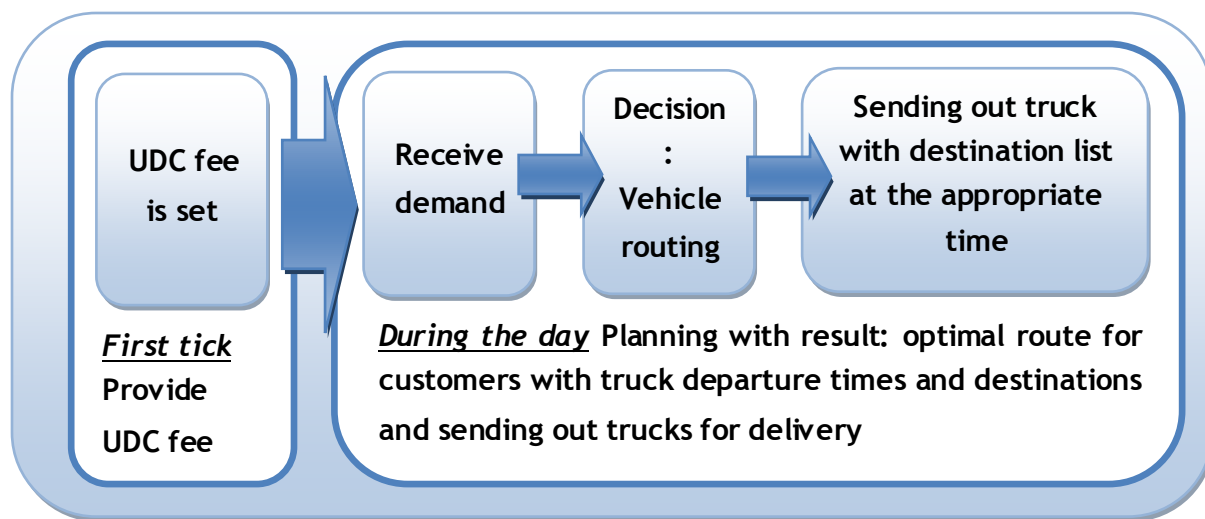


Figure 24: Schematic behavior of the UDC terminal

The vehicle routing is also a VRPTW-F solved with the genetic algorithm based on the objection function (cost minimization) just like with the Freight Carrier terminals, but with different parameters settings. Also during the day empty trucks return to the UDC terminal reporting their toll costs and km count.

5.4.3. Municipality

At the start of the program the municipality sets the average congestion rate and toll rate. At the end of the day the municipality collects the NOx emission from all the roads per time phase.

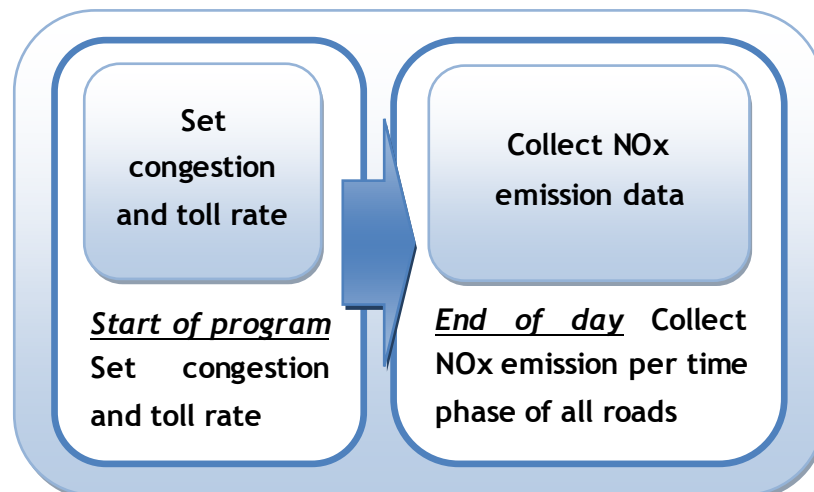


Figure 25: Schematic behavior of the freight carrier terminal

5.4.4. Trucks

Trucks are generated by either one of the freight carriers or by the UDC. The trucks stay on the road and on crossings decisions for directions are made based on destination and traffic congestion. The truck sets its speed every 2.5 minutes with information from the road it is on. The truck keeps on the road (and crossings) by checking where it is heading. When it is heading for something, which is not a road (or crossing) it will change direction. The truck can drive either to the UDC, where it will drop all its goods, or it can drive to several retailers where the truck delivers the appropriate goods. After delivery (of 1 tick) it continues to its next destination. The final destination is always the original terminal. During driving the truck emits NO_x at every time step. When the truck enters the city, it can be requested to pay toll to the road. Trucks keep track of their traveled distance.

The truck receives a route from his freight carrier. It has to follow the customer order, but it is free to choose the route in between (fixed order variable route).

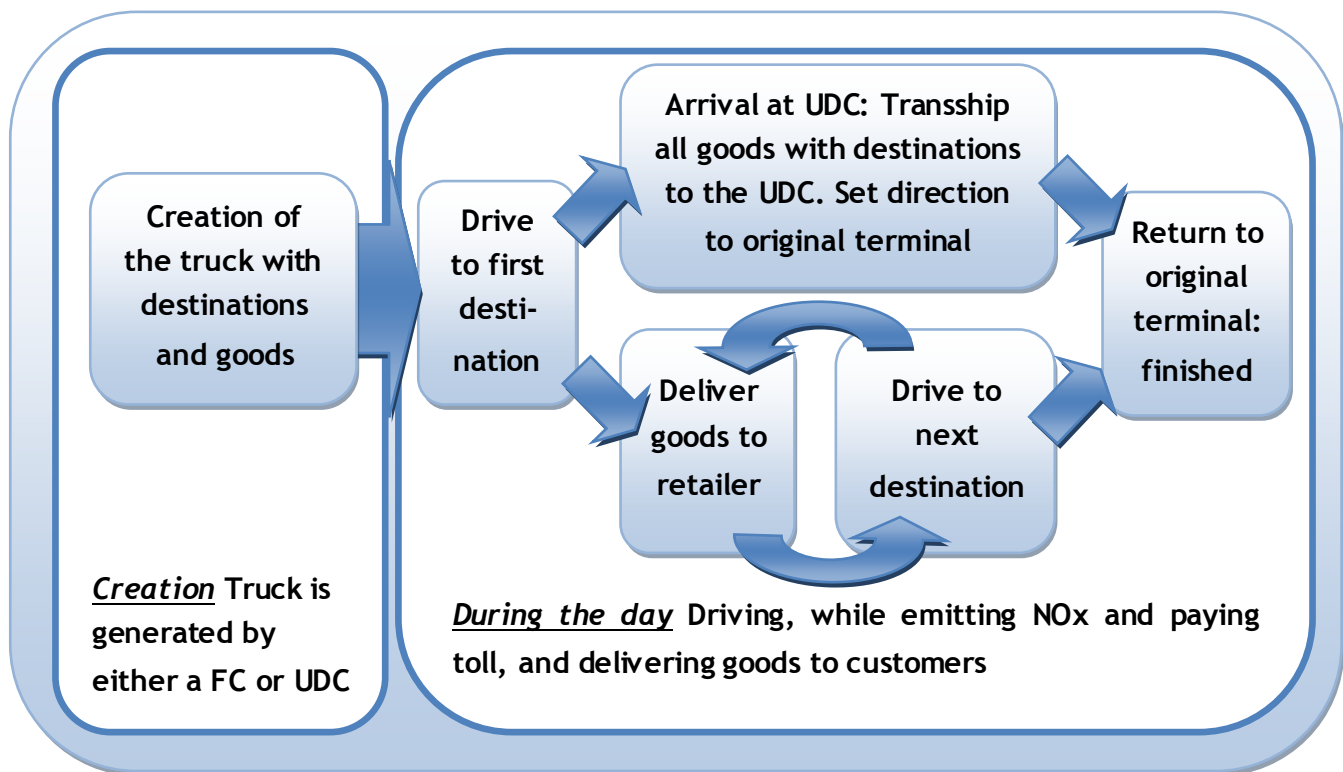


Figure 26: Schematic behavior of a truck

The driving and direction choosing will be elaborated upon in Appendix B: Multi-agent Model.

5.4.5. Retailers

At the start of the day, the freight carriers randomly select 10 retailers to place an order. During the day the retailers received their orders from trucks.

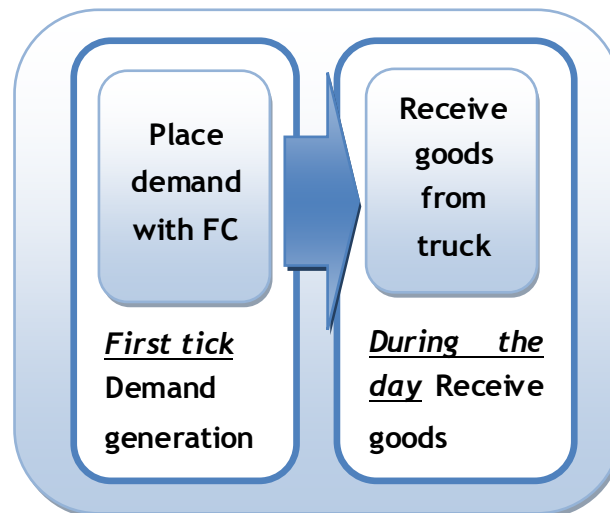


Figure 27: Schematic behavior of the retailers

5.4.6. Roads

At the start of the day the roads reset their NOx. At the end of the day, the average speed per time phase from today is taken as the current average speed per time phase for the next day. During the day the roads collect NOx from the trucks and reduce their speed when trucks are present on their road segment.

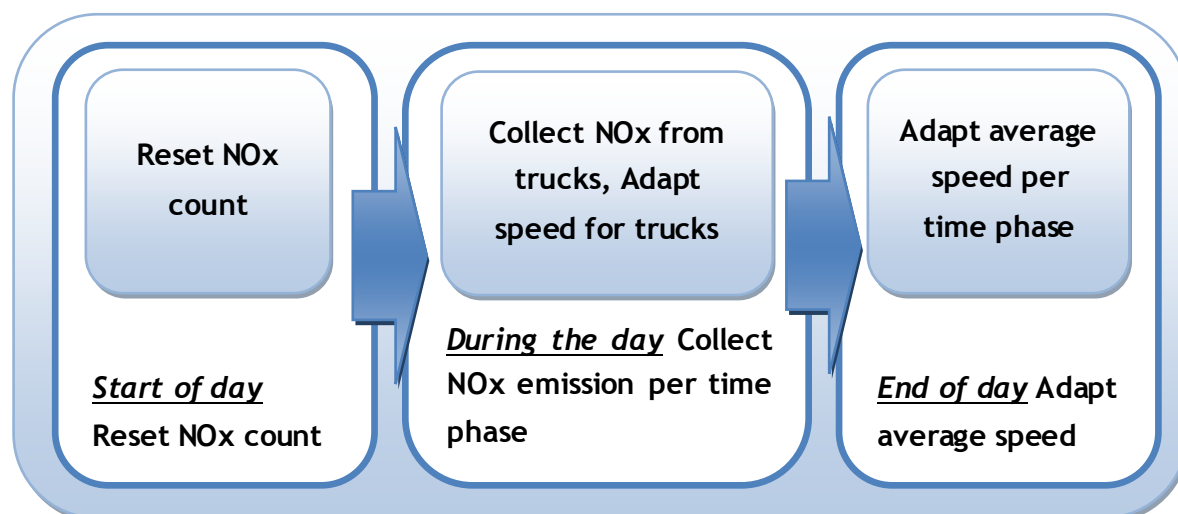


Figure 28: Schematic behavior of the roads

5.4.7. Dummy agents: Streets and nodes

At the start of the day the streets and nodes are used to order the road network information in a convenient way for the genetic algorithm calculation. The nodes are present at the crossing of the roads and the streets link these nodes. The streets request travel time and cost information from the road segments below to add this information. Next the nodes use the Dijkstra algorithm to calculate the cheapest paths to all other nodes in the system.

5.4.8. Total time line of one day of the multi-agent model

The total time line for the model is presented on the next page. In this figure the large arrows indicate continuous process over the time block, while small arrows indicate a single event.

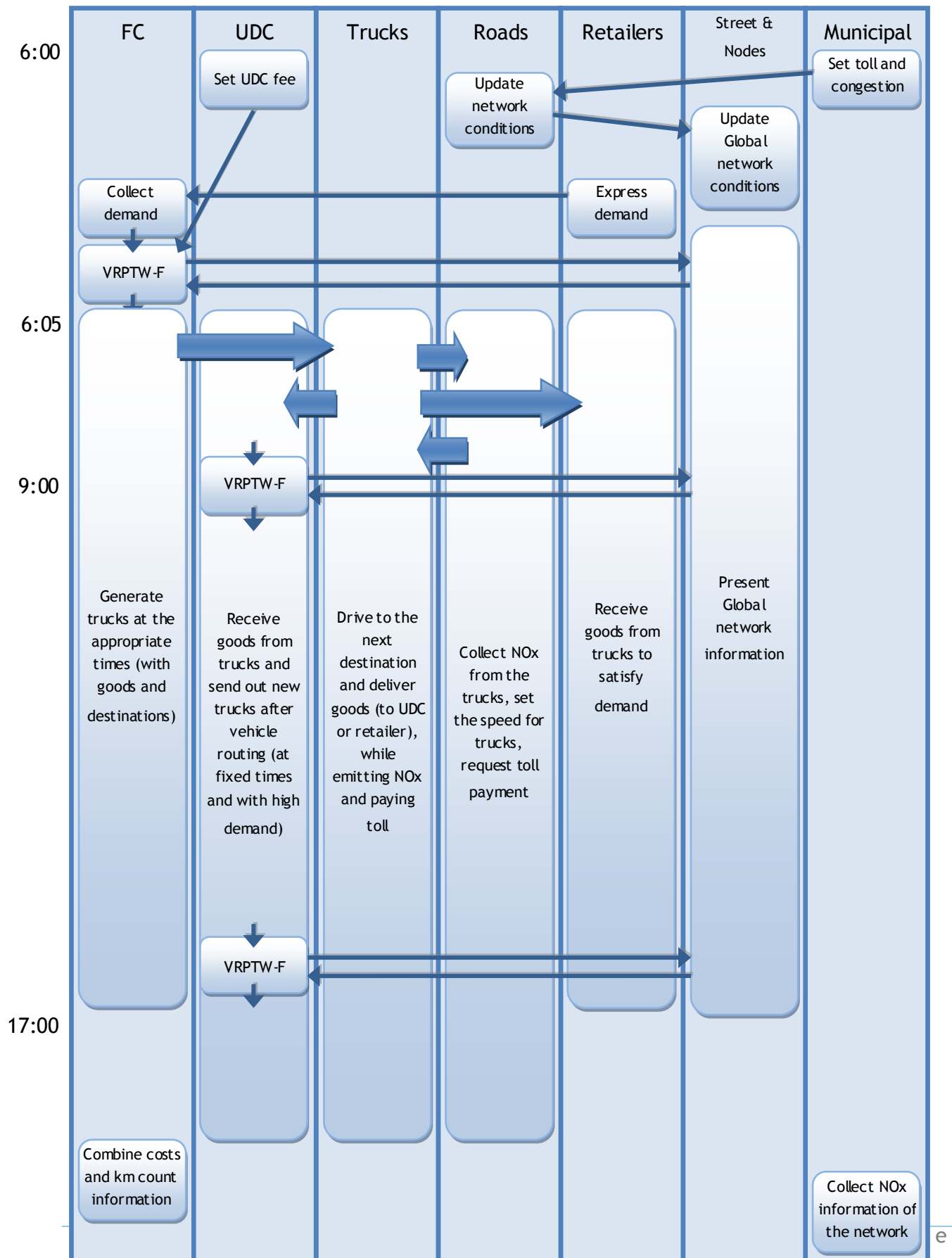


Figure 29: Timeline for one day in the multi-agent model

5.5. Formalization and Implementation

In order to further formalize the model and implement all the characteristics a pseudo code is developed. The pseudo code for the total model is given as well as the relevant decision processes.

5.5.1. Modelling assumptions

Below the assumptions for the model are described. The full description and explanation can be found in the Appendix B: Multi-agent Model.

Table 11: Modelling assumptions

Modelling assumptions
Model parameters
Time scales in the model are 2.5 minute per tick and 1 hour per phase. Active hours are between 6:00 and 19:00. The model represents a inner city with diameter 5 km and per patch the size is 1/3 km.
General assumptions
There is one type of truck. There is one type of good. Model represents an artificial city.
Freight Carriers
Freight carriers travel with an average load factor of 42%. Artificial high overload penalties exist. Artificial high delay penalties exist. The freight carrier uses the fast decision maker choice model for the choice for the UDC.
UDC
Location of UDC is near entry of the city for freight carriers. Artificial high overload penalties exist. Artificial high delay penalties exist. The UDC can have an early delivery, fixed time deliveries or full truck delivery scheme.
Trucks
Per 2.5 minute 1 unit of NOx is generated per truck. Fixed truck costs are 15624 Yen per truck per day. Variable truck costs are 1175 Yen per 10 minutes per truck. Individual trucks from the terminals in the model represent small truck companies. Trucks can only drive horizontally or vertically at one time. Trucks can choose the least congested road at a crossing. Trucks use a fixed retailer order variable route scheme.

Road Network
The average speed profile during a day to include congestion is presented in Appendix D.
Retailers
Per retailer unit on average 100 parcels are delivered.
Municipality
There can be three different congestion levels in the city. There are three states for toll.

5.5.2. Pseudo code for the total model

First of all the set-up of the model is described, followed by the general go-procedure for every time step. The bold indicated words represent submodels, which are described in the Appendix B: Multi-agent model.

Table 12: Global set-up of the model

Generate the global variables and breeds with assigned variables in Netlogo

To do the set-up

Set day 0, set time 0, set number of phases, set number of steps in a day

Check for number of steps in desired value, otherwise write error message

Set deliverytype to corresponding values.

Set toll setting to corresponding values

Set type of fee (dynamic or not) to corresponding values

Execute **Generate-Network**

Set truck shape to truck

Ask FC and UDC to set all initial parameters like toll counter, GA parameters (chromosome, newchromosome, totalfitness, ctovld, costs, numchromo, varcosts, dist)

end

Now the routine for the go-procedure is described.

TO GO routine per tick

If (time is 0) ;; many initial steps of the model

 [**Update-the-streets**

Perform-the-cheap-path-calculation

 Update the plots

 Ask fcterminals

 [**Generate-demand**

 Set demandsize

Genetic-algorithm]

 Set day(day + 1)]

Calculate the current phase with (floor (time / (daytime / phasenr)))

If (conditions for UDC delivery is met (on time or truck load))

 [ask UDC [ifelse (demandsize of UDC > mod 4)

 [do **UDC-GA]**

 [take out random truck load to have a higher load factor (if not the last delivery)

 do **UDC-GA**]]

Ask fcterminals and UDC [if (there are trucks to be send) [**generate-trucks**]]

Ask trucks

 [set kmcount (kmcount + speed)

 Move truck forward with speed

 set speed of truck equal to the speed of current road segment in this time phase

]

NOx/speed-calculation

Direction-truck

If (it is the end of the day)

 [ask municipalities [**regulations**]

 Write-outputfiles]

if ticks > 5000 [stop] ;; stop criteria for total model

tick

set time (ticks modulus daytime) ;; set the time of the day

end TO GO routine

5.6. Model verification

In this part model verification is performed with single-agent verification, interaction testing and multi-agent model testing. Most results for single-agent verification and interaction testing are in Appendix B: Multi-agent Model. Variability analysis in the multi-agent testing is displayed here.

5.6.1. Single-agent verification

With single-agent verification there is a sanity check, breaking the agent, extreme value testing and dynamic signal testing. Below a table is presented to indicate the relevance of these testing methods for the individual agents.

Table 13: Single agent verification for the agents

Agent	Sanity Check	Breaking the agent	Extreme Values	Dynamic signal
Roads	X	X	X	X
Streets/links	X	X	X	
Retailers	X	X	X	X
Trucks	X	X	X	X
FC terminals	X	X	X	X
UDC terminal	X	X	X	X
Municipality	X	X	X	

The tests are described in the Appendix B: Multi-agent model.

5.6.2. Interaction testing

For this model the generation was done in several steps, performing interaction tests at each stage. Most agents need interaction with an agent of another breed. The model is set up with the minimum number of agents necessary and it is investigated, if the basic agent interactions happen correctly. The different steps were as follows:

1. Generate the retailers and road network with boundaries
2. Add the streets and nodes
3. Generate a truck and let it drive
4. Add 1 FC terminal and generate trucks
5. Verify the GA and find the parameters
6. Add the UDC and find GA settings
7. Add the time parameter to the model

The results are described in Appendix B: Multi-agent model.

5.6.3. Multi-agent testing

In the multi-agent testing a timeline sanity check is performed to see if the steps are performed in the correct order as indicated in the timeline in Figure 29. Timelines for individual agents were checked with their expected behavior and the time line of the total model was verified. Parameters were changed for the UDC fee and for the congestion and toll rate of the municipality over 1 day and 10 days.

The variability is only tested for one case. In this scenario there are no toll, no subsidy, normal congestion rate and fee ranging from 0 to 350 Yen per parcel. All the fees are taken together. Here it is expected that variance in output is caused by different input settings and caused by the genetic algorithm, variable demand location and individual truck routing choices as shown below.

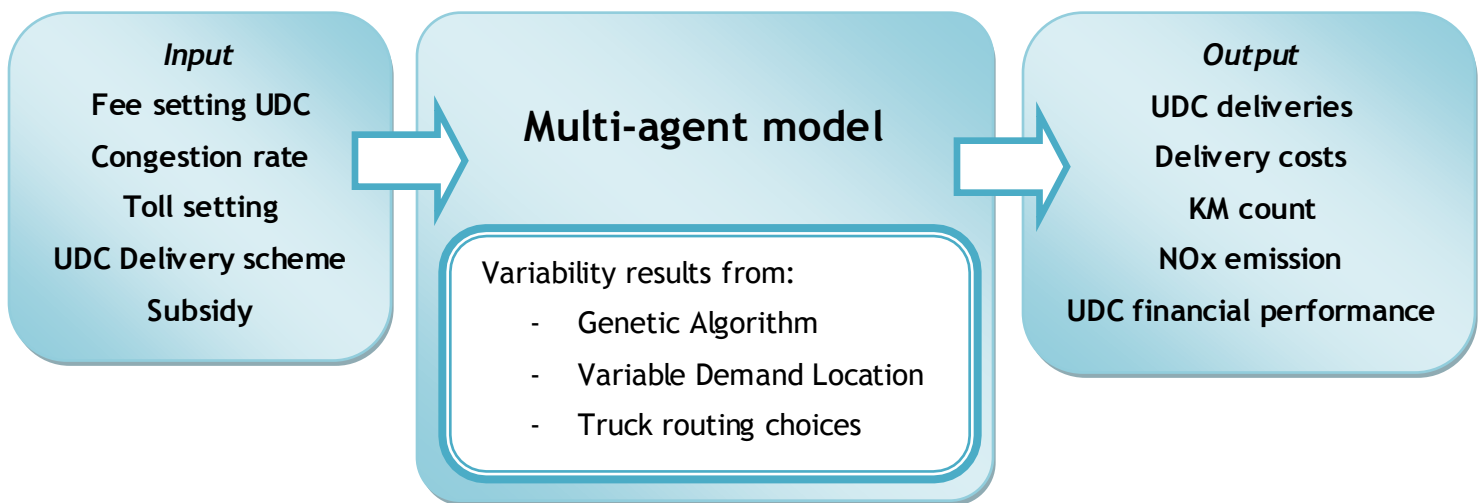


Figure 30: Schematic representation of input, output and variance account for the UDC multi-agent model

The variance was measured during one day at the end of the day for the relevant parameters: NOx reduction, km reduction, Delivery costs average for the FC terminals, Delivery costs for the UDC and the UDC financial performance.

Table 14: Table with Variability data

Parameter	Mean	Standard Deviation
NOx reduction	81.0	13.7
km reduction	81.2	12.9
Delivery costs average (FC)	195.227	94.501
Delivery costs average (UDC)	55.210	72.583
Income UDC	202.493	271.724
UDC financial performance	-565.854	236.065

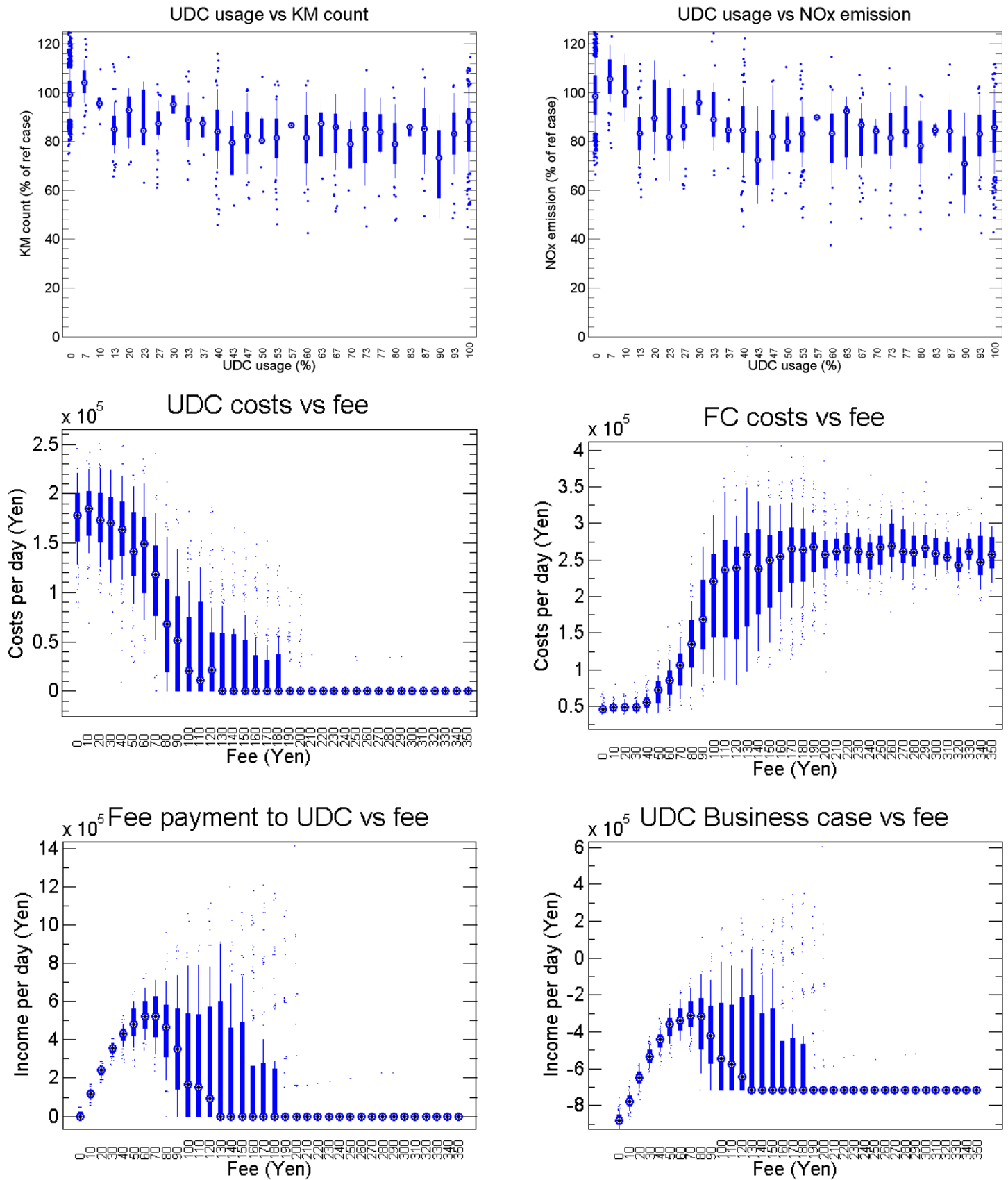


Figure 31: Here the relevant parameters are displayed for 2138 runs. Top left is the km count vs. UDC usage rate and top right is the the NOx emission vs. UDC usage rate. In the middle are the costs associated with the truck (fixed and variable) for the UDC (left) and the FC carriers combined (right). On the bottom left is the income from fee vs. fee price to the UDC and on the right is the business case for UDC vs. fee presented.

5.7. Model representation in Netlogo

The visual model representation is shown below.

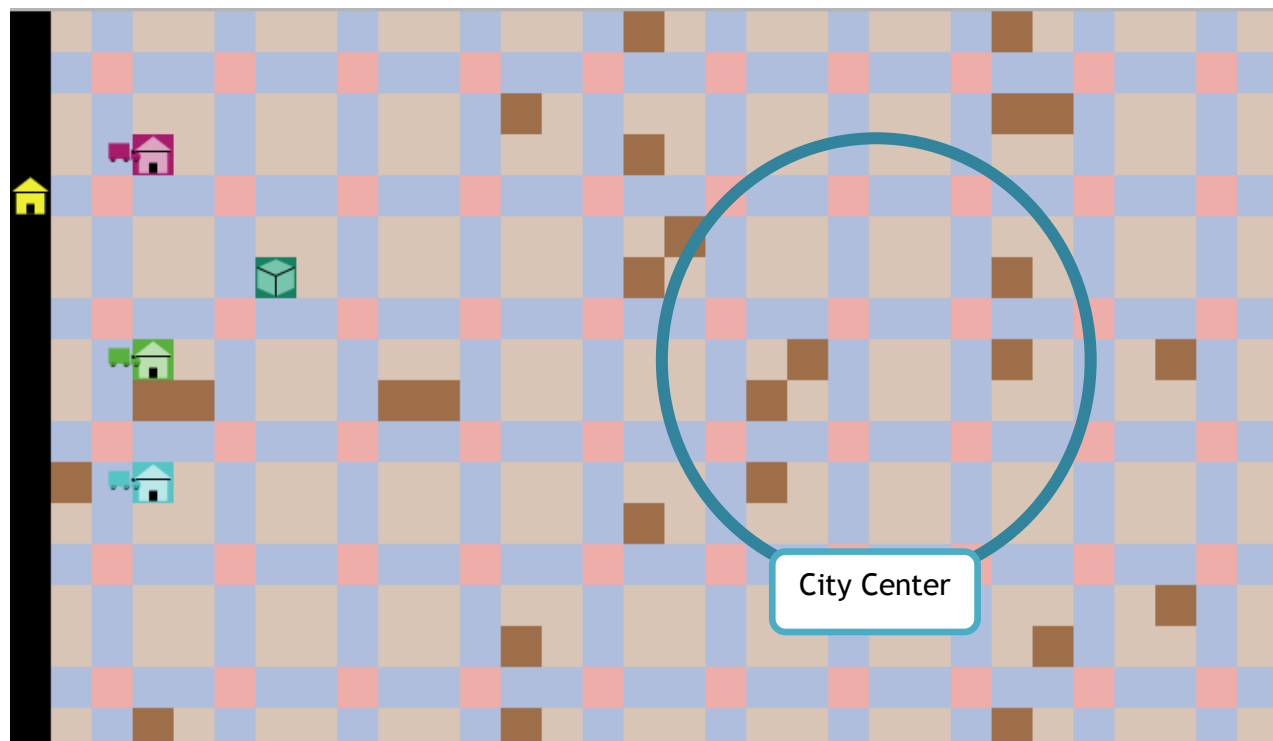


Figure 32: Model representation in Netlogo

The color coding is presented below.

Table 15: Logo/ Color representation of the various agents in Netlogo

Legend			
Logo	Representation	Number	Fixed location
	UDC centre	1	Yes
	Freight Carrier Terminal	3	Yes
	Municipality	1	Yes
	Shop (well supplied or waiting for goods)	240	Yes
	Road (Can be crossing or not)	300	Yes
	Truck	variable	No
	Not part of the model	irrelevant	Yes

5.8. Experimental Set-up

In the experimental set-up it is decided exactly what is the question to be answered with the model. For this research the following question will be answered with parameter sweeps:

- Does UDC usage have the desired impact on the NOx emission and km count?
- How does the UDC fee impact UDC usage, NOx emission and UDC finance for various UDC delivery schemes?
- How does the UDC fee impact UDC usage, NOx emission and UDC finance for various congestion conditions?
- How does the UDC fee impact UDC usage, NOx emission and UDC finance for various toll conditions?
- How does subsidy impact the UDC financial performance?
- Can a dynamic fee improve the UDC financial performance?

These questions are asked to identify scenarios where the UDC can be financially sustainable. The data analysis will be displayed in the next chapter.

5.9. Model validation and use

The further validation and use of this model will be mostly performed by other researchers. In order to allow for a good understanding this thesis can be read. Further validation with major points on the demand generation, GA performance and validation by modelling a real city is required. Current validation of this theoretical city besides logic check is difficult.

With the current model various tests and easy extensions are possible. The different settings can be retested by others. Also varying inputs parameters like fixed and variable costs for trucks can significantly impact the results. Extensions to a more complicated demand or delivery system are desired. Further implementation of time windowed deliveries are of interest. The genetic algorithm is implemented in such a way that if it can relatively easy be used for other programs, but adaption of the objection function (fitness calculation) is necessary and parameters need to be rechecked for chromosome size and specific purposes. For this goal the complete explanation of the GA is presented in Appendix E: VRPTW with the Genetic Algorithm.

5.10. Conclusion

In this chapter the multi-agent model for the UDC is developed and programmed. The model holds 3 freight carriers, an UDC, a municipality, variable trucks, 240 retailers and a road network of 300 individual road segments. The model also holds dummy agents in the form of

60 nodes and 114 streets. These agents interact as displayed in Figure 22. In this model these agents interact to see the impact of various policy measures and circumstances to potentially identify a positive business case for the UDC.

Behavior is identified for individual agents and described on the time line of 1 day in the total model. Validation and additional information of the model is described in the Appendix B: Multi-agent model. The results to the research questions are displayed in the next chapter.

6. Results of Dynamic usage of Urban Distribution Centre

In this chapter the results to answer the research question with the questions posed in experimental set-up from the multi-agent model are presented. First it is checked if the UDC has the desired effect to reduce NOx emission and the km count. Next the impact of various delivery schemes on the business case and the fee vs. UDC usage is displayed. The same is done for the congestion and toll rate. Lastly the impact of subsidy and dynamic fee (variable fee during the day) is shown followed by the conclusion.

6.1. Effect UDC on NOx and km count

First it is checked if the expected positive relation between the UDC and decreased NOx emission and decreased km count is found. The variance originates from the variable demand location, fee setting, GA and truck decision making. There was no toll requirement, normal traffic congestion and no subsidy and the fixed time delivery scheme was used. This total data set consists of 2318 model runs.

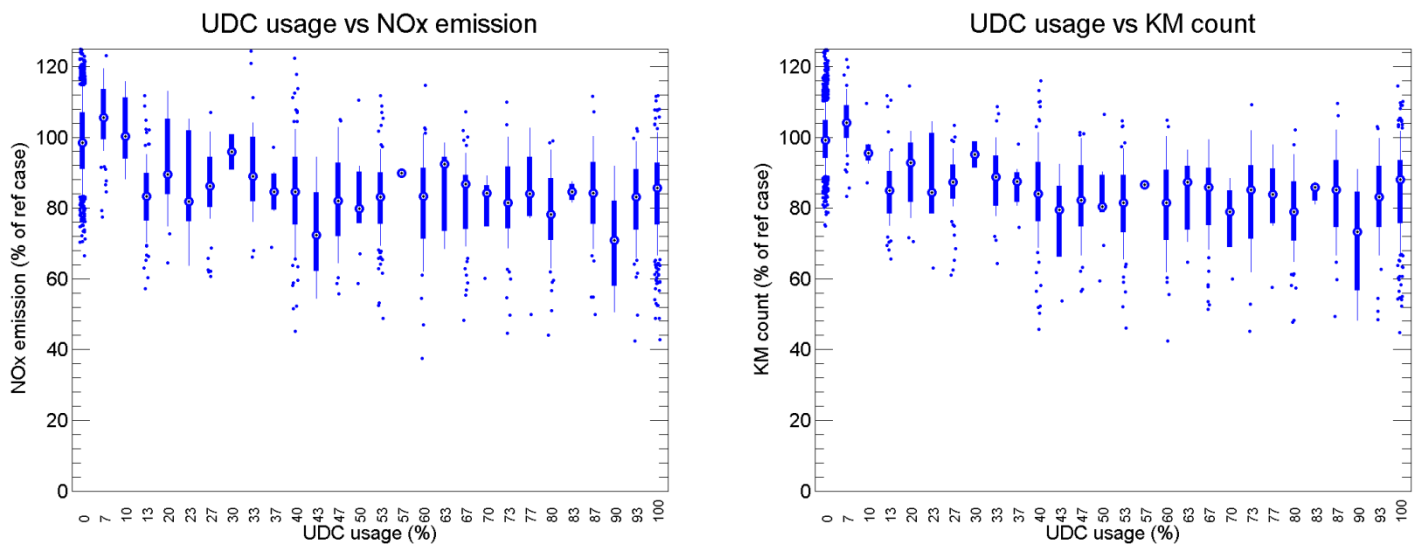


Figure 33: Relation between percentage of usage vs. NOx emission with respect to reference case (no UDC) (left) and Relation between percentage of usage vs. km count with respect to reference case (no UDC) (right)

The negative relation for increasing UDC usage percentage with both NOx emission and km count is visible in the boxplot. Though the relationship is not as strong as expected from literature, in reference with no UDC deliveries the NOx emission decreases with 19.0% with standard deviation of 13.7 % and the km count decreases with 18.8% with standard deviation of 12.9%. This was comparing the 0% UDC usage case with the cases in the interval between 70% and 90% UDC usage.

The fact that this is less than in the literature found is possibly caused by the fact that the same truck size was assumed for both the freight carriers and UDC. Normally the UDC often has larger truck sizes (as indicated in an interview by Kim Hassal).

6.2. Effect fee on UDC usage for various delivery schemes

This can be investigated under three conditions. The first condition is that the freight carriers drop their goods at the UDC early in the morning (before 9:00), thereby giving the UDC the time for the whole day to deliver. This allows for optimal routing for the UDC as it can be done with the whole set of goods for that day. The second case is that the freight carriers can drop off the goods at the UDC before 14:30, narrowing the time for UDC delivery and requiring vehicle routing before all the goods are present. The second case is more realistic and more complex from the UDC point of view, while providing additional benefit for the freight carriers of flexible drop-off. Per delivery scheme 360 runs are performed. In the Appendix A: Additional results the cases are displayed per scenario.

Upon comparing the various delivery schemes with respect to fee vs. UDC usage and financial situations all three hold a negative business case and can only charge a low fee.

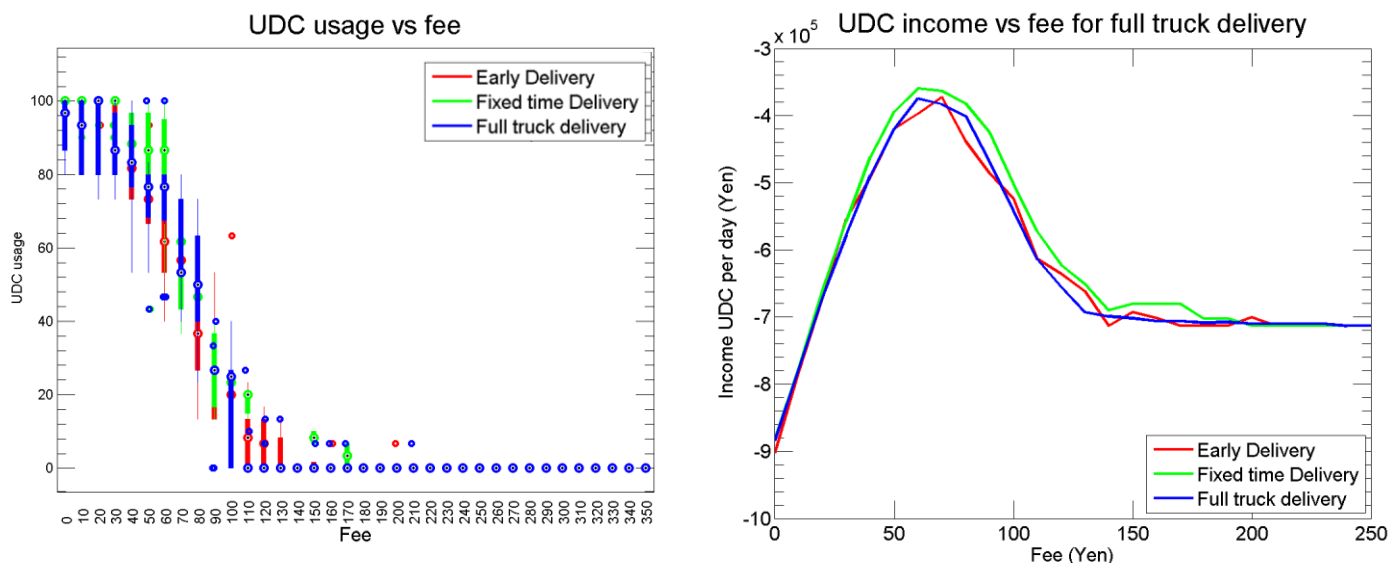


Figure 34: UDC usage vs. fee for all truck deliveries (left) UDC income vs. fee for all truck deliveries (right), (red = early delivery, green = fixed time delivery and blue = full truck delivery)

In the case of Early morning delivery scheme it is visible that the UDC will have a lot of customers below 50 Yen (80% usage border) very few customers at a rate above 100 Yen (20% usage border). Independent of the fee, the UDC business case remains negative with a peak at 70 Yen at - 370.000 Yen per day. In the fixed time delivery scheme it is visible that the UDC will have very few customers at a rate above 105 Yen (20% usage border) and a lot of

customers above 55 Yen (80% usage border). No matter what the fee is of the UDC the business case remains negative with a peak at 60 Yen at - 300.000 Yen per day. Therefore also this business case is not positive. Lastly in the full truck delivery scheme, the UDC will have very few customers at a rate above 100 Yen (20% usage border) and enough below 40 Yen (80% usage border). The UDC business case remains negative with a peak at 70 Yen at - 320.000 Yen per day. Therefore also this business case is not positive. In the NOx emission and km count data show no clear distinction between the three cases (corresponding graphs are in Appendix A: Additional Results). For the financial results no standard deviation is presented as this standard deviation contains properties from two dimensions.

Table 16: Financial impact of the delivery schemes on 4 UDC parameters

Delivery scheme	Optimal fee (Yen)	Maximum income per day (Yen)	80% usage boundary (Yen)	20% usage boundary (Yen)
Scheme 1	70	-370.000	50	100
Scheme 2	60	-300.000	55	105
Scheme 3	70	-320.000	40	100

Table 17: NOx emission and km count information for various delivery schemes

Delivery scheme	Mean difference NOx emission	Standard deviation NOx emission	Mean difference km count	Standard deviation NOx emission
Scheme 1	21.2%	10.6%	17.0%	9.1%
Scheme 2	24.8%	11.3%	19.8%	10.5%
Scheme 3	24.2%	8.2%	18.5%	7.5%

There is no significant difference in type of delivery scheme with an impact on the UDC or freight carriers. However the first scheme, where freight carriers have to deliver goods before 9:00 am is very tough. Therefore the choice is made for fixed time UDC delivery scheme. This was chosen over the full truck delivery scheme as this sometimes results in late deliveries at retailers caused by the fact that sometimes the truck is send out right before 13:30 with its 8 deliveries. Further fine tuning of this delivery scheme is required.

6.3. Effect of congestion on fee vs. usage

Here the effects of the three different traffic conditions are made (in the fixed time delivery scheme with normal congestion, no toll and no subsidy). The cases represent slow, normal and fast traffic conditions (corresponding to an average speed of 13 km/hour, 15 km/hour and 18 km/hour respectively). Per congestion rate 360 runs are performed.

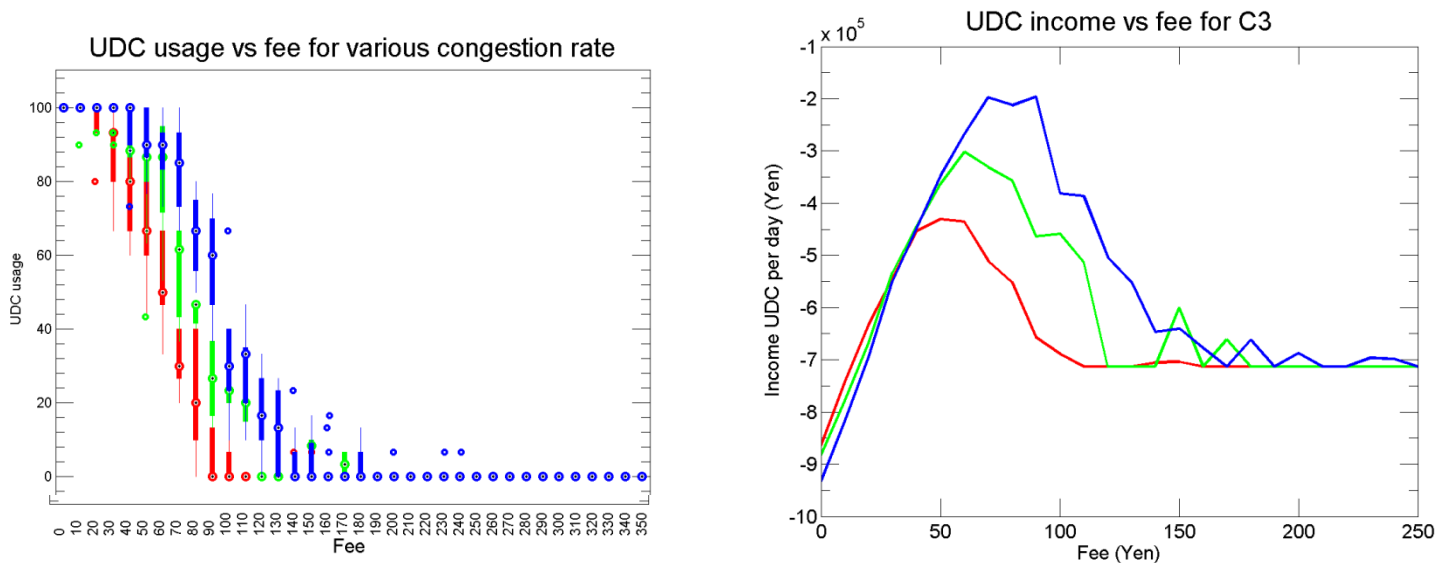


Figure 35: UDC usage vs. fee for various fees for all congestion rate (average) (left); Impact on UDC income for various fees for all congestion rate (average) (right), (red = fast traffic, green is normal traffic, blue is slow traffic)

The UDC business case remains negative with a peak at 50 Yen at - 430.000 Yen per day for low congestion rate (fast traffic). The UDC will have very few customers at a rate above 80 Yen (20% usage border) and enough customers with a fee lower than 40 Yen (80% usage border). At normal congestion rate the UDC will have a lot of customers below 50 Yen (80% usage border) and very few customers at a rate above 105 Yen (20% usage border). No matter what the fee is of the UDC the business case remains negative with a peak at 60 Yen at - 300.000 Yen per day. Even for highly congested traffic, the business case of the UDC remains negative with a peak at 80 Yen at - 200.000 Yen per day. The UDC will have very few customers at a rate above 110 Yen (20% usage border) and a lot below 80 Yen (80% usage border).

Table 18: Financial impact of congestion rates on 4 UDC parameters

Congestion rate	Optimal fee (Yen)	Maximum income per day (Yen)	80% usage boundary (Yen)	20% usage boundary (Yen)
Rate 1	50	-430.000	40	80
Rate 2	60	-300.000	50	105
Rate 3	70	-200.000	80	110

From this graph and table, it is clearly visible that the freight carriers are willing to pay more in congested cities than during less congested conditions. This relates to the fact that UDC is

more successful in highly congested cities as here the price to be charged is expected larger. However also environmental impact varies for the cases.

Table 19: NOx emission and km count information for various congestion rates

Congestion rate	Mean difference NOx emission	Standard deviation NOx emission	Mean difference km count	Standard deviation NOx emission
Rate 1	12.0%	15.7%	15.4%	13.4%
Rate 2	25.2%	15.0%	19.8%	14.4%
Rate 3	27.5%	9.5%	21.2%	9.2%

In the table above, the decrease in NOx emission and km count is presented with corresponding standard deviation. There is a significant different between rate 1 and 3 over the 31.2% significance level (assuming a normal distribution) for both NOx emission a km count. With a higher congestion rate the decrease due to UDC usage is larger. All differences with the reference case for both NOx emission as well as km count are more than 1 standard deviation. In Appendix A: Additional Results the data is displayed graphically. Therefore it is concluded that the expected benefits in km count and NOx emission decrease are larger in more congested cities. In congested cities the financial performance of the UDC is improved and a higher fee charge can be requested. This correlates with the literature research in the initial chapter that UDC are more successful in highly congested cities.

6.4. Effect of toll on fee vs. usage

Here the effects of the policy measure toll are discussed. There can be no toll, low toll level (400 Yen) or high toll level (800 Yen). The toll has to be paid upon entering the city centre. The model made 360 runs in the fixed time delivery scheme with normal congestion, no toll and no subsidy.

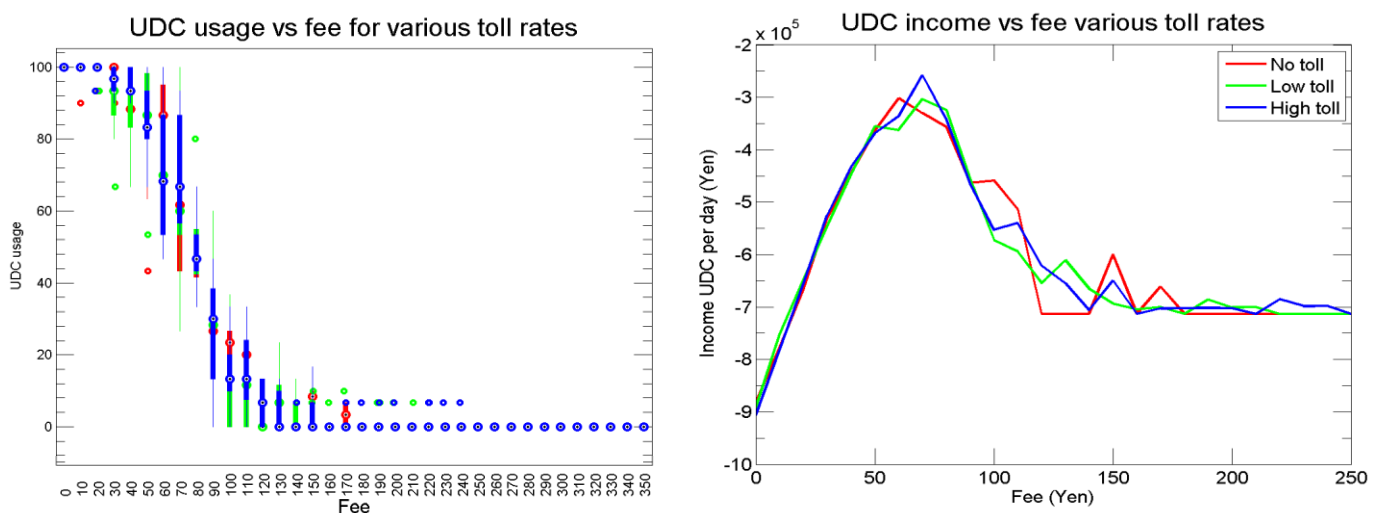


Figure 36 : UDC usage vs. fee for congestion rate 3 (left) and Impact on UDC income for various fees for congestion rate 3 (average) (right)

For the different toll rates, there actually isn't a significant difference between the cases. The UDC business case remains negative with a peak at 60 Yen at - 300.000 Yen per day no additional toll to enter the centre of the city. The UDC will have very few customers at a rate above 105 Yen (20% usage border) and enough customers with a fee lower than 55 Yen (80% usage border). At the low toll rate the UDC will have a lot of customers below 55 Yen (80% usage border) and very few customers at a rate above 105 Yen (20% usage border). No matter what the fee is of the UDC, the business case remains negative with a peak at 70 Yen at - 300.000 Yen per day. With the high toll rate the business case of the UDC is slightly better, but remains negative with a peak at 70 Yen at - 260.000 Yen per day. The UDC will have very few customers at a rate above 105 Yen (20% usage border) and a lot below 55 Yen (80% usage border).

Table 20: Financial impact of toll rates on 4 UDC parameters

Congestion rate	Optimal fee (Yen)	Maximum income per day (Yen)	80% usage boundary (Yen)	20% usage boundary (Yen)
Rate 1	60	-300.000	55	105
Rate 2	70	-300.000	55	105
Rate 3	70	-260.000	55	105

Also in the NOx emission decrease and km count decrease no important differences are found (see the table below).

Table 21: NOx emission and km count information for various toll rates

Congestion rate	Mean difference NOx emission	Standard deviation NOx emission	Mean difference km count	Standard deviation NOx emission
Rate 1	25.2%	15.0%	19.8%	14.4%
Rate 2	24.3%	11.7%	19.8%	10.9%
Rate 3	22.3%	12.5%	17.7%	12.3%

Even though additional toll in the city correlates positively with UDC performance, here the positive relation is not observed. Possibly the presence of toll indicates a highly congested city centre, where entry boundaries are posted to decrease traffic. The price of the fee 400 or 800 Yen however is too low to make a significant difference on the total costs for the freight carrier as the price to go to the UDC for all the parcels is a factor 100 more. Therefore it is concluded that toll does not affect UDC usage directly.

6.5. Impact of subsidy on the UDC business case

For the impact of subsidy the case is investigated where the municipality pays a certain percentage of the fee for the freight carrier. This was investigated for the case of normal traffic conditions, no toll and fixed time UDC delivery scheme. The subsidy can cover 0%, 25% 33% or 50% of the fee. The plots for UDC usage vs. fee are plotted separately here for practical reasons with Matlab.

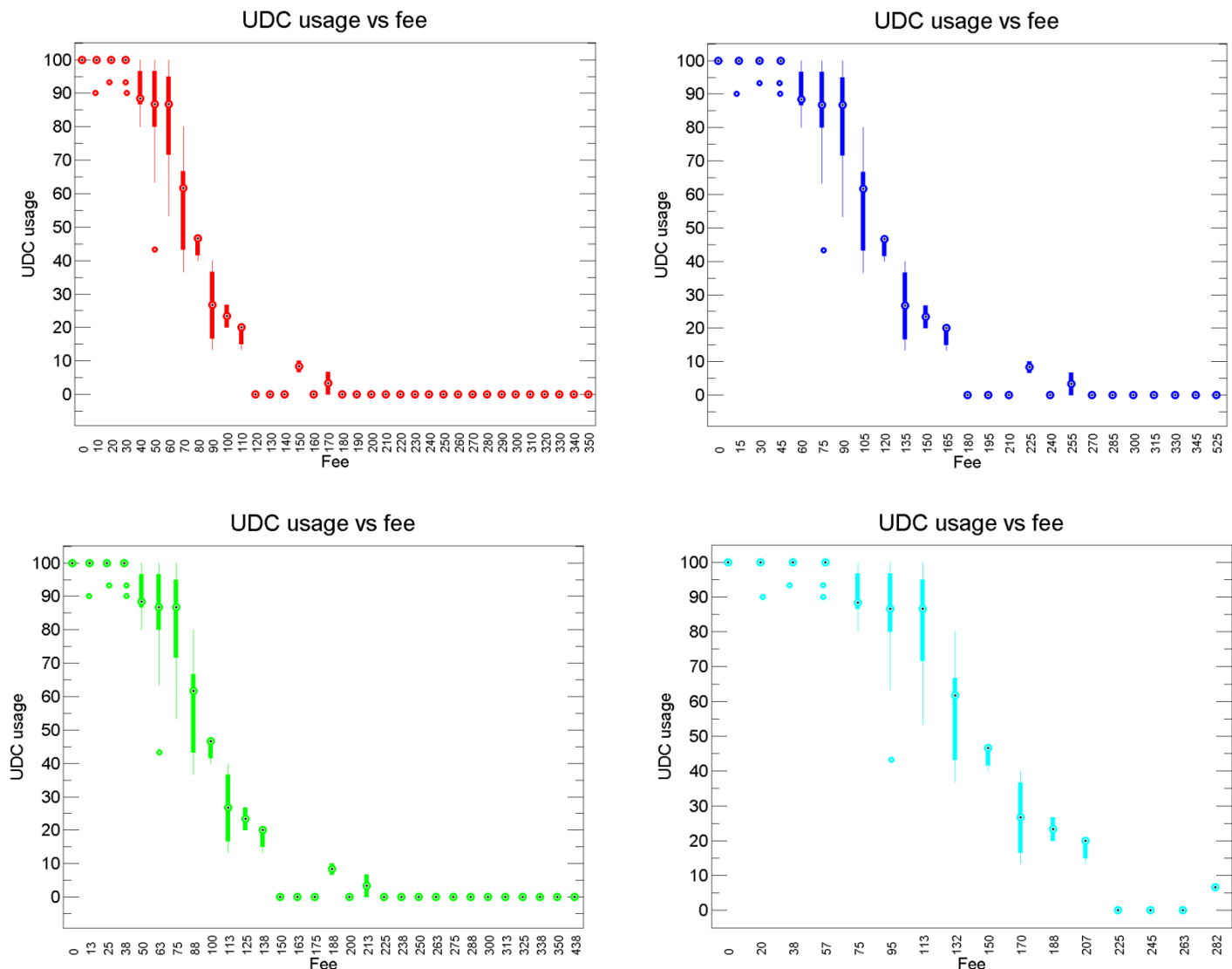


Figure 37: UDC usage vs. fee at no subsidy (red, top left); 20% of fee subsidy (green, bottom left); 33% of fee subsidy (blue, top right); 50% of fee subsidy (cyan, bottom right)

As can be seen, providing subsidy as a percentage of fee increases the price to charge for the UDC. This has positive business implications.

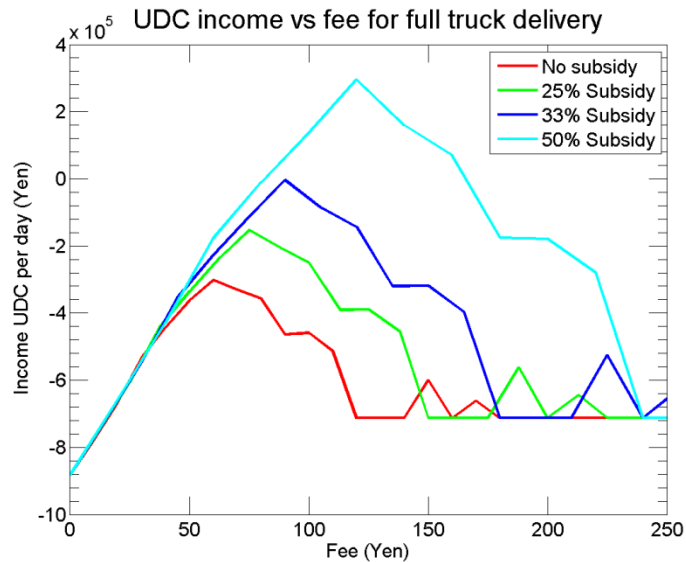


Figure 38: UDC income vs. fee at no subsidy (red), 20% of fee subsidy (green), 33% of fee subsidy (blue), 50% of fee subsidy (cyan)

The information is contained in the following table. No NOx emission decrease or km count is displayed as it has the same dependence on the UDC usage rate as displayed in the first paragraph.

Table 22: Financial impact of subsidy rate on 4 UDC parameters

Subsidy rate	Optimal fee (Yen)	Maximum income per day (Yen)	80% usage boundary (Yen)	20% usage boundary (Yen)
Rate 1	60	-300.000	50	105
Rate 2	75	-150.000	65	130
Rate 3	90	-3.400	80	160
Rate 4	120	300.000	105	210

Subsidy has the positive effect on the business case of the UDC and this matches results from the literature, where it was indicated as a main success factor.

6.6. Impact of dynamic fee on the UDC business case

For the dynamic UDC fee five cases were tested with varying fee per hour. These are displayed in the figure below.

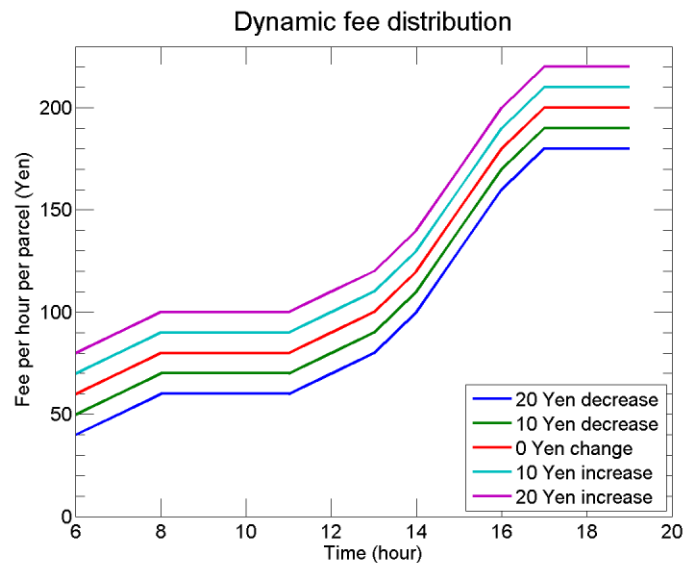


Figure 39: Dynamic UDC fee profile

For this scenario the impact of NOx emission and km count reduction correspond to the scenario with no toll, no subsidy, fixed time UDC delivery scheme and normal congestion. The average UDC usage percentage is presented per fee scenario.

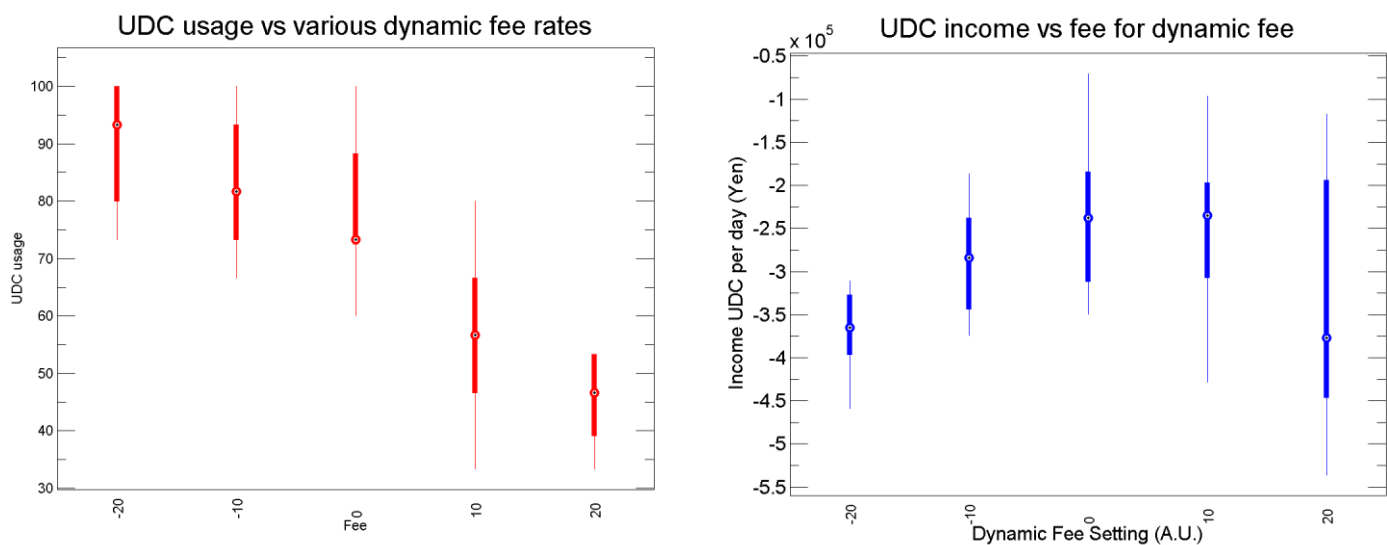


Figure 40: UDC Usage (%) vs. Dynamic fee rate (A.U.) (Left) and UDC income (Yen) vs. Dynamic fee rate (A.U.) (right)

The mean and corresponding standard deviation per fee setting are presented below.

Table 23: Mean and standard deviation of the UDC usage percentage per dynamic fee setting

Fee setting	Mean of UDC usage (%)	Standard deviation of UDC usage (%)	Mean of UDC Income (Yen)	Standard deviation of Income (Yen)
-20	90	11	-368.000	43.700
-10	83	13	-284.000	67.400
0	79	12	-239.000	86.100
10	58	14	-258.000	102.000
20	46	7.6	-342.000	144.000

The differences in mean of the UDC usage percentage are around or more than one standard deviation for the -20, 0, 10 20 dynamic fee setting. The 20 setting is not significantly different from any other fee setting due to the high standard deviation for the UDC income. Besides the 20 setting, the -20 setting is different from -10, 0 as well as 10. In total 50 runs were performed.

The fact that the UDC income is almost significantly higher than the -300.000 Yen per day in the similar situation for fixed fee, is highly encouraging. However further and full evaluation of all possible dynamic fee settings is desired to find potential better dynamic fee settings than the currently used settings.

6.7. Conclusion

NOx emission and the km count hold a negative relation with increasing UDC usage as was expected from literature and common sense. In reference with no UDC deliveries The NOx emission decreases with 19.0% with standard deviation of 13.7 % and the km count decreases with 18.8% with standard deviation of 12.9% in the interval between 70% and 90% UDC usage. This data set contains 2318 runs. The variance originates from the variable demand location, fee setting, GA and truck decision making. There was no toll requirement, normal traffic congestion and no subsidy and the fixed time delivery scheme was used. These settings are always used in the follow tests as well, except for the parameters being varied.

There is no significant difference of one standard deviation in NOx emission reduction and km reduction for the various delivery schemes. The early delivery scheme, fixed time delivery scheme and the full truck delivery scheme were tested. Also no differences are observed in the UDC business case parameters. However the first scheme, where freight carriers have to deliver goods before 9:00 am seems unsuitable for reality.

For the various congestion rates there was a significant difference between the low and high congestion rate of over one standard deviation. Therefore expected benefits in km count and NOx emission reduction are larger in more congested cities. The financial performance of the UDC is improved and a higher fee charge can be charged in congested cities. This information correlates with literature research.

The positive relation between toll in cities and the UDC business case or the environmental parameters was not found. Toll settings could be no toll, low toll or high toll. Toll was identified as a success factor for UDC performance, but seems to act indirectly. Possibly the presence of toll indicates a highly congested city centre, where entry boundaries are posted to decrease traffic. The price of the fee 400 Yen (low toll) or 800 Yen (high toll) however is too low to make a significant difference on the total costs for the freight carrier as the price to go to the UDC for all the parcels factor 100 more.

In all the previous mentioned business cases the daily income of the UDC is negative, but with a subsidy of 50% of the fee the UDC produces it first positive business case. The subsidy can be no subsidy, 25% payment of the fee price, 33% payment of the fee of 50% payment of the fee. Subsidy has the positive effect on the business case of the UDC and this matches results from the literature, where it was indicated as a main success factor.

For the five different dynamic fee settings, the differences in mean of the UDC usage percentage are around or more than one standard deviation for the -20, 0, 10 and 20 dynamic fee setting. Here 50 runs were performed. Besides the 20 setting, the -20 setting is different from -10, 0 as well as 10 for the UDC income. The 20 setting is not significantly different from any other fee setting due to the high standard deviation for the UDC income. The fact that the UDC income is almost significantly higher than the -300.000 Yen per day in the similar situation for fixed fee, is highly encouraging. However further and full evaluation of all possible dynamic fee settings is desired to find potential better dynamic fee settings than the currently used settings.

7. Policy measures and their impact on the business case

In this last chapter three different cases are compared to investigate in which situations the UDC can be financially viable. Since congestion rate and subsidy have the largest effect to improve the business case for the UDC, these will be investigated to find the positive business case for the UDC.

The three cases to be compared are: a reference case, highly congested city case and the involved municipality case. These cases have the following properties as indicated in the table.

Table 24: Parameters for the three scenario's.

Parameter	Reference case	Congested city	Involved Municipality
Congestion level	Normal	High	High
Toll settings	None	None	None
Delivery scheme	Fixed time	Fixed time	Fixed time
Subsidy	None	None	33%
Fee setting	Dynamic (0)	Dynamic (0)	Dynamic (0)

The output of this comparison is measured for the UDC in their business case. Further results on NOx emission and km reduction are presented for environmental aspects. The financial benefits for the FC carriers will also be displayed. Lastly the financial implication for the municipality is calculated with the needed subsidy and the NOx emission reduction. Hereby it is assumed that 5 gr of NOx is emitted per traveled km and the value associated with 1 kg of NOx is 12 euro per kg (Kuiper, 2006).

For these scenarios the impact of NOx emission and km count reduction corresponds to the scenario with no toll, no subsidy, fixed time UDC delivery scheme and normal or high congestion. The average UDC usage percentage is presented per fee scenario.

First the impact of these cases from various perspectives is presented. The UDC looks at the UDC usage percentage and the UDC income. The next angle is environmental and the impact on NOx emission and km count decrease is displayed. This is followed by the costs for the freight carriers. Lastly the municipality looks at the subsidy it has to pay and the money for NOx emission inside the city. Hereby it is assumed the municipality will invest 12 euro per kg NOx to decrease emissions. The presentation of the data itself per scenario and dynamic fee settings, can be found in Appendix A: Additional results.

7.1. Impact of scenarios on the UDC

Here the impact on the UDC income and UDC usage percentage is displayed for different dynamic fees and for the various scenarios.

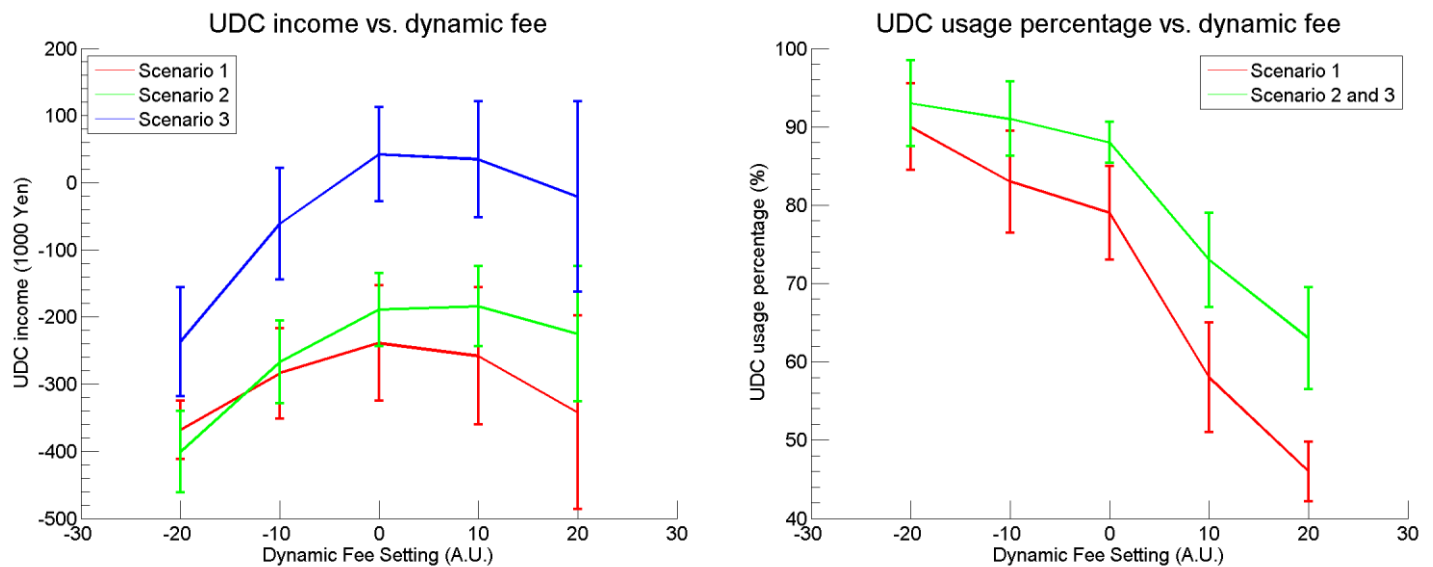


Figure 41: Income of the UDC vs dynamic fee (left) and UDC usage percentage vs dynamic fee (right)

The UDC usage rate is the same for scenario 2 and 3 as the difference lies in the additional subsidy from the municipality. The third scenario is significantly different from the first two for the UDC income. The UDC income is the highest for the dynamic fee rates of 0 and 10. Differences in UDC usage for the various fee rates are only significantly high at the dynamic fee setting 10 and 20. Within scenarios the different dynamic fee settings also have some significant differences.

7.2. Impact on environmental parameters

Scenario 2 and 3 hold the same NOx emission and km count decrease as only the subsidy is varied. The difference for NOx emission and km count decrease between scenario 1 and scenario 2/3 is not more than one standard deviation. Also variation within one scenario is not relevant; this is caused by the fact that the UDC usage percentages have a relatively high standard deviation. This information is presented below.

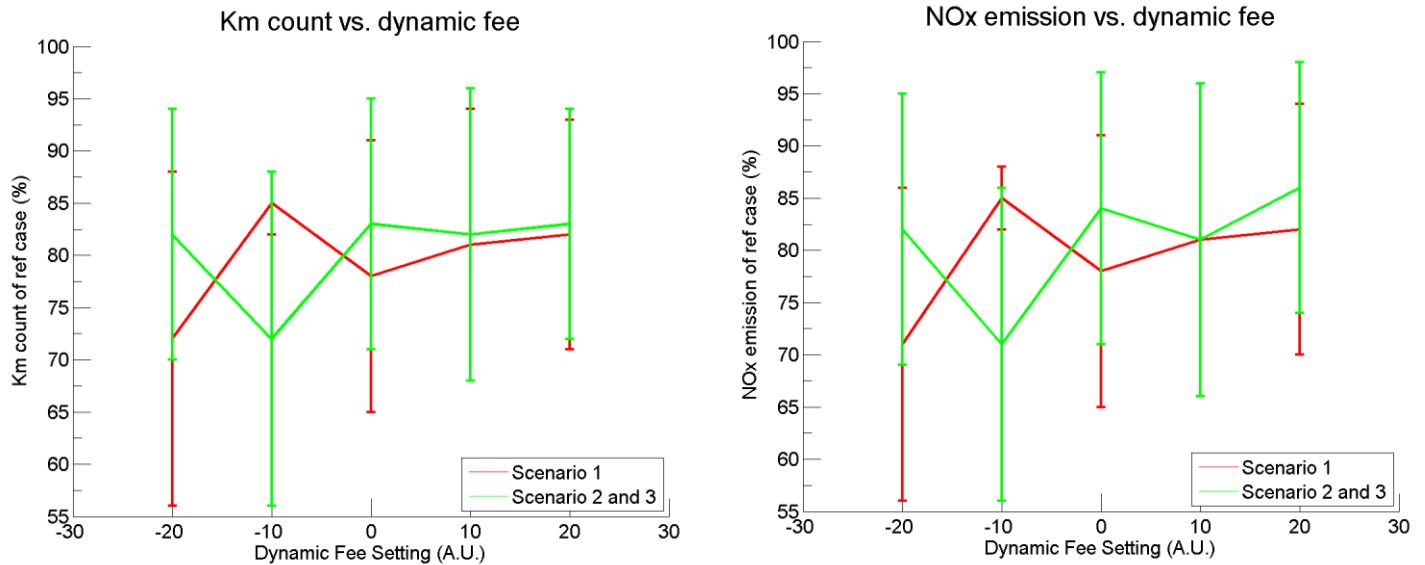


Figure 42: Impact of the scenarios on km count (left) and NOx emission (right) vs. dynamic fee setting

7.3. Impact on freight carriers

Below the impact on the freight carriers is displayed.

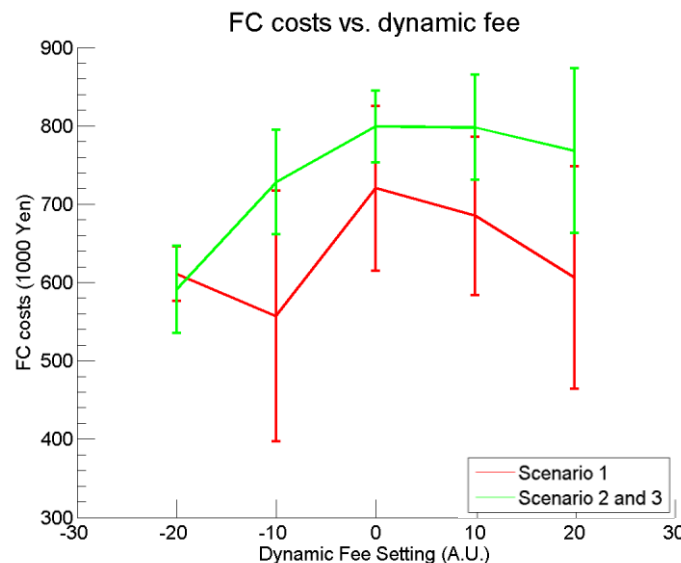


Figure 43: Variable costs for the FC terminals vs. dynamic fee for different scenarios

The freight carriers also experience the same situation in scenario 2 and 3. The costs generally increase as the UDC is not longer a cheaper alternative; the freight carriers prefer the lower dynamic fee as this decreases their costs most. Surprising here is that the only significant difference is the -20 dynamic fee setting in the second and third scenario. In this comparison freight carriers only use the UDC in case of financial benefit.

7.4. Impact on municipality

Below the impact on the municipality is displayed.

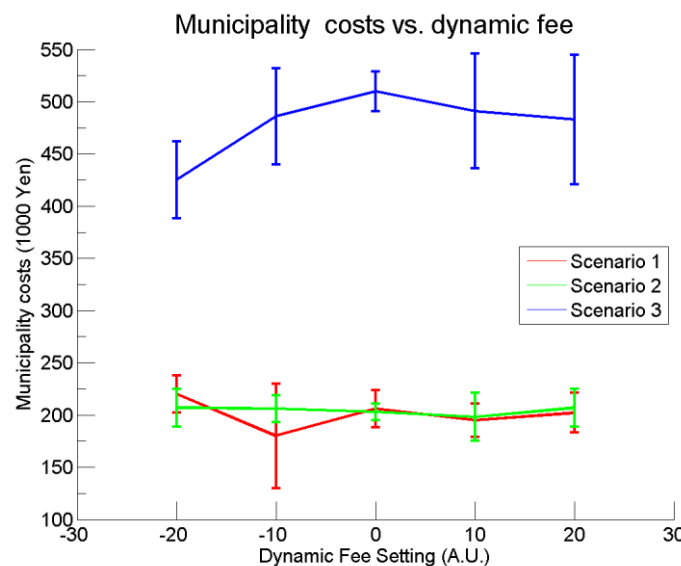


Figure 44: Costs for NOx emission and UDC subsidy per day for the municipality vs. dynamic fee for the various scenarios.

In scenario 1 and 2 the costs for the NOx emission are similar. However when the subsidy costs at the 33% level is added to the costs in the third scenario, there is a significant difference. The costs for the municipality approximately increase a factor 1.5. However since there is no significant decrease in NOx emission or km count between the second and third scenario, the municipality will need other incentives like social or corporate well-being associated with the UDC, before the municipality will participate in providing such high subsidies.

7.5. Conclusion

In this chapter the three cases are compared: a reference case, highly congested city case and the involved municipality case. In all scenarios the toll rate is zero and the delivery scheme is fixed time delivery. In the reference case the congestion rate is normal, but in the other scenarios the congestion rate is high. In the involved municipality scenario a subsidy of 33% of the fee is provided by the municipality.

Comparing the three scenarios there is no significant difference between the NOx emission or km count reduction and the freight carriers costs. For the UDC income there is a significant increase in the third scenario with respect to the scenarios. Also for the municipality there is a significant increase in costs in the third scenario.

However since there is no significant decrease in NOx emission or km count between the second and third scenario, the municipality will need other incentives like social or corporate well-being associated with the UDC, before the municipality will participate in providing such high subsidies. Also other benefits like reduction in road maintenance are possible, but from this comparison it is clear that other incentives need to be present for the municipality to participate. Therefore for a UDC to be profitable without or with a small subsidy, it seems to be necessary to find other activities like for example pre-retail activities (See Figure 11) to increase profitability.

Conclusion

In a 6 month research at Kyoto University and Delft University of Technology, the urban distribution centre (UDC) was investigated with a multi-agent model to find situations where the UDC can be financial viable. From a literature study it was understood that in this city logistics multi-agent model the following aspects were required: commodity estimation, vehicle routing, traffic environment, agent interaction and various policy measurements to test their impact on social, environmental and financial parameters. From further research it became clear that the interesting policies to test the impact on the business case of the UDC are: different congestion and toll rates, subsidy and a dynamic fee. Hereby a gap in literature is addressed by dynamic modelling with a multi-agent model with vehicle routing and incorporating dynamic usage. Dynamics are included with variable traffic conditions (resulting in dynamic travel times) and dynamic demand locations. Dynamic UDC usage was tested by various delivery schemes and a dynamic fee, which changes during the course of the day.

With this information a new multi-agent model with vehicle routing based on the genetic algorithm was constructed containing: 3 freight carriers terminals, 1 UDC centre, 12 - 18 trucks, 240 retailers and a road network (including 300 roads, 114 streets and 60 node agents).

After generating the multi-agent model it was checked if an increasing UDC usage percentage corresponds to decreasing both NOx emission and km count. The NOx emission decreases with 19.0% with standard deviation of 13.7 % and the km count decreases with 18.8% with standard deviation of 12.9% in reference with no UDC reference case. This was comparing the 0% UDC usage case with the cases in the interval between 70% and 90% UDC usage. Each presented input settings consists of 360 model runs for the 36 UDC fee settings.

Different UDC delivery schemes (early delivery, fixed time delivery or full truck delivery), congestion rates (low, normal or high), toll rates (none, low or high) and subsidy levels (none, 25%, 33% or 50% subsidized) were tested. Different delivery schemes or toll rates do not have a significant impact on NOx emission decrease or km count reduction. There is however a significant difference between the low and high congestion rate of 1 standard deviation for NOx emission. Subsidy provision doesn't impact NOx emission or km count, only UDC financial performance.

Next the impact on four UDC financial parameters is investigated for these different settings. These parameters present the fee with the highest UDC income per day, the highest average UDC income for the dynamic fee settings and the fee at the 80% and 20% UDC usage boundary. Here the only positive business case for the UDC is with a subsidy rate of 50% (rate 4), which

indicates the complexity of generating a positive business case for an UDC. For the optimal fee the fee with the highest average income per day was chosen. The fees found in the model are lower than prices indicated in literature (Ieda, 2005).

To enhance UDC financial performance the dynamic UDC fee was tested. This fee varies during the different hours of the day. Five cases were tested by increasing the general dynamic fee curve with -20, -10, 0, 10 or 20 Yen per parcel. The scenario has no toll, no subsidy, fixed time UDC delivery scheme and normal congestion. The impact of NOx emission and km count reduction corresponds to the fixed fee scenario. In total 50 runs were performed. Significant differences between various dynamic fee settings are present. The fact that the UDC income is almost significantly higher than the -300.000 Yen per day in the similar situation for fixed fee is highly encouraging. However further and full evaluation of all possible dynamic fee settings is desired to find potential better dynamic fee settings than the currently used settings.

At the end of this research the three cases are compared for dynamic fee usage: a reference case, highly congested city case and the involved municipality case. In all scenarios the toll rate is zero and the delivery scheme is fixed time delivery. In the reference case the congestion rate is normal, but in the other scenarios the congestion rate is high. In the involved municipality scenario a subsidy of 33% of the fee is provided by the municipality.

Comparing the three scenarios there is no significant difference between the NOx emission or km count reduction and the freight carriers costs. For the UDC income there is a significant increase in the third scenario with respect to the scenarios. Also for the municipality there is a significant increase in costs in the third scenario.

Since there is no significant decrease in NOx emission or km count between the second and third scenario, the municipality will need other incentives like social or corporate well-being associated with the UDC, before the municipality will participate in providing such high and long term subsidies. Other benefits like reduction in road maintenance are possible.

From this research it is clear that generating a positive business for a UDC is highly challenging. For a UDC to be profitable, it seems to be necessary to receive subsidy permanently or/and to find other activities like for example pre-retail activities (See Figure 12) to increase profitability. When collaboration in the form of subsidy is needed, other incentives are required for the municipality to participate.

This less optimistic analysis of the UDC analysis relates to the low UDC success rate of the research of Browne, Sweet et al (2005). The conditions for generating a successful business case for the UDC are hard and limited.

Further research of dynamic fee settings and investigation of the possibility to implement this in a real-life case can enhance the future prospective and success rate for urban distribution centres.

Reflection and Recommendation

Some critical notes are made on the current multi-agent model, followed by recommendations for research on extensions to the model. Lastly reflection from a personal point on the process is presented.

Methodological and multi-agent model reflection

Full dynamic usage investigation

In the current report research is performed in the dynamic fee, but full behavior space research in Netlogo with variation in all phases and prices could enhance the solution quality. However a higher performance computer is required to perform this task as multiple prices per phase with several runs per point increase the number of runs tremendously.

Demand, parcel and UDC business case estimation

Many aspects in the model are based upon rough estimations, which linearly impact the business case as a minimum. The demand estimation is based on model size and relates only one source (van Duin, Quak et al 2008). This information was also used for the business case. There are additional costs of 2 miljoen euro, which can not be traced back and determine over 65% of the total costs. The parcel estimation directly impacts the results tremendously as it enters the entry boundary for using the UDC and directly in the UDC income. Further validation with multiple sources on these parameters is needed.

Extended GA verification

The genetic algorithm is major weak point in the model. The performance rate of the GA is not up to standard in this model. Especially for larger chromosomes the optimal solutions is not always found. This can produce less optimal solutions for the UDC as relatively the freight carriers manage their routing better therefore. Further research on the severeness/impact and possible solutions is highly recommended.

Computation speed and related adaptations

Another major point in this research was computation speed. The transition from programming in Fortran to Netlogo was made and the problem was scaled up from 4 to 10-30 retailers, 20 retailers to 240 retailers and 25 to 60 nodes. Additional dynamics were included by varying demand locations and truck decision making. Some of the sophistication of the Tamagawa model were therefore lost (like Q-learning and reloading the same truck during one day) due to the different functionalities in Netlogo and needed scaling tricks. The scaling of the model is successful. Especially in the genetic algorithm many tricks were implemented

to improve performance to allow for the scaling, which took substantial amount of effort and creativity. Currently the genetic algorithm also uses a different time mutation profile that in the original documentation and the Tamagawa model. The impact of this would be highly interesting to investigate as the binary approach has a clear elegance, but the comparison of the probability profile for the time was not fully understood by me.

Recommendations on extensions to the model

Main points of interest for making extensions for future research:

- Addition of shippers to the model
- Auctioning between shippers, freight carriers and UDC
- Negotiation between shippers, freight carriers and UDC
- Various types of trucks
- Possibility for multiple trips per truck during a day
- Use time window at retailers
- Allow for waiting in the city for truck
- Returning to terminal and send out again of truck
- Modeling of stock at the UDC
- Scaling of the model

It would be highly interesting to add shippers to the UDC and generate an auctioning or negotiating process in Netlogo to simulate these complex relations. A desired addition is also the possibility to use different truck types. Now only 1 type is used, but in reality different truck types impact costs and possibilities tremendously. Also the complexity the load and reload trucks a second time during the day would be highly interested. For this requirement the GA has to be, the problem size reduced or a completely different salutation has to be found.

Additionally various time windows at retailers can be used in stat of delivery between 6:00 and 19:00. Time windows can be narrowed and varied for different retailers. Tidier time windows also require the possibility for trucks to wait inside the city for the appropriate time window. The modeling of stocking at the UDC can also be interesting to investigate additional services for the UDC; with the multi-agent model this is very convenient to do.

Furthermore it would be interesting to scale the model to a different size city for example and to see the effects of this.

Personal Reflection

The goal of this research is to generate a model for a dynamic UDC. Creating a model with vehicle routing, a multi-agent model and research for the business case as well as the modelling choices was a large and fun assignment for a person new to city logistics. This master thesis was a great project to tackle: testing a business case of a social issue, while using technical programming skills.

This model took 6 months to generate and contains complex vehicle routing and a multi-agent model. The broad possibilities of the multi-agent model require further research and therefore it is recommended to find new researchers to continue with this project.

References

Anand N, van Duin R, Tavasszy L, *Multi-agent Modeling for city logistics policy analysis: potentials and challenges*, to be published 2010.

Banks J, *Handbook of Simulation: Principles, Methodology, Advances, Applications and Practice*, New York, John Wiley, 1998.

Benjelloun A, Crainic T, Bigras Y, *Towards a Taxonomy of city Logistics Projects*, The 6th International Conference on City Logistics, 30th June - 2nd July 2009, Puerto Vallarta, Mexico

Bestufs, Various articles from <http://www.bestufs.net/> visited last on 06-04-2010

Boerkamps J, van Binbergen A, *A new approach for modelling and evaluating urban goods distribution*, chapter 2, pag 175, City Logistics I, Institute of systems science research, 1999.

Browne M, Sweet M, Woodburn A and Allen J, University of Westminster, Transport group, *Urban Freight Consolidation Centres Final Report*, November 2005, More information: <http://www.freightbestpractice.org.uk/imagebank/Consolidation%20centres%20Finalreport%20Nov2005.pdf>

Browne M, Piotrowska M, Allen J, Woodburn A, *The potential use of urban consolidation centres in the hotel industry in London*, Chapter 6, Innovations in City Logistics, 2008, Nova Science Publishers, Inc.

Chopra S, *Designing the distribution network in a supply chain*, Transportation Research Part E 39, 123-140, 2003.

Christian A, Meimbresse B, Routhier R, Sonntag H, *Urban freight policy-oriented modeling in Europe*, Chapter 12, Innovations in City Logistics, 2008, Nova Science Publishers, Inc.

Chrobok R, Kaumann O, Wahle J, Schreckenber M, *Different methods of traffic forecast based on real data*, European Journal of Operational Research 155, pag 558-568, 2004.

COST321, *Urban Goods Transport*, Final Report, Brussels, European Commission, 1998

Crainic T, Ricciardi N and Storchi G, *Models for evaluating and planning city logistics systems, transportation science*, vol 43, nr 4, p 432-454, Nov 2009.

Dablanc L, *Goods transport in large European cities: Difficult to organize, difficult to modernize*, Transportation Research Part A 41, 280-285, 2007

Davidson P, Henesey L, Ramstedt L, Törnquist J and Wernstedt W, *An analysis of agent-based approaches to Transport Logistics*, Transportation research Part C 13 (4), p 255-271, 2005.

Delaître L, *A new approach to diagnose urban delivery areas plans*, International Conference on Computers & Industrial Engineering, p 991-998, 2009.

Encyclopedia Britannica, word City,
<http://www.britannica.com/EBchecked/topic/118952/city>, visited 02-04-2010

Freight Best Practice Benchmarking Guide, *Key Performance Indicators for the Next-day Parcel Delivery Sector*, www.frieghtbestpractise.org.uk, 2006

Germain R, Droge C, Spears N, *The implications of just-in-time for logistics organization management and performance*, 01-01-1996, Journal of Business Logistics

Hochbaum D, *Approximation Algorithms for NP-Hard Problems*, ACM publishers, New York, 1997. (note: book was not fully read, but presented for back ground information)

Holguín-Veras J, Patil G, *A Multicommodity Integrated Freight Origin-destination Synthesis Model*, Netw Spat Ecn 8:309-326, 2008

Ieda H. et al, Joint distribution centre research, City logistics conference, 2005.

ISI Web of Knowledge search on City Logistics and Urban Logistics, performed on 12-3-2010.

Marcucci E, Danielis R, *The Potential Demand for an urban freight consolidation centre*, Transportation, vol. 35, nr. 2, pp. 269, 2008

Meimbresse, B, Sonntag H, *Modelling urban commercial traffic with the model Wiver*, Proceedings of Jacques Cartier Conference, Montreal, 2000.

Muñuzuri J, Cortés P, Onieva L and Guadix J, *Modeling Freight Delivery Flows: Missing Link of Urban Transport Analysis*, September 2009, pag 90, Journal of Urban Planning and Development.

Ministry of Land Infrastructure and Transport Japan, Calculation with *Monthly Statistical Report on Motor Vehicle Transport*, geppou-e.xps, <http://www.mlit.go.jp/k-toukei/56/56b0excel.html> visited on 2 April 2010.

Nakamura Y, Taniguchi E, Yamada T, Ando N, *A macroscopic traffic simulator for evaluating urban transport measures for heavy vehicles*, Chapter 12, *Innovations in City Logistics*, 2008, Nova Science Publishers, Inc.

OECD (Organisation for Economic Co-operation and Development), some examples: *OECD Territorial Reviews: Randstad Holland, Netherlands* Policy Brief, March 2007 and *Urban Trends and Policy in China*, Papers 2009/1.

Patier D, Browne M, *A methodology for the evaluation of urban logistics Innovations*, The 6th International Conference on City Logistics, 30th June - 2nd July 2009, Puerto Vallarta, Mexico

Quak H, de Koster M, *Delivering Goods in Urban Areas: How to deal with urban policy restrictions and the environment*, *Transportation Science*, Vol 43, Nr 2, pag 221-227, 2009

Quak H, van Duin R, Visser J, *City Logistics over the years... Lessons learned, Research Directions and Interests*, Chapter 3, *Innovations in City Logistics*, 2008, Nova Science Publishers, Inc.

Reeves C, *Modern Heuristic techniques for combinatorial problems*, Halsted Press, 1993.

van Rooijen T., Quak H., *Binnenstadservice.nl - a new type of urban consolidation centre*, ETC proceedings, 2009.

Roorda M, Cavalrcante R, McCabe S, Kwan H, *A conceptual framework for agent-based modeling in logistics services*, Volume 46, Issue 1, January 2010, Pages 18-31, *Transportation Research Part E: Logistics and Transportation Review*

Routhier J, *Urban goods transport modelling with FRETURB*, Séminaire MATAARI, Aix-en-Provence, 26-10-2006

Ruesch M, Hegi P, Haefeli U, Matti D, Schultz B, Ruetsche P, *Sustainable goods supply and transport in Conurbations: Freight patterns and developments in Switzerland*, Chapter 8, *Innovations in City Logistics*, 2008, Nova Science Publishers, Inc.

Russo F, Comi A, *A modelling system to link end-consumers and distribution logistics*, European Transport. n. 28, 6-19, 2004

Russo F, Comi A, *City Logistics Measures: Comparison and Obtainable Results*, The 6th International Conference on City Logistics, 30th June - 2nd July 2009, Puerto Vallarta, Mexico.

Shannon R, *Introduction to the art and science of simulation*, Proceedings of the 1998 Winter Simulation Conference, D.J. Medeiros, E.F. Watson, J.S. Carson and M.S. Manivannan, eds, 1998.

Tamagawa D, Taniguchi E, Yamada T, *Evaluating city logistics measures using a multi-agent model*, The 6th International Conference on City Logistics, 30th June - 2nd July 2009, Puerto Vallarta, Mexico

Taniguchi E, Noritake M, Yamada T, Izumitani T, *Optimal size and location planning of public logistics terminals*, Transportation Research Part E 35, p 207-222, 1999

Taniguchi E, Shimamoto H, *Intelligent transportation system based dynamic vehicle routing and scheduling with variable travel times*, Volume 12, Issues 3-4, June-August 2004, Pages 235-250, Intelligent Transport Systems: Emerging Technologies and Methods in Transportation and Traffic

Taniguchi E, Thompson R, *City Logistics I*, first International Conference on City Logistics 1999 Cairns, Australia, Institute of Systems Science Research, 1999.

Taniguchi E, Thompson R, Yamada T, *Modelling the behaviour of stakeholders in City Logistics*, Chapter 1, Innovations in City Logistics, 2008, Nova Science Publishers, Inc.

Taniguchi E, Thompson R, Yamada T, van Duin R, *City Logistics, Network Modelling and Intelligent Transport Systems*, Chapter 1 and 2, 2001, Pergamon

Taylor M, Editorial, *Intelligent transport systems: emerging technologies and methods in transportation and traffic*, Transportation Research Part C 12, 167-169, 2004.

Tseng Y, Lin H, Taylor M, *Provision of real-time traffic information for the VRPTW Model*, Chapter 11, Innovations in City Logistics, 2008, Nova Science Publishers, Inc.

van Duin J, Quak H, Munuzuri J, *Revival of cost benefit analysis for evaluating the city distribution centre concept*, Chapter 7, Innovations in City Logistics, 2008, Nova Science Publishers, Inc.

van Duin J, Tavasszy L, Taniguchi E, *Real time simulation of auctioning and re-scheduling processes in hybrid freight markets*, 2007 pag 1050-1066, Transportation Research Part B 41

Vleugel J., *Modelling goods city distribution in the Netherlands*, European Transport n28, pag 20-33, 2004.

Wang Q, Holguín-Veras J, *Tour-based Entropy Maximization Formulations of Urban Freight demand*, Transportation research board 88th annual meeting, 19 feb 2009.

Watkins C, Dayan P, *Technical note Q learning*, Machine Learning, 8, 279-292, 1992.

Wiki SPM 955x, Information from the TU Delft Wiki (with TU Delft netID log-in) on courses SPM9550 and SPM9555.

Wisetjindawat W, Sano K, Matsumoto S, Rathachonkun P, *Micro-simulation model for modeling freight agents interactions in urban freight movement*, 86th Annual meeting of the transportation research board, Washington D.C., January 21-25 2007.

Wooldridge M., *An Introduction to MultiAgent Systems*, Second Edition, John Wiley & Sons, May 2009.

Woudsma C, *Understanding the Movement of Goods, not People: Issues, Evidence and Potential*, Urban Studies, Vol 38, No 13, 2439-2455, 2001.

Zargayouna M, Balbo F, Seemama G, *A multi-agent approach for the dynamic VRPTW*, esaw, 2008.

Appendices

A Additional results

This appendix is an addition to the Chapter Results. The different results for all the individual scenarios are displayed here: delivery schemes, congestion rate, toll rate, subsidy.

A.1 Various delivery schemes

Since the relation with NOx emission and km count for normal traffic congestion is already presented, the impact on the specific delivery schemes are displayed in this appendix. The data is represented per individual scenario as in the main chapter only the combined information is presented.

Early morning drop off by freight carriers

First here the data for the case were the freight carriers have to deliver their goods to the UDC before 9:00 and afterwards the UDC will deliver these goods during the day.

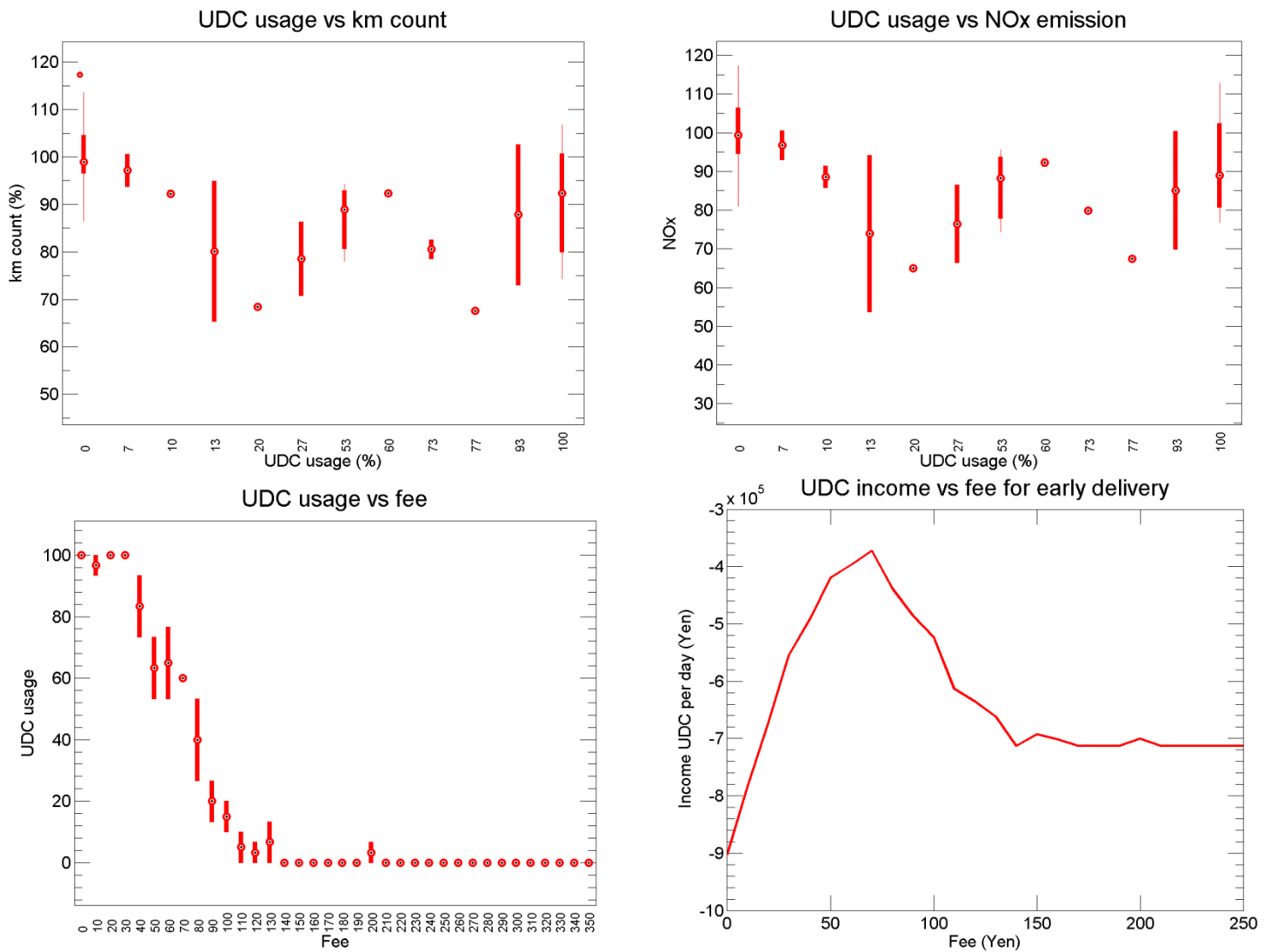


Figure 45: UDC usage vs km count (top left); UDC usage vs. NOx emission (top right); UDC usage vs. fee (bottom left); Impact on UDC income (bottom right). All data is for the early delivery scheme.

Corresponding number and important parameters are presented in Chapter 6.

Fixed time delivery by freight carriers

Here is the data for the case where the UDC sends out trucks at fixed times and the freight carriers have to deliver their trucks before the last time slot. The time slots were at 9:00, 11:00 and 13:30. At the initial times only full (or almost full trucks are sent out), but at 13:30 all trucks are sent out and with a load factor of 4 as otherwise goods will be delivered late to the retailers.

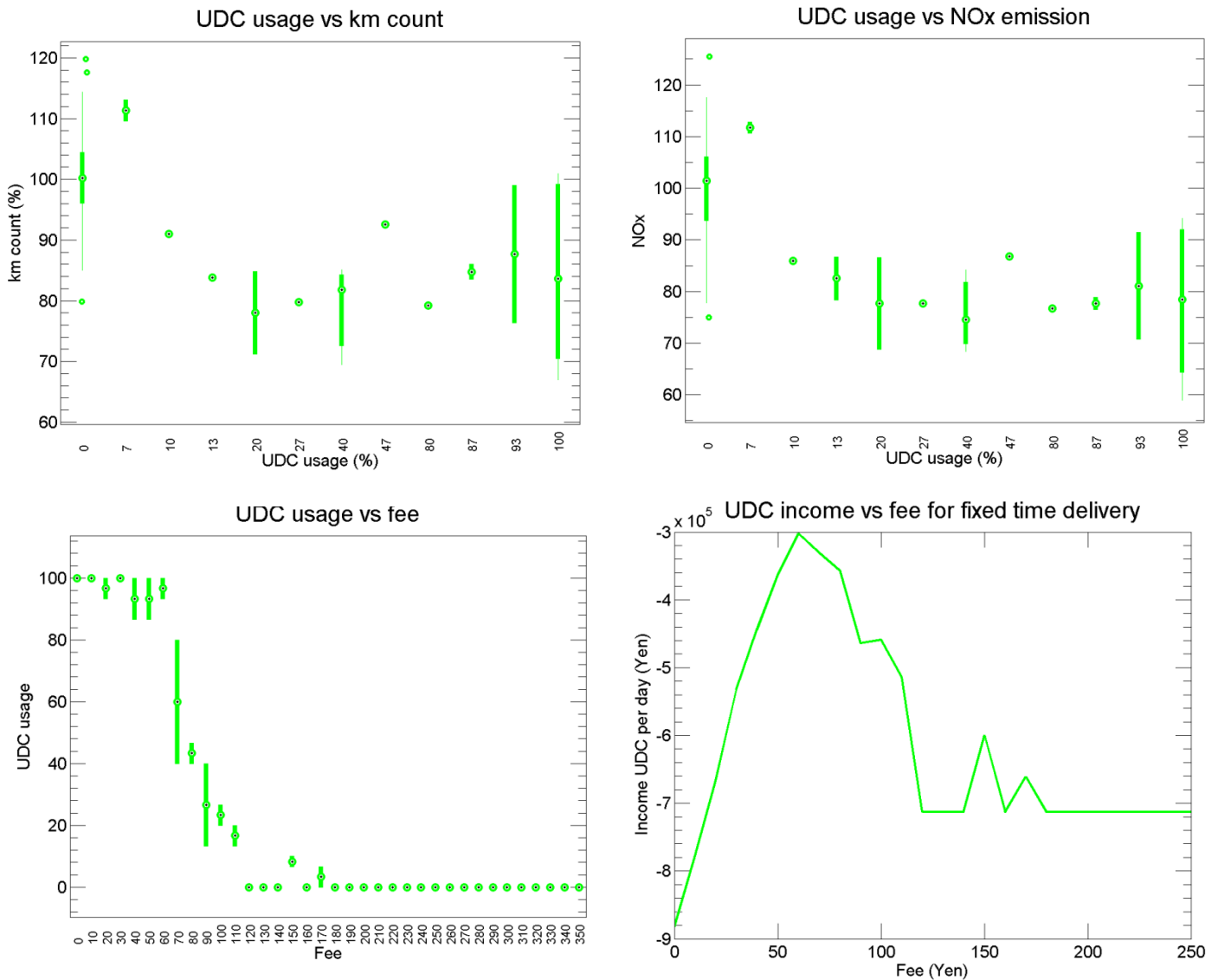


Figure 46: UDC usage vs km count (top left); UDC usage vs. NOx emission (top right); UDC usage vs. fee (bottom left); Impact on UDC income (bottom right). All data for fixed time delivery scheme.

Full truck delivery by freight carriers

The last delivery scheme consists of sending out full trucks, as soon as there is a full truck. At 13:30 the last trucks are send out with load factor 4 as otherwise the delivery will be late at the retailers. Freight carriers have to bring deliveries before this time.

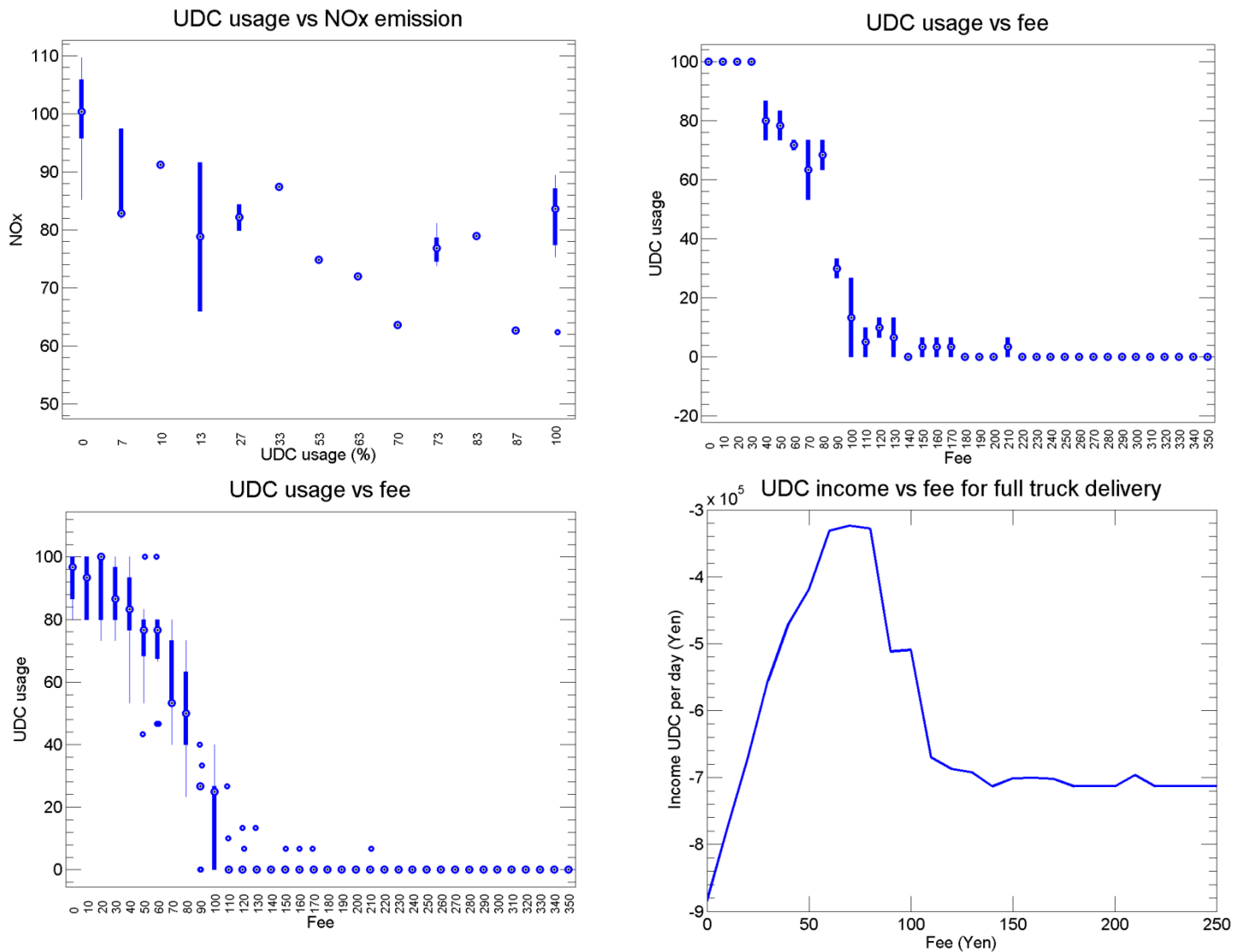


Figure 47: UDC usage vs km count (top left); UDC usage vs. NOx emission (top right); UDC usage vs. fee (bottom left); Impact on UDC income (bottom right). All data for full truck UDC delivery scheme.

A.2 Various congestion rates

Here the data per congestion rate scenario is presented.

A.2.1 Low congestion rate

This congestion rate has the lowest average speed in the city. The various parameters are presented here.

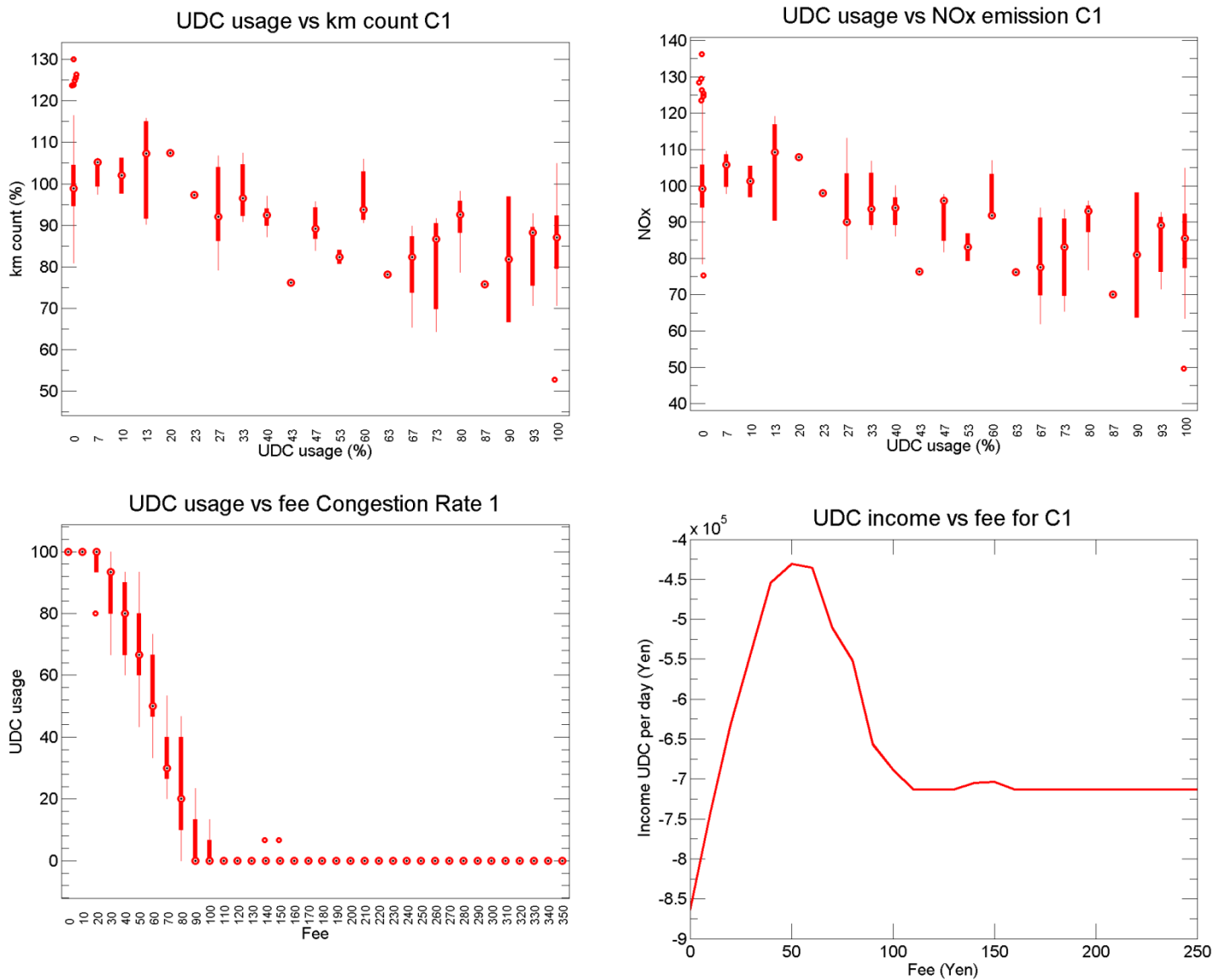


Figure 48: UDC usage vs km count (top left); UDC usage vs. NOx emission (top right); UDC usage vs. fee (bottom left); Impact on UDC income (bottom right). All data for low congestion rate.

A.2.2 Normal congestion rate

This congestion rate has the normal average speed in the city. Below the various parameters are displayed.

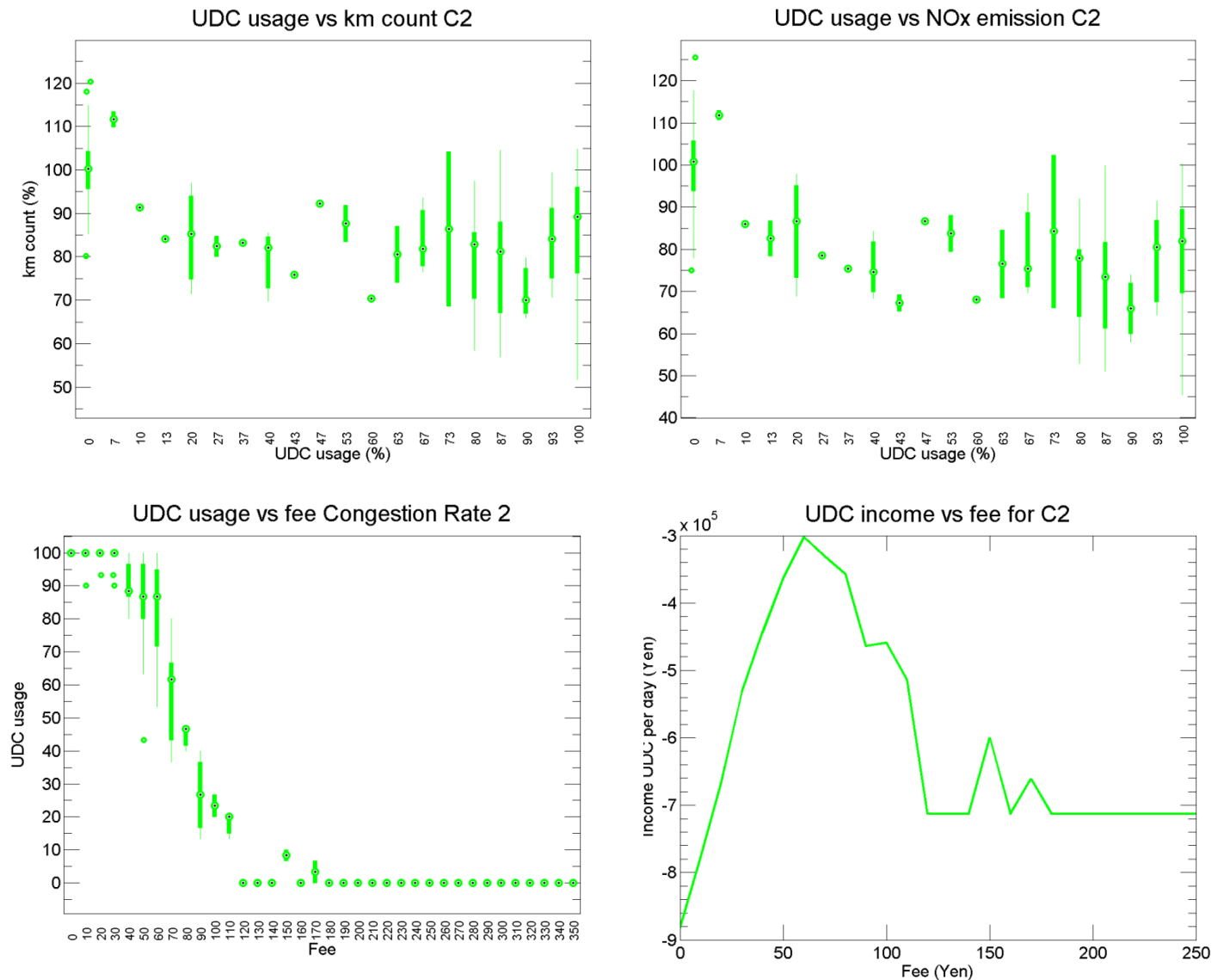


Figure 49: UDC usage vs km count (top left); UDC usage vs. NOx emission (top right); UDC usage vs. fee (bottom left); Impact on UDC income (bottom right). All data is for the normal congestion rate.

A.2.3 High congestion rate

This congestion rate has the normal average speed in the city. Below the various parameters are displayed.

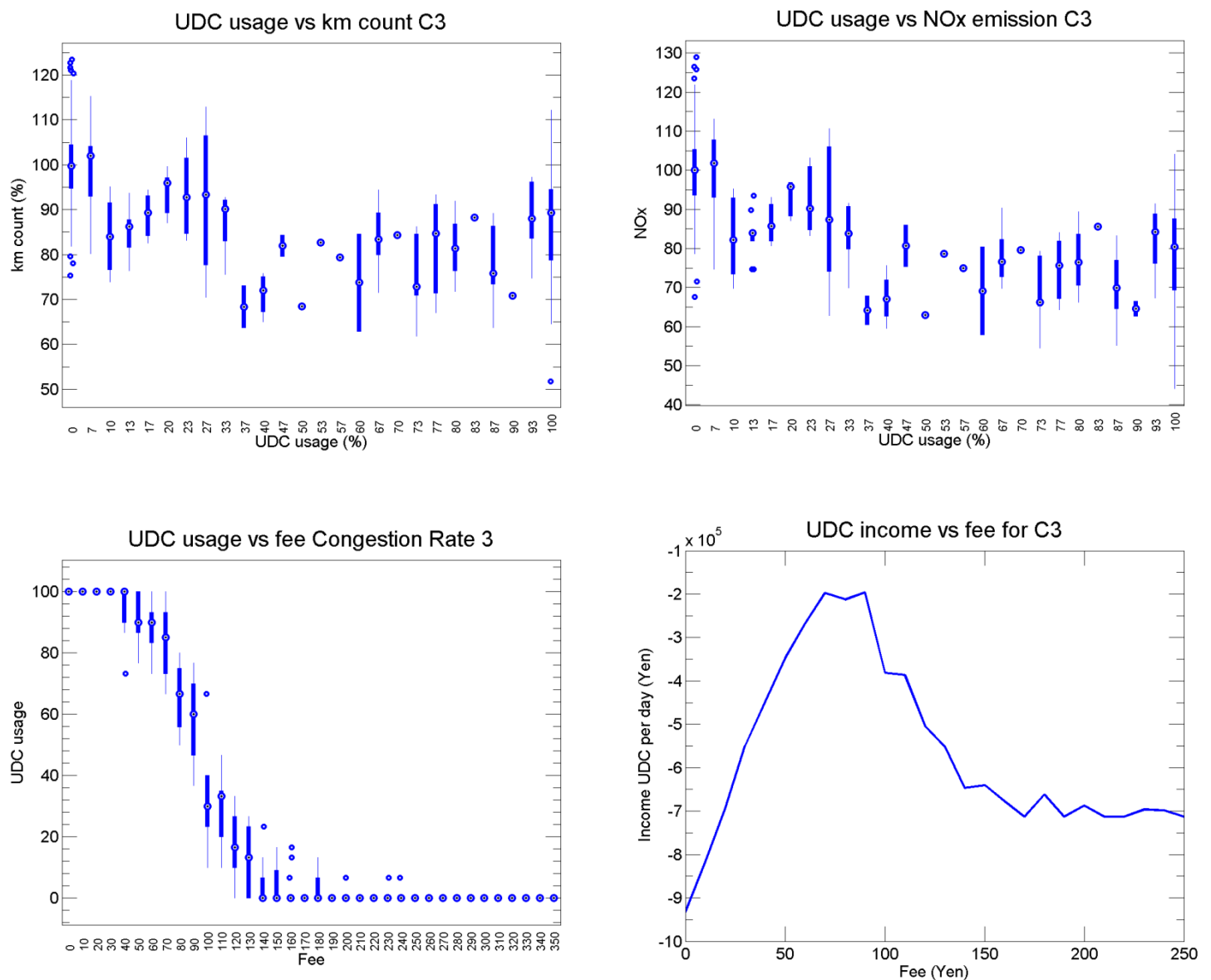


Figure 50: UDC usage vs km count (top left); UDC usage vs. NOx emission (top right); UDC usage vs. fee (bottom left); Impact on UDC income (bottom right). All data is for the high congestion rate.

A.3 Various toll rates

Here the data per toll rate scenario is presented.

A.3.1 No toll

This congestion rate has the lowest average speed in the city. The various parameters are presented here.

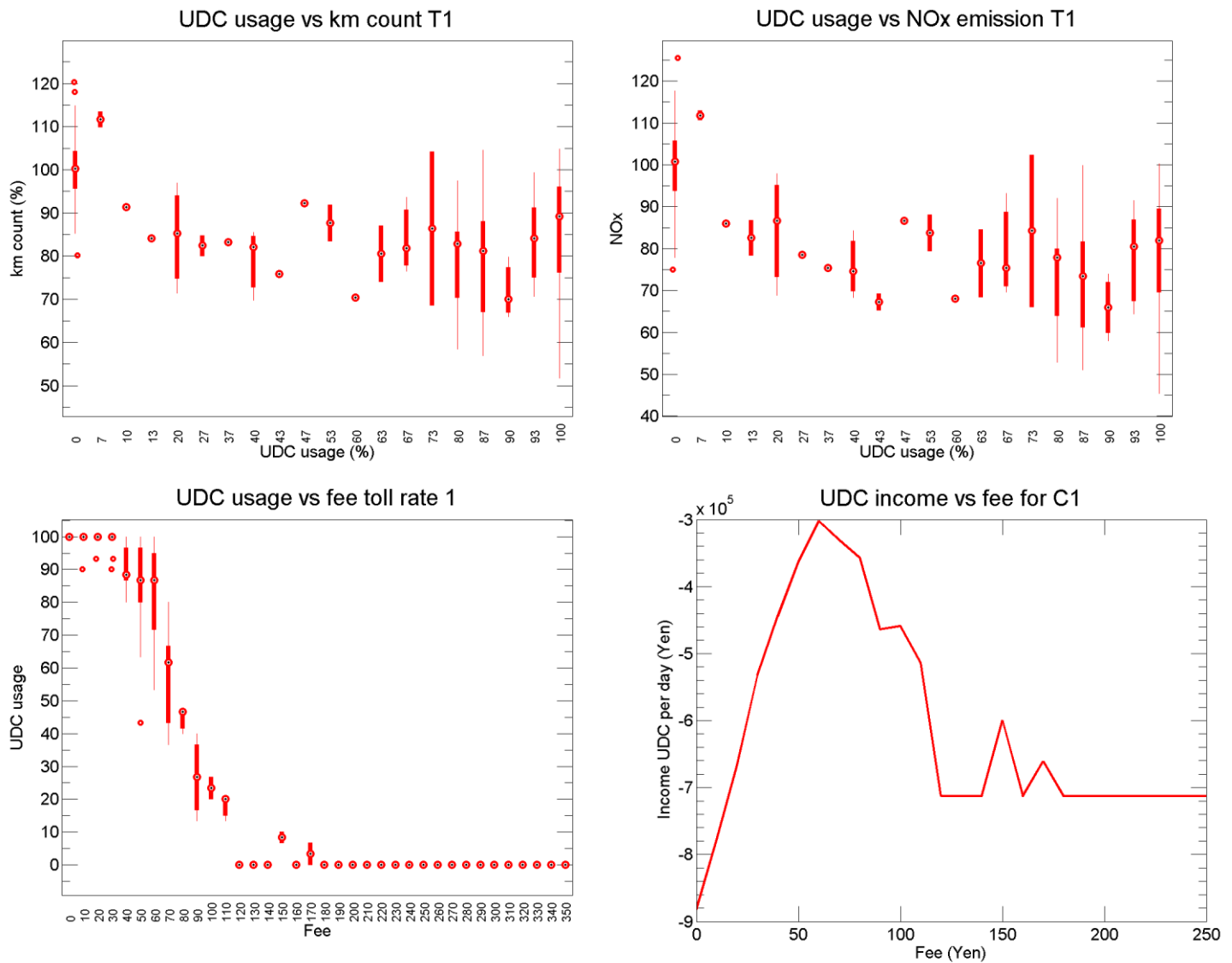


Figure 51: UDC usage vs km count (top left); UDC usage vs. NOx emission (top right); UDC usage vs. fee (bottom left); Impact on UDC income (bottom right). All data for scenario without toll.

A.3.2 Low toll rate

This toll rate request 400 Yen upon entering the city centre. Below the various parameters are displayed.

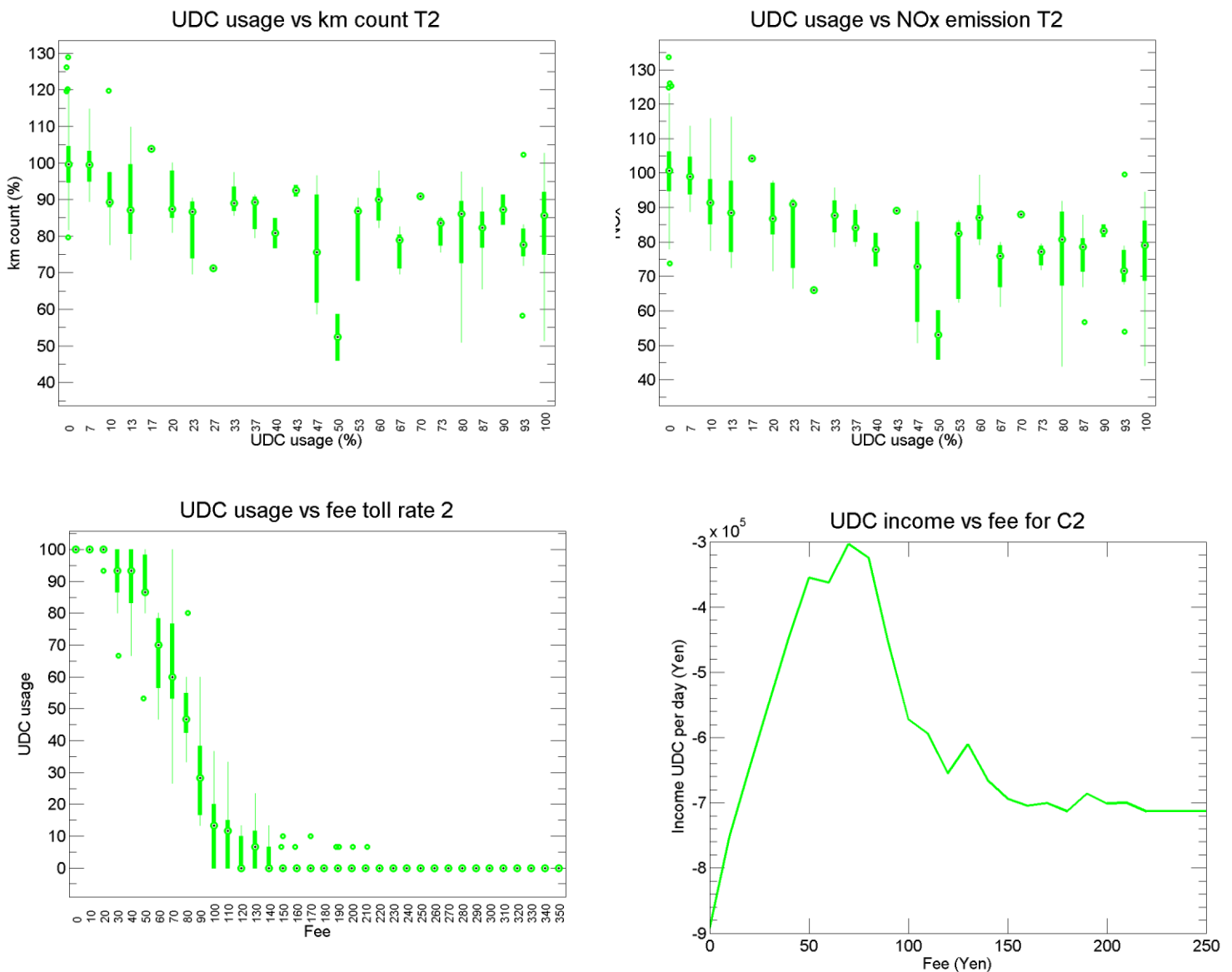


Figure 52: UDC usage vs km count (top left); UDC usage vs. NOx emission (top right); UDC usage vs. fee (bottom left); impact on UDC income (bottom right). All data is for low toll rate.

A.3.3 High toll rate

This toll rate request 800 Yen upon entering the city centre. Below the various parameters are displayed.

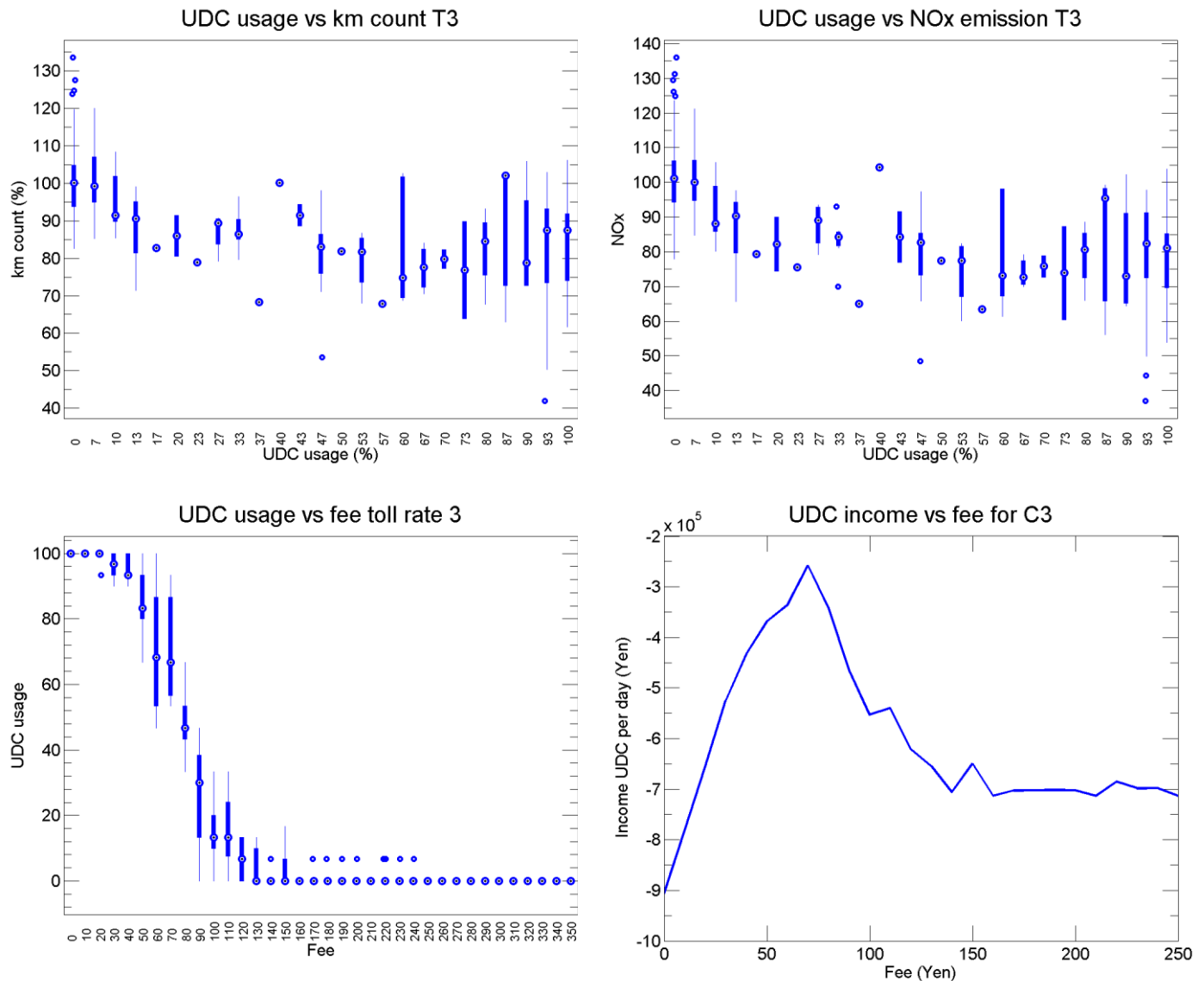


Figure 53: UDC usage vs km count (top left); UDC usage vs. NOx emission (top right); UDC usage vs. fee (bottom left); Impact on UDC income (bottom right). All data is for high toll rate.

A.4 Subsidy for the UDC

The impact of subsidy on the UDC business case is here presented per scenario. The NOx emission and km count hold similar relations as displayed in the scenarios with normal traffic conditions, no toll and the fixed time delivery scheme.

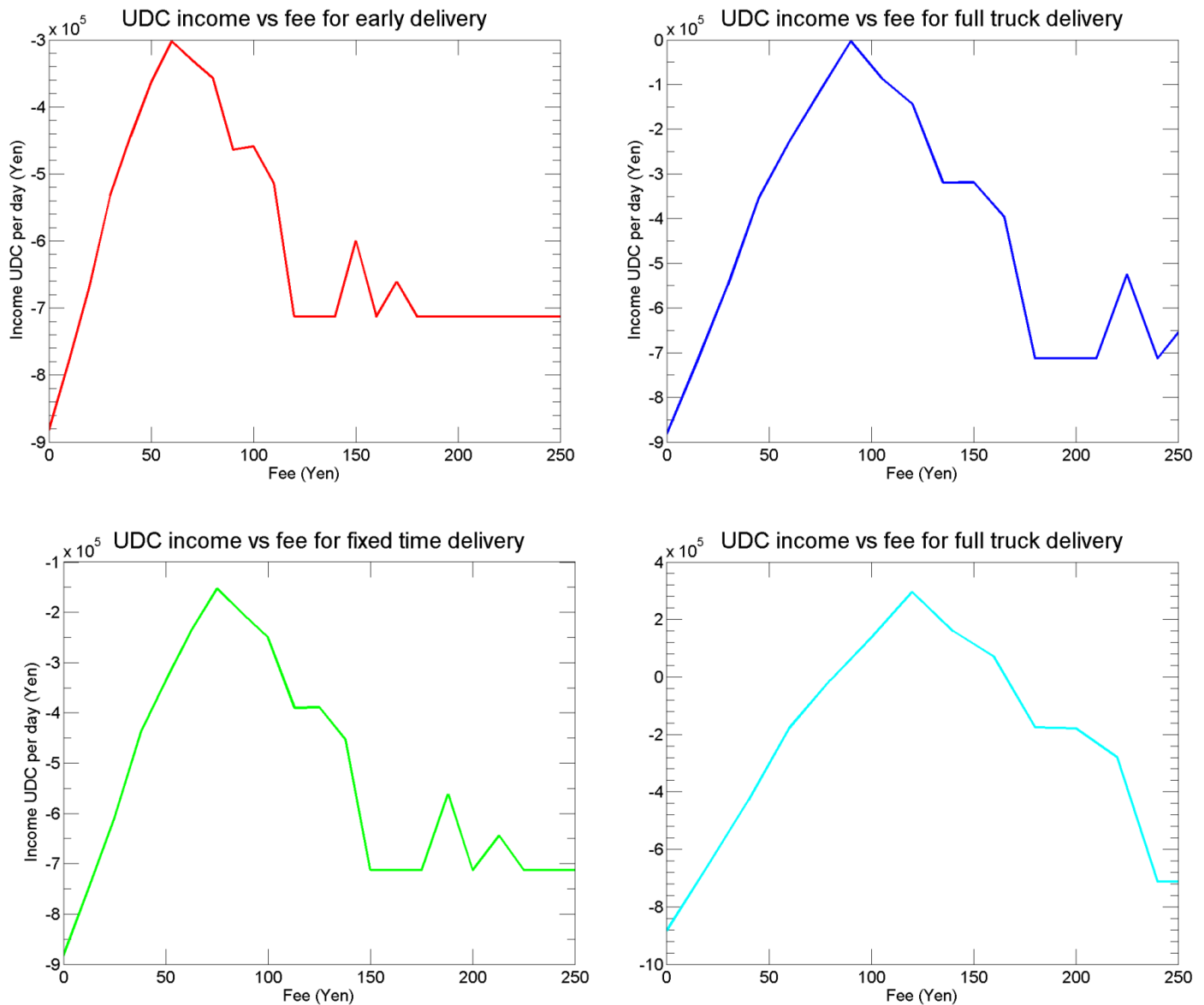


Figure 54: UDC income vs. fee at no subsidy (red, top left); 20% of fee subsidy (green, bottom left); 33% of fee subsidy (blue, top right); 50% of fee subsidy (cyan, bottom right)

A.5 Scenarios

Here the data is presented as used in Chapter 7 for the various scenarios.

Table 25: Mean and standard deviation of the UDC usage percentage per scenario

Fee setting	Mean of UDC usage (%)	Standard deviation of UDC usage (%)	Mean of UDC Income (Yen)	Standard deviation of Income (Yen)
Scenario 1: Reference case				
-20	90	11	-368.000	43.700
-10	83	13	-284.000	67.400
0	79	12	-239.000	86.100
10	58	14	-258.000	102.000
20	46	7.6	-342.000	144.000
Scenario 2: Congested city				
-20	93	11	-401.000	60.600
-10	91	9.5	-267.000	61.600
0	88	5.3	-189.000	54.300
10	73	12	-184.000	60.000
20	63	13	-225.000	101.000
Scenario 3: Involved Municipality				
-20	93	11	-237.000	80.900
-10	91	9.5	-61.500	83.100
0	88	5.3	42.200	70.000
10	73	12	34.800	86.000
20	63	13	-20.800	142.000

The UDC usage rate is the same for scenario 2 and 3, but the received fee increases as both the freight carrier and the municipality finance part of it. In the model the fee presented to the freight carrier is used as the standard and the subsidy is added afterwards. The UDC income is highest for the dynamic fee rates of 0 and 10. Both these cases seem plausible to use for comparison and their differences per scenario are within one standard deviation. Differences for the various fee rates for scenario 1 and 2 are not significant, but the difference of scenario 3 with both scenario 1 and 2 is more than one standard deviation.

Table 26: Mean and standard deviation of the NOx emission and km count compared to reference per scenario

Fee setting	Mean NOx emission (%)	Standard deviation NOx emission (%)	Mean km count (%)	Standard deviation km count (%)
Scenario 1: Reference case				
-20	71	15	72	16
-10	85	3	85	3
0	78	13	78	13
10	81	15	81	13
20	82	12	82	11
Scenario 2: Congested city				
-20	82	13	82	12
-10	71	15	72	16
0	84	13	83	12
10	81	15	82	14
20	86	12	83	11
Scenario 3: Involved Municipality				
-20	82	13	82	12
-10	71	15	72	16
0	84	13	83	12
10	81	15	82	14
20	86	12	83	11

Scenario 2 and 3 hold the same NOx emission and km count decrease as only the subsidy is varied. The difference for NOx emission and km count decrease between scenario 1 and scenario 2/3 is not more than one standard deviation. Also variation within one scenario is not relevant; this is caused by the fact that the UDC usage percentages have a relatively high standard deviation.

Table 27: Impact on FC terminal costs for the various scenarios.

Fee setting	Mean FC costs (Yen)	Standard deviation FC costs (Yen)
Scenario 1: Reference case		
-20	611.000	35.000
-10	557.000	162.000
0	720.000	105.000
10	685.000	101.000
20	606.000	142.000
Scenario 2: Congested city		
-20	591.000	56.000
-10	728.000	67.000
0	799.000	46.000
10	798.000	67.000
20	768.000	105.000
Scenario 3: Involved Municipality		
-20	591.000	56.000
-10	728.000	67.000
0	799.000	46.000
10	798.000	67.000
20	768.000	105.000

The freight carriers also experience the same situation in scenario 2 and 3. Except for the dynamic fee setting of -20; the costs increase relative to scenario 1 for the various dynamic fee settings. Only the difference for the 10 and 20 dynamic fee setting are relevant. The freight carriers prefer the lower dynamic fee as this decreases their costs most. Stunning here is that in scenario 2 the only significant difference is in the -20 dynamic fee setting.

Table 28: Impact on municipality for the various scenarios.

Fee setting	Mean FC costs (Yen)	Standard deviation FC costs (Yen)
Scenario 1: Reference case		
-20	220.000	18.000
-10	183.000	36.000
0	206.000	18.000
10	195.000	16.000
20	202.000	19.000
Scenario 2: Congested city		
-20	207.000	18.000
-10	206.000	13.000
0	203.000	8.000
10	198.000	23.000
20	207.000	18.000
Scenario 3: Involved Municipality		
-20	425.000	37.000
-10	486.000	46.000
0	510.000	19.000
10	491.000	55.000
20	483.000	62.000

Now the scenario were the UDC income is the highest are compared from all perspectives. However when the subsidy costs at the 33% level is added to the costs in the third scenario, there is a significant difference. The costs for the municipality approximately increase a factor 1.5. However since there is no significant decrease in NO_x emission or km count between the second and third scenario, the municipality will need other incentives like social or corporate well-being associated with the UDC, before the municipality will participate in providing such high subsidies.

B Multi-agent model

In MAS the following steps are identified to generate a valid model.

Table 29: Step to generate a multi-agent model (Ref: Wiki SPM 955x)

Step	Explanation
Problem formulation and Actor identification	Here the problem under investigation is formulated in accordance with the research goal. Important actors are identified.
System identification and decomposition	In this step a time frame, various agents, interactions, behaviors, objects, environment, flows, concepts and states are identified and structured.
Concept formalization	In this step all the relevant information from the previous steps is organized and put together in one model by linking the agent together with flows and interactions.
Behavior identification	In behavior identification the input, output and states are investigated per actor to make logic decisions for each individual. Main modelling assumptions are presented.
Model narrative	In model narrative the complete model is written as a story to identify the order of agent reaction and knowledge input.
Model formalization	In model formalization the model narrative is converted into the actual programming language.
Model verification	Here the model is tested to check if agents display the desired behavior under various simplistic cases. Main modelling assumptions are presented.
Experiment setup	In the experimental set-up the desired research question is posed to be answered.
Data analysis	The results are analyzed and need to be presented clearly.
Model validation	Internal validation is required and external validation is possible with either real world data or other models.
Model use	The model can now be used to answer further research questions.

Problem formulation and Actor identification are well described in the Chapter 5, just as System identification and Decomposition and finally the concept formalization.

B.1 Problem formulation and Actor identification

In recent decades cities are getting larger and becoming denser, which generates tremendous demand on business deliveries and passenger transport. As a result traffic jams increase in number and severity. The increased importance of environmental and social impact of such issues adds additional dimensions to the city's problem. Therefore efforts are made to improve the traffic flow inside the city. Passenger transport is investigated relatively well inside the city (Bestuufs website), but a gap in knowledge was present on the freight transport inside the city. Freight transport contributes an estimated 14% of the traffic flow (in Japan, Ministry of Land Infrastructure and Transport Japan), but an estimated 40% in pollution and noise (COST321, 1998). A proposed solution for this has been the urban distribution centre, where freight carriers decrease costs by outsourcing the last mile delivery as UDC combine shipments and increase efficiency of truck traffic inside the city.

The UDC would increase the load factor and allow for easier time-windowed operation (avoid traffic jams) (Quak, de Koster, 2009). A higher load factor in the city can decrease harmful effects associated with city logistics. In practice the concept has failed due to stakeholder behavior as freight transport itself is under high pressure with Just-In-Time delivery system (Germain, Droge, 1996) and inadequate price setting. The unsuccessful implementation of any urban distribution centre makes self-sufficient operation impossible (van Duin and Quak, 2008). In theoretical business case analysis the urban distribution centre appeared to be successful in many cases (Marcucci and Danielis, 2008). This difference between theory and practice could be caused by the fact that static representations use average values for calculation and thereby do not take statistical effects and individual choices into account.

B.2 Model assumptions and choices

Here various additional assumptions and choices are described for the model. This is followed by the pseudo code.

B.2.1 Modelling assumptions

Below the assumptions for the model are described more indepth.

Modelling an artificial City

First of all is the choice to model a real city or to make an artificial city. The goal is off course to mimic a real city to compare the model with reality. This however requires a lot of data on the city to model travel times accurately. Currently commodity flow still remains unknown for all cities. To reduce complexity and to deduce effects of policies on parameters a simple city will be mimicked.

Time/space scale

The inner city (inside highway) has a diameter of 3 km to 10 km (estimation from Google maps based on Osaka, Kyoto, Amsterdam, Rotterdam, New York and Paris), so this is approximately the distance the model should cover.

In city logistics normally planning for freight trips are done one day before. And since impact on traffic is on the scale of 15 minutes, the appropriate time scale for the model is 2.5 minutes.

Delivering hours are normally between 6:00 till 19:00 according to (Freight Best Practice, Key performance indicators for the Next-day Parcel Delivery Sector). Vehicles travel on average with 15 km/hour (including traffic lights etc).

Location choice for FC and UDC in the model

In this research a small part of the city will be modeled with a UDC located close to the highway entry, where the FC terminals are expected to be. In real life the choice for the UDC is a highly important one with large impact on the success factor of the UDC. Research indicates that UDCs located near highway entry. So therefore the UDC as well as the FC terminals are located close to the city border. Since choosing the appropriate location is a complete research question in itself, advice was asked from a modelling expert in this field (mr. Yamada) as proper positioning of the UDC is an elaborate task outside the scope of this research. For simplicity the high way itself is not modeled, but the starting point is right after the highway.

One type of truck and one type of good in the model

In this model for complexity reduction only 1 type of trucks and freight carrier will be assumed. The complexity of various types of goods (like fresh products or clothes) is also reduced by assuming 1 type of goods with no temperature demand.

Load factor assumptions

To account for the 30-40% of empty truck trips (Holguín-Veras and Patil, 2008) (Freight Best Practice Benchmarking Guide, 2006), the average load factor of the truck of the freight carrier is adjusted to match this. The UDC can assume a higher load factor based on deliveries by the freight carriers.

NOx emission

NOx emission represents general harmful emission as correlates to driving time.

Average speed assumption

Here the average speed profile over one day is relates to the following profile. More information can be found in Appendix D: Road network Traffic Conditions.

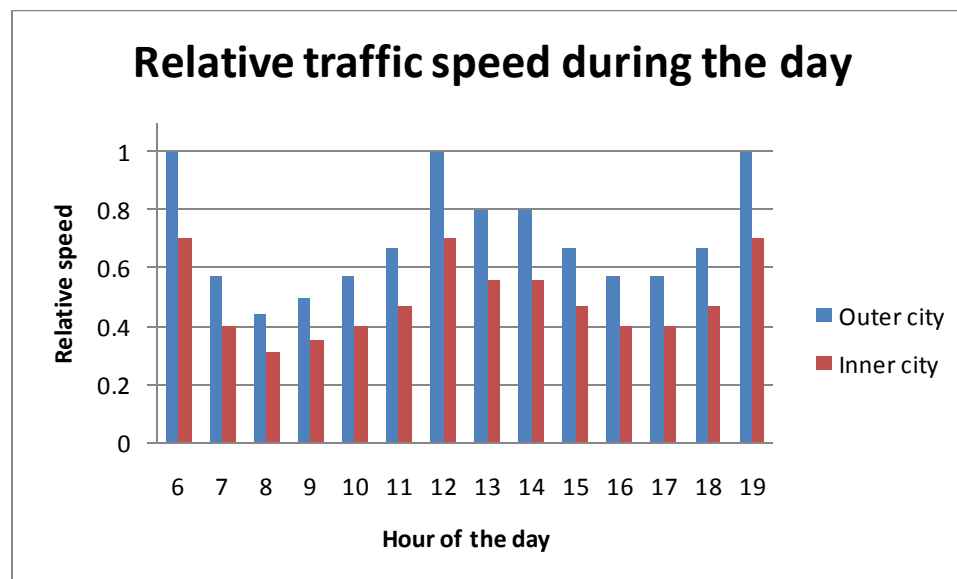


Figure 55: Relative speed during the day for the Netlogo model

Variable and fixed truck costs

From the Tamagawa model the estimation based on real data is that variable costs are 887 Yen per 10 minutes and fixed costs at 11799 Yen per vehicle. Variable costs include labor costs, fuel costs, depreciation costs for the vehicles etc. The mentioned numbers for fixed costs include (fixed) staff costs and vehicle costs. The staff costs include fixed salaries, labor insurances costs, welfare expanses etc. The vehicle costs include tax and insurance for vehicle.

These costs estimates are from interviews conducted before 1996 approximately. Since fixed costs for the UDC are based on analysis made in 2008, at least account for inflation need to be taken to match these prices with the Dutch UDC prices. The new prices for variable and fixed costs are therefore: 1175 Yen per 10 minutes variable costs per truck and 15624 Yen per truck per day for fixed costs. The inflation analysis can be found in Appendix C: Freight carrier prices recalculation with inflation.

Overload penalty on trucks

For the load penalty the value is assumed to be 200.000 Yen per customer, which is an artificial high value to suppress overloading a truck as this does not happen often in reality.

Delay penalty for freight carriers

In the model there is also an artificial high penalty for arriving late at the customer side as often in reality there are no direct penalties paid for late arrival. Here it is assumed that the goods can be delivered the whole day until 19:00.

Decision process of a freight carrier to use the UDC

In this model the Freight carrier uses a fairly conservative cost based choice mechanism to make the decision whether or not to use the service of the UDC called the fast decision maker freight carrier. This freight carrier will not consider going to the UDC unless the costs for delivering a truck load of goods himself is more expensive then going to the UDC. When the UDC is cheaper in total, he will just drop all the goods there.

Company type

It is assumed that the individual trucks represent small companies whole offered to take some freight to a certain area of the city. Large freight companies act as an UDC themselves as they normally have terminals to facilitate combining goods efficiently. The smaller companies however compete fiercely on price and do not have the facilities to have such efficient deliveries. These companies normally accept several freight options in a certain direction of the city, but normally drive with a relatively low load factor to various sides of the city. Therefore the optimization from the freight carrier terminal in the model produces small trucks who represent this segment of the highly competitive freight market, where using a UDC can add social and economic advantage to outweigh the additional time penalty.

Truck driving

The trucks only drive horizontally and vertically in the model city in a fixed retailer order with variable route. The truck can choose, when at a crossing between the two directions (left or right) bringing them closer to their retailer based on speed on these roads. There are the following options:

1. Left/right random: Trucks randomly choose to go right or left on a crossings.
2. Left/right speed: Trucks look into the two roads and choose the path with the highest speed (the lowest congestion).

3. Preprogrammed route: The original must used alternative is to use a preprogrammed route between customers. However to increase the level of dynamics in the model, this option will not be executed.

The choice is made for option 2 that trucks choose the path with the least congestion after the crossing.

Parcel estimation

In order to gain somehow realistic price settings for the UDC, the price per parcel is estimated to compare with literature. It will be assumed that trucks are used with a volume load capacity of 3m by 2 m by 2m (guess based on various truck companies and online secondhand truck offers). This leads to a cubic load of 12 m³. Knowing that the average number of stops per trip is 57 (Freight Best practice, KPI for the Next-day Parcel delivery sector) leads to two calculations on the number of parcels per stop. This calculation is performed for the Freight Carrier truck as their behavior relates more to the data than for the UDC truck (which is aimed at a higher load factor).

Volume based estimation

Assuming that each parcel has a volume of 4 liters (guess) and knowing that on average 25% of the truck cubic fill is delivered (Freight Best practice, KPI for the Next-day Parcel delivery sector). This leads to

$$Nr\ of\ delivered\ parcels = \frac{Truck\ volume * cubic\ fill\ factor}{average\ parcel\ size}$$

and putting in the mentioned values, gives:

$$Nr\ of\ delivered\ parcels = \frac{12\ m^3 * 1000\ litres / m^3 * 0.25}{4\ litres} = 750\ parcels\ per\ truck$$

Stops based estimation

Assuming that per stop 10 parcels are delivered on average (guess) and knowing that the average number of stops per trip is 57 (Freight Best practice, KPI for the Next-day Parcel delivery sector). This leads to a total of 10 * 57 = 570 parcels delivered per trip.

Therefore is it assumed that per trip 600 parcels are delivered. Since in the Netlogo model the number of stops is 6 and not 57, it is assumed that 100 parcels are delivered per customer.

Different models for delivery by the UDC

The UDC can have various delivery schemes. The first one is when all the freight carriers are forced to deliver the goods to the UDC before 9:00 to ensure optimal delivery for the UDC. Hereby the assumption is made that the freight carriers can deliver this early to the UDC.

In the second case the freight carriers can deliver their goods to the UDC in a time range of 4 hours centered around their current departure time from the terminal. The UDC receives goods during the whole day and sends out truck when the demand is higher than 14 retailers or at fixed times (9:00, 1300 and 14:15).

In the last case the delivery from the FC terminals is the same, but the UDC sends out trucks when there is a full truck and late in the day (14:15) to ensure all package are delivered.

Fixed and variable costs with inflation calculation

However Japan has had serious deflation problems for several years now, which does not represent the situation in most countries. Therefore the Japanese prices will be recalculated with Dutch inflation rates over the last 14 years to improve compatibility with the business case information presented in Chapter 4. The Dutch inflation rate per year is also presented in the Appendix C: Freight Carrier prices recalculation with inflation. They are calculated for a 2008 value as the results on the cost benefit analysis in (van Duin, 2008) were published in 2008. The new prices for variable and fixed costs are therefore: 1175 Yen per 10 minutes variable costs per truck and 15624 Yen per truck per day for fixed costs. When transferring the variable costs to the Netlogo model with a minimum time scale of 2.5 minutes the costs become $(1175 / 4)$ Yen per 2.5 minute.

Decision options for the freight carrier to use the UDC

In this model the Freight carrier uses a fairly conservative cost based choice mechanism to make the decision whether or not to use the service of the UDC. There are several options to decide whether or not the use a UDC:

Full calculation.

In the first option the freight carrier will calculate for all his freight all the options to bring it to the UDC or not and the impact of this on his routing and costs. This is however a very complex decision model with 2 to the power n (number of destinations) calculations.

UDC averse freight carrier.

This freight carriers has a high barrier to consider going to the UDC, which is only when the costs of the UDC are lower than delivering everything himself. When the freight carrier then considers to go to the UDC he decides per customer if he will or not.

If (ave costs > *UDC path costs* + *fees*) then
 (if ($x_1 + x_2 > \text{UDC path} + \text{UDC fee} + x_3$) then
 (Bring customer good to the UDC))

For other customers this adds rerouting to the UDC first and then taking out one customer one the route.

Fast decision maker freight carrier.

This freight carrier will not consider going to the UDC unless the costs for delivering a truck load of goods himself is more expensive then going to the UDC. When the UDC is cheaper in total, he will just drop all the goods there.

If (ave costs > *UDC path costs* + *fees*) then
 (Bring customer good to the UDC)

Individual customer choice freight carrier.

This freight carrier will check per individual customer whether it is worthwhile to bring the goods to the UDC, starting from the first customer on its route. For the first customer this leads to:

if ($x_1 + x_2 > \text{UDC path} + \text{UDC fee} + x_3$) then
 (Bring customer good to the UDC)

For other customers this adds rerouting to the UDC first and then taking out one customer one the route.

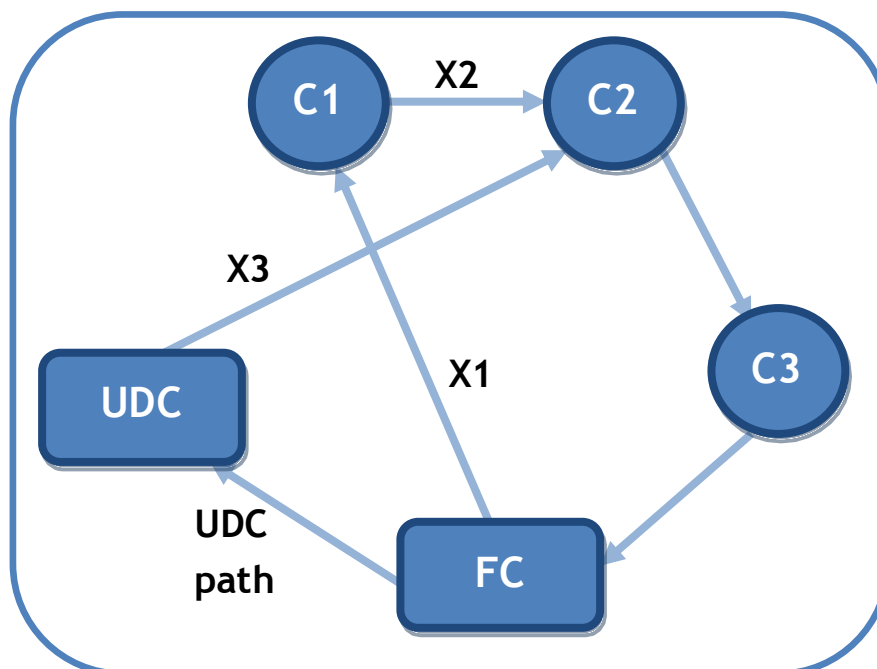


Figure 56: Schematic representation to assist choices for the UDC in the cost calculation

It is highly interesting to see how different choice models for freight carriers impact the amount of goods which are sent to the UDC. Therefore it is recommended that this is investigated further in continuing research with the model.

However for the current model the choice is made for fast decision maker freight carrier. It is assumed that the individual trucks represent small companies whole offered to take some freight to a certain area of the city. Large freight companies act as an UDC themselves as they normally have terminals to facility combining goods efficiently. The smaller companies however compete fiercely on price and do not have the facilities to have such efficient deliveries. These companies normally accept several freight options in a certain direction of the city, but normally drive with a relatively low load factor to various sides of the city. Therefore the optimization from the freight carrier terminal in the model produces small truck who represent this segment of the highly competitive freight market, where using a UDC can add social and economic advantage to outweigh the additional time penalty.

B.2.2 Pseudo code for the total model

First of all the set-up of the model is described, followed by the general go-procedure for every time step. The bold indicated words represent submodels, which are described below.

Table 30: Global set-up of the model

Generate the global variables and breeds with assigned variables in Netlogo

To do the set-up

Set day 0, set time 0, set number of phases, set number of steps in a day

Check for number of steps in desired value, otherwise write error message

Execute **Generate-Network**

Set truck shape to truck

Ask FC and UDC to set all initial parameters like toll counter, GA parameters (chromosome, newchromosome, totalfitness, ctovld, costs, numchromo, varcosts, dist)

end

Table 31: Generate-Network subroutine of the set-up

To Generate-Network

```

let grid-x 3    let grid-y 3    ;; the amount of patch distance between two roads
ask patches [set pcolor brown + 3]
ask patches with-max or with-min [pxcor] [set pcolor black]
;; these are the edges of the world

set roads patches with      ;; make the roads for trucks to travel on
[(pcolor = brown + 3) and ((floor((pxcor + max-pxcor - floor(grid-x - 1)) mod grid-x) = 0) or
(floor((pycor + max-pycor + 1) mod grid-y) = 0))]
ask roads [set pcolor blue + 3 set NOx []
  set toll [] set maxspeed [] set realspeed []
  ifelse (outside city centre with diameter 15 patches)
  [set NOx and toll 0 for all phases
    set maxspeed and realspeed to appropriate values
    set maxspeed (maxspeed / congestionrate from municipality)
    set realspeed (realspeed / congestionrate from municipality) ]
  [set NOx and toll 0 for all phases
    set maxspeed and realspeed to appropriate values with
    congestionrate] ]

set crossings patches with    ;; make crossings, here the trucks change direction
[((pcolor = blue + 3) and (x & y is mod 3))]
ask crossings [set pcolor red + 3
label the crossings from left to right, top to bottom and generate a node on each crossing.
Hide the nodes ]
ask the nodes      ;; for streets between the nodes
[create-streets-with other nodes in-radius 3.5]
ask streets [hide set linktime and linkcost 0 for all time phases]

ask three specific patches to generate a fcterminal and one to make a UDC, ask one black patch to
generate a municipality with a different color and the appropriate shape each
ask fcterminals [ifelse UDC [set loadfactor 8] [set load factor 4]]
ask fcterminals [set GA parameters: set numchromo 220 set generation 200 set elites 15 set
crossoverrate 55 set mutationrate 6 and set kmcounter 0]
Now input the GA parameters for the UDC with the demandtable. Here information on the needed
parameters per demandsize are presented. For the fcterminals there is only one set of parameters,

```

since in this model they have fixed number of retailer orders.

;; Note: since many characteristics are the same for the fcterminal and the udc terminal, both are named fcterminal with a parameter UDC (which is boolean) to separate the two and put different parameters.

set shops patches with [pcolor = brown + 3] ;; remaining brown patches are shops
ask shops [set realspeed 0 set good_A 0] ;; in order to see trucks who are driving on the shops, we stop them there and the demand is put to 0 now.

end

Now the routine for the go-procedure is described.

Table 32: The Go routine

TO GO routine per tick

If (time is 0) ;; many initial steps of the model

 [**Update-the-streets**

Perform-the-cheap-path-calculation

 Update the plots

 Ask fcterminals

 [**Generate-demand**

 Set demandsize

Genetic-algorithm]

 Set day(day + 1)]

Calculate the current phase with (floor (time / (daytime / phasenr)))

If (conditions for UDC delivery is met (on time or truck load))

 [ask UDC [ifelse (demandsize of UDC > mod 4)

 [do **UDC-GA]**

 [take out random truck load to have a higher load factor (if not the last delivery)

 do **UDC-GA**]]

Ask fcterminals and UDC [if (there are trucks to be send) [**generate-trucks**]]

Ask trucks

 [set kmcount (kmcount + speed)

 Move truck forward with speed

 set speed of truck equal to the speed of current road segment in this time phase

```

]
NOx/speed-calculation
Direction-truck

If (it is the end of the day)
    [ask municipalities [regulations]
    Write-outputfiles ]

if ticks > 5000 [stop] ;; stop criteria for total model
tick
set time (ticks modulus daytime) ;; set the time of the day
end TO GO routine

```

Table 33: Update the street subroutine

```

TO Update-the-streets (with calculation subroutines)
ask roads [reset NOx]
ask streets [set distance-in-between about 1 km and reset linktime and costs + reset count
variables]
Now calculate backwards and therefore set phase (phasenr - 1)
while [phase >= 0] [set i -1 set j 0
    while [i < nr of horizontal crossing patches]
        [set i (i + 1)
            while [(j modulus nr of vertical crossing patches) < (nr of vertical crossing patches)]
                [ask node i [xside yside ;(calculate horizontal links and vertical link speed)
                    Calculate xtime0 and xtime1 and xtime2 and xtime3
                    Calculate ytime1 and ytime2 and ytime3
                    ask street i (i + 1) [update linktime = xtime0 / 2 + xtime1 + xtime2 + xtime3 / 2]
                    ask street i (i + nr of vertical crossing patches)
                        [update linktime = xtime0 / 2 + xtime1 + xtime2 + xtime3 / 2]
                    Calculate xsidecost ysidecost
                    ask street i (i + 1) [update linktime = xtime0 / 2 + xtime1 + xtime2 + xtime3 / 2]
                    ask street i (i + nr of vertical crossing patches)
                        [update linktime = xtime0 / 2 + xtime1 + xtime2 + xtime3 / 2] ]
                set i (i + 1) ]
            set j (j + 1) ]

```

Repeat procedure again for the most right column (different streets naming)
 Repeat procedure again for the last row (different streets naming)
 Repeat procedure again for the last row right column position (different streets naming)
 Continue to next phase]

end Update-the-streets

For all xtime0 etc calculations: [set xtime0 (1 / item phase realspeed)] etc for all patches below the streets. Do this calculation with time and with costs (in case of km toll collection).

Table 34: Dijkstra algorithm

To Perform-the-cheap-path-calculation (aka Dijkstra algorithm)

set j count nodes

ask all nodes [start to make table netw ;; this is preparation for dijkstra

 check nr nodes [put empty table location for each node

 for all phases [put at table position (list 3000000 0 3)]]

Put settings to zero for own node in all phases and status to 1 (searched).

ask all nodes [set k who set starttime 0

 for all phases [set itera 0 set linkend -5 set searchnode who

 while [itera < nr of nodes - 1]

 [ask node searchnode [ask the connecting node of my connecting

 [set linkend nr of the other node and ask node k

 [if (this node is not finished searching (aka status 1))

 [set starttime (phase + traveltime to the node in hours)

 ifelse (starttime < phasenr)

 [set temp1 (arrival time at old link + time to new link)

 set temp (arrival costs + cost to new link)]

 [set starttime one phase less set time to start of last link

 set temp1 (arbitrary high value + arrival costs + cost to new link)]

 if (current price > temp1)

 [replace with new value and set status to searching (2)]]

]]]]

;; find the next node for searching cheap path

Find point in table with lowest price AND status 2. (Status 1 is finished)

Set this point to status 1 as there is not cheaper way to get there.

end

Table 35: Generate demand subroutine

to generate-demand

ifelse (variable or fixed demand)

[set x 10

ask x of the shops to [set good_A -100 and change color

[ask fcterminal [add xcor, ycor, demandgoods to demand list]]]

[set demand [[1 1 100] [-5 7 100] [-7 -3 100] [-25 3 100] [-11 -3 100] [-20 -2 100] [-8 9 100]

[-2 -5 100] [-10 -5 100] [-22 7 100]]; [-14 3 100] [-5 1 100] [-17 9 100]]; [-25 7 100]

[-14 -3 100] [-20 -5 100] [-8 3 100] [-4 1 100] [2 6 100] [-22 4 100]]; [-25 -8 100] [-16 -6 100]

[-11 6 100] [-17 1 100]]]

end

Table 36: Genetic Algorithm subroutine

To Genetic Algorithm

chromosomegeneration

set lengthchr (length chromosome)

set bestroute (list) set bestroutecots to arbitrary high value

set gen 0 (nr of generations)

while [gen < generation]

[**fitnesscalculation****elitechoice****selection****crossover****mutation**

gen = (gen + 1)]

UDCdelivery choice**bestroute-organisation**end

The full description for the genetic algorithm can be found in the Appendix E: VRPTW with Genetic Algorithm.

Table 37: Genetic Algorithm subroutine for the UDC

To Genetic Algorithm for the UDC

```

    set demandsize (length demand)
    if (demandsize = 1) [set demandsize 2] (otherwise computation error as no parameter exist)
    if (demandsize > 1) [set corresponding number of chromomes, generations, elite percentage,
        crossoverrate and mutationrate]
    if (demandsize > 1) [Perform Genetic Algorithm]
    while [(length bestroute > 1)] [generate-truck] ;; when send out everything at once
    reset demand []
end

```

Table 38: Generation of a truck subroutine

```

To generate a truck

%% generating a truck is done from the best route at the right time.
set UDCcheck true if passes to UDC or false when not
remove first item of bestroute
set temp [] set temp1 []
while [not next time/truck item] [
    set temp add x direction bestroute
    set temp1 add y direction bestroute
    set bestroute but-first bestroute]
set temp add pxcor terminal
set temp1 add pycor terminal
ask patch pxcor pycor (aka the terminal)
    [sprout-trucks 1 [set color [pcolor] of patch-here
        set dirx temp      set diry temp1
        set costs 0        set status 1
        set tollt 0        set good_At 0
        set heading 270    set speed 0.7
        if (UDCcheck) [set dirx add at start -20 set diry add at start 3 (-20 3 is location UDC)] ]

```

Table 39: NOx collection subroutine

```

TO NOx calculation
ask roads [set phase NOx (item phase NOx + count (trucks-here))
    set phase realspeed (item phase maxspeed –
        (0.05 * count (trucks-here)) + ((random 19) / 200))
    if (item phase realspeed < 0.05) [set phase realspeed 0.05] ]

```

Table 40: Truck direction determination

To move-truck

ask trucks [

ifelse (status = 1) ;status 1 is a normal road ahead

 [if (retailer is next to road) [**deliver**]

if (road ahead != non crossing road)

[ifelse (patch ahead = black (end of world))

[turn around (no other option)]

[if (not road ahead is any road (crossing or not) [face crossing at straight angle]

if (distancexy retailer < distancexy (next patch) (y coordinate retailer))

[set heading (180 + heading)]]]

if (patch ahead is crossing) [set status 2]]

[if (status = 2) ;status 2 means crossing ahead

[face patch (first dirx) (first diry)

if (goal is far away (more than 1 crossing))

[;;choose optimal direction

set temp left direction set temp1 right direction

ask patch-at-heading-and-distance temp 1 [set xtime0 [item phase realspeed]]

ask patch-at-heading-and-distance temp1 1 [set xtime1 [item phase realspeed]]

ifelse (xtime0 > xtime1)

[set heading temp]

[set heading temp1] (aka choose fastest route)]

[ifelse (distancexy (first dirx) (ycor) > distancexy (xcor) (first diry))

%% targe is close in one direction

[send in x direction]

[send in y direction]]

ifelse ([pcolor] of patch-ahead speed = red + 3) [set status 2] [set status 1]

%% choose correct status for next round]

if ([pcolor] of patch-ahead 1 = black) [set heading (-1 * heading)] (check for edge, because with these choices maybe it is possible to hit the edge)

end

to **deliver**

ifelse (color of delivery patch) != [color] of my truck)

[ifelse ((first dirx = -20) and (first diry = 3)) %% Check for UDC delivery

```

[set dirx butfirst dirx set diry butfirst diry  %% transfer the goods to the UDC
ask FCterminals in-radius 1.6 [                %% find the UDC
  while (length directions > 1)
    [set demand of UDC item 0 dirx of truck and diry of truck
      ask truck [remove first dirx and diry from list] ] ] ] ]
[ask patch dirx diry [set pcolor (brown + 3) again set good_A (good_A + 100) ]
set dirx but-first dirx set diry but-first diry] ]
[if (length dirx < 2)
  [ask FCterminals in-radius 1.6 [add my driving kms and drivingscosts]
  die ]

```

```

%% directions for next delivery
face patch (first dirx) (first diry) set heading at 90 multiple
ifelse ([pcolor] of patch-ahead speed = blue + 3)
  [set status 1]
  [ifelse (edge of world)
    [turn around 180 degrees]
    [if (direction is not a road/crossings)
      [face the closest crossing at straight angle
        if (this crossing is further than other crossing at end of this street)
          [turn around 180 degrees] ] ] ] ]
set status 1
set speed 0 for unloading time
end

```

Table 41: Regulations subroutine

```

To regulations
set temp [0000000000000000]
set NOxemission []
set phase 0
while [phase < total phases] [
  ask roads [add NOx emission to temp per phase]
  set phase (phase + 1)]
set NOxemission temp
end

```

B.3 Model verification

B.3.1 Single-agent verification

With single-agent verification there is a sanity check, breaking the agent, extreme value testing and dynamic signal testing. Below a table is presented to indicate the relevance of these testing methods for the individual agents.

Table 42: Single agent verification for the agents

Agent	Sanity Check	Breaking the agent	Extreme Values	Dynamic signal
Roads	X	X	X	X
Streets/links	X	X	X	
Retailers	X	X	X	X
Trucks	X	X	X	X
FC terminals	X	X	X	X
UDC terminal	X	X	X	X
Municipality	X	X	X	

In testing the roads, streets, links and retailers, first the total initial grid is generated.

Roads

For the sanity check the road received several trucks on it and it was checked, if he counted them correctly and generated the correct NOx count. Secondly it was tested if the speed on the road decreases correctly with the number of trucks on the road, but stayed below the maximum speed parameter and above 0. Lastly it was tested if the road responded correctly to input on the congestion rate and changed its toll setting when requested to.

In order to break the agent the maximum speed of the road was set below 0, thereby creating the problem that the speed cannot be below maximum speed and above 0. This resulted either in almost zero speed when a truck was presented as input.

A second way to break the agent could be by putting the amount of trucks on the road to infinity, but the demand by the retailers is finite.

Extreme values were mimicked by putting 30 trucks on the road, to which the road reacted with a very low travel speed for the trucks. Also the NOx count was very high, but reacted well. The minimum speed of the road was set to almost zero, to eliminate the problem of getting negative speed, when many trucks are present (as speed decreases linearly with the number of trucks). The congestion rate holds a division for the truck, but it bound between 1.1 and 1.5. The toll can be zero to 600 Yen in steps of 150 Yen and this provides no problems.

Dynamic signal analysis showed good responses to adding and removing trucks from the road over time.

Streets and nodes

In the sanity check for the streets, it was checked if the calculation of the travel time and speed with and without toll was correct. Sanity check for an individual node is not possible as its main purpose is to calculate costs and times to other nodes.

While testing the breaking of the agent, it became apparent that when the speed of the road is zero, there are problems with a division through zero. Therefore input is not allowed to be zero and roads have a minimum speed of an arbitrary small value of 0.05 (which corresponds to approximately 0.35 km/hour). The number of roads below a street is fixed to four.

As input for the streets the speed settings are limited between 0.05 and 1, the toll is between 0 and 600 Yen and within these boundaries the streets respond well to all combinations.

Retailers

Sanity check for the retailers consisted of checking the response to express a demand of 100 goods (with reaction a decrease of 100 in goods) and an increase of 100 in goods when receiving goods. In order to verify that the agent received the needed goods at the end of the day, all non-satisfied agents are marked with a color, which is put to zero when the right amount of goods is received.

In order to break the agent, input a value of 0 goods was presented, to which the retailer responded as expected with zero increase or decrease. Secondly a negative order or delivery was presented, which corresponds to the opposed action. Therefore it was concluded that the nr of goods for demand and deliveries always has to be positive.

The parameter space consists only of packages of 100 goods (as it is fixed for this model). The extreme value testing therefore consisted of the case that two packages were delivered of requested, to which the retailer responded by decreasing two times with the value 100 in goods.

In the dynamic signal testing, it was checked how the retailer responded over several days. During the day the retailer could place an order, which he received during the day. This process repeated over days. In order to prevent problems when there was no delivery, an error message was put in place.

Trucks

Single-agent verification for the truck is complicated as the truck has a lot of interaction with other agents. In the sanity check it is verified that the truck adapts its speed when requested. Direction setting and driving is not possible in a single-agent verification method. It is possible to mimic an artificial delivery and the truck is able to deliver the goods and to eliminate the delivering address from its destination list. Normally the last item on the destination list is always the terminal from which the truck came. Upon arrival at this terminal the truck passes km count information to the terminal and dies.

The truck can be broken by eliminating all destinations from the list as then it does not know where to go.

In extreme value testing it was proven possible for the truck to hold 30 destinations (all retailer demand for one day). In an extreme case when the truck speed is zero for longer time period, the truck does not move, but this is in principle no problem. This can happen when the truck is placed upon a retailer (with speed 0).

Dynamic signal processing is highly important for a truck, but impossible without a surrounding. Therefore the testing of the truck will be done more extensively in the interaction tests.

FC terminal

Main part of the sanity check for the FC terminals as well as the UDC is the verification of the GA as the GA needs to return a route after receiving the input about the demand. This is described here and GA Verification UDC and GA Verification UDC.

Sanity check for the FC terminal consisted of checking the correct response to demand, the formation of chromosomes for the GA and the generation of the truck with the correct information. It needs to request and collect information of 10 shops and to form diverse chromosomes with this information. After performing the GA a truck has to be formed with the correct routing information and correct load factor.

The FC terminal can be broken if there is zero demand. Additionally when GA parameters become so large, the agent needs a substantial amount of time to complete its tasks. Physical limits of requiring too large memory space were not reached. Attention was given to the time at which the trucks left the terminal to ensure departure of trucks scheduled for the same time.

In extreme values first off all the limits for the GA parameters were tested. Ranging the crossover, mutation and elite rate from 0 to 100 percent, posed no problems (besides a meaningless solution occasionally). For the GA it is necessary that the number of chromosomes is a multiple of two and above zero. Also the number of generations needs to be above zero as otherwise no fitness calculation with a resulting best route is performed. Maximum for these numbers did not reach problem providing numbers. Demand size was fixed with ten retailers. Also input from the nodes remains bound and positive (no zeros or infinity). Input from the UDC fee is also bound and can possess negative as well as positive values. Trucks with a large number of kilometer and toll payment do not pose a problem as long as they are bound.

Every day the FC terminals receive new demand from the shops and during the day trucks leave and return. A large number of trucks (4) departing at the same time or returning at the same time posed no problems, even though the choice was made to leave truck at least with one tick in between to account for leaving time. Since fee price and network conditions are only required during one time of the day, variation in this is only taken into account on the next day.

UDC

The full check of the GA is performed in the Appendix G: GA verification UDC. The UDC and FC terminal have many similar characteristics. So the previous steps for the UDC were also performed, except the different demand collection. Here only additional tests are mentioned.

Additionally in the sanity check the collection of demand was tested by providing demand input in the same way trucks would provide it.

The UDC can withstand a zero demand (unlike the FC terminal), but computation time increases significantly for demand of larger size (16 to 30 retailers).

For the UDC dynamics are highly important as trucks with demand arrive during the whole day. This will be mostly explored in the interaction testing as problems result from late deliveries for example. For the individual agent this means that upon providing all demand in one tick, larger chromosomes are needed than with more continuous FC truck arrivals as no time is present to send out some trucks in between.

Municipality

The municipality sanity check consisted of checking that the congestion rate and toll output was provided correctly. NOx count checking was performed in the interaction testing.

It is very difficult to break the municipality agent as both congestion rate and toll are fixed parameters. Subsidy providence is performed outside the Netlogo model. NOx counts cannot be infinity was resulted in the extreme values check. Dynamic signals are signals presented to the municipality on various days.

B.3.2 Interaction testing

For this model the generation was done in several steps, performing interaction tests at each stage. Most agents need interaction with agent of another breed. The model is set up with the small number of agents necessary and it is investigated, if the basic agent interactions happen correctly. The different steps were as follows:

1. Generate the retailers and road network with boundaries
2. Add the streets and nodes
3. Generate a truck and let it drive
4. Add 1 FC terminal and generate trucks
5. Verify the GA and find the parameters
6. Add the UDC and find GA settings
7. Add the time parameter to the model

Generate the retailers and road network with boundaries

In the initial step a network with retailers, roads and black boundaries was formed to check the formation of the city. First of visual inspection was performed for the layout by requesting color changes. Secondly demand request to 10 out of the XX retailers were made and checked, by requesting color changes and automatic counting. Sanity checks were performed again on the group of roads and retailers to set the speed and collect NOx by the streets. Also it is checked if crossings are formed on the correct places. Also the municipality was added and it was checked if the parameters of speed and costs changed according to the congestion and toll rate.

No additional tests to break these agents, extreme values or dynamic signals were performed the situation is the same for the individual agent as for this group as there is not so much interaction yet.

Add the streets and nodes

Next the nodes are formed on the crossings. Location and numbering is checked before there are hidden. Between the nodes streets are formed to the right and below for proper organization. In the sanity check it is checked if the streets manage to represent the roads correctly. With extreme values on some roads, it is checked if the streets use this correctly. Dynamic testing is performed for several days, but without extreme cases as the process is the same each day. The interaction of each street with the roads is correct.

Next the nodes are tested. First the Dijkstra algorithm was rechecked from theory and from the Tamagawa model. Next the calculation takes place and the costs/times are displayed on the nodes after requesting the information of one node. Some paths are recalculated by hand to verify correct interaction. Extreme values were inserted in some roads to check the response and calculation. The interaction of nodes with the streets is correctly.

Generate a truck and let it drive

In the elaborate process to check the interaction of the truck with other agent various interaction were distinguished: Delivery to retailer, direction determining and driving on the road. In delivery to the retailer the truck presents goods to the retailer, eliminates the current direction coordination of this shop and looks at the new coordinates. In the sanity check it was checked if this interaction was as planned looking at the list of directions at the truck and seeing the delivery at the retailer side. This was performed with visual color inspection and numeric checking. Next the truck has to find the correct direction on crossings to find the retailer. The truck has a maximum of two directions it can travel without making addition km to reach the customer. This was checked numerically with the angle of direction and travel as well as travel speed information. It is checked that trucks do not drive on other patches but roads, but setting the other speeds to zero, thereby getting the truck stuck (thereby “breaking” the agent). This was checked visually and in multiple runs by counting the trucks left behind.

Currently the truck can respond to the various situations correctly, also complex situations as direction determination and driving at edges of the world. Besides checking the logic model for the reasoning of the truck, the zero speed (and error message) outside roads ensure error detection. This was all checked in dynamic behavior with multiple retailer deliveries as this is the normal situation.

Add 1 FC terminal and generate trucks

Next step is the adding of an FC terminal to provide real input for the trucks. For this interaction a sanity check was performed as extreme values and dynamic are covered with

previous tests. The FC terminal creates the truck with the correct values from the best route results of the GA. After having delivered all the goods, the truck will return to the terminal. When arriving at the original terminal, the truck is finished. It was checked by the terminal and at the end of the day by identifying unsatisfied retailers, if all goods were delivered.

In extreme values the case was tested to produce an empty truck. This was possible, but in the next round it disappeared as it immediately was at the final terminal as well. Even though this case should not take place, an error message was incorporated for the detection.

Verify the GA and find the parameters

In the sanity check with the GA was performed with artificial chromosomes, while calculating the results to see if the GA took the desired actions. Before being able to verify the success of output, optimization of the parameters is required. In order to check the GA parameters dependent performance, the behavior space in terms of number of chromosomes, number of generations, mutation rate, crossover rate and elite rate had to be searched for the optimal solution. Previous methods described sequential search of the optimal parameters, but a simultaneous search is preferred as all parameters impact each other and the solution quality. This search is first performed for the FC terminal with an average load factor of 40%. Searching the behavior space and finding the correct parameters takes several hours per scan and required multi-scans as the values are not the same as expected. For example the crossover rate is substantial lower than expected (90%), most likely caused by the fact that no separate of chromosomes in two groups is used. Also the new problem size and larger solution space might affect these parameters. Below the optimal parameters for the GA for the FC terminals are presented.

Table 43: Overview of parameters settings for the UDC GA

Demand size	Chromosomes	Generations	Elite rate (%)	Crossover rate (%)	Mutation rate (%)
10	220	200	15	55	6

Besides parameter searches, visible inspection with color coding and/or numbering of various fixed demand schemes was performed to check if indeed the GA delivered the correct solution. The GA does not always reach the most optimal solution, so it is recommended for future to carry out further tests on the performance of the GA and the parameters settings. Repeatability is not 100%, when researching the parameter space slightly different optima appear in the least square methods, which appears to have a dependence on parameter size. More information is found in the corresponding appendices.

Now the FC terminal and its trucks are able to deliver goods to the retailers.

Add the UDC and find GA settings

Next the UDC agent is added to the model. In the sanity check it is verified that the truck arriving at the UDC transfers the goods with according destinations correctly. Next it is tested if the FC makes correct choices for using the UDC by one full recalculation of the path costs compared to the UDC costs. Other tests included numeric checks of UDC costs comparing to the costs of the optimal route.

In extreme values it was found out that it is not a problem when the truck arrives with no goods for the UDC and neither the truck or the UDC break. However in practice this case will not occur (as it requires an empty truck to drive to the UDC).

Next the GA parameters have to be tested for the UDC agent. This is an extensive work as for all different demand sizes different size chromosomes are generated and different parameters are required. In order to reduce the work not all demand size were tested and at higher values several behavior spaces searches with some fixed parameters were performed. In the future further research on this is recommended as for some parameters the success factor of finding the correct route is not high. This can found in the Appendix G: GA verification UDC.

Add the time parameter to the model

The last addition to the model was the time parameter with 14 phases during the day. This means that all above mentioned steps for single agent and interaction testing were repeated from as many parameters changed for single value to a 14 value long list. All agents experienced change due to this. It might appear easier to implement the time from the start, but due to the highly complex GA and complex interactions, the choice was made to add time in the last phase and repeat these steps with additional tests. Additional tests in the sanity check ensured the phase never got out of bound (0-13) and if the correct phase was selected.

In the sanity check it was also checked if the trucks departure at logical times.

In the results above for the GA, the time implemented results are mentioned as these represent the final solution and therefore are more relevant.

B.4 Experiment setup

In the experiment design the question of interest is how the UDC fee input delivers different output on UDC finances. This is answered in Chapter 5 like data analysis, model verification and model use.

C Freight carrier prices recalculation with inflation

In this appendix the information is presented as used for the inflation calculation for the truck fixed and variable prices.

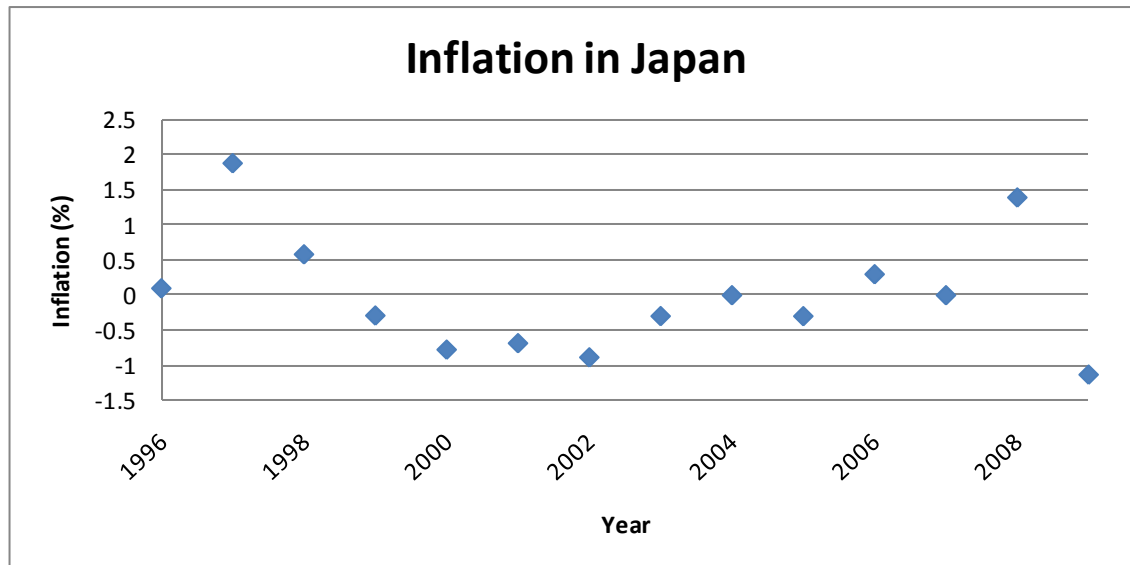


Figure 57: Japanese inflation rates from 1996 (Ref: IMF, 2009 Economic outlook)

Here the Dutch inflation rate per year is noted and the new prices for variable and fixed costs are presented here. They are calculated for a 2008 value as the results on the cost benefit analysis in (XX) were published in 2008.

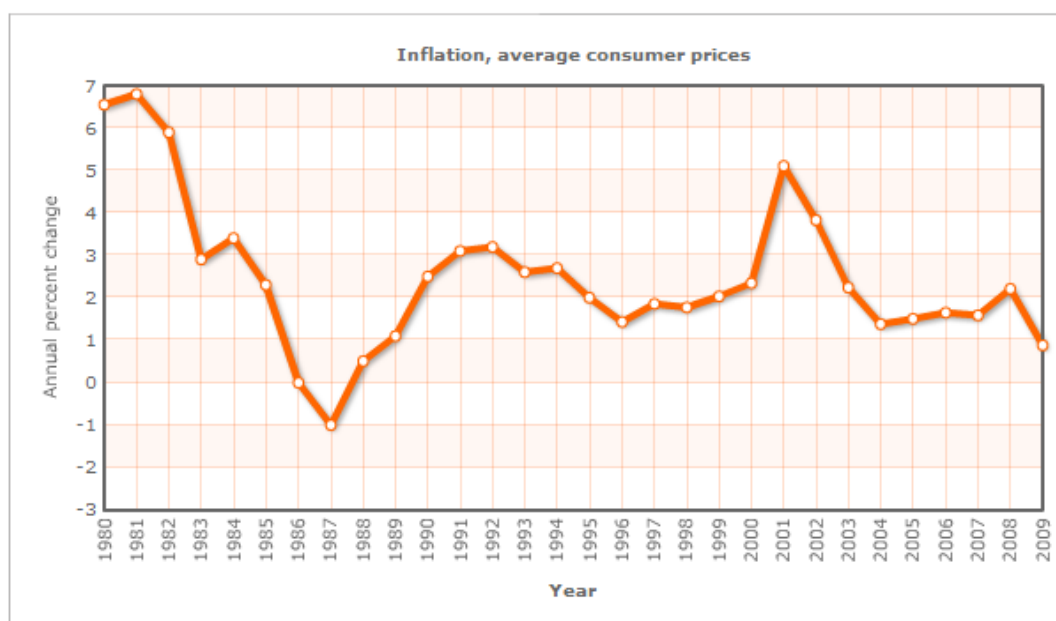


Figure 58: Inflation rate in the Netherlands from 1980 to 2009
(ref: IMF, 2009 World Economic Outlook)

Table 44: Inflation and effect on fixed and variable costs in the model

Year	Inflation (%)	Variable costs (¥/10 minutes)	Fixed costs (¥/vehicle)
Start		887	11799
1996	1.35	899	11958
1997	1.85	916	12180
1998	1.75	932	12393
1999	2	950	12641
2000	2.3	972	12931
2001	5.1	1022	13591
2002	3.8	1061	14107
2003	2.2	1084	14418
2004	1.35	1098	1461
2005	1.45	1114	1482
2006	1.6	1132	15061
2007	1.55	1150	15295
2008	2.15	1175	15624

D Road network traffic conditions

In order to obtain realistic traffic conditions the model by Tamagawa, mr Ando and Chrobok/Kaumaann are consulted. From this information a general traffic flow trend will be deducted and used as input into the Netlogo model basic conditions. The Tamagawa model travel times are based on real data as was indicated in conversation. The used data is presented in the table below.

Table 45: Travel times in minutes for roads segments of 5 km long as used in the Tamagawa model

	0-3 am	3-6 am	6-9 am	9-12 am	0-3 pm	3-6 pm	6-9 pm	9-12 pm
Highway	5	5	5	5	5	5	5	5
Road A	10	10	12	12	10	12	10	10
Road B	10	10	15	15	10	15	10	10
Road C	10	10	20	20	10	20	10	10

The second resource is the article from (Chrobok and Kaumann, 2004), where traffic forecasts are made and traffic flows are measured. Even though the data represents nr of cars vs time, it can be assumed that the nr of cars is related to the travel speed of the cars (though not linearly). Strong peaks are observed around 7:30 and between 16:00 and 18:00.

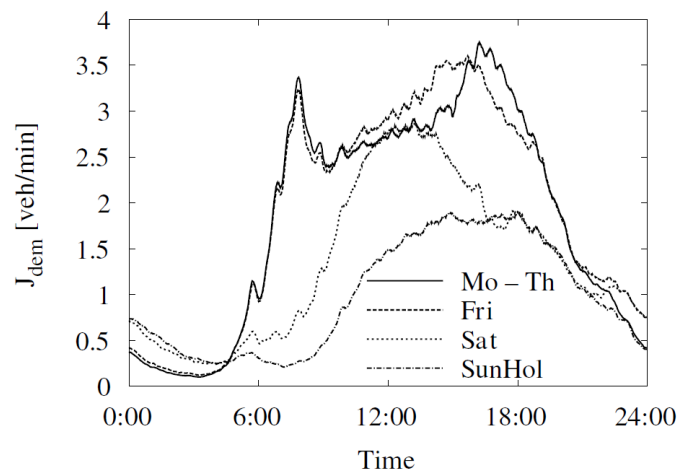


Figure 59: Nr of cars vs time. The data are stemming from all inductive loops and are averaged over days with similar traffic demand. The four classes are distinct. (Ref: Chrobok and Kaumann, 2004)

Similar data is obtained from mr. Ando (collega of prof. dr. Taniguchi at Kyoto University).

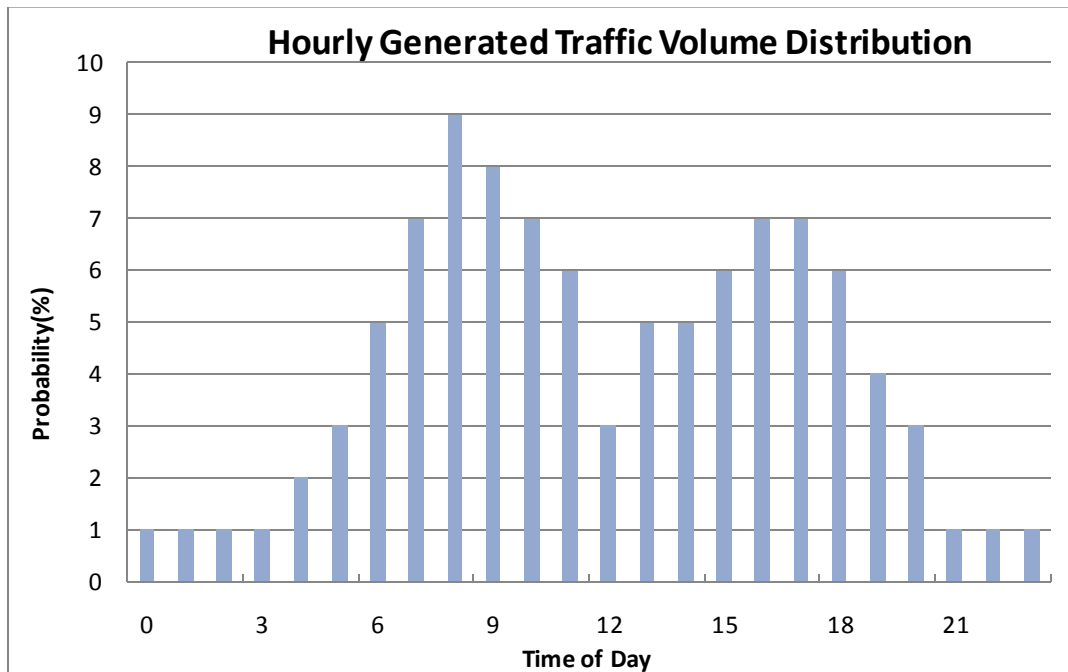


Figure 60: Probability of hourly generated traffic volume distribution by mr. Ando (Kyoto University)

Here peak activity is seen in the morning at 8:30 and again in the evening between 16:00 and 18:00. Combining this information will make the basic speed profile for the Netlogo model. Hereby it is assumed in the Tamagawa model that the worst congestion increases travel time with a factor 2, in the Chrobok/Kaumann model the number of vehicles / min increases with 40% compared to the afternoon on Monday to Thursdays and in the data from mr Ando the hourly generated traffic volume distribution increases a factor 3 at peak morning congestion compared to 12:30. In mr. Tamagawa the intensity of congestion is the same for morning and evening, while in mr Ando data the morning has higher peak congestion and Chrobok/Kaumann model had worse evening congestion conditions. Therefore equal increase is assumed for morning and evening for the Netlogo model with increase of 100% compared to the afternoon hour.

Computation was only carried out for relevant times between 6:00 and 19:00 am. This is based on freight carrier working hours as can be seen in the figure below, where collecting and delivering takes place mostly between 7:00 and 19:00.

E VRPTW with the Genetic algorithm

The freight carriers and the UDC need to determine the optimal route for their trucks. Vehicle routing and scheduling is the optimization process of delivering goods to customers with a set of trucks by finding an optimum route. This is a non-deterministic polynomial-time hard² combinatorial optimization process with variable traffic conditions and various customers locations.

Without algorithms the search in the numerous possibilities of this so-called Travelling Salesmen Problem, the computation time increases exponentially. Even with just solving the problem of the order of the locations and not taking into account the time dependent traffic conditions, the computation time scales with $n!$ and for a higher number of customers exceeding years(!) even is one computation only takes 10^{-8} seconds (Taniguchi and Thompson, 1999).

Therefore it is necessary to use heuristics methods to find approximate solutions. Several algorithms exists for this class of problems: Genetic Algorithm (GA), Simulated Annealing (SA) and Tabu Search (TS) are the main categories used in City Logistics.

E.1 Algorithms for the NP-hard travelling sales man problem

The three mentioned algorithms: Genetic Algorithm (GA), Simulated Annealing (SA) and Tabu Search (TS), will be explained here briefly (Reeves, 1993).

Genetic Algorithm

The GA computation techniques use biological evolution as the theory behind the search for the optimal solution. Various solutions experience mutation and crossover with a probability rate like in DNA reproduction. These solutions are called individuals and each individual represents one possible solution. Every individual has a fitness, which is based upon desired properties in an objection function. This whole set of individuals is called the population. By employing mutation and crossover with a probability the individuals change. By selecting a percentage of population with the highest fitness and randomly adding individuals/chromosomes to fill the next generation population, evolution takes place.

The critical factors of the GA are therefore the mutation rate, the crossover rate, the population size, the number of generation and the elite percentage. They each influence

² Type of problems in the general classification for computational complexity theory. For more information: (Hochbaum, 1997)

each other. For example a high elite percentage leads quickly to a minimum, but this can be a local minimum due to the quick convergence of the solution space.

Tabu Search

Tabu search is a local search method and tries to find the next solution close to the current solution in the solution space, but certain adaptations are "tabu" or prohibited. The list of prohibited moves is updated as the search processes. Investigated solutions becomes forbidden. Every time the close solution space of the solution is redesigned by modifying the neighborhood structure.

Critical factors in TS are number of solutions on the tabu list, possibly the tabu attributes and the number of iterations.

Simulated Annealing

Simulated Annealing is a local search method based on probabilistic functions in physics and got his name from the controlled heating and cooling of metallurgy to influence defects. Here a energy probability function as similar for chemical potential for example determines the probability to impact the objection function with respect to the input parameter.

$$p(\partial E) = e^{\frac{-\partial E}{kT}}$$

where

$p(\partial E)$	Probability to make this "energy" change
E	Energy
k	Boltzmann constant
T	Temperature

Here the energy is de objection function with its constraints and the temperature is the dependent variable. In analogy to metallurgy the cooling down process to find the minimal solution takes place by changing the variable according to for example

$$T(i + 1) = \beta T(i) ,$$

where β is the cooling parameter and i the iteration number. This and the above mentioned algorithms are well described in (Taniguchi and Thompson, 1999) and on Wikipedia.

Conclusion for choosing GA

All three types have been used in City Logistics models for probabilistic vehicle routing and scheduling and location finding of public terminals for example (Taniguchi and Thompson, 1999). Results from the different techniques and comparison therefore depend on the size of the problem and specific conditions of the problem. However for time windowed problems the Genetic Algorithm is recommended.

The topic of the research is not to find and improve the vehicle routing by investigating different algorithms. At Kyoto University the Genetic algorithm was present and would thereby ease the implementation in the Netlogo model by serving as an example. Since a lot research in Kyoto is performed with the GA, it was decided to choose the GA to have the possibilities for discussion.

VRPTW-F vs VRPTW-D

The VRP-TW model will determine the optimal route for the freight carrier by minimizing the total transport cost of the freight carrier. For this vehicle routing problems time window are used to ensure on-time delivery at the retailers. There is the possibility to use dynamic of fore-casted routing. Here the choice is made to use fore-casted modelling by the Freight Carriers and then an asymmetry of information by the truck drivers, who do not have perfect knowledge of exact speed of the entire network, when they perform the actual run. They can just see streets from the crossing. Therefore dynamic rerouting is not used and VRPTW-F is chosen.

E.2 Main components of the Genetic Algorithm

The genetic algorithm contains the following parts: initialization and GA body (fitness calculation, elite creation, mutation and crossover). In the Netlogo model the Initialization consists of the demand collection of number of goods and the locations, a update by the roads on their conditions (network conditions update), calculation of the cheapest paths from the nodes in the system based on toll and operational costs per time phase & the generation of the chromosomes/individuals. During the body of the GA calculation per generation the fitness is calculated based on the objection function and several elite chromosomes are selected. Next the chromosomes experience mutation and crossover and the process is repeated. During the Netlogo model the choice is also made to keep track of the best chromosome so far, which will represent the final solution.

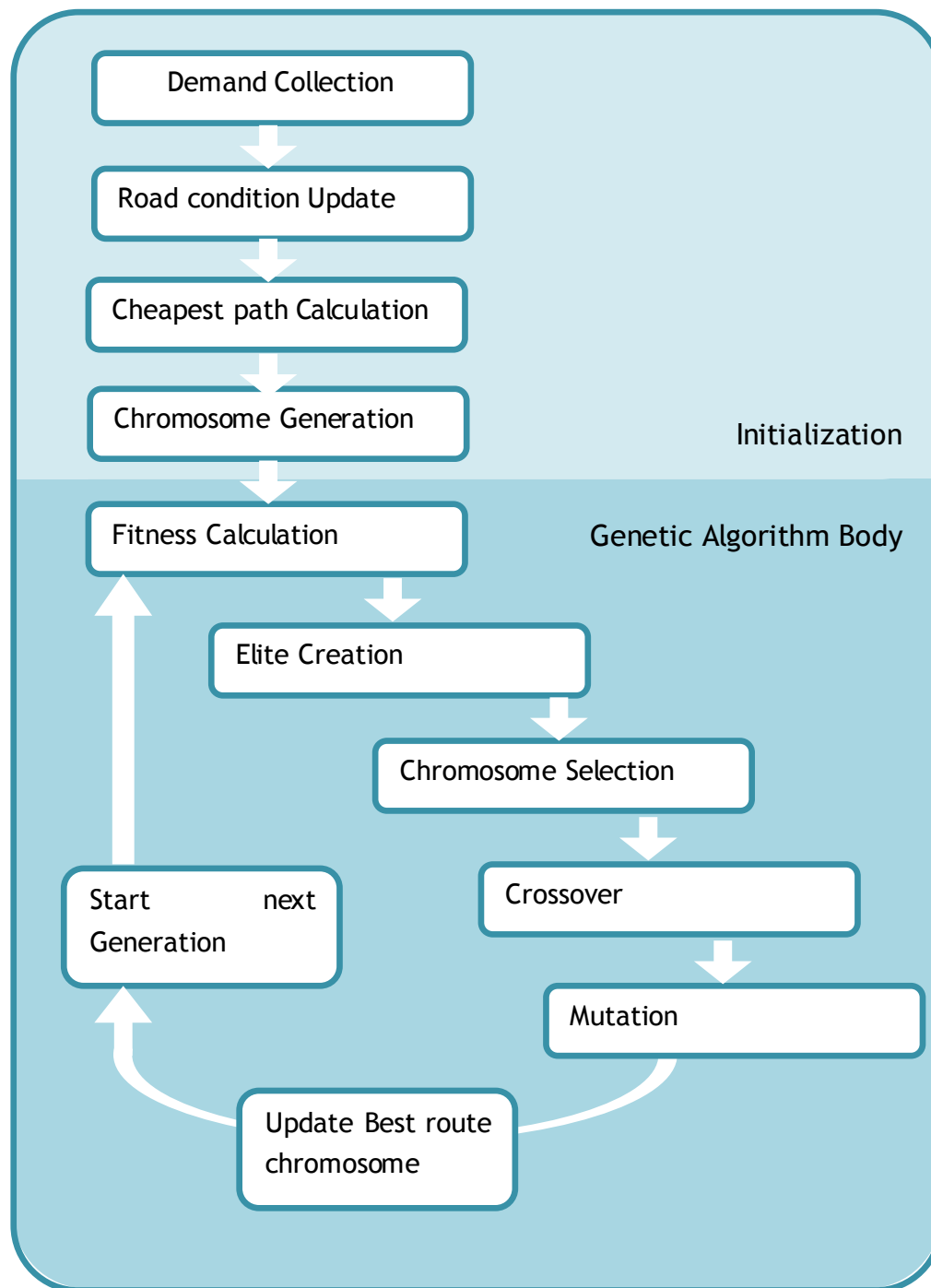


Figure 61: Schematic representation of the Genetic Algorithm (based on the version of Tamagawa and Taniguchi 2009)

The set-up for the GA is based completely on the article (Tamagawa and Taniguchi, 2009). However adaptations are required to deal with increased computation time. The computation

time will increase as the grid size is significantly larger (and flexible) and written in Netlogo. In the above article a fixed customer demand of 4 customers is present on a grid of 5 by 5 nodes. In the Netlogo model the grid size is flexible and 6 * 10 nodes with 6 * 60 patches on a grid of 240 potential customers and 300 road segments. The number of customers is also flexible (currently set to 10) and higher than before. This will lead a significantly larger chromosome and path calculations. Therefore tricks to reduce complexity are required, which are described below at the specified parts of the GA.

E.3 Initialization

In the initialization phase the main choices are made on how to model the problem and how to form the chromosome. The individual steps with the relating choices are explained below per component.

E.3.1 Demand collection

In order to model the UDC more dynamically the number of customers is flexible and customer location is variable. For most other analysis the location is variable, but the number is fixed at 10 customers per freight carrier. At the start of each day a number of shops decide to request a certain demand. The FC terminal collects the demand for their products by location and amount.

E.3.2 Network conditions update

The roads contain information about the average travel speed per time window. With Netlogo writing large output files on network conditions is not required as the network conditions are within the multi-agent model. In order to reduce computation time a network of nodes and links is laid out over the road network of patches. Now the calculation is not made per patch, but per crossing (node). This reduces the number of points in the cheap patch calculation by a factor 5, which currently with time windows is a large part of the computation time. The nodes are present on top of the roads at the location of crossings. These nodes are connected by hidden agents (links) called streets. The streets collect the information of the four roads below them to calculate the average travel time and costs in a certain time window.

E.3.3 Cheapest path calculation with the Dijkstra algorithm

Now the information from the street links and corresponding nodes is used to calculation the cheapest path from all nodes to other nodes with the Dijkstra algorithm. It is called cheapest path calculation as the cheapest path between nodes is searched, as the business is highly

price sensitive. However it is expected that often the cheapest path will also be the fastest one.

The Dijkstra algorithm contains the following concepts:

1. In the system of nodes and links set the price to travel to the nodes to an artificial high value. The Dijkstra algorithm improves them step by step.
2. Name the starting node the initial node in the network and set the cost to zero as this is the current location.
3. Mark all nodes as unsearched and set the initial node as searched and current node. (The status can be unsearched, searching or searched. The current node is the current location from where searching will take place.)
4. For current node, consider all its unsearched and searching neighbors (by travelling the available links) and calculate their travel cost (plus the cost from the current node). Then if the calculated price is lower than the currently node price, replace the price (and travel time for phase purposes) and set the node to searching status.
5. When all the neighboring nodes have been calculated, mark the current node as searched. Now it will not be searched again.
6. If all nodes have been searched, the program is finished. Otherwise, select the node in the total system with the lowest price and status searching, and set this the current node. Now repeat from step 3.

Now this process is repeated for all the times window and for all the nodes in the network. An attempt to find a smarter and faster computation based on binary choices was made, but proved less successful on this larger grid in a short analysis. Therefore the Dijkstra algorithm is implemented in Netlogo for cheapest path calculation. Now for all nodes and all time phases the costs are calculated as this is needed later in the fitness calculation.

E.3.4 Chromosome generation

Since the number of customers is increasing, the chromosome length will increase and thereby computation time. Therefore efforts are made to decrease the length of the chromosome as this is an important factor determining the computation time for the GA. By assuming each departing truck from the terminal is a new identical truck and automatically assuming the route home to the terminal, the current chromosome length is the number of trucks + number of customers. Previously this was $\text{customers} * 2 + \text{trucks} + 1$. Distinction between trucks is automatically made on the basis of a new departure time, which includes the return to FC terminal for the previous truck. This eliminates the intrinsic possibility to send out a truck two times per day. If artificial terminal nodes are included, this can be

solved, but currently no further research is made on this as this requires that the amount of artificial nodes and new time additions need to be investigated before implementation.

This leads to the following form of the chromosome.

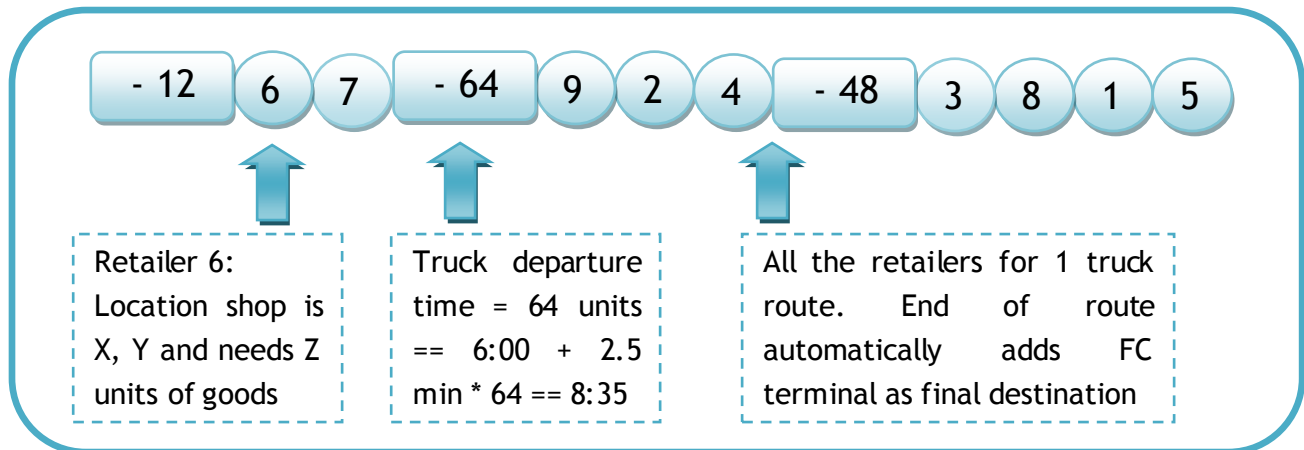


Figure 62: Schematic representation of the chromosome

Genes are not randomly generated, but contain trucks filled with the average load factor to improve the quality of the starting population for faster solving. This was checked by comparing both methods for the freight carrier case. This means that the amount of trucks is decided by the demand size and is not fixed for the freight carriers or the UDC. The conclusions are shown in Appendix F: GA verification FC and for the UDC in Appendix G: GA verification UDC.

E.4 Body of the Genetic Algorithm

The previous steps were needed for the initialization of the GA. The steps below are repeated for many generations until the final generation is reached. These steps include fitness calculation, elite creation and selection, mutation and crossover.

E.4.1 Fitness calculation

The fitness calculation is based completely on the costs. The various costs are described fully in the multi-agent model and include fixed costs per truck, overloading penalty (variable cost), operating costs based on travel time (variable cost) and toll by the municipality (variable cost).

These are calculated per chromosome for the whole population. During the generations information about the fitness of the total population and fittest individual is recorded to

monitor progress. In order solve the problem of vehicle routing for the freight carriers as well as the UDC, the following cost function needs to be minimized:

$$C(T, R) = \sum_{n=0}^m f * \delta(n) + \sum_{n=0}^m C_{t,l}(T, R, n) + \sum_{n=0}^m P_{t,l}(T, R, n) \quad (1)$$

where

$C(T, R)$	Total cost in ¥ for set trucks with departure time T and route of customers R
T	Set of trucks with departure time
R	Set of routes with customers per truck
n	Truck number, m is the maximum of trucks
f	Fixed cost per truck (identical for all trucks as only 1 type is used)
$\delta(n)$	Delta function, which is 1 for a used truck and 0 for non-used truck
$C_{t,l}(T, R, n)$	Operating cost per truck, which depends on the time of day and its corresponding network conditions
$P_{t,l}(T, R, n)$	Penalty costs per truck for late delivery to customers

$$C_{t,l}(T, R, n) = \sum_{p=1}^q (c(p, t, n) + w(n)) + c(p, t_0, n) \quad (2)$$

where

$c(p, t, n)$	Cost from current location to the next customer p from customer 1 to q on phase t of the day for truck n. This information is contained in the network matrix per node.
$w(n)$	Costs of waiting time at loading/unloading at the customer site for truck n.
$c(p, t_0, n)$	Cost from the last customer back to the terminal location based on phase p of the day, location of the terminal t_0 for truck n.

The penalties are based on the following function:

$$P_{t,l}(T, R, n) = \sum_{p=1}^q (a(p, n)) + l(n) \quad (3)$$

Where

$a(p, n)$	Artificial high value of 200.000 yen when arriving late and zero when on time per customer p and for truck n
-----------	--

$l(n)$ Artificial high value of 200 * (number of overloaded parcels) when overload and zero when within load limits per truck n .

Simultaneous the travel time is also calculated, besides travel costs in a similar matter.

$$T_{t,l}(T, R, n) = \sum_{p=1}^q (t(p, t, n) + w_t(n)) + t(p, t_0, n) \quad (4)$$

where

$t(p, t, n)$ Time from current location to the next customer p from customer 1 to q on phase t of the day for truck n . This information is contained in the network matrix per node.

$w_t(n)$ Waiting time at loading/unloading at the customer site for truck n .

$t(p, t_0, n)$ Cost from the last customer back to the terminal location based on phase p of the day, location of the terminal t_0 for truck n .

Here the travel costs and travel time were calculated in the Cheapest Path Calculation (based on the Dijkstra algorithm). The solution with the lowest costs is compared to the current best solution and copied as the current best route. The final output of the model will be this best route.

E.4.2 Elite creation and selection

After determining the fitness per chromosomes the elites are created. A certain percentage of the chromosomes with the highest fitness (lowest costs) will be elites and they automatically continue to the next generation. The remaining part of the next generation (100 % minus the elite rate) of chromosomes is randomly selected out of all chromosomes (including elites). There is a possibility that some elite chromosomes therefore appear twice at this stadium, but this is not a problem as crossover and mutation will take place. By using elite chromosomes the good solutions have a higher presence in the next generation, thereby increasing the average fitness of the total population. The solution with the lowest costs is compared to the current best solution and copied as the current best route.

E.4.3 Mutation

In the current set of chromosomes mutation and crossover takes place. First mutation is explained here. There is the mutation for the general chromosome called customer mutation and a special mutation for the departure times of the trucks.

Customer Mutation

In the chromosome one specific gene moves to another location, while the rest stays in the relative same position. First two genes are randomly selected. The gene on the right will be the moving gene and goes to the location of the other gene, while the genes in between all move one to the right (see figure). This has the probability of the mutation rate of happening.

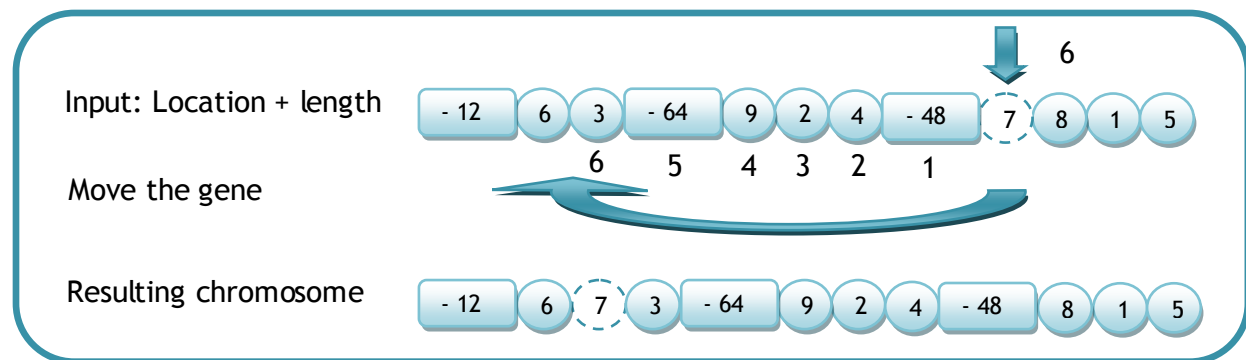


Figure 63: Concept of customer mutation

Departure time mutation

During the departure time mutation all the departure times are selected out of the chromosome. Then these departure times have the chance with probability mutation rate to change their departure time within 5 hours of the current departure time.

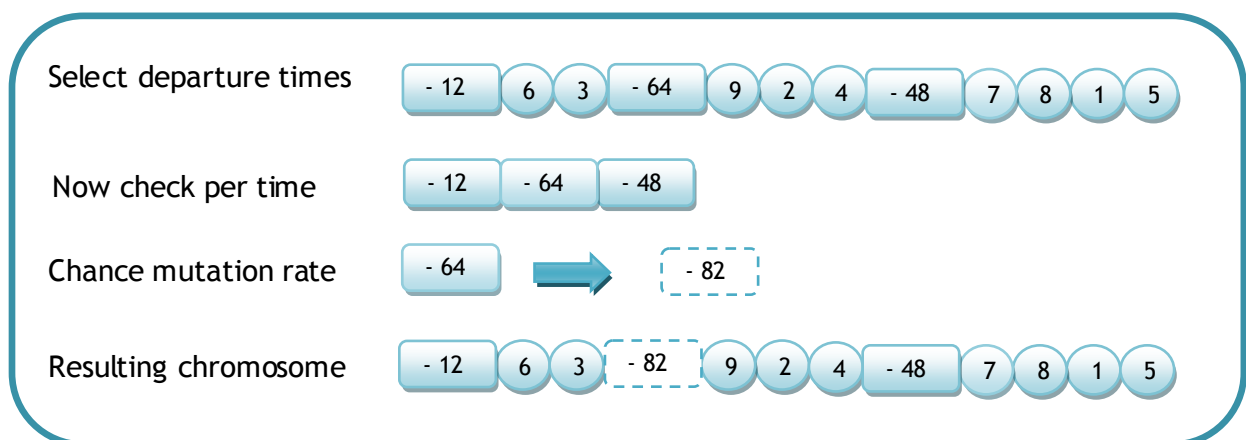


Figure 64: Schematic representation of time mutation

E.4.4 Crossover

While the time genes are separated from the chromosome they also experience with probability crossover rate to have time crossover. Later customer crossover is explained.

Time crossover

In time crossover the genes of the departure times are cut a random place (though the same place for a pair of chromosomes) and the position is swapped.

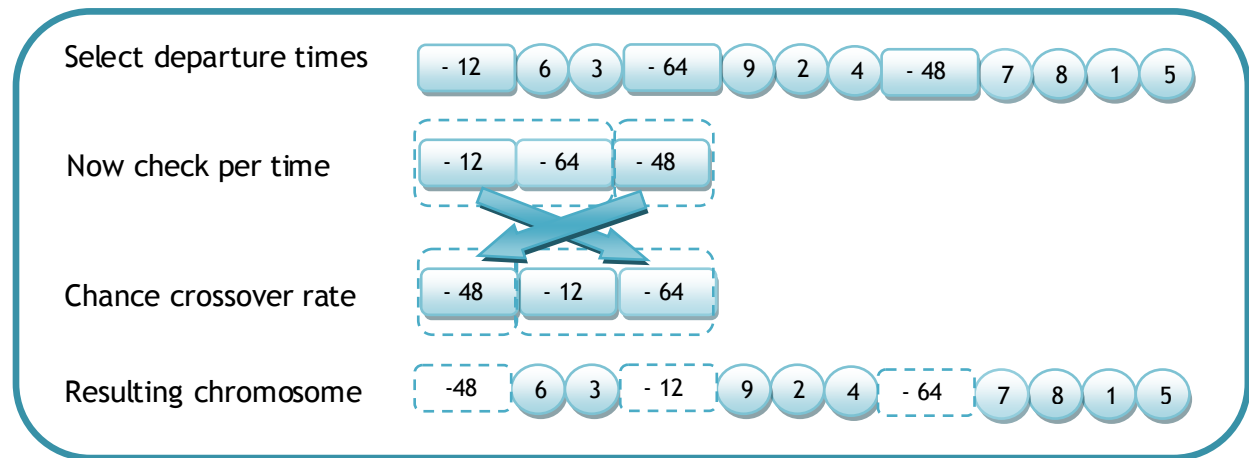


Figure 65: Schematic representation of time crossover

Customer Crossover

Customer cross is very complex, so it is first explained with biology analogy before showing the GA chromosome example. In biology the crossover between chromosomes looks schematically like:



Figure 66: Biology schematic crossover between two chromosomes (the orange circle represents the location of crossover).

However, if the exact same principle is used for the chromosomes in the GA, some chromosomes will have some customers double, while missing some customers of the list. Therefore crossover will take place with two chromosomes, but while changing only its own order.

Therefore crossover takes place as shown in the next figure. First two locations are randomly chosen inside the chromosome. In between is the “safe zone”. First on the right side of the “safe zone” a search is made for chromosomes which are the same for both chromosomes. In our example none is found. Next the “left” side is searched from the first position of chromosome until the same gene is found on chromosome 1. If the same gene is found, it is

moved to the same location. Then the next location on the second chromosome is searched on the first chromosome, etcetera. This process is repeated until the “safe zone” is found. Now the process is repeated for chromosome 2 with the original chromosome 1.

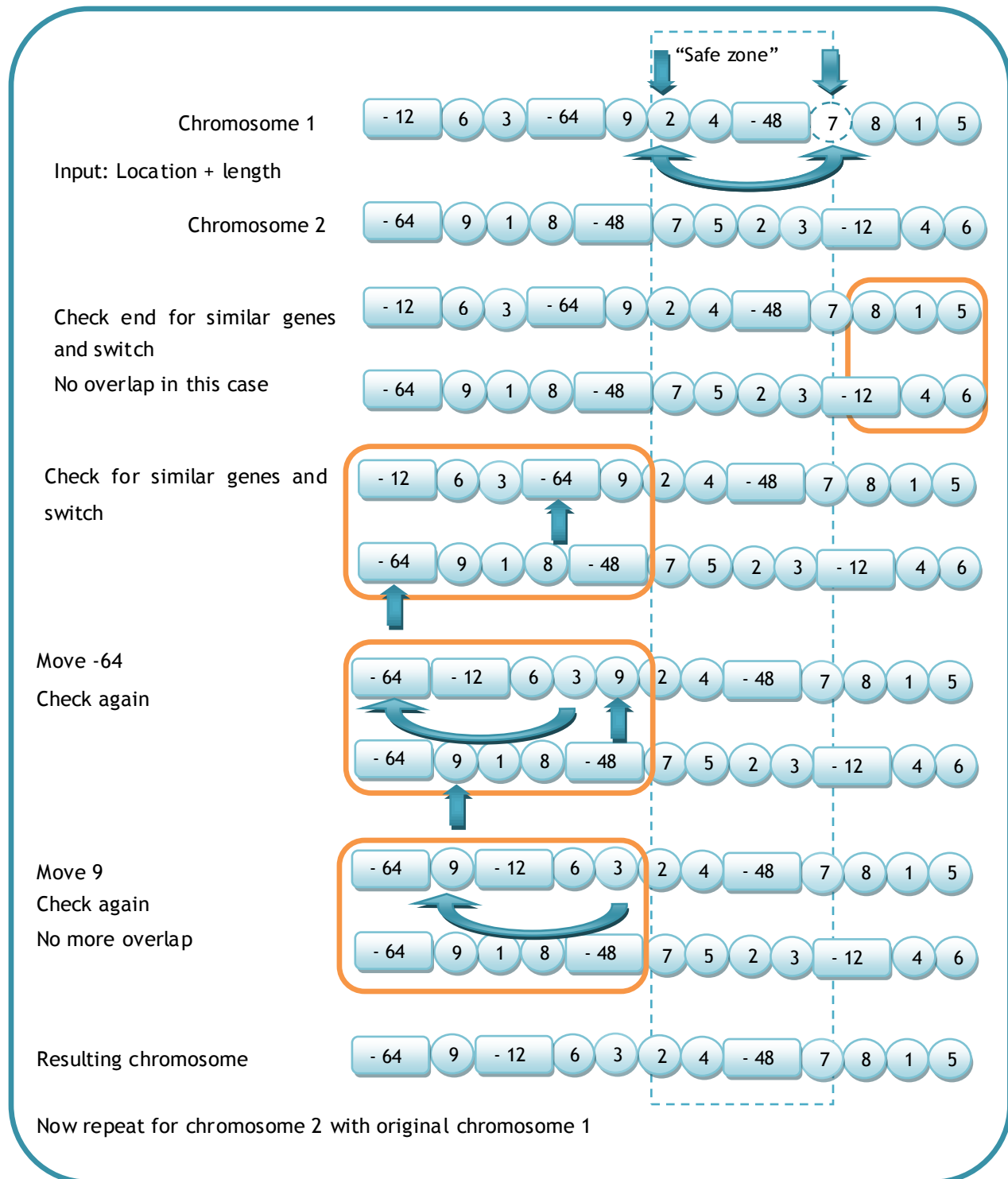


Figure 67: Schematic representation of the customer crossover of chromosome 1

E.4.5 Short overview main changes with original model

The main changes with respect to the original model are:

- Different chromosome build-up
- Two layered network with patches as well as links and nodes
- Predefined chromosome order
- Best route solution track

E.5 Genetic algorithm Validation in Netlogo

In the original model the optimal parameters were searched one after another. However all parameters are correlated. Therefore the Behavior space of Netlogo is used to find the optimal set of parameters. Search parameters for the Freight carriers are presented here with plotted results. For the UDC the same analysis has been performed, but for the different demand sizes. These results can be found in Appendix G: GA verification UDC.

Table 46: Parameter search for the freight carrier parameters

Parameter	Value
Number of chromosomes (Numchromo)	220
Generations	200
Percentage of elites	15
Crossover rate	55
Mutation rate	6

E.6 Conclusion

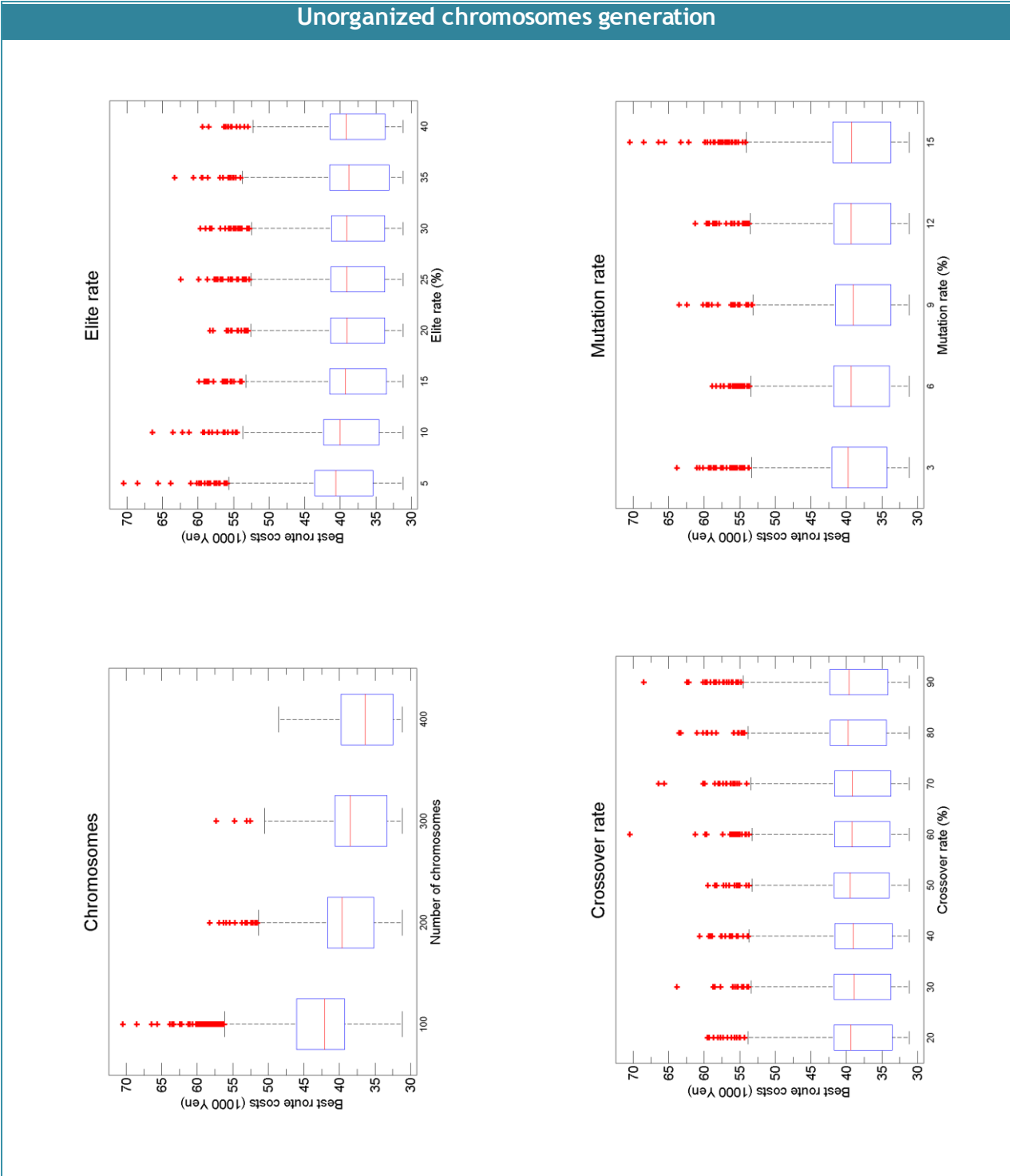
The vehicle routing in the model is a non-deterministic polynomial-time hard combinatorial optimization process. In this chapter the genetic algorithm was chosen over Tabu search or Simulated annealing as it is said to have better time window operation. The genetic algorithm uses concepts like chromosomes, crossover, mutation and generation from biology to find the optimal solution. The general lay-out of the genetic algorithm consists of initialization (with demand collection, road network conditions update, cheapest path calculation and chromosome generation) and the genetic algorithm body (with fitness calculation, elite creation and selection, mutation and crossover), which is repeated many times. For the fitness calculation the full objective function for vehicle routing is presented.

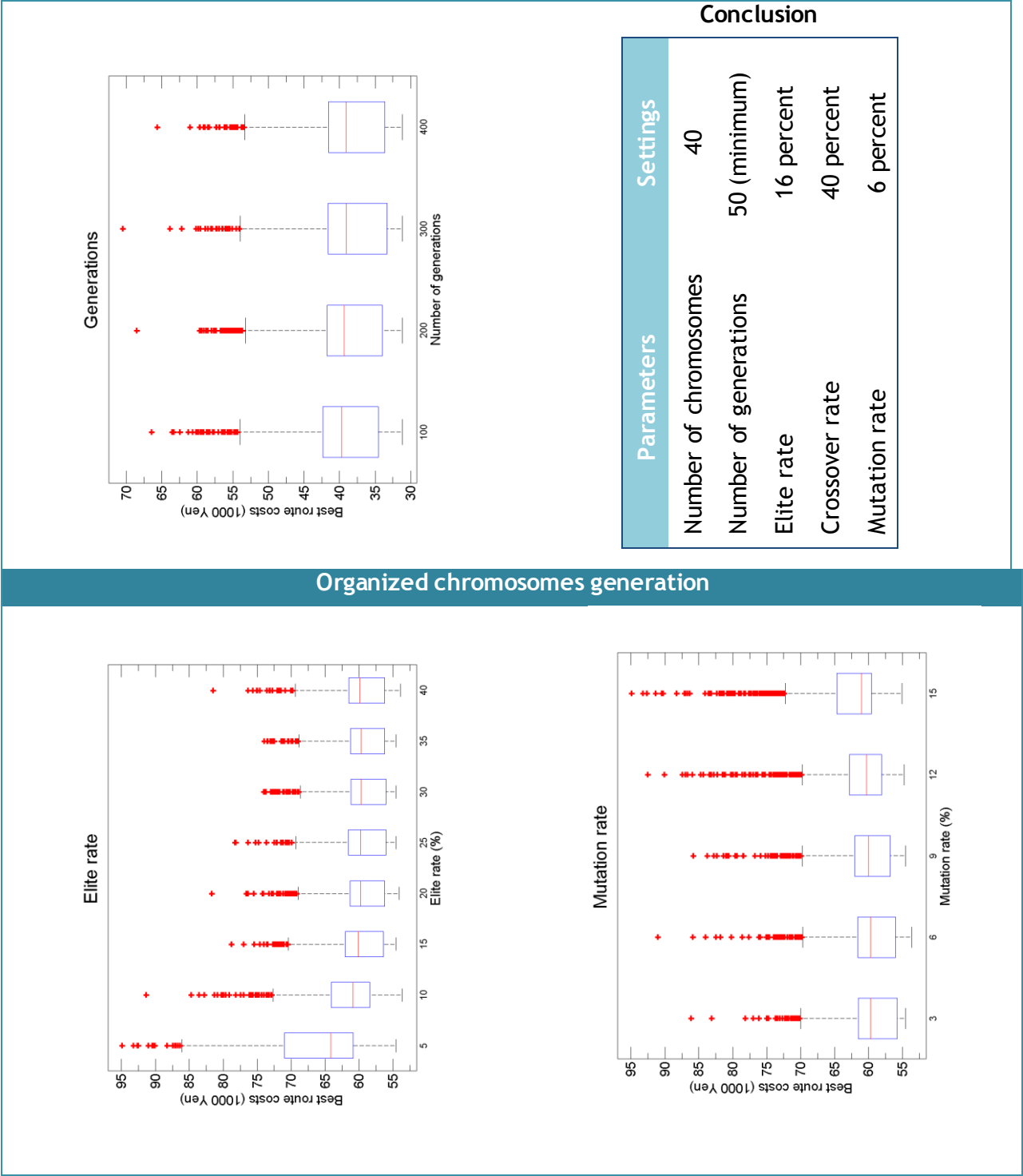
Validation of the genetic algorithm was done by examining the behavior space for the parameters and searching for the optimal solution. The results for the freight carriers were presented in this chapter. The results for the UDC can be found in the Appendix G:GA verification UDC.

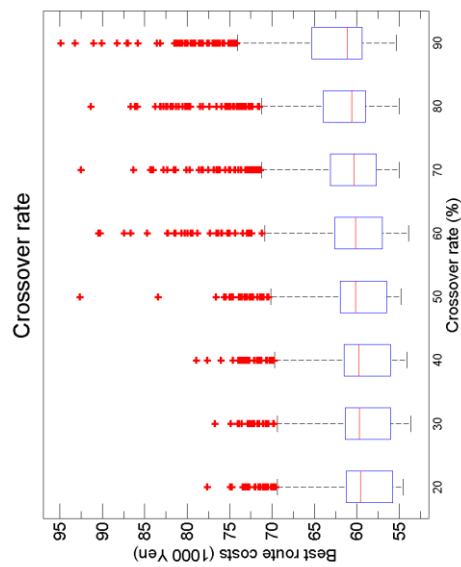
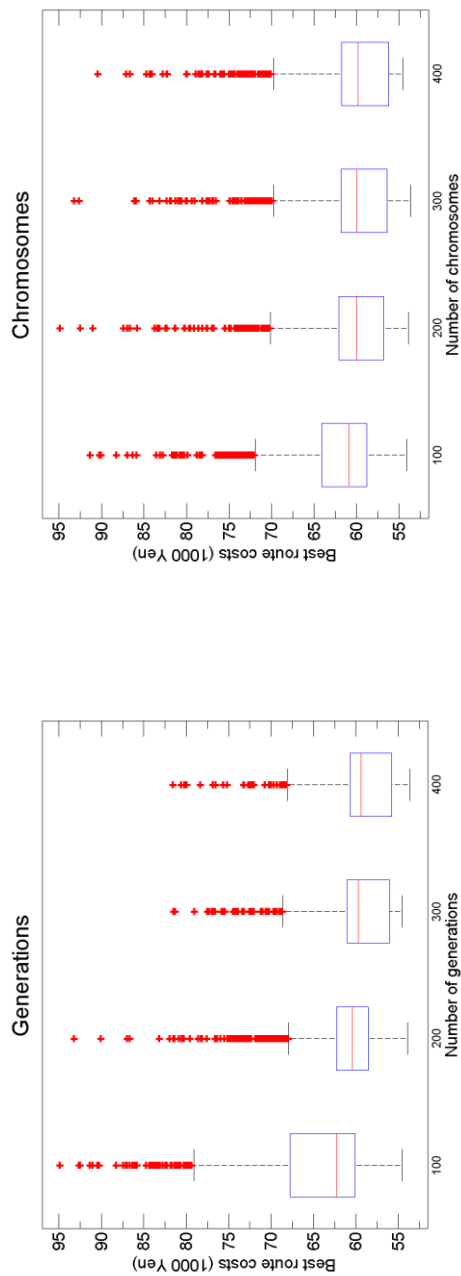
F GA verification FC

For the freight carriers the correct settings for the GA had to be found. This is performed for the ordered and unordered trucks to indicate if ordering has a benefit.

Table 47: GA parameter search for organized and unorganized chromosome generation







Conclusion

Parameters	Settings
Number of chromosomes	220
Number of generations	200
Elite rate	15 percent
Crossover rate	55 percent
Mutation rate	6 percent

G GA verification UDC

The UDC has a variable demand varying between 2 and 30 retailers. Each demand size forms a different length chromosome. For each demand size different parameters are required for the genetic algorithm. The results for these parameters searches are presented for these various sizes in this appendix.

Since the calculation time of these parameters went up to several days for the longer chromosomes, not all the exact sizes were calculated. The demand sizes, which were calculated, are: 2, 5, 8, 10, 13, 16, 20, 24 and 28. Since larger demand requires more genes and therefore a longer chromosome and normally also more generations, it is assumed that values between tested values assume the parameters of the higher value. For the elite, crossover and mutation rate it is assumed that the parameters can remain the same and that the extra number of chromosomes and generations will ensure a good outcome. All measurements were performed assuming a load factor of 8 units.

In the Tamagawa model (Tamagawa, Taniguchi et al, 2009) the approach was taken to search the optimum parameters by varying one parameter, keeping other constant and finding the optimal setting for the individual parameter. However since all parameters impact each other (for example high elite rates require high cross over/ mutation rates), the benefits of Netlogo are used and global minima are searched with behavior space. This increases the search time significantly as before per parameter various values were tested sequentially for all parameters, but now various values for all parameters are tested simultaneously where possible. This is compensated slightly by the advantage that Netlogo does this most efficiently by using all processors. For larger chromosomes the number of generations was put to a high value and repeated separately later to reduce calculation time.

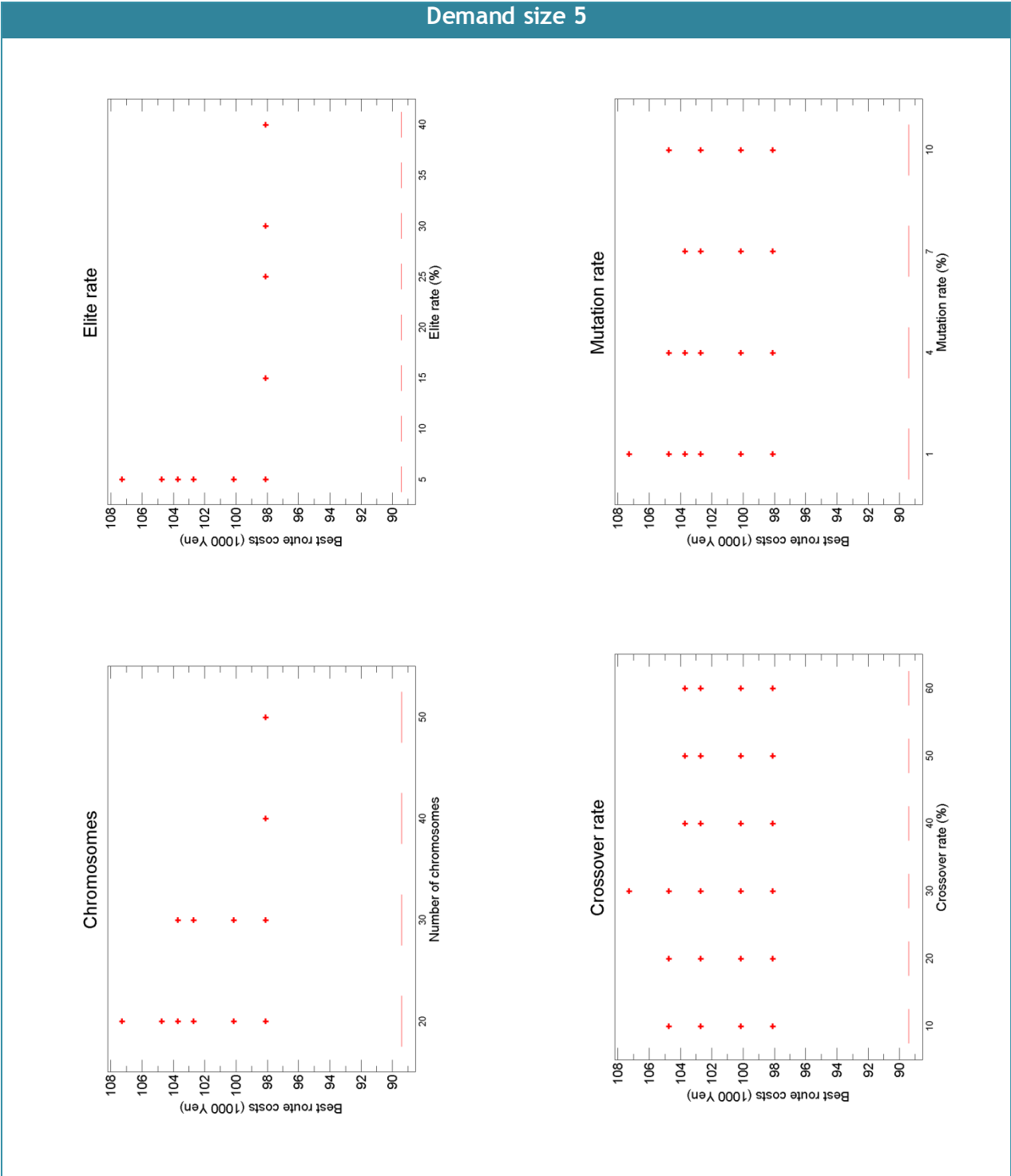
Below the final results are presented for the various chromosomes lengths (aka demand sizes). Below this table specific information is presented per demand size to show the results on the results.

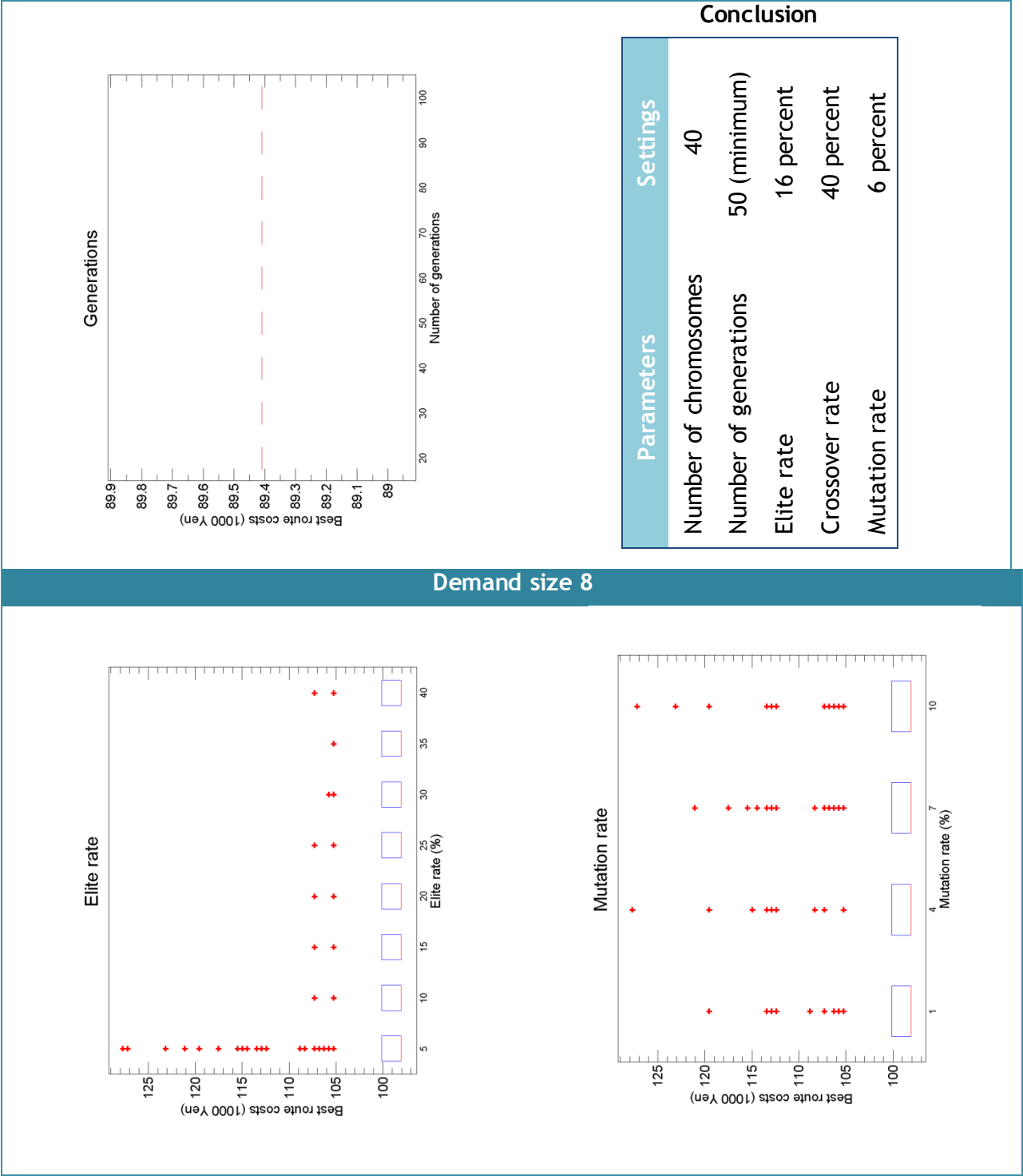
Table 48: Overview of parameters settings for the UDC GA

Demand size	Chromosomes	Generations	Elite rate (%)	Crossover rate (%)	Mutation rate (%)
2	20	50	10	40	6
5	40	50	16	40	6
8	120	200	25	45	6
10	350	450	25	55	6
13	600	700	26	85	7
16	650	800	22	65	9
20	800	850	30	55	7
24	1000	1200	24	45	3
28	1200	1600	22	24	4

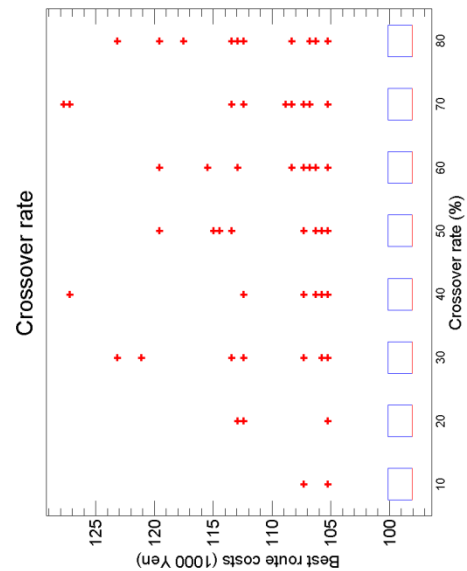
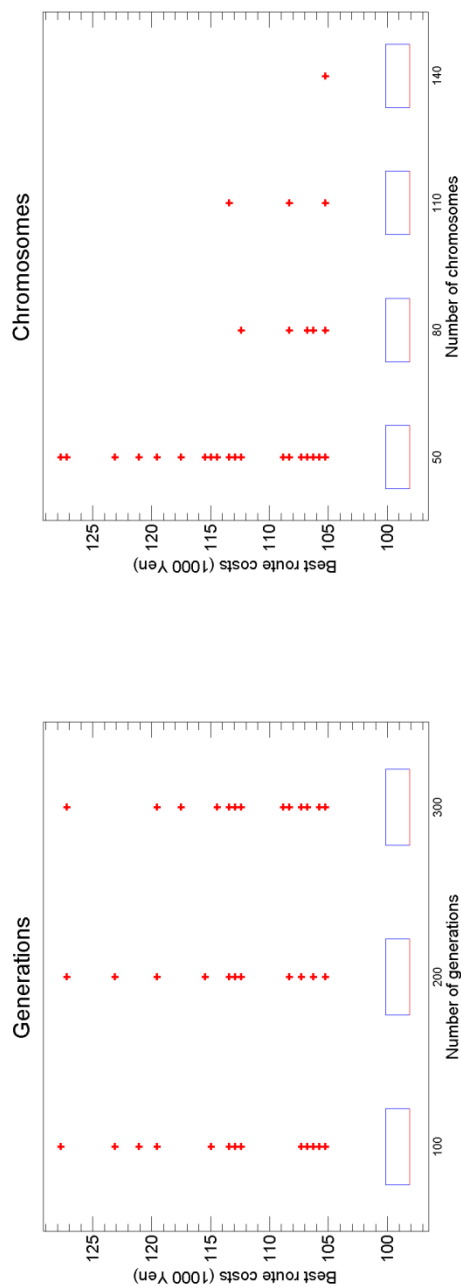
Table 49: Results from Behavior Space Analyses in Netlogo for various demand sizes

Demand size 2													
Conclusion													
	<table> <tr> <th>Parameters</th><th>Settings</th></tr> <tr> <td>Number of chromosomes</td><td>20 (minimum)</td></tr> <tr> <td>Number of generations</td><td>50 (minimum)</td></tr> <tr> <td>Elite rate</td><td>10 percent</td></tr> <tr> <td>Crossover rate</td><td>40 percent</td></tr> <tr> <td>Mutation rate</td><td>6 percent</td></tr> </table>	Parameters	Settings	Number of chromosomes	20 (minimum)	Number of generations	50 (minimum)	Elite rate	10 percent	Crossover rate	40 percent	Mutation rate	6 percent
Parameters	Settings												
Number of chromosomes	20 (minimum)												
Number of generations	50 (minimum)												
Elite rate	10 percent												
Crossover rate	40 percent												
Mutation rate	6 percent												
All settings meet the requirement.													



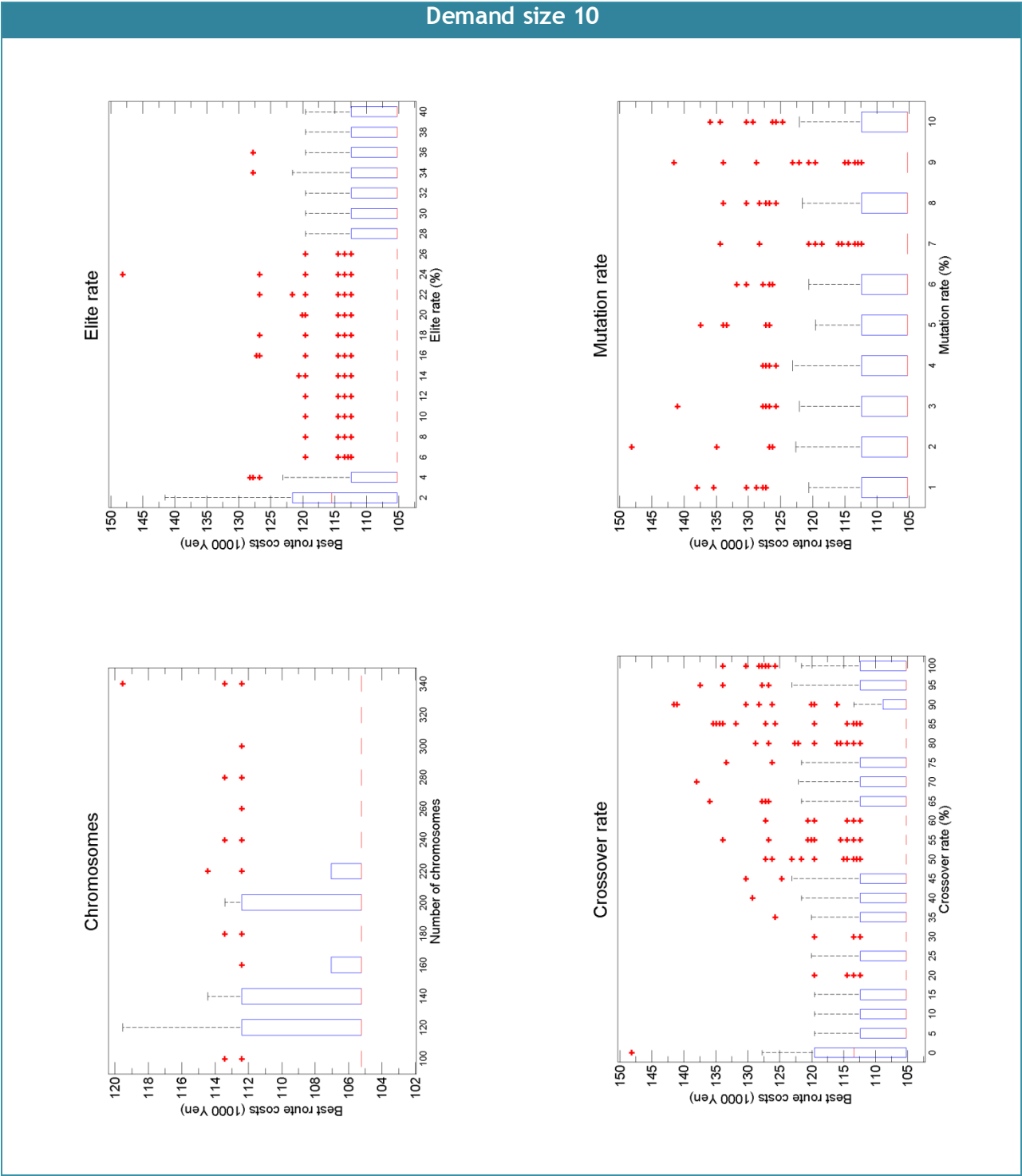


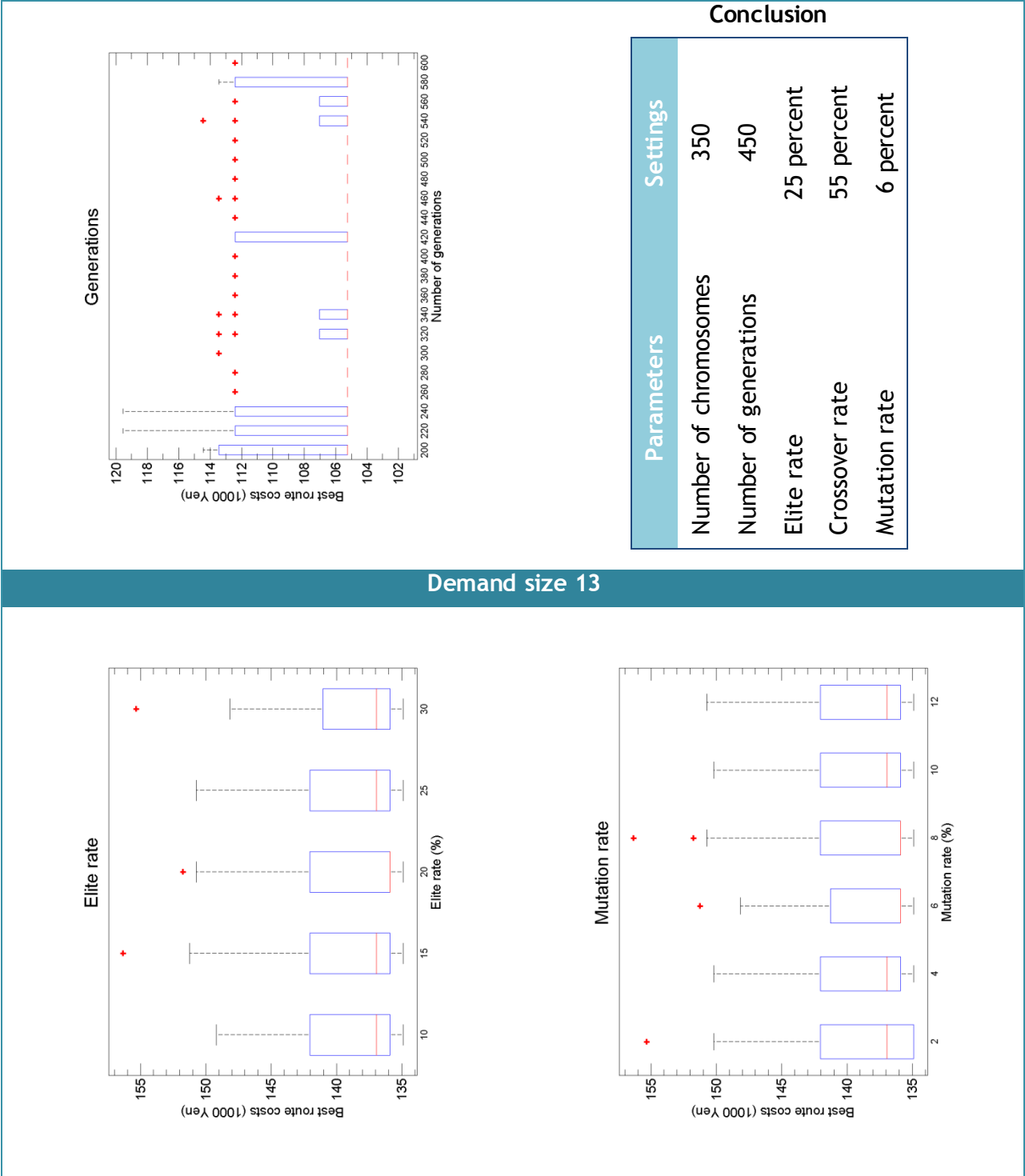
Demand size 8



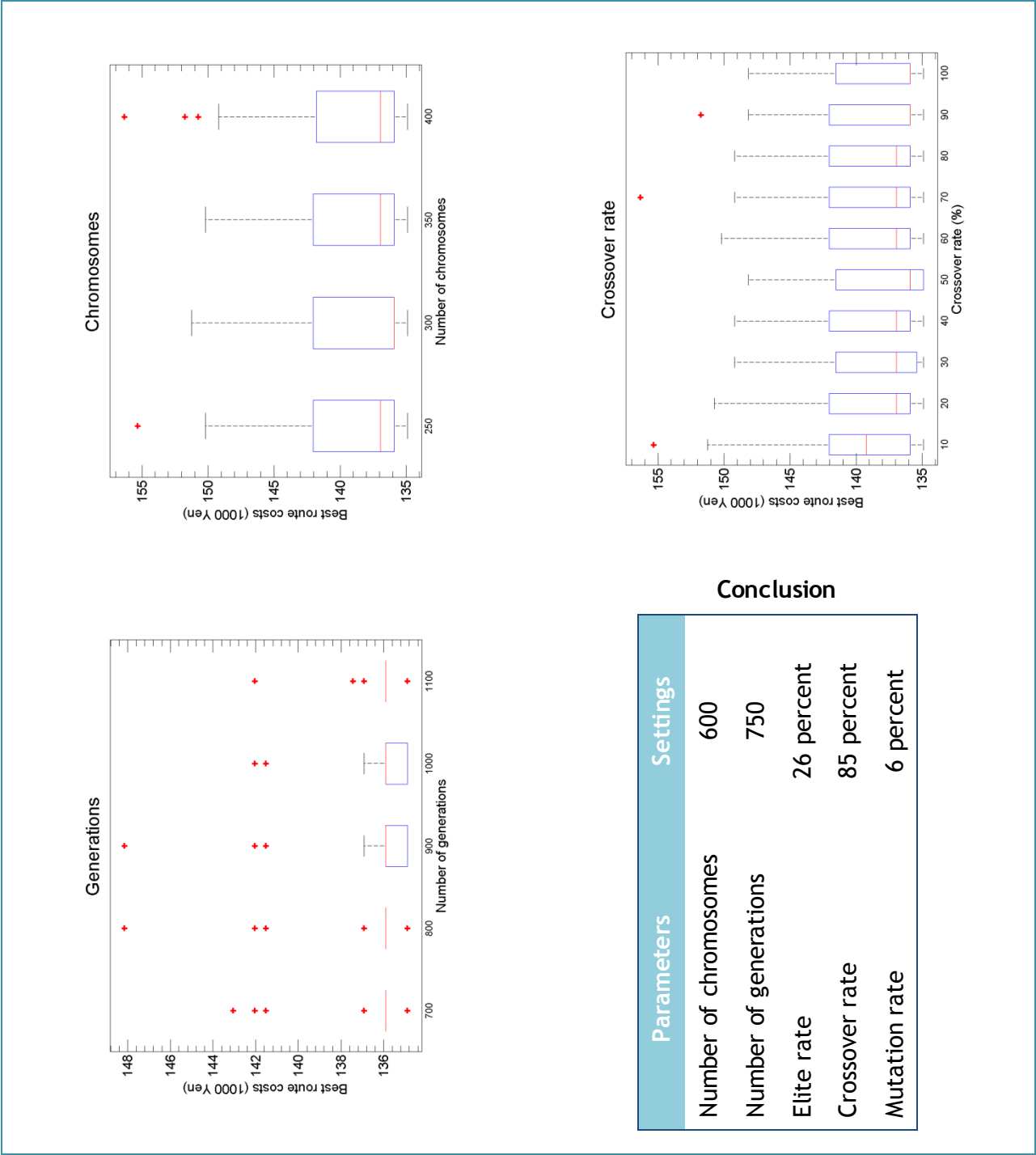
Conclusion

Parameters	Settings
Number of chromosomes	120
Number of generations	200
Elite rate	25 percent
Crossover rate	45 percent
Mutation rate	6 percent

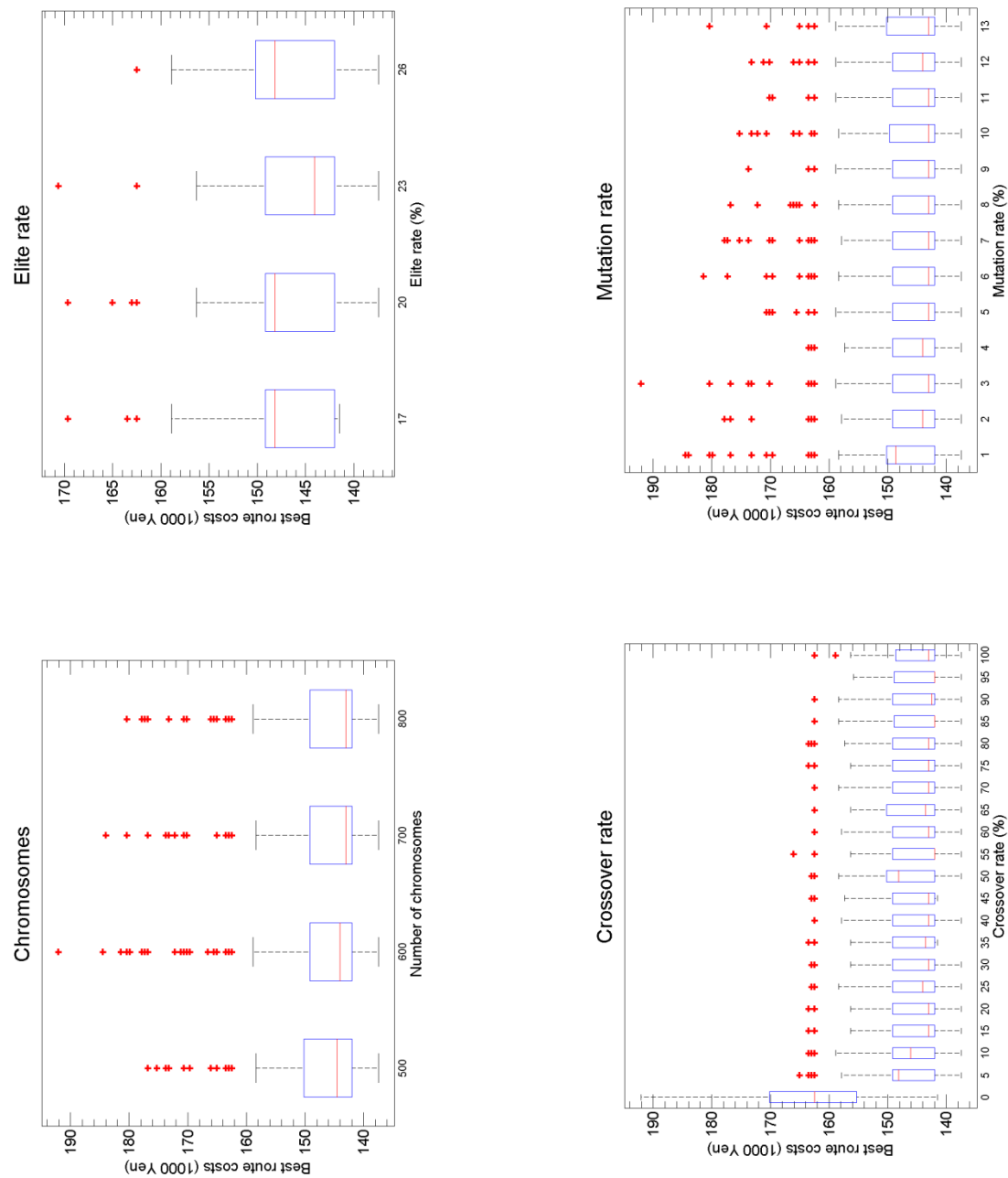


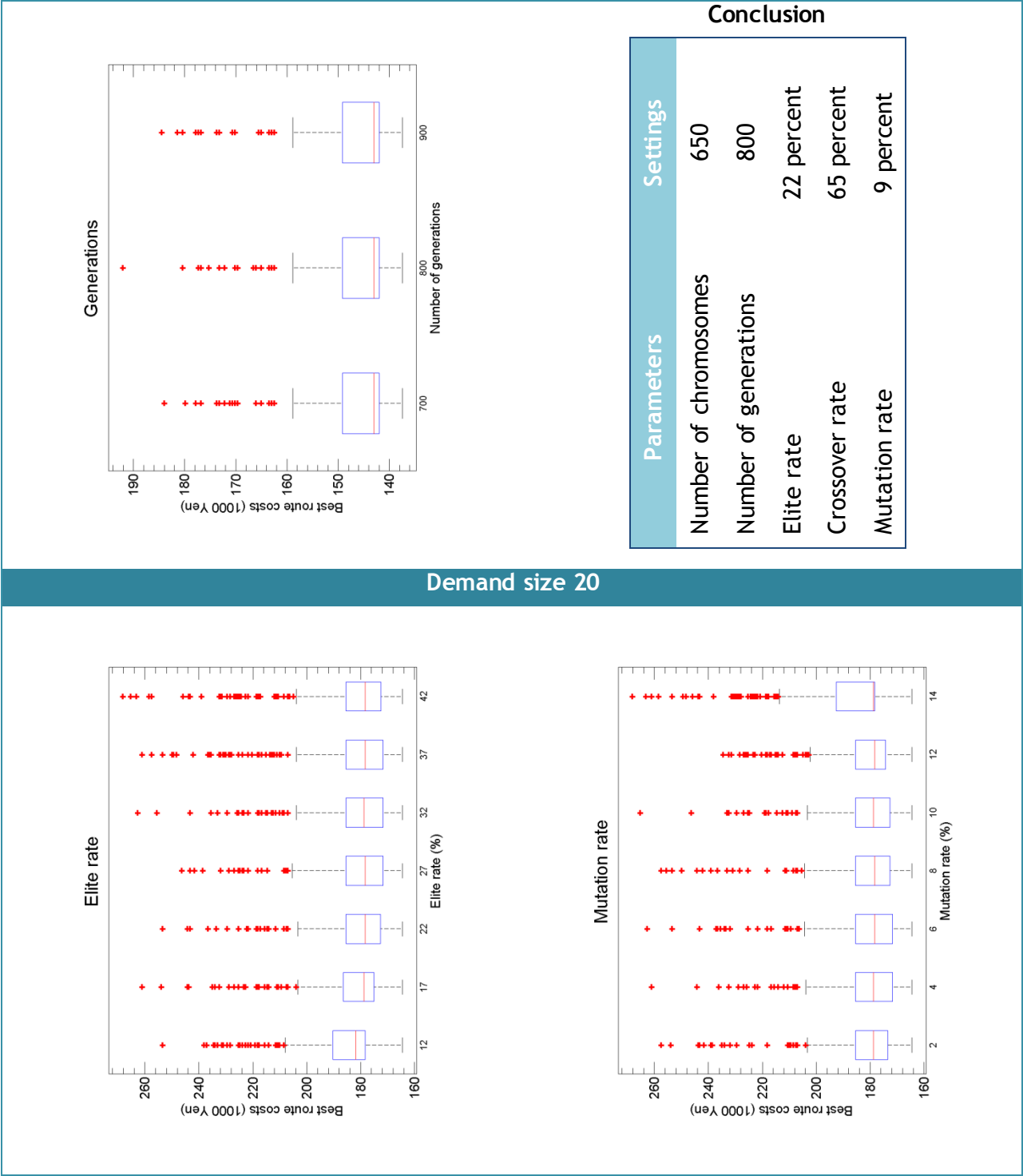


Demand size 13



Demand size 16





Demand size 20

Elite rate

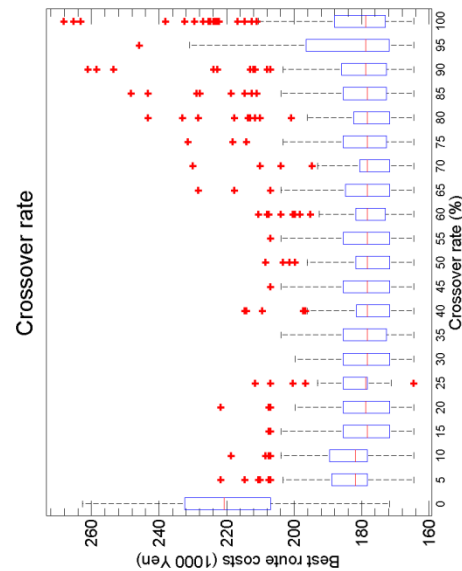
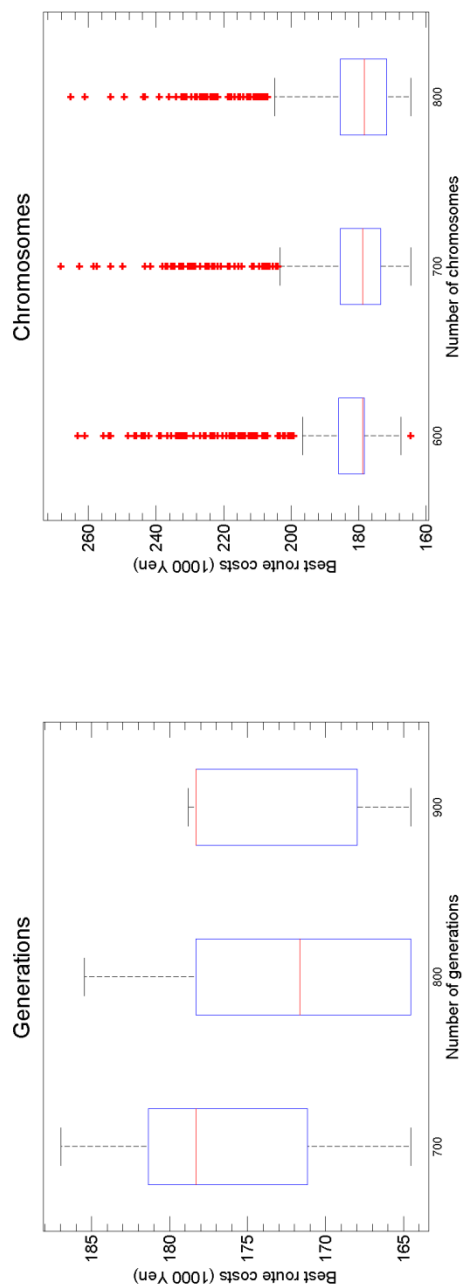
Best route costs (1000 Yen)

Elite rate (%)

Mutation rate

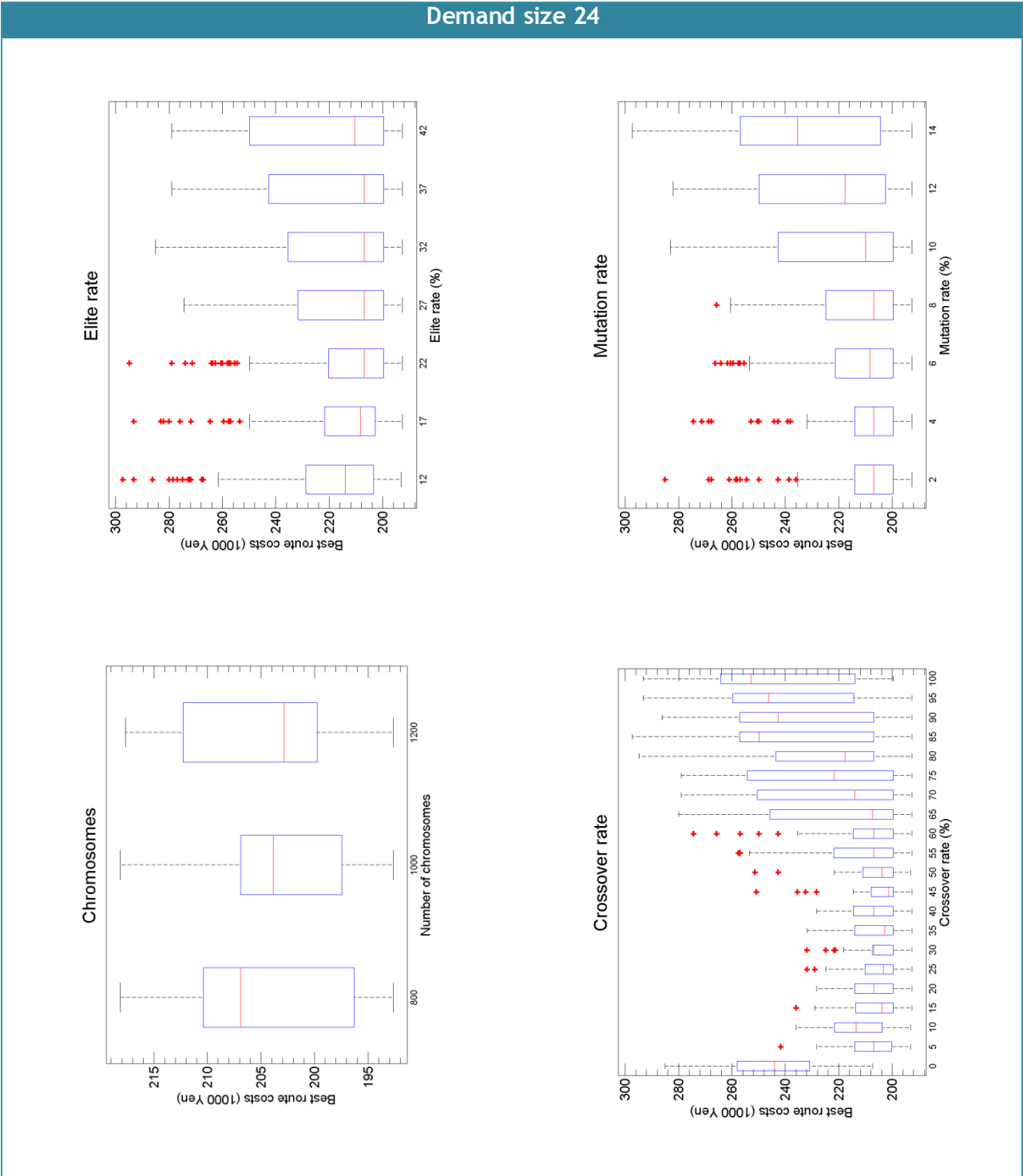
Best route costs (1000 Yen)

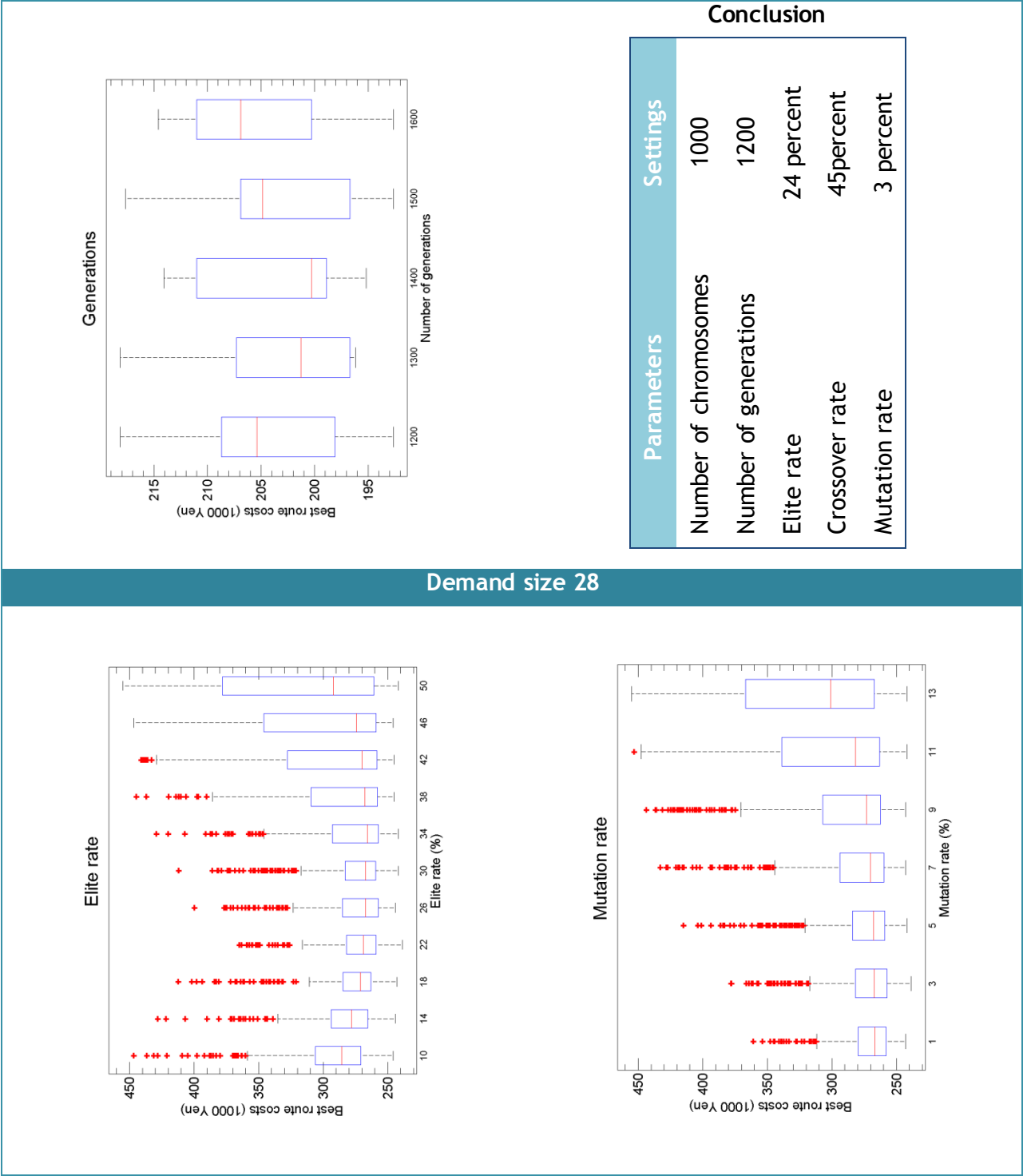
Mutation rate (%)



Conclusion

Parameters	Settings
Number of chromosomes	800
Number of generations	850
Elite rate	30 percent
Crossover rate	55 percent
Mutation rate	7 percent





Demand size 28

Elite rate

Best route costs (1000 Yen)

Elite rate (%)

Mutation rate

Best route costs (1000 Yen)

Mutation rate (%)

Parameters	Settings
Number of chromosomes	1200
Number of generations	1600
Elite rate	22 percent
Crossover rate	24 percent
Mutation rate	4 percent

G.1.1 Proceedings

For demand size 2 the solution is always contained in the initial set. For demand size 3-5 the optimal solution is also always found quickly.

1. From demand size 6-8 the searching for the right parameters gets more complicated. No more are parameters settings observed the value is always at a minimum. Minimum settings need to be found with quadratic fits for elite, cross over and mutation rate. The number of chromosomes and generations is determined with a quadratic fit to indicate the point where increasing the chromosomes and generation has most benefit; the function itself should display the properties of an exponential decrease. This is scheme is performed for demand size 9-10.

From demand size 11-13 first rough test are performed to find the ranges of the parameters search as the parameters and computation time are increasing fast. With higher chromosome and generation numbers the problem is much more complex to solve for the GA and the right answer is not always found with the correct parameters.

Next is the demand size up to 16, which indicates two completely full trucks. From this size on many rough tests are needed and generation is put to a high value and retested later with different values as computation time is now a serious issue; running these behavior spaces takes 1 to 3 days. Therefore the choice is made to assume a high generation number and test this later with the other specified parameters values. Often the perfect solution is not found anymore.

For demand size 17 to 30 it was extremely difficult to find the right parameter set. The same procedure is taken as for demand size 16. For demand size 24 and 28 the number of chromosomes was also hold at constant high value and tested later. It is assumed that even higher number of chromosomes and generations are required, but due to computation time a choice is made to accept a suboptimal solution. This leads to a suboptimal solution, but computation time to search the entire behavior space was problematic.

G.1.2 Conclusions

The following GA parameters settings are chosen for the UDC based on behavior space searches in Netlogo for the various parameters, where the minima were sought.

Table 50: Overview of parameters settings for the UDC GA

Demand size	Chromosomes	Generations	Elite rate (%)	Crossover rate (%)	Mutation rate (%)
1-2	20	50	10	40	6
3-5	40	50	16	40	6
6-8	120	200	25	45	6
9-10	350	450	25	55	6
11-13	600	700	26	85	7
14-16	650	800	22	65	9
17-20	800	850	30	55	7
21-24	1000	1200	24	45	3
25-30	1200	1600	22	24	4

For future research it is recommended to redo these tests more extensively and investigated repeatability as well as impact of behavior space borders and granularity. Inner consistence is present for rerunning the same behavior space at low demand sizes (with a uncertainty due to random choices in the GA and low success rate), but results are highly dependent on parameters search settings (as all the rates are determent by the minimum in the quadratic fit). More research will fine tune the parameters setting results and improve the GA performance. Other computers could allow for the possibility to use larger chromosome and generation numbers for high demand sizes, but the current choice is to limit computation time.

H Q learning

The agent has actions $\forall a_t \in A$ and is in a surrounding of state $\forall s_t \in S$. After each state a reward or punishment is given to the agent based on his chosen action. The goal of the agent is to maximize its total reward. Therefore the agent learns which action is optimal for each state. In Q-learning the Quality function for each state-action combination. Initially these values are given as input, but during the course of the model new values are calculated for state-action combinations and the Quality function is updated.

$$Q(s_t a_t) \leftarrow \underbrace{Q(s_t, a_t)}_{\text{old value}} + \underbrace{\alpha(s_t, a_t)}_{\text{learning rate}} \times \left[\underbrace{\overbrace{r_{t+1} - \gamma \max_{a_{t+1}} (Q(s_{t+1}, a_{t+1}))}}_{\text{expected discount rate}} - \underbrace{Q(s_t, a_t)}_{\text{old value}} \right] \quad 1$$

Table 51: Parameters from basic Q-learning system

Parameter	Meaning
$Q(s_t a_t)$	Expected profit obtained from state $s_{f,t}$, to the last state when selected the behavior $a_{f,t}$, in state $s_{f,t}$
$\forall s_t \in S$	State from a set of all states
$\forall a_t \in A$	Behavior from a set of all possible behaviors
$\alpha(s_t, a_t)$	Learning rate
r_{t+1}	Expected reward obtained in state $s_{f,t}$, when selected behavior $a_{f,t}$, in state $s_{f,t}$

Here the learning rate and the discount rate range from 0 including 1.

Q-learning for the freight carriers

In the article two methods of unsupervised learning are compared, namely Monte Carlo method and Q-learning (Watkins and Dayan, 1992).

As example the Q-learning for the freight carriers in article XXX will be discussed. In the case of the freight carriers the expected transport profit is used as the value of action-value function for freight carriers with the following function:

Behavior choice

$$\rightarrow Q_{fc}(s_{fc,t} a_{fc,t}) + \alpha_{fc} [r_{fc,t+1} - \gamma_{fc} * \max (Q(s_{fc,t+1} a_{fc,t+1})) - Q(s_{fc,t} a_{fc,t})] \quad 2$$

with

$$r_{fc,t+1} = fee_t \times n_t - lc_t \quad 3$$

The mentioned parameters are given in the next table.

Table 52: Parameters needed for Q-learning for the freight carriers.

Parameter	Meaning
$Q_{fc}(s_{fc,t}a_{fc,t})$	Expected total transport profit obtained from state $s_{f,t}$, to the last state when freight carrier selected the behavior $a_{f,t}$, in state $s_{f,t}$
$\forall s_{fc,t} \in S_{fc}$	State from a set of all states for the freight carrier
$\forall a_{fc,t} \in A_{fc}$	Behavior from a set of all possible behaviors for the freight carrier
α_{fc}	Learning rate for freight carrier
$r_{fc,t+1}$	Actual transport profit obtained in state $s_{f,t}$, when freight carrier selected behavior $a_{f,t}$, in state $s_{f,t}$
γ_{fc}	Discount rate for freight carrier
fee_t	Charge that freight carrier offered to shippers in state $s_{f,t}$
n_t	Number of obtained shippers when freight carrier selected behavior $a_{f,t}$, in state $s_{f,t}$
lc_t	Transport cost when freight carrier selected behavior $a_{f,t}$, in state $s_{f,t}$

I Netlogo code

```

;; generate breeds
extensions [table]
globals
[ roads      ;; agentset of all road patches
  shops      ;; agentset of all shop patches
  crossings  ;; agentset of all crossing patches
  node1 node2
  i j m imax jmax  ;; counter for cheapcalc level 2 variable
  temp temp1
  tt
  totalfitness
  costvar overweighttemp
  day UDCcheck
  time
  nrgoods
  phase phasenr daytime
  xtime0 xtime1 xtime2 xtime3 ytime1 ytime2 ytime3
]

;; major agents
breed [municipalities muni]      ;; municipality agent
breed [FCterminals FCterminal]   ;; terminal breed

;; minor agents
breed [nodes node]              ;; generate breed nodes
breed [trucks truck]            ;; truck breed
undirected-link-breed [streets street] ;; streets
breed [cars car]                ;; regular other traffic

;; properties
trucks-own
[ dirx diry status speed good_At tollt kmcount costs ] ;; the shops need goods
patches-own
[ NOx maxspeed realspeed good_A toll speednow ] ;;
streets-own [afstand linktime linkcost] ;;
FCterminals-own
[ UDC demand bestroute bestroutecots loadfactor demandsize
  chromosome newchromosome lengthchr lowcosts bestchromosome costs fitness varcost dist crosspoint crosslength
  demandtable numchromo generation elites crossoverrate mutationrate
  kmcounter drivingcosts
]
nodes-own [netw]
municipalities-own [congestion NOxemission]
cars-own [speed]

```

```

.....
;;;; Setup
.....

```

```

to setup                                ;; setup level 0
clear-all
set nrgoods 0 set vacost (1175 / 4)
set day 0 set time 0 set phasenr 14 set daytime 168 * 2
if (daytime != 168 * 2) [write "problem with phases and road input, please check"]
ifelse (deliverytype = "Early UDC delivery")
    [set start true set choice false ]
    [ifelse (deliverytype = "Fixed time UDC delivery")
        [set start true set choice true]
        [set start false set choice true]]
ifelse (tollsetting = "No toll")
    [set toll 0 ]
    [ifelse (tollsetting = "Low toll")
        [set toll 400 ]
        [set toll 800 ]]

ifelse dynamicfee
    [set fee [60 70 80 80 80 80 90 100 120 150 180 200 200 200]
    set fee map [ ? + feesetting] fee ]
    [set fee array:from-list n-values phasenr [feeprice]
    set fee array:to-list fee]

generate-network
set-default-shape trucks "truck"
ask FCterminals [
    set chromosome table:make set newchromosome table:make set totalfitness table:make
    set costs table:make table:put costs "totlen" 0 table:put costs "ctovld" 0
    set i 0 set fitness table:make set varcost table:make set dist table:make
    while [i < numchromo]
        [table:put fitness (i) 0 table:put varcost (i) 0 table:put dist (i) 0 set i (i + 1)]    ]

ask one-of fcterminals with [UDC] [set-current-plot "Fitnessplot"
set-plot-x-range 0 generation
set-plot-y-range (1300 * 3 * numchromo) (2000 * 3 * numchromo)]
end

to make-movie
user-message "First, save your new movie file (choose a name ending with .mov)" ;; prompt user for movie location
let path user-new-file
if not is-string? path [stop] ;; stop if user canceled

setup                                ;; run the model
movie-start path
movie-grab-view
while [ticks < 500] [go movie-grab-view]

movie-close
user-message (word "Exported movie to " path)
end

```

```

.....
;;;; Main routine
.....
to go
;; go level 0

if (time = 0)
;; Start of new day for FCterminals
[filewriting
ask municipalities [regulations]
updatestreets print "streets"
;; first update current speed per link
cheapathcalc print "calc"
;; calculate the cheapest path from a to b
; set-current-plot "UDCusage"
; plotxy fee ((sum [demandsize] of fcterminals with [color = 77]) / (sum [demandsize] of fcterminals with [color != 77]))
set-current-plot "Fitnessplot"
ask FCterminals [
if (not UDC)
[generate-demand
set demandsize (length demand)
GeneticAlgorithm
]]

set day (day + 1)]

set phase (floor (time / (daytime / phasenr)))

if (time = round (daytime / 7.5)) [ask FCterminals
;; new deliveries for the UDC
[if ((UDC) and (length demand > 7)) [UDCGA] ] ]
; export-plot "Bestroutrcosts" (word "c:/Users/Others/Desktop/Antal/Netlogotests/100813/UDCGAparameters" demandsize ".csv")

ask fcterminals [if (UDC and length demand > 6) [UDCGA]]
; if (time = 190) [ask fcterminals [if ((UDC) and (length demand > 0)) [UDCGA]]]
if (time = 240) [ask fcterminals [if ((UDC) and (length demand > 0)) [UDCGA]]]
if (time = 290) [ask fcterminals [if ((UDC) and (length demand > 0)) [write "undelivered goods"]]]

set phase (floor (time / (daytime / phasenr)))
;; regular delivering
ask fcterminals [if (not UDC and (length bestroute > 1) and (item 0 first bestroute > (-1 * time))) [generate-truck]]
ask trucks [set kmcount (kmcount + speed)
fd speed
if ([pcolor] of patch-here = brown + 3) [fd (-1 * speed) set status 1 move-truck]
setspeed ([item phase realspeed] of patch-here)
set costs (costs + (1175 / 4) + [item phase toll] of patch-here)
]
NOxspeedcalculation
move-truck
;; set direction for trucks

if (ticks mod daytime = 329) [
; print "time 329"
ask municipalities [regulations]]
; print "day"
; print [demandsize] of FCterminal 63

```

```

;          print [kmcounter] of fcterminal 60
;          print [kmcounter] of FCterminal 61
;          print [kmcounter] of FCterminal 62
;          print [kmcounter] of FCterminal 63]

;; check for finishing
if ticks > 5000 [stop]
tick
set time (ticks mod daytime)
end

to UDCGA
set demandsize length demand
set nrgoods (nrgoods + demandsize) if (demandsize = 1) [set demandsize 2]
if (demandsize > 1) [
    set numchromo first table:get demandtable demandsize
    set generation item 1 table:get demandtable demandsize set elites item 2 table:get demandtable demandsize
    set crossoverrate item 3 table:get demandtable demandsize set mutationrate item 4 table:get demandtable demandsize]
ifelse (demandsize > 1) [GeneticAlgorithm] [set bestroute list -1 false]
while [length bestroute > 1] [generate-truck]
set demand []
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
;;;; Subroutines
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

to move-truck          ;; level 1
ask trucks [
    ifelse (status = 1) ;status 1 in a normal road ahead
        [if (([patch-at 0 1 = patch first dirx first diry] or ([patch-at 0 -1 = patch first dirx first diry] or
            ([patch-at 1 0 = patch first dirx first diry] or ([patch-at -1 0 = patch first dirx first diry])) [deliver]
        ifelse ([pcolor] of patch-ahead speed = blue + 3)
            [] ; is the next patch a road?
            [ifelse ([pcolor] of patch-ahead speed = black) [set heading (-1 * heading)]
                [if (not (([pcolor] of patch-ahead speed = blue + 3) xor ([pcolor] of patch-ahead speed = red + 3)))
                    [face one-of (patches in-radius 2 with [pcolor = red + 3]) set heading 90 * (round (heading / 90)) ;else (abs (first dirx
                    - round (xcor)) < abs (first diry - round (ycor)))
                    if (distancexy first dirx first diry < distancexy ([pxcor] of patch-ahead speed) (first diry)) [set heading (180 + heading)] ] ]
        ; is the next patch a shop/ terminal? yes, please change direction
    ]
    if ([pcolor] of patch-ahead speed = red + 3) [set status 2] ; is the next patch a crossing?
    ]
    [ifelse status = 2
        ;status 2 means crossing ahead
        [face patch (first dirx) (first diry)
            ifelse ((distancexy (first dirx) (ycor)) > 3 and (distancexy (xcor) (first diry) > 3)) ;; if goal is far away
                ; [set heading (round (heading / 90) * 90)] ; way to travel next closest
                [set temp ((ceiling (heading / 90)) * 90) set temp1 ((floor (heading / 90)) * 90)
                    ask patch-at-heading-and-distance temp 1 [set xtime0 [item phase realspeed] of self]

```

```

ask patch-at-heading-and-distance temp1 1 [set xtime1 [item phase realspeed] of self]
ifelse (xtime0 > xtime1)[set heading temp][set heading temp1]
]
[ifelse (distancexy (first dirx) (ycor) > distancexy (xcor) (first diry))
  [ifelse (round (heading / 90) = 1) [set heading 90] [set heading 270]]
  [ifelse (round (heading / 180) = 1) [set heading 180] [set heading 0]]
]
ifelse ([pcolor] of patch-ahead speed = red + 3) [set status 2] [set status 1]
[print "Error undefined status"]
]
if ([pcolor] of patch-ahead speed = black) [set heading (-1 * heading)]
]
end

to deliver ;; procedure to deliver the goods and set the next destination level 2
ifelse ([pcolor] of (patch first dirx first diry) != [color] of self)
  [ifelse ((first dirx = -20) and (first diry = 3)) ;; transfer the goods to the UDC or normal delivery?
    [set dirx butfirst dirx set diry butfirst diry ;; transfer the goods to the UDC
    ; ask FCterminals with [color = [color] of myself] [print demand]
    ask FCterminals in-radius 1.6 [
      while (((length dirx) of myself) > 1)
        [set demand fput (list [first dirx] of myself [first diry] of myself 100) demand
        ask myself [set dirx butfirst dirx set diry butfirst diry]]]
    ;ask fcterminals [if UDC [print demand]]]
    [ask patch first dirx first diry [set pcolor (brown + 3) set good_A 0] ;; normal delivery
    set dirx but-first dirx set diry but-first diry] ; make next customer the goal
    ; if ([pcolor] of (patch first dirx first diry) != [color] of self) [ask (patch first dirx first diry) [set pcolor (green + 3)]] ] ;; upon arrival bak at
the terminal, die if no more deliveries are needed
    [if (length dirx < 2) [ask FCterminals in-radius 1.6 [set kmcounter (kmcounter + [kmcount] of myself)
      set drivingcosts (drivingcosts + [costs] of myself)
      die]]

face patch (first dirx) (first diry) set heading (90 * round (heading / 90))
ifelse ([pcolor] of patch-ahead speed = blue + 3) [] ; is the next patch a road?
  [ifelse ([pcolor] of patch-ahead speed = black) [set heading (-1 * heading)]
    [if (not (((pcolor] of patch-ahead speed = blue + 3) or ([pcolor] of patch-ahead speed = red + 3)))
      [face one-of (patches in-radius 2 with [pcolor = red + 3]) set heading 90 * (round (heading / 90)) ;else (abs (first dirx -
round (xcor)) < abs (first diry - round (ycor)))
      if (distancexy first dirx first diry < distancexy ([pxcor] of patch-ahead speed) (first diry))
        [set heading (180 + heading)] ] ] ;is the next patch a shop/ terminal? yes, please change direction
      if (not (((pcolor] of patch-ahead speed = blue + 3) or ([pcolor] of patch-ahead speed = red + 3))) [write "error no road ahead"]]
set status 1
set speed 0 ;; unloading time
end

to NOx speed calculation ;; calculate the amount of NOx emission by counting trucks level 1
ask roads [set NOx replace-item phase NOx (item phase NOx + count (trucks-here))
  set realspeed replace-item phase realspeed (item phase maxspeed - (0.05 * count (trucks-here)) + ((random 19) / 200))

```

```

        if (item phase realspeed < 0.05) [set realspeed replace-item phase realspeed 0.05]
        ask trucks-here [set tollt (tollt - ([item phase toll] of myself))]
    end

    ;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
    ;;;; Subroutines Cheapest path ;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
    ;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

to generate-truck ;; level 1
    ;; Here the truck is generated with its directions.
    ; print who
    ; print bestroute
    while [(item 1 first bestroute = true) or (item 1 first bestroute = false)] [set UDCcheck item 1 first bestroute set bestroute but-first bestroute if
(length bestroute = 0) [stop]]
    set temp [] set temp1 []
    if (length bestroute = 0) [stop]
    while [not (item 1 first bestroute = true or item 1 first bestroute = false)] [
        set temp fput item 0 first bestroute temp set temp1 fput item 1 first bestroute temp1
        set bestroute but-first bestroute]

    set temp fput pxcor temp set temp1 fput pycor temp1

    ask patch pxcor pycor
    [sprout-trucks 1 [
        set size 0.8 set color [pcolor] of patch-here
        set dirx reverse temp set diry reverse temp1
        set costs 0
        set heading 270 set status 1
        set speed 0.7 set tollt 0 set good_At 0
        ifelse (UDCcheck) [set dirx fput -20 dirx set diry fput 3 diry
            set i 1 set imax length dirx
            while [i < (imax - 1)] [ask patch item i dirx item i diry [set pcolor 78] set i (i + 1)]]
        [set i 0 set imax length dirx
            while [i < (imax - 1)] [ask patch item i dirx item i diry [set pcolor ([pcolor] of myself + 3)] set i (i + 1)]]

    ]]
end

to GeneticAlgorithm ;; calculate the cheapest path between two nodes for all nodes. level 1

    gengeneration ;; make the initial genes needed for the calculation
    set lengthchr (length (table:get chromosome 0))
    set bestroute (list ) set bestroute costs 10000000000
    table:clear total fitness
    set-current-plot "Bestroute costs" plotxy 0 (bestroute costs / 100000) plot-pen-down
    set-current-plot "Fitness plot" plotxy 0 (bestroute costs / 100000 * numchromo) plot-pen-down
    set tt 0
    while [tt < generation]
        [fitnesscalc ;print "fit"]

```

```

        elitechoice
        selection
        crossovermutation
        set-current-plot "Fitnessplot" plot-pen-down
        plotxy tt (table:get totalfitness (tt))
        print tt
        set tt (tt + 1) ]
;print bestroute print 333
UDCdelivery
; print bestroute print 334
bestroute-organisation
; print bestroute print 335
set-current-plot "Fitnessplot" plot-pen-up
set-current-plot "Bestroutecots" plot-pen-up
end

to crossovermutation
set i 0
while [i < numchromo] [      ;; for all chromosomes
    set crosspoint random (lengthchr - 1)
    set crosslength (1 + random ((lengthchr - 1) - crosspoint - 1))
    let g1 table:get chromosome i    let g1first []    set g1first fput (-1 * (item 0 first g1)) g1first    set g1 but-first g1    ;; chromosome i    ;;
    use lists to decrease calculation time by not entering matrixxtimes.
    let g2 table:get chromosome (i + 1) let g2first []    set g2first fput (-1 * (item 0 first g2)) g2first    set g2 but-first g2    ;; chromosome i + 1
    let g1new g1 let g2new g2      ;; make newchromosome

;; crossover part
set j 0 set temp1 0
while [j < (- crosslength - 1)] [      ;; string 1
    if (random 10 < (crossoverrate / 10)) [set m temp1
        while [m < ((lengthchr - 1) - crosslength)] [
            if (item (j) g1 = item (m) g2new )
                [set temp1 (temp1 + 1)
                    ifelse ((temp1 - 1) = m)
                        [set m ((lengthchr - 1) - crosslength)]
                        [set temp (item j g1new)
                            set g1new replace-item j (g1new) (item (m) g1new)
                            set g1new replace-item (m) (g1new) (temp)
                            set m ((lengthchr - 1) - crosslength)
                        ] ]
            set m (m + 1)]
        ]
    set j (j + 1)]

set j 0 set temp1 0
while [j < ((lengthchr - 1) - crosslength)] [      ;; string 1
    if (random 10 < (crossoverrate / 10)) [set m temp1
        while [m < ((lengthchr - 1) - crosslength)] [
            if (item (j) g2 = item (m) g1new )

```

```

        [set temp1 (temp1 + 1)
        ifelse ((temp1 - 1) = m)
            [set m ((lengthchr - 1) - crosslength)]
            [set temp (item j g2new)
            set g2new replace-item j (g2new) (item (m) g2new)
            set g2new replace-item (m) (g2new) (temp)
            set m ((lengthchr - 1) - crosslength)
        ] ]
        set m (m + 1)]
    ]
    set j (j + 1)]

set g1 g1new set g2 g2new set g1new [] set g2new []

;; mutation part
if (random 100 < mutationrate)
    [let ex1 random (lengthchr - 1) let ex2 random (lengthchr - 1)
    while [item ex1 g1 = 0] [set ex1 random (lengthchr - 1)]
    while [item ex2 g1 = 0] [set ex2 random (lengthchr - 1)]
    if (ex1 <= ex2) [let swop ex1 set ex1 ex2 set ex2 swop
        if (ex1 = ex2) [ifelse (ex1 != 0) [set ex2 (ex2 - 1)] [set ex1 (ex1 + 1)]] ; assure that the value are not identical and organize
    set g1 reverse g1
    set temp item (ex2) g1 set j 0
    while [j < (lengthchr - 1)]
        [ifelse ((j = ex1) or (j = ex2))
            [ifelse (j = ex2)
                [set g1 butfirst g1 ]
                [set g1new fput (first g1) g1new
                set g1new fput (temp) g1new
                set g1 butfirst g1
            ] ]
        [set g1new fput (first g1) g1new
        set g1 butfirst g1]
    set j (j + 1)
    ]
    set g1 g1new
    ]
if (random 100 < mutationrate)
    [let ex1 random (lengthchr - 1) let ex2 random (lengthchr - 1)
    while [item ex1 g2 = 0] [set ex1 random (lengthchr - 1)]
    while [item ex2 g2 = 0] [set ex2 random (lengthchr - 1)]
    if (ex1 <= ex2) [let swop ex1 set ex1 ex2 set ex2 swop
        if (ex1 = ex2) [ifelse (ex1 != 0) [set ex2 (ex2 - 1)] [set ex1 (ex1 + 1)]] ; assure that the value are not identical and organize
    set g2 reverse g2
    set temp item (ex2) g2
    set j 0
    while [j < (lengthchr - 1)]
        [ifelse ((j = ex1) or (j = ex2))
            [ifelse (j = ex2)
                [set g2 butfirst g2]
                [set g2new fput (first g2) g2new

```

```

        set g2new fput (temp) g2new
        set g2 butfirst g2
      ] ]
      [set g2new fput (first g2) g2new
        set g2 butfirst g2]
      set j (j + 1)
    ]

    set g2 g2new
  ]

; set g1 fput (list 0 false) g1  set g2 fput (list 0 false) g2

;; time mutation
ifelse (not UDC) [
  set j 0
  while [j < (lengthchr - 1)]
    [if (item 1 item j g1 = false) [set g1first lput (-1 * (item 0 item j g1)) g1first] set j (j + 1)]
  set j 0
  while [j < (lengthchr - 1)]
    [if (item 1 item j g2 = false) [set g2first lput (-1 * (item 0 item j g2)) g2first] set j (j + 1)]
  set j 0
  while [j < length g1first]
    [if (random 100 < mutationrate)
      [set g1first replace-item j g1first ((item j g1first + (random (daytime / 7) * ((2 * random 2) - 1))) mod daytime)]
      set j (j + 1)]

  set j 0
  while [j < length g2first]
    [if (random 100 < mutationrate)
      [set g2first replace-item j g2first ((item j g2first + (random (daytime / 7) * ((2 * random 2) - 1))) mod daytime)]
      set j (j + 1)]

  if (random 100 < (crossoverrate) and (length g2first > 1))
    [let k 1 let l random length g2first
      while [k = l][set l random (length g2first)]
      set temp item k g1first set temp1 item k g2first
      set g1first replace-item k g1first item l g1first
      set g1first replace-item l g1first temp
      set g2first replace-item k g2first item l g2first
      set g2first replace-item l g2first temp1
    ]

  set g1 fput (list (-1 * first g1first) false) g1  set g2 fput (list (-1 * first g2first) false) g2
  set g1first butfirst g1first      set g2first butfirst g2first
  set j 1
  while [j < (lengthchr)]
    [if (item 1 item j g1 = false) [set g1 replace-item j g1 (list (-1 * first g1first) false) set g1first butfirst g1first] set j (j + 1)]
  set j 1

```

```

while [j < (lengthchr)]
  [if (item 1 item j g2 = false) [set g2 replace-item j g2 (list (-1 * first g2first) false) set g2first butfirst g2first] set j (j + 1)]
if (length g1first > 0 or length g2first > 0) [write "error g1/g2first is not 0"]
] [set g1 fput (list (-1 * first g1first) false) g1 set g2 fput (list (-1 * first g2first) false) g2]

table:put newchromosome i g1
table:put newchromosome (i + 1) g2
set i (i + 2)] ; go to the next pair of chromosomes

set chromosome table:make
foreach sort table:keys newchromosome [table:put chromosome ? table:get newchromosome ?]
set newchromosome table:make

end

to selection
  set j (int (elites / 100 * numchromo) - 1)    ;; select non-elite genes
  if (j <= 0) [set j 0]
  set chromosome shuffle table:to-list chromosome

  while [j < numchromo] [
    table:put newchromosome j (item 1 (first chromosome))
    set chromosome butfirst chromosome
    set j (j + 1)]

  set newchromosome table:to-list newchromosome
  set newchromosome shuffle newchromosome    ;; shuffle the chromosomes

  set chromosome table:make
  set i 0 while [i < numchromo] [table:put chromosome i (item 1 first newchromosome) set newchromosome butfirst newchromosome set
i (i + 1)]
  set newchromosome table:make

end

to elitechoice

set varcost sort-by [item 1 ?1 < item 1 ?2] table:to-list varcost
set i 0 set imax (int (elites / 100 * numchromo))
while [i < imax] [
  table:put newchromosome i (table:get chromosome (item 0 item 0 varcost))
  set varcost but-first varcost
  set i (i + 1) ]
set varcost table:from-list varcost
end

to fitnesscalc
  let loc1 0 let loc2 0 let totfit 0 let overload 0

```

```

table:put costs "ctovld" 0 set fitness table:make set varcost table:make set dist table:make
set i 0 set j 0

;; include fixed costs for the trucks
while [j < lengthchr - 1] [if (item 1 item j table:get chromosome 0 = false or item 1 item j table:get chromosome 0 = true) [set i (i + 1)] set j (j + 1)]
table:put costs "fixedc" (i * 11799)

set i 0
while [i < numchromo]
  [set j 0 let weight 0 set overload 0 set temp 0 set costvar 0 set loc2 (list pxcor pycor) set node2 node 31
  let UDCcount 0 let costvartemp 0 let hour 0 ;; overweight calculation + travel costs calculation
  ; print chromosome
  while [j < lengthchr] [set loc1 loc2 set node1 node2
  ifelse (item 1 item j table:get chromosome i = false)
    [
      if (weight > (100 * loadfactor)) [set overload (overload + 200000 * (weight - (100 * loadfactor)))]
      set weight 0
      set hour ((item 0 item j table:get chromosome i) * -1)
      set phase floor (hour / (daytime / phasenr)) if (phase >= phasenr) [set phase (phasenr - 1)]
      set loc2 (list pxcor pycor) ]
    [set loc2 (item j table:get chromosome i)
    set weight (weight + item 2 (item j (table:get chromosome i))) ]
  ask patch (first loc2) (item 1 loc2) [set node2 (nodes in-radius 2)]
  ask node2 [set temp1 who]
  ask node1 [set hour (hour + (item 1 item phase table:get netw temp1))
    set phase floor (hour / (daytime / phasenr)) if (phase >= (phasenr - 4)) [if (phase >= (phasenr - 1)) [set phase (phasenr - 1)] set
costvar costvar + 10000000 ]
    set costvar (costvar + ((1175 / 4) * (item 0 item phase table:get netw temp1)))
  ]
  ask node2 [set hour (hour + (([item phase realspeed] of patch - here) * 3 / 4))
    set phase floor (hour / (daytime / phasenr)) if (phase >= (phasenr - 4)) [if (phase >= (phasenr - 1)) [set phase (phasenr - 1)] set
costvar costvar + 10000000 ]
    set costvar (costvar + ((3 / 4) * (1175 / 4) * ((1 / ([item phase realspeed] of patch - here)) + ([item phase toll] of patch - here))))
  ]
  set j (j + 1)]

  if (weight > (100 * loadfactor)) [set overload (overload + 200000 * (weight - (100 * loadfactor)))]

  set node1 node2 set node2 (nodes in-radius 2) set temp1 21 ;; last node
  ask node1 [set hour (hour + item 1 item phase table:get netw temp1))
    set phase floor (hour / (daytime / phasenr)) if (phase >= (phasenr - 4)) [if (phase >= (phasenr - 1)) [set phase (phasenr - 1)] set
costvar costvar + 10000000 ]
    set costvar (costvar + ((1175 / 4) * (item 0 item phase table:get netw temp1))) ]
  table:put costs "ctovld" overload
  table:put fitness (i) (costvar + overload)
  ; print "chromosomecheck" print i print (costvar + table:get costs "fixedc" + table:get costs "ctovld") print table:get chromosome i
  table:put varcost (i) (costvar + overload)

  set i (i + 1)]

```

```

;; total fitness calculation and route selection
set i 0 set temp [] while [i < numchromo] [set temp lput (table:get fitness i) temp set i (i + 1)]
set lowcosts (precision (min (temp)) 2) set temp1 (position (min temp) temp)
set totfit sum temp table:put totalfitness (tt) (totfit)

if (lowcosts < bestroute costs) [
  ; print "checkbestroute" print tt print lowcosts print temp1 print table:get chromosome temp1
  set bestroute table:get chromosome temp1 set bestroute costs lowcosts
  if (item 1 first bestroute != false) [write "error no first truck" set bestroute fput (list 0 false) bestroute]]

set-current-plot "Bestroute costs"
plotxy tt bestroute costs
; set j 0 while [j < numchromo] [write table:get fitness j print table:get chromosome j set j j + 1]
end

to UDCdelivery
; print bestroute
if (not UDC) [set bestroute lput (list 0 false) bestroute]
  let loc2 (list pxcor pycor)

  ask patch (first loc2) (item 1 loc2) [set node2 (nodes in-radius 2)]
  ask node2 [set temp1 who] set temp temp1
  set j 0 let UDCcount 0 set costvar 0 let hour 0

  while [j < length bestroute] [
    set node1 node2
    ifelse ((item 1 item j bestroute = false))
      [set hour ((item 0 item j bestroute) * -1)
       set phase floor (hour / (daytime / phasenr)) if (phase >= phasenr) [set phase (phasenr - 1)]
       set loc2 (list pxcor pycor) ]
      [set loc2 (item j bestroute)
       set UDCcount UDCcount + 1 ]

    ask patch (first loc2) (item 1 loc2) [set node2 (nodes in-radius 2)]
    ask node2 [set temp1 who]
    ask node1 [set hour (hour + (item 1 item phase table:get netw temp1))
               set phase floor (hour / (daytime / phasenr)) if (phase >= phasenr) [set phase (phasenr - 1) set costvar (costvar + 200000)]
               set costvar (costvar + ((1175 / 4) * (item 0 item phase table:get netw temp1)))]

    if ((item 1 item j bestroute = false) and (j > 0)) ;; check for cheaper route
    [set phase 0
     if ((costvar > ((UDCcount * fee * 100) + ((1175 / 4) * 2 * (((item 0 item phase table:get netw temp) of node 20) + (((1 / item phase
realspeed) + item phase toll)) of patch -21 3))))))
       [set j (j - UDCcount - 1)
        let phases []
        set hour ((item 0 item j bestroute) * -1)
        set phase floor (hour / (daytime / phasenr)) if (phase >= phasenr) [set phase (phasenr - 1)]
        if (phase < 2) [if (phase = 0) [set phases (list 0 1 2)] if (phase = 1) [set phases (list 0 1 2 3)]]
        if (phase > 1 and phase < 7) [set phases (list (phase - 2) (phase - 1) (phase) (phase + 1) (phase + 2))]]

```

```

    if(phase > 7)[set phases (list 5 6 7)]

    set temp10 let temp11 []
    while [temp1 < length phases][set phase item temp1 phases
        set temp11 lput ((UDCcount * fee * 100) + ((1175 / 4) * 2 * (((item 0 item phase table:get netw temp) of node 20) + (((1 /
item phase realspeed) + item phase toll)) of patch -21 3)))) temp11
        set temp1 temp1 + 1
    ]

    let temp2 (position (min temp11) temp11)
    set temp11 item temp2 phases
    set temp11 -1 * (round (temp11 * 168 * 2 / 14) + random (12))
    set bestroute replace-item j bestroute (list temp11 true)
    set j (j + UDCcount + 1)
    ]
    set costvar 0
    if((item 1 item j bestroute = false)) [set UDCcount 0]
    set j (j + 1)] ]
; print bestroute
end

```

to gengeneration

```

set i 0 table:clear chromosome table:clear newchromosome set newchromosome table:make
while [i < numchromo] [table:put newchromosome i (0) set i (i + 1)]
set i 0 set chromosome table:make
while [i < numchromo]
    [set temp shuffle demand set temp1 []
    ifelse UDC [set temp1 fput (list (-1 * (ticks + 1 + random 10)) false) temp1] [set temp1 fput (list (-1 * random daytime) false) temp1]
    set j 0 set jmax ((length temp - length temp mod loadfactor) / loadfactor)
    while [j < jmax]
        [let k 0
        while [k < loadfactor] [set temp1 fput (first temp) temp1 set temp but-first temp set k (k + 1)]
        ifelse UDC [set temp1 fput (list (-1 * (ticks + 1 + random 10)) false) temp1]
            [set temp1 fput (list (-1 * random daytime) false) temp1]
        set j (j + 1)]
    while [length temp > 0] [set temp1 fput (first temp) temp1 set temp but-first temp]
    table:put chromosome (i) (reverse temp1)
    set i (i + 1)]
end

```

```

to cheappathcalc ;; Dijkstra algorithm
set j count nodes

```

```

ask nodes [set i 0 set netw table:make ;; prepare for search algorithm
    while [i < j] [table:put netw (i) [] set phase 0

```

```

        while [phase < phasenr] [table:put netw (i) fput (list 3000000 0 3) table:get netw (i) set phase (phase + 1)]
        set i (i + 1)]
    table:put netw (who) [] set phase 0 while [phase < phasenr] [table:put netw (who) fput (list 0 0 1) table:get netw (who) set phase (phase + 1)]]

ask nodes [set phase 0 let k who let starttime 0
while [phase < phasenr] [let itera 0 let linkend -5 let searchnode who
while [itera < (j - 1)]
[ask node searchnode [ask my-links [ask both-ends
[if (who != searchnode)
[set linkend who
ask node k
[if ((item 2 item phase table:get netw (linkend)) != 1)
[set starttime (phase + floor (item 1 item phase table:get netw (searchnode) / (daytime / phasenr))]
ifelse (starttime < phasenr)
[set temp1 (((item starttime linkcost) of street searchnode linkend) + (item 1 item phase table:get netw (searchnode)
))]
set temp (((item starttime linktime) of street searchnode linkend) + (item 0 item phase table:get netw (searchnode)
))]

[set starttime (phasenr - 1) set temp 200
set temp1 (10000 + ((item starttime linkcost) of street searchnode linkend) + (item 0 item phase table:get netw
(searchnode) ))]

if ((item 0 item phase table:get netw (linkend)) > temp1 )
[table:put netw (linkend) replace-item phase (table:get netw linkend) (list temp1 temp 2)]]
]
]]]
;; find the next node for searching cheap path
set temp 500000000 set i 0 let temp2 -1
while [i < j ]
[ask node k
[if (((item 2 item phase table:get netw (i)) = 2) and ((item 0 item phase table:get netw (i)) < temp) )
[set temp (item 0 item phase table:get netw (i)) set temp2 i ] ]
set i (i + 1)]
ask node k [table:put netw (temp2) replace-item phase (table:get netw temp2) (list (item 0 item phase table:get netw (temp2)) (item 1
item phase table:get netw (temp2)) 1)]
set searchnode temp2
set itera (itera + 1) ]
set phase (phase + 1)
] ]
; ask node 34 [set i 0 while [i < 60] [set temp item 0 table:get netw i ask node i [set label precision temp 2] set i i + 1]]
end

```

```

to updatestreets ;; level 2
ask roads [set NOx [] set i 0 while [i < phasenr] [set NOx fput 0 NOx set i (i + 1)]]
ask streets [set a fstand 3 set linktime [] set linkcost [] set i 0 ]
set imax(count patches with [(pcolor = red + 3) and (pycor = max-pycor - 1)])
set jmax(count patches with [(pcolor = red + 3) and (pxcor = (max-pxcor - 2))])
set xtime00 set xtime1 0 set xtime2 0 set xtime3 0 set ytime1 0 set ytime2 0 set ytime3 0

```

```

set phase (phasenr - 1)
while [phase >= 0] [set i -1 set j 0
  while [j < (jmax - 1)]
    [set i (i + 1)
      while [(i mod imax) < (imax - 1)]
        [ask node i [
          xside yside ; calculate horizontal links and vertical link speed from node i
          askstreet i (i + 1) [set linktime fput ((xtime0 / 2) + xtime1 + xtime2 + (xtime3 / 2)) linktime]
          askstreet i (i + imax) [set linktime fput ((xtime0 / 2) + ytime1 + ytime2 + (ytime3 / 2)) linktime]
          xsidecost ysidecost
          askstreet i (i + 1) [set linkcost fput ((xtime0 / 2) + xtime1 + xtime2 + (xtime3 / 2)) linkcost]
          askstreet i (i + imax) [set linkcost fput ((xtime0 / 2) + ytime1 + ytime2 + (ytime3 / 2)) linkcost]
        ]
        set i (i + 1) ]
      set j (j + 1) ]

set i (i + 1) ;; now for the last row
while [(i mod imax) < (imax - 1)]
  [ask node i [ ; calculate horizontal links and vertical linkspeed from node i
    xside yside
    askstreet i (i + 1) [set linktime fput ((xtime0 / 2) + xtime1 + xtime2 + (xtime3 / 2)) linktime]
    askstreet i (i - (imax * (jmax - 1))) [set linktime fput ((xtime0 / 2) + ytime1 + ytime2 + (ytime3 / 2)) linktime]
    xsidecost ysidecost
    askstreet i (i + 1) [set linkcost fput ((xtime0 / 2) + xtime1 + xtime2 + (xtime3 / 2)) linkcost]
    askstreet i (i - (imax * (jmax - 1))) [set linkcost fput ((xtime0 / 2) + ytime1 + ytime2 + (ytime3 / 2)) linkcost] ]
  set i (i + 1) ]

set i 0 ;(imax - 1) ;; now for the right side
while [i < jmax - 1]
  [ask node (i * imax + imax - 1) [
    yside
    ask patch-here [set xtime0 (1 / item phase realspeed)]
    askstreet (i * imax + imax - 1) (i * imax + 2 * imax - 1) [set linktime fput ((xtime0 / 2) + ytime1 + ytime2 + (ytime3 / 2)) linktime]
    ysidecost
    ask patch-here [set xtime0 ((1 / item phase realspeed) + item phase toll)]
    askstreet (i * imax + imax - 1) (i * imax + 2 * imax - 1) [set linkcost fput ((xtime0 / 2) + ytime1 + ytime2 + (ytime3 / 2)) linkcost] ]
  set i (i + 1) ]

ask node ((jmax) * imax - 1) [ ; last street of network
  ask patch-here [set xtime0 (1 / item phase realspeed)] ; calculate horizontal links and vertical link speed from node i
  yside
  askstreet ((jmax) * imax - 1) (imax - 1) [set linktime fput ((xtime0 / 2) + ytime1 + ytime2 + (ytime3 / 2)) linktime]
  ask patch-here [set xtime0 ((1 / item phase realspeed) + item phase toll)] ; calculate horizontal links and vertical linkspeed from node i
  ysidecost
  askstreet ((jmax) * imax - 1) (imax - 1) [set linkcost fput ((xtime0 / 2) + ytime1 + ytime2 + (ytime3 / 2)) linkcost] ]
set phase (phase - 1)]

;ask streets [show-linkset label precision linktime 2]
end

```

```

to xside
  ask patch-here [set xtime0 (1 / item phase realspeed)]
  ask patch-at 2 0 [set xtime2 (1 / item phase realspeed)]
  ask patch-at 1 0 [set xtime1 (1 / item phase realspeed)]
  ask patch-at 3 0 [set xtime3 (1 / item phase realspeed)]
end

to xsidecost
  ask patch-here [set xtime0 ((1 / item phase realspeed) + item phase toll)]
  ask patch-at 2 0 [set xtime2 ((1 / item phase realspeed) + item phase toll)]
  ask patch-at 1 0 [set xtime1 ((1 / item phase realspeed) + item phase toll)]
  ask patch-at 3 0 [set xtime3 ((1 / item phase realspeed) + item phase toll)]
end

to yside
  ask patch-at 0 -1 [set ytime1 (1 / item phase realspeed)]
  ask patch-at 0 -3 [set ytime3 (1 / item phase realspeed)]
  ask patch-at 0 -2 [set ytime2 (1 / item phase realspeed)]
end

to ysidecost
  ask patch-at 0 -1 [set ytime1 ((1 / item phase realspeed) + item phase toll)]
  ask patch-at 0 -3 [set ytime3 ((1 / item phase realspeed) + item phase toll)]
  ask patch-at 0 -2 [set ytime2 ((1 / item phase realspeed) + item phase toll)]
end

to bestroute-organisation
  set j 0 set i 0
  if ((item 1 last bestroute = false) or (item 1 last bestroute = true)) [set bestroute butlast bestroute]
; while [j < length bestroute]
;   [if (item 1 item j bestroute = true) [set bestroute replace-item j bestroute (list i true) set i (i - 1)]
;   set j (j + 1)]
  set j 1 set temp [] set temp1 []
  set temp fput first bestroute temp
  while [j < length bestroute]
    [ifelse ((item 1 item j bestroute = true) or (item 1 item j bestroute = false))
      [set temp1 fput temp temp1
        set temp [] set temp fput item j bestroute temp ]
      [set temp lput item j bestroute temp]
    set j (j + 1)]
  set temp1 lput temp temp1
  set temp sort-by [item 0 first ?1 > item 0 first ?2 ] temp1
  set j 0 set temp1 length temp set bestroute []
  while [j < temp1] [let k first temp set temp butfirst temp
    while [length k > 0] [set bestroute fput first k bestroute set k butfirst k ]
    set j (j + 1)]
  set bestroute reverse bestroute
  set bestroute lput (list 0 false) bestroute
; print bestroute
end

to generate-demand ;; Generate the demand of the shops (currently random, later it will be purposeful) level 2
  ;; size of demand
; set demand [[1 1 100] [-5 7 100] [-7 -3 100] [-25 3 100] [-11 -3 100] [-20 -2 100] [-8 9 100] [-2 -5 100] [-10 -5 100] [-22 7 100] [-14 3 100] [-5 1
100] [-17 9 100]]; [-25 7 100] [-14 -3 100] [-20 -5 100] [-8 3 100] [-4 1 100] [2 6 100] [-22 4 100]]; [-25 -8 100] [-16 -6 100] [-11 6 100] [-17 1
100]]
; set temp demand

```

```

; while [length temp > 0]
; [ask patch (item 0 first temp) (item 1 first temp) [set pcolor brown] set temp butfirst temp]

; real one
;write "demand generation"
  if UDC [set demand []]
  let xint (10)
  set demand []
  ask n-of xshops [set good_A -100 set pcolor brown
    ask myself [set demand fput (list ([pxcor] of myself) ([pycor] of myself) (100)) demand]]
end

to regulations
  set temp [0 0 0 0 0 0 0 0 0 0 0 0]
  set phase 0 while [phase < phasenr][

; ask roads [set temp fput item phase realspeed temp]
; ask municipalities [set congestion mean temp]
  ask roads [set temp replace-item phase temp (item phase NOx + item phase temp)]
  set phase (phase + 1)]
  set NOx emission temp ;print NOx emission
end

to filewriting

end

.....
;;;; Subroutines Set-up                               ;;;;
.....

to generate-network          ;; set up the network with edges
  let grid-x 3    let grid-y 3    ;; the amount of patches in between two roads in the x/y direction
  ask patches [set pcolor brown + 3]
  ask patches with-max [pxcor] [set pcolor black]      ;; these are the edges of the world
  ask patches with-min [pxcor] [set pcolor black]

  set roads patches with [(pcolor = brown + 3)          ;; make the roads for trucks to travel on
    and ((floor((pxcor + max-pxcor - floor(grid-x - 1)) mod grid-x) = 0)
    or (floor((pycor + max-pycor + 1) mod grid-y) = 0))]
  ask roads [set pcolor blue + 3 set NOx [] set toll [] set maxspeed [] set realspeed []
    ifelse (abs(pxcor) > 8 or abs(pycor) > 8)
      [set i 0 while [i < phasenr] [set NOx fput 0 NOx set toll fput 0 toll set i (i + 1)]
        set maxspeed [1 0.57 0.44 0.50 0.57 0.67 1 0.8 0.8 0.67 0.57 0.57 0.67 1]
        set realspeed [1 0.57 0.44 0.5 0.57 0.67 1 0.8 0.8 0.67 0.57 0.57 0.67 1]
        set maxspeed map [ ? / 1.1 ] maxspeed
        set realspeed map [ ? / 1.1 ] realspeed
      ]
      [set i 0 while [i < phasenr] [set NOx fput 0 NOx set toll fput 5 toll set i (i + 1)]
        set maxspeed [0.7 0.4 0.31 0.35 0.40 0.47 0.7 0.56 0.56 0.47 0.4 0.4 0.47 0.7]

```

```

set realspeed [0.7 0.4 0.31 0.35 0.4 0.47 0.7 0.56 0.56 0.47 0.4 0.40 0.47 0.7]
set maxspeed map [ ? / 1.1] maxspeed
set realspeed map [ ? / 1.1] realspeed ] ]

set crossings patches with          ;; make crossings, here the trucks change direction
[(((pcolor = blue + 3)
and (floor((pxcor + max-pxcor - floor(grid-x - 1)) mod grid-x) = 0)
and (floor((pycor + max-pycor + 1) mod grid-y) = 0))]
ask crossings [set pcolor red + 3]

let n 0 foreach sort crossings
[ask ? [set plabel n set n (n + 1) set plabel " " sprout-nodes 1          ;; make nodes to generate links for the vehicle routing calculation
[set size 1 set color red + 3 hide-turtle]]]
ask nodes [create-streets-with other nodes in-radius 3.5] ;; now we have links with 90 and 180 degrees
ask streets [hide-link set linktime [] set linkcost []
set i 0 while [i < phasenr] [set linktime fput 0 linktime set i (i + 1)]]

ask patch (min-pxcor + 3) (max-pycor - 8) [sprout-FCterminals 1
[set UDC false set color green + 3 ask patch-here [set pcolor green] set shape "house" set loadfactor 4 set demandsize 1] ] ;; make FC
terminal
ask patch (min-pxcor + 3) (max-pycor - 11) [sprout-FCterminals 1
[set UDC false set color 85 + 3 ask patch-here [set pcolor 85] set shape "house" set loadfactor 4 set demandsize 1] ] ;; make FC terminal
ask patch (min-pxcor + 3) (max-pycor - 3) [sprout-FCterminals 1
[set UDC false set color 125 + 3 ask patch-here [set pcolor 125] set shape "house" set loadfactor 4 set demandsize 1] ] ;; make FC
terminal

ask patch (min-pxcor + 6) (max-pycor - 6) [sprout-FCterminals 1
[set UDC true set color 74 + 3 ask patch-here [set pcolor 74] set shape "box" set loadfactor 8 set bestroute (list) set demand [] set
demandsize 0] ] ;; make UDC

ask fcterminals [set numchromo 220 set generation 200 set elites 15 set crossoverrate 55 set mutationrate 6 set kmcounter 0]

;; GA parameter setting
ask fcterminals [if (not UDC) [set demandtable table:make table:put demandtable 10 (list 220 200 15 55 6)]] ;; numchromo generations elites
crossoverrate mutationrate
ask fcterminals [if (UDC) [set demandtable table:make
table:put demandtable 2 (list 20 50 10 40 6) table:put demandtable 12 (list 600 700 26 85 7) table:put demandtable 23 (list 1000
1200 24 45 3)
table:put demandtable 3 (list 40 50 16 40 6) table:put demandtable 13 (list 600 700 26 85 7) table:put demandtable 24 (list 1000
1200 24 45 3)
table:put demandtable 4 (list 40 50 16 40 6) table:put demandtable 14 (list 650 800 22 65 9) table:put demandtable 25 (list 1200
1600 22 24 4)
table:put demandtable 5 (list 40 50 16 40 6) table:put demandtable 15 (list 650 800 22 65 9) table:put demandtable 26 (list 1200
1600 22 24 4)
table:put demandtable 6 (list 120 200 25 45 6) table:put demandtable 16 (list 650 800 22 65 9) table:put demandtable 27 (list 1200
1600 22 24 4)
table:put demandtable 7 (list 120 200 25 45 6) table:put demandtable 17 (list 800 900 30 55 7) table:put demandtable 28 (list 1200
1600 22 24 4)
table:put demandtable 8 (list 120 200 25 45 6) table:put demandtable 18 (list 800 900 30 55 7) table:put demandtable 29 (list 1200
1600 22 24 4)

```

```

    table:put demandtable 9 (list 350 450 25 55 6)    table:put demandtable 19 (list 800 900 30 55 7)    table:put demandtable 30 (list 1200
1600 22 24 4)
    table:put demandtable 10 (list 350 450 25 55 6)    table:put demandtable 20 (list 800 900 30 55 7)
    table:put demandtable 11 (list 600 700 26 85 7)    table:put demandtable 21 (list 1000 1200 24 45 3)
                table:put demandtable 22 (list 1000 1200 24 45 3)    ] ]

set shops patches with [pcolor = brown + 3]           ;; the remaining brown patches are shops
ask shops [set realspeed 0 set good_A 0]

ask one-of patches with [(pcolor = black)] [sprout-municipalities 1
[set color yellow set shape "house" set congestion [] set i 0 while [i < phasenr] [set congestion fput 0 congestion set i (i + 1)] ]] ;; make
municipality
end

```