

# Logistic Control In Automated Transportation Networks

Mark Ebben

2001

Ph.D. thesis  
University of Twente



Twente University Press

Also available in print:

[www.tup.utwente.nl/uk/catalogue/management/logistic-control](http://www.tup.utwente.nl/uk/catalogue/management/logistic-control)

# **Logistic Control in Automated Transportation Networks**



Research School for Operations  
Management and Logistics

This thesis is number D-40 of the thesis series of the Beta Research School for Operations Management and Logistics. The Beta Research School is a joint effort of the departments of Technology Management, and Mathematics and Computing Science at the Technische Universiteit Eindhoven and the Centre for Production, Logistics and Operations Management at the University of Twente. Beta is the largest research centre in the Netherlands in the field of operations management in technology-intensive environments. The mission of Beta is to carry out fundamental and applied research on the analysis, design and control of operational processes.

### **Dissertation committee**

Chairman / secretary	Prof.dr. W. van Rossum
Promotor	Prof.dr. A. van Harten
Assistant Promotor	Dr. M.C. van der Heijden
Members	Prof.dr.ir. W.M.P. van der Aalst (Technische Universiteit Eindhoven)
	Prof.dr.ir. M.B.M. de Koster (Erasmus Universiteit Rotterdam)
	Prof.dr.ir. M.F.A.M. van Maarseveen
	Prof.dr. R.A. Stegwee
	Dr.ir. A. Verbraeck (Technische Universiteit Delft)
	Prof.dr. W.H.M. Zijm

### **Acknowledgement**

Herewith we express our gratitude to Connekt (the Dutch knowledge centre for traffic and transport) for their funding of the simulation study that was the basis of our research results. Connekt coordinates a project for the pre-design of an underground logistic system around Amsterdam Airport Schiphol, which was used as a case study in this thesis.



Twente University Press

Publisher: Twente University Press  
P.O. Box 217, 7500 AE Enschede, the Netherlands  
[www.tup.utwente.nl](http://www.tup.utwente.nl)

Cover design: Hana Vinduška, Enschede  
English correction: Lorraine van Dam, Delft  
Print: Grafisch Centrum Twente, Enschede

© M.J.R. Ebben, Enschede, 2001

No part of this work may be reproduced by print, photocopy or any other means without the permission in writing from the publisher.

ISBN 9036515912

# **LOGISTIC CONTROL IN AUTOMATED TRANSPORTATION NETWORKS**

PROEFSCHRIFT

ter verkrijging van  
de graad van doctor aan de Universiteit Twente,  
op gezag van de rector magnificus,  
prof.dr. F.A. van Vught,  
volgens besluit van het College voor Promoties  
in het openbaar te verdedigen  
op vrijdag 8 juni 2001 te 16.45 uur

door

Mark Jacobus Richard Ebben  
geboren op 5 juli 1973  
te Wierden

Dit proefschrift is goedgekeurd door de promotor

prof.dr. A. van Harten

en de assistent-promotor

dr. M.C. van der Heijden

# Preface

This thesis is the result of several years of research. It could not have been completed without the help of many others. A lot of people contributed to the final result, by in-depth discussions, interesting ideas, mental support, jokes or coffee. Therefore, I would like to thank several people for their contribution.

First of all I want to thank my supervisors Aart van Harten and Matthieu van der Heijden, both for the fruitful discussions about the research and the pleasant cooperation within the OLS-project. Furthermore, I enjoyed working together with Noud Gademann and Durk-Jouke van der Zee, discussing specific research topics and writing papers.

Speaking about the OLS-project. This project provided an environment with multiple disciplines, in which a lot of enthusiastic people worked on the latest technological developments. I want to thank Connekt, one of the initiators of the OLS-project, for their funding of the simulation study that was the basis of our research results. It was fun working on such an interesting project. I want to thank all participants in the OLS-project for the pleasant cooperation, in particular the simulation group at Delft University of Technology, Alexander Verbraeck, Yvo Saanen and Edwin Valentin. I would also like to thank the students, Marc Bakhuis, Marcel Brinkman and Gerben Verstegen, whose work contributed to the completion of this thesis.

Besides work there are also other things. I want to thank my colleagues of the OMST department for the pleasant atmosphere and relaxing coffee breaks. In the last period of writing it was of great help to meet PhD students in the same phase of writing. They understand the pressure and obstacles in these last months. Especially I want to mention Frank de Bakker. We both worked many evenings on the last details of our theses, but there was still time for diner or coffee to discuss also other things than research.

Last but certainly not least, I want to thank family and friends for their interest in my research, and even more for the pleasant and enjoyable holidays, trips, weekends, etc.

THANKS!

Enschede, April 2001

Mark Ebben



# Table of contents

<b>PREFACE.....</b>	<b>I</b>
---------------------	----------

<b>TABLE OF CONTENTS.....</b>	<b>III</b>
-------------------------------	------------

<b>1 INTRODUCTION.....</b>	<b>1</b>
----------------------------	----------

1.1	MOTIVATION .....	1
1.1.1.	<i>Need for innovative transport and logistic concepts.....</i>	<i>1</i>
1.1.2.	<i>Dealing with congestion .....</i>	<i>2</i>
1.1.3.	<i>Underground freight transportation .....</i>	<i>3</i>
1.1.4.	<i>Research motive .....</i>	<i>5</i>
1.2	RESEARCH DESIGN.....	6
1.2.1.	<i>Research setting.....</i>	<i>6</i>
1.2.2.	<i>Research objectives and research questions.....</i>	<i>7</i>
1.2.3.	<i>Demarcation and assumptions.....</i>	<i>10</i>
1.2.4.	<i>Research approach .....</i>	<i>11</i>
1.3	POSITIONING THE RESEARCH PROBLEM AND OUR CONTRIBUTIONS .....	13
1.3.1.	<i>Logistic modeling.....</i>	<i>13</i>
1.3.2.	<i>Physical system design.....</i>	<i>14</i>
1.3.3.	<i>Planning and control .....</i>	<i>16</i>
1.4	THESIS OUTLINE .....	21

<b>2 OBJECT ORIENTED MODELING OF PHYSICAL PROCESSES, THEIR LOGISTIC CONTROL STRUCTURE AND THE INFORMATION EXCHANGE.....</b>	<b>23</b>
---	-----------

2.1	INTRODUCTION .....	23
2.2	OBJECT ORIENTED ANALYSIS AND DESIGN.....	24
2.3	PHYSICAL PROCESSES AND PHYSICAL OBJECTS .....	27
2.3.1.	<i>Process description.....</i>	<i>27</i>
2.3.2.	<i>Basic objects and inheritance structure.....</i>	<i>29</i>
2.3.3.	<i>Object hierarchy (“part of”, aggregation).....</i>	<i>33</i>
2.3.4.	<i>Class diagrams.....</i>	<i>34</i>
2.4	CONTROL STRUCTURE .....	36



2.4.1.	<i>Primary process control</i> .....	37
2.4.2.	<i>Secondary process control</i> .....	40
2.5	DESIGN OF THE CONTROL OBJECTS .....	40
2.6	INFORMATION STRUCTURE AND INFORMATION OBJECTS.....	45
2.6.1.	<i>Information exchange and communication lines</i> .....	46
2.6.2.	<i>Performance information</i> .....	50
2.7	SUMMARY .....	51

### **3 CASE DESCRIPTION OLS: THE ROLE OF SIMULATION....53**

3.1	INTRODUCTION .....	53
3.2	PROJECT STRUCTURE .....	54
3.2.1.	<i>Feasibility study</i> .....	54
3.2.2.	<i>Definition study</i> .....	54
3.2.3.	<i>Pre-design phase</i> .....	55
3.3	SUB-PROJECT TRANSPORTATION AND CONTROL TECHNOLOGY.....	56
3.3.1.	<i>System design</i> .....	56
3.3.2.	<i>Functional requirements</i> .....	59
3.3.3.	<i>Vehicles</i> .....	60
3.3.4.	<i>Terminal and dock design</i> .....	61
3.3.5.	<i>Operating and information system</i> .....	64
3.3.6.	<i>Simulation</i> .....	65
3.3.7.	<i>Connekt Test Site</i> .....	67
3.4	MODEL INPUT: TRANSPORTATION FLOWS .....	68
3.5	OPTIMIZATION VERSUS SIMULATION: ESTIMATION OF THE RESOURCE REQUIREMENTS .....	73
3.5.1.	<i>An analytical model</i> .....	73
3.5.2.	<i>Example: Stommeer layout</i> .....	78
3.5.3.	<i>Resource requirements for the 3 different layouts</i> .....	80
3.6	CONCLUSIONS.....	81

### **4 CONTROL OF THE PRIMARY PROCESS: VEHICLE MANAGEMENT.....83**

4.1	INTRODUCTION .....	83
4.2	MODEL AND ASSUMPTIONS.....	84
4.2.1.	<i>Integer linear programming formulation</i> .....	85
4.2.2.	<i>Dynamic programming formulation</i> .....	87
4.2.3.	<i>The difficulties of vehicle management and the available information</i> .....	89
4.3	LITERATURE .....	91
4.4	HEURISTICS FOR EMPTY VEHICLE MANAGEMENT .....	92
4.4.1.	<i>Local coordination using dispatching rules (EVM1 and EVM2)</i> .....	93
4.4.2.	<i>Hierarchical coordination (EVM3)</i> .....	95
4.5	INTEGRATED PLANNING (EVM4).....	97
4.6	LOGISTICS QUEUING NETWORK (EVM5) .....	101

4.6.1.	<i>Reformulating the ILP model</i> .....	103
4.6.2.	<i>Calculating the gradients</i> .....	107
4.6.3.	<i>Calculating the upper bounds</i> .....	108
4.6.4.	<i>Summary of the LQN algorithm</i> .....	110
4.6.5.	<i>Application of the LQN algorithm</i> .....	112
4.6.6.	<i>Modifications of the LQN algorithm</i> .....	113
4.6.7.	<i>Drawbacks of the LQN algorithm with possible solutions</i> .....	114
4.7	NUMERICAL INVESTIGATION.....	115
4.7.1.	<i>Experimental setting</i> .....	115
4.7.2.	<i>Initial experiments</i> .....	116
4.7.3.	<i>Numerical results</i> .....	120
4.8	CONCLUSIONS.....	123

## **5 CONTROL OF AN INFRASTRUCTURE BOTTLENECK: TWO-WAY TRACK CONTROL.....125**

5.1	INTRODUCTION .....	125
5.2	LITERATURE REVIEW .....	128
5.2.1.	<i>Traffic literature</i> .....	128
5.2.2.	<i>Queuing models</i> .....	129
5.2.3.	<i>Machine scheduling</i> .....	130
5.3	CONTROL RULE CONCEPTS.....	131
5.3.1.	<i>Assumptions and notation</i> .....	131
5.3.2.	<i>Characterization of control rules</i> .....	131
5.4	PERIODIC CONTROL .....	133
5.4.1.	<i>Introduction and assumptions</i> .....	133
5.4.2.	<i>Stability conditions</i> .....	136
5.4.3.	<i>Waiting times if the succession time is negligible (<math>\delta=0</math>)</i> .....	136
5.4.4.	<i>Numerical results for <math>\delta = 0</math></i> .....	139
5.4.5.	<i>Waiting times in the case of strictly positive succession time (<math>\delta &gt; 0</math>)</i> .....	140
5.4.6.	<i>Numerical experiments for <math>\delta &gt; 0</math></i> .....	147
5.4.7.	<i>Generalization to compound Poisson arrivals</i> .....	151
5.5	ADAPTIVE CONTROL RULES .....	151
5.5.1.	<i>Adaptive local control</i> .....	152
5.5.2.	<i>Adaptive look-ahead control</i> .....	153
5.5.3.	<i>Dynamic programming</i> .....	155
5.6	DESIGN OF THE SIMULATION STUDY .....	157
5.7	ANALYSIS OF SIMULATION RESULTS .....	158
5.7.1.	<i>Default settings</i> .....	158
5.7.2.	<i>Alternative system configurations</i> .....	159
5.7.3.	<i>Sensitivity for the information horizon and planning frequency</i> .....	161
5.7.4.	<i>Practical use of the DP-approach</i> .....	162
5.7.5.	<i>AGV length and safety precautions</i> .....	162
5.8	A TWO-WAY TRACK IN A CLOSED SYSTEM .....	163
5.8.1.	<i>The effects of convoys on the system</i> .....	163

5.8.2.	<i>Numerical results for the OLS-case</i> .....	166
5.9	CONCLUSIONS.....	167
<b>6</b>	<b>SECONDARY PROCESSES BATTERY AND FAILURE MANAGEMENT, AND THE CONTROL OF RESTRICTED STORAGE CAPACITIES.....</b>	<b>169</b>
6.1	INTRODUCTION .....	169
6.2	BATTERY MANAGEMENT .....	170
6.2.1.	<i>Introduction</i> .....	170
6.2.2.	<i>Implications of battery constraints for the logistic control</i> .....	171
6.2.3.	<i>Logistic performance in case of battery constraints</i> .....	177
6.2.4.	<i>Structure for a cost trade-off</i> .....	180
6.2.5.	<i>Conclusions</i> .....	182
6.3	FAILURE MANAGEMENT.....	183
6.3.1.	<i>Model for failure management</i> .....	184
6.3.2.	<i>Parameters for the OLS-case</i> .....	190
6.3.3.	<i>Experiments with a single controlled failure</i> .....	191
6.3.4.	<i>Experiments with an integrated failure model</i> .....	194
6.3.5.	<i>Robustness of two-way track control with respect to failures</i> .....	199
6.3.6.	<i>Conclusions and further research</i> .....	201
6.4	RESTRICTED STORAGE CAPACITIES OF TERMINALS .....	201
6.4.1.	<i>Adjustments in logistic control</i> .....	202
6.4.2.	<i>The OLS-case</i> .....	203
6.4.3.	<i>Effects of limited storage capacities in the OLS-case</i> .....	204
6.4.4.	<i>Conclusions</i> .....	206
<b>7</b>	<b>CONCLUSIONS AND FURTHER RESEARCH.....</b>	<b>209</b>
7.1	INTRODUCTION .....	209
7.2	CONCLUSIONS.....	210
7.3	FURTHER RESEARCH.....	214
	<b>REFERENCES.....</b>	<b>217</b>
	<b>SAMENVATTING (SUMMARY IN DUTCH).....</b>	<b>225</b>
	<b>CURRICULUM VITAE.....</b>	<b>229</b>

# Chapter 1

## Introduction

### 1.1 Motivation

#### ***1.1.1. Need for innovative transport and logistic concepts***

The worldwide growth of passenger and cargo flows has severe repercussions in terms of traffic congestion problems, especially at and near main traffic hubs such as airports and harbors. In many urban centers, the highway system is also approaching saturation, while the congestion costs met by business in the Netherlands have risen to about 1 billion dollar a year (Ministry of Transport, Public Works and Water Management, 2000a). The attractiveness of top industrial areas gives rise to ongoing concentration of activities. In combination with good facilities for transit cargo, this leads to rapidly increasing inbound- and outbound transportation volumes. In the Netherlands the growth percentages in these volumes can easily surpass those of the GNP by a factor two, i.e. 6 to 8 percent annually (ECMT, 1995). In order to accommodate these increasing flows, the development of new infrastructure has to keep pace. Reliable accessibility of a main hub and its surroundings is essential and realizing sustainable growth is a major challenge, since land is an extremely scarce commodity around a main hub. Given this growth problem, innovative proposals for the extension of the transportation infrastructure should have a high priority. Here public and private interests go hand in hand.

### **1.1.2. Dealing with congestion**

The Dutch Government has initiated and proposes several initiatives to deal with the congestion problems. We distinguish several ways to handle congestion and the resulting problems with respect to reliable throughput times in the transportation sector:

- a) Increasing the transportation capacity
- b) Improving infrastructure utilization by traffic control
- c) Improving infrastructure utilization by peak shaving
- d) Transport prevention
- e) Prioritizing particular transportation flows

#### *a) Increasing the transportation capacity*

A classical solution is to simply extend the transportation infrastructure (roads, railways, terminals, etc.). Such a capacity increase is complicated by the scarcity of suitable surface space and by environmental constraints. Nevertheless, one large project with respect to increasing the rail capacity has been initiated in the Netherlands. This project comprises the construction of the “Betuwelijn”, a dedicated rail connection for freight transportation between Rotterdam and the German border. Once it starts operating, it should lead to a modal shift from road to rail transportation. In addition to such a classical solution, it is important to study new transportation systems that can increase the total transportation capacity and that experience few or none of the aforementioned drawbacks. Underground construction is an option, as it does not require surface space and has less environmental impact. Several cities and companies, for example Amsterdam Airport Schiphol, are interested in such underground transportation systems. We discuss this in more detail in Section 1.1.3.

#### *b) Improving infrastructure utilization by traffic control*

It is difficult to increase the capacity of road and rail facilities, but it may be possible to improve the utilization of the existing infrastructure. Many schemes to improve the road and rail capacity have been proposed. Examples are:

- The dynamic allocation of driving directions to traffic lanes (i.e. 4 lanes to the city and 2 lanes from the city in the morning and vice versa in the early evening).
- Automated speed and distance control on highways. Tests with respect to this technology were performed in the Netherlands and several projects have been initiated in other countries (Ministry of Transport, Public Works and Water Management, 2000b).
- The idea to increase the capacity usage on the rail network by considering new control mechanisms. Instead of the static block control, which is currently used, one could use a dynamic control rule, which allows a shorter headway between trains.

*c) Improving infrastructure utilization by peak shaving*

Another option, which is related to improving the utilization by traffic control, is to influence the user behavior instead of making technical adjustments. An initiative to diminish the peak usage and thereby reducing the congestion problems is road pricing, i.e. installing a time-dependent fee, which is high during rush hours and decreases in hours with lower traffic intensity. The aim of road pricing (congestion charging) is to stimulate a modal shift and/or to spread the vehicles more evenly over the day. Previous implementations in Singapore, Oslo and Trondheim showed a decrease in road usage (Ministry of Transport, Public Works and Water Management, 2000c).

*d) Transport prevention*

The goal of transport prevention is to reduce the transportation volume in terms of ton-kilometers by avoiding unnecessary cargo moves. For example, stimulating companies to settle near their customers and/or suppliers reduces mileage. Information and communication technologies provide opportunities for working at home. Since most traffic in congestion hours results from commuters, working at home may considerably reduce the number of trips, as is the case with car-pooling. Another concept designed to prevent transportation is the Goods Clearing House (cf. Verduijn and Broens, 2000). Whereas traditionally, goods are moved each time the ownership is transferred, the concept of a Goods Clearing House aims to postpone transportation until the ownership is transferred to the final customer. In that case, only one move from producer to the final customer is required. For example, at an auction products are normally transported to the auction and are sold to a particular customer, who transports the goods from the auction to the home location. In case of a Goods Clearing House the goods are not transported to the auction but when the goods are sold they are directly transported to the customer.

*e) Prioritizing particular transportation flows*

In the current situation on the roads all users have more or less the same throughput time. Prioritizing a specific user group, e.g. commercial transporters, may decrease the throughput times of this user group, while increasing the throughput times of other user groups. Target lanes can be used to separate freight and passenger transportation, which increases the safety of both, even though it may lead to less efficient capacity usage.

### **1.1.3. Underground freight transportation**

One of the solutions to the congestion problems is underground transportation. We focus on freight transportation. Freight pipelines constructed underground have only limited environmental impact. These systems can be fully automated and do not interfere with human movement. Pipeline systems are closed and can thus be operated regardless of weather conditions. In itself, underground infrastructure is just a way of increasing transportation capacity. In combination with a high level of automation – loading/unloading and planning and control – it facilitates high

infrastructure utilization (both traffic control and peak shaving) and prioritized transportation flows (e.g. priority for time-critical products as perishables and spare parts for emergency repair). High labor costs are also avoided by using an automated transportation system and there is less need to find qualified personnel, which is a serious problem nowadays. In this setting a rather radical innovation with high potential is the use of Automated Guided Vehicles (AGVs) in underground tube systems. The strength of such a solution arises from the combination of underground construction, advanced transportation technology and logistics.

#### *International developments*

Underground transportation systems already exist. We distinguish capsule pipeline systems (Liu, 2000) and systems where vehicles drive through tunnels. An existing capsule pipeline system is the Sumitomo Capsule Liner in Japan (Liu, 2000), which has been in operation since 1983. This single line is used for the transportation of limestone over a length of 3.2 kilometer. The diameter is about 1 meter and it has been delivering 2 million tons of freight a year. From a logistic point of view, this is a simple system connecting two locations, not a complex network. An existing system with drive-through tubes is the Mail Rail system in London, which has been in operation since 1927 (Bliss, 2000). Small trains deliver the mail through a simple tube network to different mail offices in the center of London according to a fixed schedule. Hence, its logistic control is simple and lacks flexibility. This does not cause problems, because the capacity is ample. This system is currently only used for mail, but one could think of a public underground transportation network, which can be used by different parties. The existing underground infrastructure (subway tubes) may be used for transporting goods from outside the city to department stores in the center of London.

Other studies in this area are being performed on an underground transportation system in the German “Ruhrgebiet” (Stein and Schoesser, 2000) and on city distribution in Japan (Taniguchi et al., 2000). In the Netherlands, underground logistic systems are indicated by the term “OLS”, “Ondergronds Logistiek Systeem” (= Underground Logistic System). Notice that the existing systems consist of only one tube (Sumitomo) or are simple networks with a fixed train schedule (Mail Rail). Therefore, to date no intelligent logistic control structures for large automated transportation networks, possibly with hundreds of AGVs and underground tubes, exist.

#### *City distribution*

Several Dutch cities, such as Leiden, Tilburg and Utrecht, are considering constructing an underground transportation system for city distribution. A large distribution center at the city boundary would link the underground transportation system with other transportation modes, such as road transportation and inland shipping. Technical and economical feasibility studies of such transportation systems have recently been performed in the Netherlands (e.g. Dynavision, 1999 and Buck et al., 1999). All studies indicate that an underground logistic system is

feasible if some preconditions are satisfied. A connection with a national network of freight transportation is essential. Such a connection is necessary to attract enough transportation flows on a local underground transportation system. Further research in the Netherlands has focused on the feasibility of an integrated national network for freight transportation with underground transportation in certain areas (local city networks and/or industrial parks), see e.g. Iding and Van der Heijden (2000).

#### *OLS-Schiphol*

The initiative that has made most progress is a planned underground logistic system near Amsterdam Airport Schiphol, connecting the airport with the world's largest flower auction market in Aalsmeer and a planned rail terminal (near Hoofddorp or Schiphol Airport). The reason for such a system is to relieve the congestion on the roads around Schiphol Airport in order to provide a flexible service depending on particular load priorities and to guarantee reliable throughput times. There are concrete plans to realize this underground logistic system, including technical, logistical and economic aspects. The system is supported by the major business partners involved, these being Amsterdam Airport Schiphol and the flower auction market in Aalsmeer. The OLS-project was one of the main motives for this research. Although our research focuses on *general* automated transportation networks, we have used the OLS-Schiphol to test the logistic planning and control designed in this thesis. To this end, we consider several potential layouts. For a more detailed description of the OLS-case, we refer to Chapter 3.

#### **1.1.4. Research motive**

Automated transportation networks provide a possible solution to the congestion problems, whether underground or not. The Dutch Government increasingly talks about *Undisturbed* Transportation Systems instead of Underground Transportation Systems, which are also automated and require good logistic control. These networks ensure a reliable connection between the locations in the network, such that throughput times can be guaranteed. An in-depth investigation of its merits for reliable logistic service, technical feasibility, environmental benefits and cost performance is worthwhile. To ensure that the system will work properly innovative logistic concepts will also be necessary.

Specific characteristics of the networks that we consider are:

- 1) Both transportation and docking are automated. Automated Guided Vehicles (AGVs) transport the loads from origin to destination.
- 2) The automated transportation system consists of a network structure. Several terminals are connected by a tube or track network, thus we consider larger networks instead of a single line passing through different terminals (cf. Mail Rail, Bliss, 2000).
- 3) In contrast to traditional AGV systems, designed for internal transportation in a warehouse or production environment, we focus on *external* transportation systems for physical distribution. Hence, we consider network layouts with long



distances between the terminals. The size of these AGV networks (and number of vehicles) makes specific aspects of planning and control, such as pre-positioning of empty vehicles, failure management and battery management, more important than it is in existing AGV systems.

- 4) The terminals in the network have a given internal structure (consisting of docks, tracks, parking places, buffers, etc.) and limited capacities. This leads to a hierarchy in the network, which should be taken into account in the design of a logistic control structure.
- 5) The transportation flows consist of a large number of jobs with strongly varying and sometimes tight time-windows. Therefore, a flexible control structure is required and a fixed timetable is not very appropriate.
- 6) Many independently moving AGVs should efficiently share the same infrastructure without collision or deadlock risks.

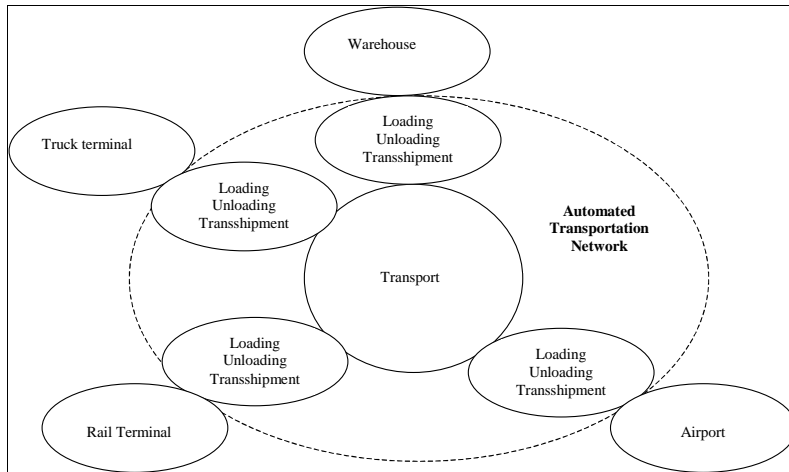
The development of logistic planning and control structures for automated transportation networks is essential to the design and implementation of future automated transportation systems, such as the OLS. The capacity requirements and system performance largely depend on this logistic planning and control. Therefore, we have investigated planning and control in these automated transportation networks. The contributions of this thesis consist of:

- 1) The design of a framework for logistic control on network and terminal levels.
- 2) The development and evaluation of control rules within this framework.
- 3) The development and evaluation of planning variants for some crucial decisions in the planning hierarchy.

## **1.2 Research design**

### ***1.2.1. Research setting***

In this thesis we consider a closed transportation network consisting of a fixed number of origin/destination locations. In this context closed means that the vehicles do not leave the system and that no vehicles from outside enter the system. An asymmetric road or tube network connects these locations. Automated Guided Vehicles (AGVs) transport loads from the origin to the destination. The choice fell on AGVs because of the flexibility required in the system. Advantages of the use of AGVs are the low labor cost, 24-hour availability, and the computer integration and control of the material handling function. For an overview of AGV systems refer to Hollier (1987). We define an *Automated Transportation Network* as a fully automated system for transportation, loading, unloading and transshipment of goods, supported by an advanced automated planning and control system. Such a network and the system boundaries are represented in Figure 1.1.



**Figure 1.1.** *System boundaries of an automated transportation network*

There is a clear hierarchy in the system, since the terminals have an internal structure, with tracks, docks and parking places. This internal structure of the nodes makes the control function more complicated. For example, space at the terminals is limited and the terminals contain a restricted number of docks at which the vehicles can load and unload. In consequence of this finite capacity a limited number of vehicles can be in a node at a certain time so waiting times occur at docks and terminals. In the design of an automated transportation network not only should the number of vehicles be minimized; required capacities or space at the nodes should also be minimized because of the large investments required. The logistic performance is expressed as a service level, i.e. the percentage of all transportation jobs that is delivered at its destination on time (within a specified time window). There should be sufficient resources and a good logistic planning to achieve a high service level.

### ***1.2.2. Research objectives and research questions***

In Section 1.1 we discussed why it is important to investigate automated transportation networks and especially the logistic control. We mentioned the possibilities for automated transportation networks and the need for an intelligent logistic control structure. Because of the importance of logistic control for the future implementation of a proposed automated transportation system, our research goal is the

***design and evaluation of a logistic control structure for automated transportation networks, which guarantees high logistic performance in real-time with acceptable resource requirements.***

We now discuss the different aspects of our research goal. In an automated transportation network several activities have to be performed, including planning the vehicles and terminal operations. All these control activities should be covered by the *logistic control structure*. This is a structured set of objects together with the relationships between them, which is responsible for the systematic planning, coordination and execution of the logistic activities to ensure that particular performance targets are reached, using the relevant information available. Customers demand certain throughput times and since the cargo is transported onward by train or plane, the cargo should reach its destination on time (within a specified time window). Therefore, we should try to minimize the number of late jobs, thus optimizing the customer service. *Logistic performance* is defined as the percentage (e.g. 98%) of transportation jobs that should be delivered at its destination on time. In order to be able to use the designed logistic control structure in future implementations of automated transportation networks, the control should be *real-time*. It should not lead to delays in other processes performed within the network. It should also have sufficient *flexibility* to react quickly to relevant events such as the arrival of rush orders or equipment failure. These real-time and flexibility requirements impose restrictions on the control methods that can be used. The *resource requirements*, the number of vehicles and docks, should be such that the high logistic performance can be guaranteed. Because these resource requirements also influence the cost of the system one might want to minimize these required resources given the constraint of a sufficiently high logistic performance.

### ***Research questions***

To be able to reach our research goal we define a number of research questions. For each question, we indicate the chapter(s) in which the specific question will be answered.

#### **1. Which logistic control activities can be distinguished in automated transportation networks?**

First we need to define the logistic activities that have to be controlled in automated transportation networks. In Section 1.3 we discuss the important control aspects. A more detailed description follows in Chapter 2.

#### **2. Which criteria can be used to evaluate various logistic control concepts and rules?**

As the purpose is to attain high logistic performance, we have to define appropriate logistic performance measures. We discuss this question in Chapter 2.

#### **3. Which logistic control structures are appropriate for an automated transportation network?**

In the previous section we gave a definition of a logistic control structure. In designing the control structure we should keep in mind possible practical aspects because of the usefulness for future implementations. We discuss this question in Section 1.3 and Chapter 2.

#### **4. Which methods can be used to perform the different control activities and how do they interact?**

Given the control structure, we have to specify the methods that give a structured way to take each decision. These control methods can be based on available literature and adapted to automated transportation networks. In some cases, dedicated control rules have to be developed for this special case. In Chapter 2, we propose a preliminary set of control methods for each decision. In subsequent chapters (4 and 5), we examine a few key decision areas in depth.

#### **5. What is the impact of disturbances, such as equipment failures, on the logistic performance?**

Fully automated transportation systems are subject to disturbances. A control rule should be robust to ensure that the system keeps running in the case of these disturbances and these disturbances should be handled properly. We discuss this topic in Chapter 6, where we also present a few additional control rules in order to deal with equipment failure.

#### **6. To which extent can (prior) information enhance logistic performance?**

Transportation jobs can be announced to the system some time before they actually arrive. The planning procedure can take this information into account. This may lead to performance improvements compared to planning based on no prior information, due to the rather long reaction times of AGVs (long distances). With more information, the control objects can anticipate these future arrivals. Several control activities, such as vehicle management, dock functionality, two-way track control, and terminal management could use (prior) information on AGV or transportation job arrivals. (Chapters 4 and 5)

#### **7. What is the effect of the use of batteries on the logistic performance?**

Battery constraints can have serious consequences for the logistic control. Locations for battery stations have to be determined, and decisions about *when* and *where* to change or charge a battery have to be made. Charging or changing batteries has a direct impact on the availability of the AGVs. We discuss the control methods required for battery management in Chapter 6.

#### **8. Which control methods are most appropriate with respect to our research goal: high logistic performance in real-time?**

After comparing several alternative system designs and control activities, recommendations can be made about which control methods are most appropriate in certain situations (Chapter 7).

In Chapter 3, we describe the OLS-case and the role of simulation within the OLS-project. The simulation experiments for the OLS-case are used to answer the aforementioned research questions.

### **1.2.3. Demarcation and assumptions**

To keep the research project manageable, some choices have been made with respect to the research focus:

- a) Order acceptance is part of our control framework, but specific control procedures for order acceptance have not been developed. This can be considered as an additional activity that has only limited impact on the other control methods. When evaluating alternative decision procedures, we assume that all orders under consideration have already passed the acceptance phase. All transportation jobs have a release time and a due time before which they should be delivered at the destination terminal. Priorities can be included by setting different due times. We focus on the automated transportation network and do not consider subsequent logistic handling (see Figure 1.1). For example, flight or train timetables are not taken into account; these can be incorporated in the due time of the transportation job. The transportation jobs leave the system after unloading at the terminal of destination, possibly after a certain time in a terminal buffer.
- b) We focus on control activities at network and terminal level. This implies that we do not experiment with control methods for the lower level control activities, such as traffic control, AGV distance control and dock control. We only include elementary traffic control, required for preventing deadlocks in our simulations.
- c) We design several alternative control methods for vehicle management and two-way track control (see Section 1.3), because these two aspects can have a large impact on system performance. Positioning the vehicles at the right location in the network is important because of the long driving times and a two-way track can be a serious bottleneck in the system.
- d) We focus on the transportation of cargo. Load-bearers are taken into account in the control framework, but not in the logistic control and the simulation model. We assume that there are sufficient load-bearers present, when needed, and that there is sufficient capacity to reposition load-bearers, given the imbalance in transportation flows.
- e) Consolidation of cargo to unit AGV loads is a separate terminal activity outside our scope. Given these unit loads, the transportation jobs are specified by exactly one origin and one destination. The capacity of an AGV is equal to 1 unit load.

Assumptions:

- f) We assume that the layout structure of the system and terminals is given, although during the design phase several of these system or terminal layouts can be compared and the dimension (size, capacity given a layout structure) may be varied. In this context, the dimension of a terminal refers to its geometrical dimensions as well as to its capacity in terms of the number of docks and parking places.
- g) We use simplified vehicle behavior, no acceleration and deceleration and no distance control. Differences in speed as a result of driving loaded or empty are

- also not taken into account. Nevertheless, specific control methods take the possible negative effects on system performance into account (see Chapter 2).
- h) We assume that all AGVs are identical.

#### **1.2.4. Research approach**

Several steps have to be taken to reach our research goal and to answer the research questions. The word *design* in the research goal already indicates that the research is design oriented. This research contains explorative elements too, such as the comparison of different alternatives of system layouts and control methods, and determination of the influence of information.

The amount of information used and the level of planning coordination can affect the logistic performance significantly. It is likely that under ideal circumstances the best system performance can be attained if all major decisions are taken at a central level, using all system information available. This implies that one central organizational unit should be responsible for integrated system planning and that all relevant information should be available at this central level. In consequence of frequent data alteration, extensive and reliable data exchange is essential, which leads to an extensive (and possibly vulnerable) information and communication system. It may also be more difficult for a central authority to react quickly to unexpected events such as equipment failure and the arrival of rush orders.

Local responsibility and authority can be more flexible in this respect. A prerequisite is that the hierarchic layers communicate in a simple, yet efficient way. In a decentralized control concept as many control decisions as possible are taken at the local level. Local control does *not* necessarily mean significant loss of performance, provided that appropriate information exchange between objects takes place. In the case of local control, the responsible local controller bases planning decisions and control activities on local information. Insofar as other information is useful to optimize local decisions, this information can be supplied by communication with other information objects. Preferably, local controllers also communicate with a common global controller to guarantee necessary coordination. For the sake of robustness as well as extendibility, we decided to focus on a *local control* concept, but with the possibility of central coordination. We use a fixed domain structure. Within each sub-domain a controller is responsible for the activities in that sub-domain (cf. Section 2.3.2). For example, a terminal controller is responsible for all control activities within the terminal, but can receive job definitions from a higher level controller to ensure coordination between the sub-domains. In Chapter 4 we analyze the effect of different levels of coordination on the logistic performance.

Once the control activities are known and a control structure has been developed, the different control methods and different levels of coordination should be evaluated. A *discrete event simulation model* seems well suited for this purpose, hence we mainly choose for this approach. Van der Zwaan (1995) mentions that the

strong point of simulation-based research is that it captures the dynamics in organizations, as opposed to other types of research. A simulation model is the ideal environment for experiments with different system layouts, control structures and control activities. Simulation is especially popular in logistics and operations research and it is often used for routing problems, facilities planning, materials handling and operational planning. Below we mention some advantages and disadvantages of simulation (Law and Kelton, 1991).

#### *Advantages of simulation*

1. Most complex, real-world systems with stochastic elements cannot be described accurately by a mathematical model that can be evaluated analytically.
2. Alternative proposed system designs (or alternative operating policies for a single system) can be compared via simulation to see which best meets a specified requirement.
3. Simulation allows us to study a system with a long time frame in compressed time, or alternatively to study the detailed workings of a system in expanded time.

The first advantage is valid for our research problem; formulating a mathematical model is not straightforward, and solving such a model would be extremely hard. Therefore, simulation is probably the only method to answer all research questions. The second advantage describes our research objective exactly; we want to compare alternative system designs and operating policies. Besides these advantages another possible advantage is the visual aspect. Most simulation models have animation potential, which can increase the understanding of the processes and highlight the differences between alternative solutions.

#### *Drawbacks of simulation*

1. A stochastic simulation model only produces estimates of a model's true characteristics for a particular set of input parameters. An analytical model, if appropriate, can produce the exact true characteristics of that model for a variety of sets of input parameters. Thus, if an analytical model can be developed it will generally be preferable to a simulation model.
2. Simulation models are often expensive and time-consuming to develop.
3. If a model is not a "valid" representation of a system under study, the simulation results, no matter how impressive they appear, will provide little useful information about the actual system.

The first drawback mentioned is that a simulation model does not give an optimal solution. It may be possible to formulate an analytical model for some logistic control activities, but these are not suitable for real-time planning, as is the focus of our research. Simulation models may be expensive compared with analytical models, but of course they are still much cheaper than real life experimentation. Once developed, simulation models are powerful in the sense that they can be adapted and/or extended relatively easily.

### **1.3 Positioning the research problem and our contributions**

Several research areas are relevant in relation to automated transportation networks and the planning and control of such networks. In this section we discuss the relevant literature. First, we look at the aspect of logistic modeling and the selection of a logistic modeling concept (1.3.1). Next, we investigate possible methods which can be used in the physical design of an automated transportation network (1.3.2). Finally, we look at the different control aspects in an automated transportation network (1.3.3).

#### **1.3.1. Logistic modeling**

The approach to logistic modeling that we are looking for should be able to model a local control concept, as discussed in Section 1.2.4. In this distributed concept, responsibility is given to the manager of a specific control activity. This does not mean that there can be no coordination between the different controllers. It is even possible that a central manager coordinates all control activities. But the concept of local control enables comparisons of different levels of coordination and should lead to a robust control structure.

The logistic modeling framework (LMF) developed by Van der Zee (1997) provides guidelines to develop a model which is transparent from a logistic point of view. It distinguishes itself from many alternative approaches through its explicit notion of control structures. Often the control of physical processes is only implicitly modeled. The logistic modeling framework is a powerful tool for the modeling and analysis of logistic control systems. It conforms to the object-oriented paradigm because of the generally well-structured models and the closeness of these models to real-life applications.

Another tool for the design of logistic systems is SERVICES (Evers, 2000). Evers introduces the concept of service-oriented agile logistics and presents a generic tool for the design of such systems. The logistic system is conceived as a society of interacting, self-responsible, intelligent service-producing actors. This modeling framework is applied to a high-performance deep-sea container terminal. The concept of SERVICES is rather similar to the LMF, in the sense that control is explicitly modeled. The logistic activities are performed by a “society” of interacting “autonomous actors”. Another aspect these methods share is that they both favor distributed control.

A different approach is the use of agent technology (Jennings and Wooldridge, 1998). An agent is an autonomous unit with the required capabilities to execute given actions and with enough intelligence to assess the consequences of these actions for reaching particular objectives (Espinasse, 1998). Agents can be used for the different control activities. They will communicate with each other and through negotiations they try to reach their specific objectives.



Because of the explicit modeling of the control function in the LMF approach we decided to use this approach to develop our logistic control structure. A model is constructed from an object library, whose components can be classified as physical, control and information objects. These objects are structured in a hierarchical way. The control objects use the available information to ensure the efficient use of the physical objects (resources). We return to this issue in the next chapter.

### **1.3.2. Physical system design**

#### *System layout*

The layout of the track network determines the distances between the different pickup and drop-off locations, as well as possible congestion locations (crossings, junctions, etc.). In consequence, the layout can significantly affect the operational system performance. Available literature on the layout of automated transportation systems was found only on internal AGV systems as used in warehouse and manufacturing environments. Literature on the layout of train systems may also be relevant in this respect. There are several options in designing track layouts for AGV systems (Majety and Wang, 1995), ranging from unidirectional single loop layouts to bi-directional layouts. Bi-directional layouts in AGV systems require fewer vehicles and lead to improved system performance over unidirectional layouts, but traffic control becomes much more difficult. Egbelu and Tanchoco (1986) describe different flow path models and investigate the potentials for bi-directional track layouts for AGV systems. In designing the terminal layouts, which largely determine the capacity and the travel times on the terminal, several of these methods may be used. The layout of terminals is not our research focus, for a study on terminal layouts we refer to Verbraeck et al. (2000). We used terminals with a unidirectional layout, which simplifies the required traffic control. The special aspect in this thesis with respect to system layout is the size of the automated transportation network. The distances between two terminals are significant, resulting in long travel times. Once AGVs are sent in the wrong direction it may take a long time before they can be returned. Furthermore, we investigate the possibility of using only one tube for traffic in two directions. The driving direction within this tube has to be alternated to be able to serve the traffic from both ends (see Chapter 5).

#### *System dimensioning*

The required resources determine a significant part of the costs in an automated transportation network. The required number of docks can be roughly estimated by using the projected transportation flows, the dock time distribution and the estimated capacity utilization (see Chapter 3). The number of vehicles required by an AGV system depends on several factors, including order patterns, travel times, docking time, charging time, the layout of the system and the intelligence of the logistic planning and control system. Simulation is by far the most common approach to determine the number of vehicles required, see e.g. Wysk et al. (1987), Cheng (1987) and Tanchoco et al. (1987). However, an analytical approach might be more

appropriate in the initial design phase when only a rough estimate is needed. The complexity of an analytical model merely lies in estimating empty rides and the waiting times incurred at congestion locations in the system. Maxwell and Muckstadt (1982) present a time-independent mathematical model to find the minimum number of AGVs required for a unidirectional system. The total time includes loaded and empty travel time, where empty travel time is determined by solving a transportation problem. This method applies to an automated transportation network with deterministic travel times between the terminals and no waiting times. We can use a similar approach when the travel times in the network are deterministic. In this thesis we present an analytical model (cf. Section 3.5) which minimizes the maximum workload over the time periods for a model with deterministic travel times. Empty trips and loading/unloading times are incorporated. The peaks in workload are spread as much as possible over the different time periods (peak shaving). Afterwards, given a fixed number of docks, a multi-server queuing model can be used to determine the expected waiting time for a dock operation.

When travel times are very unpredictable, e.g. because of two-way track sections in the system (cf. Chapter 5), such an approach is no longer applicable. Arifin and Egbelu (2000) describe a regression model to estimate the required number of vehicles for manufacturing and assembly facilities. As independent variables they tested the number of workcenters, total vehicle routing distance, number of intersections, maximum machine utilization, total loaded and empty travel distance and the layout complexity. A simulation study was used to estimate the parameters in the regression model. We do not consider an internal transportation system and therefore factors other than those tested, e.g. two-way tracks and failures, could play an important role and new parameter estimates should be obtained (by simulation). Other analytical approaches can be found in Egbelu (1987), Mahadevan and Narendran (1990, 1993) and Ilic (1994). A combination of an analytical model and a simulation approach can be found in Mahadevan and Narendran (1994) and Rajotia et al. (1998). All of the latter approaches focus especially on flexible manufacturing systems. Most of the models mentioned consider travel time as the main factor in determining the vehicle requirements and some even neglect the empty travel time. Furthermore, they are sometimes only applicable to simple system layouts (e.g. a loop layout). In our model empty travel times determine a significant number of the required vehicles and because of capacity restrictions waiting times occur at various locations. In this thesis simulation will be used to compare the results of the simulation model with the analytical model for the case of deterministic travel times and to find the resource requirements for the case of stochastic travel times. Another special aspect with respect to system dimensions is the high number of AGVs; hundreds of AGVs may be required, whereas in existing systems the number of AGVs ranges from a few to 30-40 (cf. McHaney, 1995).

### **1.3.3. Planning and control**

For the discussion of the relevant literature on planning and control, we distinguish the following control areas (cf. Chapter 2 for a more detailed specification):

- a) Vehicle management
- b) Terminal management
- c) Traffic control
- d) Battery management
- e) Failure management

#### **a) Vehicle management**

Unbalanced transportation flows, together with long travel times and tight time windows of transportation jobs, make it very important that AGVs are positioned or pre-positioned at the right terminals at the right time. Depending on known and expected transportation jobs, and their priorities, the vehicle manager has to relocate empty AGVs from terminals with an excess of AGVs to terminals with an AGV shortage. We refer to this planning problem as vehicle management. Due to the long distances and the large number of AGVs in an external transportation network, vehicle management is much more important than in traditional internal AGV-systems. Several research areas are related to this vehicle management, from vehicle routing and scheduling, to fleet management and AGV dispatching. In Chapter 4 several control rules for vehicle management, partly based on the available literature, are developed.

#### *Vehicle routing and scheduling*

The vehicle routing problem is to determine  $K$  vehicle routes, where a route is a tour that begins at the depot, traverses a subset of the customers in a specified sequence and returns to the depot. Each customer must be assigned to only one of the  $K$  vehicle routes and the total size of deliveries assigned to each vehicle must not exceed the vehicle capacity. The routes should be chosen to minimize total travel cost. Applications of the vehicle routing problem are for example the collection of mail from mailboxes and the pickup of children by school buses. More information on vehicle routing can be found in Golden and Assad (1988), and Ball et al. (1995).

Without considering the internal structure and capacity restrictions of the nodes of an automated transportation network, our research problem can be described as a vehicle routing problem with homogeneous vehicles, time windows, asymmetric travel times and unit-load vehicles. This special case is known in the literature as the multiple traveling salesman problem with time windows (Solomon and Desrosiers, 1988; Desrosiers et al., 1995). Because of the unit-load vehicles the loads are transported directly from origin to destination, without intermediate stops. Therefore, all transportation jobs can be represented by a node with a service time equal to the travel time from origin to destination. In this case the problem is reduced to minimizing the empty travel time, which is equal to the travel times between these nodes. A prerequisite for using these mathematical formulations is that all transportation jobs (customers) are known and all vehicles are at the depot

(not travelling). Van der Meer (2000) compares the performance of an off-line control method, in which all information is known, with on-line dispatching for internal transport. It appears that the performance gap between these methods depends on the throughput, where the gap decreases with increasing throughput. In an automated transportation network, we have a dynamic context in which the transportation jobs become available over time and are not known up-front. Furthermore, some vehicles are in the depot while others are moving around loaded and still others are moving around empty with or without a specific job. Because of the real-time constraint and the fact that the internal structure of the nodes is neglected, we decided not to use this type of formulation.

#### *Fleet management*

Fleet management considers the problem of managing a fleet of vehicles over time to serve a set of loads with known origin and destination and a specified time window in which they must be served. Clearly the notion of time is included in this formulation. One of the first papers in this area was by White and Bomberault (1969), who model the allocation of empty freight cars in a railroad system. For an extensive review in the area of fleet management we refer to Powell et al. (1995a). Powell et al. (1995b) introduced a new formulation of the fleet management problem, called a logistic queuing network. The approach reformulates a classical linear programming formulation into a recursive dynamic program. Several papers on this topic were published by Powell et al. (1998a, 1998b), and showed promising results. Such an approach might be appropriate for vehicle management in automated transportation networks, although restricted capacities of terminals are not taken into account. In this thesis we have used the logistic queuing network approach to solve the vehicle management problem (Chapter 4). We added some problem-specific modifications, tested several revenue functions and applied the method in a rolling horizon approach.

#### *AGV dispatching*

The flexibility of AGV systems makes the task of controlling the AGVs very difficult. The issues of controlling AGVs may include dispatching, routing and scheduling. Dispatching involves deciding about the assignment of a particular AGV to a particular transportation job. The need for dispatching occurs when a load arrives or when an AGV becomes idle. This operational control problem significantly affects the performance of the whole system. Egbelu and Tanchoco (1984) classified the vehicle dispatching rules into two categories: vehicle-initiated and workcenter-initiated rules, and evaluate some heuristic rules. An example of a vehicle-initiated rule is a shortest travel time rule, when the vehicle is idle it will pickup the next load at the nearest location. An example of a workcenter-initiated rule is the random vehicle rule. Klein and Kim (1996) propose multi-criteria dispatching rules, and compare these with single criterion dispatching rules. A simulation study showed that the multi-criteria dispatching rules outperform the single criterion ones. Hwang and Kim (1998) use a bidding concept for vehicle dispatching. The information on work in progress in incoming and outgoing buffers

of a machine center and the travel time of an AGV are incorporated in these functions. Other studies can be found in Yim and Linn (1993), Russel and Tanchoco (1984), Sabuncuoglu and Hommeltzheim (1992) and Cheng (1987). The previous rules do not take into account the precise timing of operations. Because timing is important in the case of limited dock capacities, a scheduling approach can be used. Ulusoy and Bilge (1993) schedule machines and AGVs simultaneously, while Akturk and Yilmaz (1996) propose an analytical model to incorporate the AGV-system into the overall decision-making hierarchy. They develop a solution procedure for the AGV scheduling problem that can consider the interaction of the AGV module with the rest of the decision making hierarchy, the current loads of AGVs and the criticality of the jobs. The method is compared with existing AGV dispatching rules and showed good results. The basic dispatching rules seem more appropriate for local vehicle management, i.e. on a terminal, while the scheduling approach might be more appropriate for the total vehicle management. For such scheduling approaches one can also include methods from resource constraint project scheduling, such as Kolisch (1996). In this thesis we use a serial scheduling method, as described by Kolisch (1996), to solve the vehicle management problem. This scheduling method is tested in a dynamic context and results are compared with several heuristics and with the logistic queuing network approach (Chapter 4).

#### *Positioning of idle vehicles*

When an AGV completes a delivery task and is not assigned directly to another pickup task, it becomes idle. The AGV should be located or positioned in anticipation of future pickup calls. Objectives for selecting a position can be (Egbelu, 1993; Kim, 1995):

- Minimization of maximum vehicle response time
- Minimization of mean vehicle response time
- Even distribution of idle vehicles in the network

Kim and Kim (1997) propose a procedure to determine the home location of idle vehicles as a way of minimizing the mean response time for an arbitrary delivery job. Hu and Egbelu (2000) present a framework to determine the optimal home locations in a unidirectional AGV system. They evaluate an exact solution approach and a heuristic algorithm. Gademann and Van de Velde (2000) show that the problem of determining the home location of idle vehicles in a loop layout in order to minimize maximum response time is solvable in polynomial time for any number of AGVs. The same is true when the criterion minimum *average* response time is used. More on positioning idle vehicles can be found in Egbelu (1993), Kim (1995) and Co and Tanchoco (1991). The focus in these papers is on AGV systems for internal transport in a warehouse or production facility. Usually such networks are relatively small. However, we consider *external* transport between various facilities, in which case the networks are much larger. In consequence, the response time for an empty vehicle to arrive at its destination is substantial, and the system status may change significantly in this period (high priority order arrivals, equipment failure or recovery, etc.). Hence, here idle vehicle management differs considerably from the

usual idle vehicle positioning in AGV networks. The positioning of idle vehicles, as described here, can be used in terminal management.

### **b) Terminal management**

At a terminal several control activities have to be performed. As well as local vehicle management, as discussed above, relevant decisions are the assignment of loads to docks, the assignment of vehicles to docks and the assignment of vehicles to local parking places. Specific methods that can be used within the terminal are similar to the methods described above in relation to AGV dispatching and the positioning of idle vehicles. The problem at a terminal is rather similar to those of the existing internal transportation systems. For the operational control of internal transport refer to Van der Meer (2000). Rules such as “take the ‘nearest or fastest available dock’ (workstation) first” can be used here (Egbelu and Tanchoco, 1984). In order-release the priority of the different orders has to be taken into account, i.e. the time window that is available for transportation. In Chapter 2 we describe the methods that will be used in this thesis for the control activities of terminal management. In Chapter 6 we present an approach which takes limited storage capacity at a terminal into account. Since we focus on the higher level logistic control, we did not investigate options for terminal management in this thesis; we implement simple heuristics for the control activities at a terminal.

### **c) Traffic or infrastructure control**

#### *AGV routing*

Routing is the selection of the specific path taken by vehicles to reach their destination. The choice of routes can be based on distance or expected travel time. Shortest path algorithms (Dijkstra, 1959) can be used to determine the shortest route between two locations. But shortest distance does not imply shortest travel time because of possible congestion or disruptions on the routes. The routing of a vehicle can be either static or dynamic (Seifert et al., 1998). With static routing, the path taken by an AGV between two given nodes is always the same. When routing is dynamic different paths can be taken at different times, depending on the current expected travel times along each route. At the time the vehicle is dispatched a route is selected. During travel the route may be modified (Taghaboni and Tanchoco, 1995). Seifert et al. (1998) compare several dynamic routing strategies using a simulation study. In the case of congestion and disruptions a dynamic strategy is superior to a static strategy based on the shortest travel distance path. Of course, sufficient alternative paths should be present to achieve these benefits. Routing algorithms should also reduce the possibility of deadlocks and congestion, and maximize throughput (Soh et al., 1996). In the networks we investigated there is only one direct link between two terminals. Probably a static routing approach with shortest paths will be sufficient. Only in the case of failures or on terminals might one wish to deviate from the shortest path.

### *Two-way track*

One of the possible objects in an automated transportation network is a two-way track. A two-way track, i.e. a track that is used for traffic in two directions, naturally results in waiting times at both ends. The control of a two-way track is related to several research areas. In dynamic traffic control traffic lights are used to control traffic in a conflict area, see e.g. Haight (1963) and Heidemann (1994). Another relation can be made with machine scheduling as described in Uszoy et al. (1992, 1994) and Van der Zee (1997). The two-way track control should try to minimize the influence of the two-way tracks on system performance. The consequences of using two-way tracks in an automated transportation network depend on the flows between the terminals and the number of AGVs in the system. High occupancy leads to long waiting times for a two-way track. In this thesis we design several control methods for two-way track control, ranging from a simple periodic rule to a dynamic programming approach (see Chapter 5). Furthermore, for periodic control we derive theoretical approximations of the average waiting time in case of Poisson arrivals, for both the symmetric and asymmetric cases.

### **d) Battery constraints**

Battery modeling is a commonly omitted aspect in the modeling or simulation of AGV systems (McHaney, 1995). Of course this aspect is only relevant when AGVs use batteries. In a warehousing environment or flexible manufacturing system, the use of batteries may not have serious impact on the system and system availability. McHaney (1995) mentions some situations in which battery modeling can be omitted. Yet the main reason for omitting the effects of the use of batteries is that it is incorrectly believed to have minimal impact on system operation. In an automated transportation network travel times are long and therefore frequent battery changing or charging is required, e.g. once every two trips. When there is only little idle time, and battery changing or charging takes a significant proportion of the time, the use of batteries may make a serious impact on throughput times and the number of AGVs required. We developed control rules to take these battery constraints into account in automated transportation networks. Furthermore, we investigated the effects on system performance of several options for battery replenishment and battery types, and we present an approach to assess the costs of these different alternatives. Battery management is one of the topics in Chapter 6.

### **e) Failure management**

Equipment failures (AGVs, docks) are neglected in most literature on automated transportation systems. Taghaboni and Tanchoco (1995) noted that routing flexibility allows a quick recovery to breakdowns and other disruptive events, but failures are not modeled. Failure modeling is not always necessary. Failures can be neglected in AGV systems when the occupation of AGVs is low and failures can be resolved quickly. In these cases failures have little impact on system performance. AGV failures will also be rare in a system with only a few AGVs. In an automated external transportation network, failures of AGVs can have serious consequences on system performance due to the long travel times and the number of AGVs in the

system (several hundreds in the OLS-case). As a consequence, AGV failures may occur daily. Because of these frequent failures and the possible consequences of these failures, they cannot be ignored in determining the performance of a proposed network. We design control methods to handle AGV and dock failures. The model is used to evaluate the effects of different failure rates and to determine acceptable failure rates, i.e. failure rates that do not significantly affect the logistic performance and could be used as a target in the design process of docks and AGVs. Failure management is one of the topics in Chapter 6.

## **1.4 Thesis outline**

The remainder of this thesis is structured as follows. In Chapter 2 we first discuss the principles of object-oriented design. Then we design a control structure according to the Logistic Modeling Framework of Van der Zee (1997) and describe the various control activities. The case study that we use to test our design, the OLS-case, is subject of Chapter 3. In this chapter we also describe the role of simulation within the OLS-project and present an analytical model to determine the resource requirements. For several control activities only one alternative is implemented, but for others several options will be compared in subsequent chapters. Several options for the primary process of vehicle management are discussed in Chapter 4. The control of a two-way track, which is a potential bottleneck in the system, is the subject of Chapter 5. These aspects have a major impact on system performance and the capacity requirements. The impact of secondary processes, equipment failures and the robustness of our control structure with respect to this source of uncertainty, and battery management are discussed in Chapter 6. In the same chapter we discuss a first approach to incorporating a storage capacity restriction on one of the terminals. In Chapter 7 we present the conclusions and give recommendations for further research.





# Chapter 2

## Object oriented modeling of physical processes, their logistic control structure and the information exchange

### 2.1 Introduction

In Chapter 1 we described the research questions. Here we cover questions 1, 2, and 3. Basic to our approach is that we use a well-structured modeling technique as a tool to provide answers to the research questions. Now we can discuss in more detail what well-structured means. The starting point is that we decided to model an automated transportation network according to the logistic modeling framework of Van der Zee (1997). Basic to this framework is a systematic definition of the logistic resources and their control and the information exchange. Before we can design a control structure, we have to define the activities that must be managed by the logistic control structure (research question 1). In order to distinguish the control activities, we have to describe the physical elements and the processes in an automated transportation network. All entities involved are objects in an *object-model* and we have to design these objects. The objects are translated to building blocks in an object-oriented simulation library. We used the object-oriented simulation package eM-Plant (Tecnomatix, 2000) to implement these building blocks. With such a simulation library we can quickly construct models for alternative layouts or control structures. The simulation library should, in order to

answer the remaining research questions (cf. Section 1.2.2), be suitable for the following types of analyses:

- Comparison of different control structures, varying with respect to the level of coordination (local, central).
- Comparison of different control rules for a control activity for some sub-processes.
- Construction and comparison of several system layouts with respect to their logistic performance.
- Investigation of the robustness of the model, i.e. the impact of input factors (order patterns) and uncertainty (job arrivals, equipment failures) on the performance.

In the context of modeling and simulation, flexibility is increased by the ability to quickly construct a great variety of models from a basic set of building blocks. Such blocks are needed when studying variants of the infrastructure, terminals, docks, buffers, or tracks. Given the importance of an appropriate logistic control structure, the same sort of flexibility is required for the building blocks for the various decisions to be taken in the system, such as vehicle routing, traffic rules and order release. Information exchange blocks are required to supply the control objects with the information needed to optimize the control decisions. The required flexibility and extendibility can be achieved by using a strict object-oriented approach (Booch, 1994) based on a general framework for logistic and transportation agents and their control, cf. Van der Zee (1997). Before we describe the physical elements and processes of an automated transportation network, we first highlight the benefits and elements of object-oriented analysis and design in Section 2.2. In Section 2.3 we describe the physical process, together with the class and object hierarchy of the physical objects in an automated transportation network. The logistic control structure used to optimize the physical processes is described in Section 2.4. Because we focus on logistics, technical control objects such as the technical details of AGV control are outside the scope of our framework. Next, we specify the control objects for which no alternative options are considered in the remainder of this thesis (Section 2.5, cf. research question 4). In Section 2.6, we describe the information structure and information-objects supporting the control structure and delivering management information on performance.

## **2.2 Object oriented analysis and design**

Object orientation requires the analyst to model the world in a natural way as a set of cooperating objects. A clear relationship should exist between an object in a model and some real world entity. We highlight the most important aspects of the object-model. For more theory on object-oriented analysis and design we refer to Booch (1994).

Booch (1994) distinguishes four major elements of the object-model: abstraction, encapsulation, modularity and hierarchy. “An *abstraction* denotes the essential characteristics of an object that distinguish it from all other objects and thus provides crisply defined conceptual boundaries, relative to the perspective of the viewer” (Booch, 1994). Only real-life details that are relevant for the user should be included in the model. An example of such an abstraction is an AGV. Relevant details are the speed and capacity for cargo, while color is irrelevant for our model. Furthermore, the behavior of the AGV with regard to acceleration and deceleration should be observed, but the technical details governing this behavior are not relevant for the user of the model focussing on logistics. Whereas abstraction focuses on the observable behavior of an object, *encapsulation* focuses upon the implementation that gives rise to this behavior. Encapsulation hides the implementation of an abstraction, and in this way allows program changes to be made with limited effort. No part of the system should depend on the internal details of another part. The elements that do not contribute to the essential characteristics of an object are hidden by *information hiding*. This entails that each class has two parts: an interface and an implementation. For example, the interface of an AGV includes the required speed and acceleration whereas the operation of the motor is hidden. Clearly encapsulation and information hiding are very important in comparing different alternatives for one control activity so a basic requirement for a flexible model is the standardization of communication between objects. As long as the interfaces of the objects remain the same, both the physical objects, with their behavior, and the corresponding control objects, are allowed to change internally in any way, as long as the overall functionality of the object remains the same. This leads to a powerful object library with reusable objects and makes a distinction between physical, control and information objects very useful. “*Modularity* is the property of a system that has been decomposed into a set of cohesive and loosely coupled modules” (Booch, 1994), each being a package of abstractions. One of the reasons for introducing modularity is to manage the complexity of the system. Another justification for partitioning is that it creates a number of well-defined boundaries within the system. Modularity provides the means to describe systems by specifying its subsystems instead of having to address the whole system. For example, a terminal can be seen as a module (sub-system) in the total transportation system. Modularity is basic to decentralized modeling concepts and is represented by an object hierarchy. “A *hierarchy* is a ranking or ordering of abstractions” (Booch, 1994). The two most important hierarchies are the class hierarchy and the object hierarchy. A class is a set of objects that share a common structure and a common behavior. Attributes are used to describe the state of the object belonging to a class. Every object is the instance of some class, and every class may have multiple instances. The class hierarchy (“is a” hierarchy) indicates the inheritance relationships between classes. Inheritance relationships show how classes share or re-use the structure and behavior defined in other classes. For example, a branch is a track, but with specific properties (several directions) and/or specific methods (send an AGV in the right direction). A branch shares the property length with the track. The object hierarchy (“part of” hierarchy) shows in which way objects are part of a larger whole; it

describes aggregation relationships. For example, docks are part of the terminal and the terminal is part of the total system.

Booch (1994) distinguishes three benefits of the object-model:

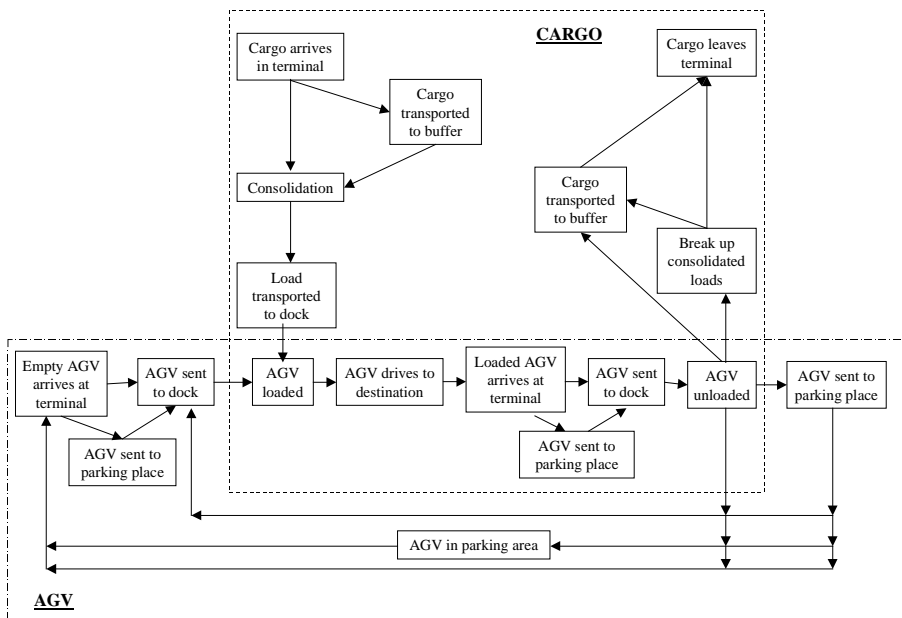
1. The use of the object-model helps us to exploit the expressive power of object-based and object-oriented programming languages, because it appeals to the working of human cognition.
2. The object-model encourages the reuse not only of software, but also of entire designs.
3. The use of the object-model produces systems that are built upon stable intermediate forms, which are more resilient to change.

In real-life, physical, information and control processes can be distinguished. In modeling a logistic system we could make the same classification. By distinguishing physical, information and control objects we can change the control and information infrastructure without modifying the physical system and processes. It is also possible to change a physical object, while the control and information infrastructure remains unchanged. This classification results in a model that is closer to reality. Different implementations of a specific control activity can easily be exchanged, which may have different impacts on system performance. This is very useful, but it is not always simple to achieve. The performance of a specific control object may depend upon the specific behavior of another control object; there are always interactions between some control objects. Because of the abstraction and encapsulation, the model will still work because of the well-defined interfaces, but the performance can be significantly influenced. Therefore, to retain acceptable logistic performance, it may be necessary to change the implementation of several control objects simultaneously. Object orientation is very useful, but not always as straightforward as sometimes suggested, due to these interactions. For example, when the coordination level is increased and responsibilities shift from the terminal to the system manager, the terminal manager should be able to execute the instructions of the system manager, such as handling the transportation jobs in the sequence specified by the system manager. Therefore, such a change in responsibilities affects both the system and terminal manager object. But even in these situations with mutual dependence, modularity is helpful in enforcing flexibility.

## 2.3 Physical processes and physical objects

### 2.3.1. Process description

The primary function of an automated transportation network is the transportation and transshipment of cargo. In describing these processes, we investigate activities from the arrival of an order in the system to its fulfillment. We distinguish primary and secondary processes. The primary process is the transportation, loading/unloading, and storage of cargo. As secondary processes we distinguish load-bearer, battery and failure management. We describe the processes using Figure 2.1, which only shows the primary process.



**Figure 2.1.** Physical primary process in an automated transportation network

In this description, all physical objects are denoted in *Italics* when encountered for the first time. The process starts with the arrival of a transportation request. Given the work in progress and the available capacities, the order acceptance procedure determines whether this transportation job can be completed. When the transportation job has been accepted, the physical process starts when the *cargo* arrives at the *terminal*. After arrival, the cargo may have to wait in a *cargo buffer* for an available *loading dock* and/or for an available *AGV*. Cargo from several transportation jobs may be consolidated to increase the capacity utilization. When a loading dock and an *AGV* are available, the cargo can be loaded on the *AGV*. At the terminals, automated handling equipment is available to load and unload *AGVs*. After loading, the *AGV* drives to the destination terminal of the cargo. A *track*

*system* is required to enable vehicles to reach the destinations. *Junctions*, *branches*, *two-way tracks* and *roundabouts* may be included in this track system. A special type of track is the two-way track, which is a single track along which traffic in two directions is possible (although not simultaneously). Hence, when the track is occupied by vehicles travelling in the opposite direction vehicles have to wait, thereby causing additional delay. Once the AGV arrives at the destination terminal, it needs to unload at an *unloading dock*. When an unloading dock is available, the AGV drives to it, otherwise the AGV can be parked in a *parking place* or wait at the terminal entrance. When an unloading dock is assigned to the AGV, it drives to that dock and the cargo is unloaded. Consolidated loads are broken up according to the original transportation jobs. After spending some time in the terminal cargo buffer, the cargo eventually leaves the terminal. The empty AGV can be sent to a loading dock, a parking place, another terminal or to a *parking area*. A loading and an unloading dock can be two different physical objects, but these two functions can also be incorporated into one physical object.

#### *Secondary processes*

Secondary, supporting processes may be associated with the logistics of shared movable resources, such as *batteries* and *load-bearers*. If the battery of an AGV is explicitly modeled, it can follow a similar process to that of the cargo (not represented in Figure 2.1). In this case a battery is charged at a *battery station* and has to wait in a *battery buffer* until it is needed by an AGV. When an AGV needs a fully charged battery, it is sent to a battery station. The AGV has to be assigned to a *battery dock* where the battery can be changed. The battery is transported from the battery buffer to the battery dock, where it is exchanged for the nearly empty battery of the AGV. The nearly empty battery is sent to a *charging station* and when sufficiently recharged, is transported to the battery buffer and ready for reuse. Another option is to recharge the battery inside the AGV, in which case the AGV is sent to a battery dock, where it is coupled to chargers. There the AGV has to wait until the battery is sufficiently recharged.

To transport cargoes of different dimension *load-bearers* may be required. A load-bearer might also be required to secure the cargo. Empty load-bearers have to be moved between the terminals to ensure that there are sufficient load-bearers available at every terminal to transport the transportation jobs present at that terminal. When temporarily not needed, the load-bearer is stored in a *load-bearer buffer*.

Another secondary process that interferes with the processes in Figure 2.1 is the failure process. Equipment, such as AGVs and docks, can break down. Because AGVs move through the system and may block traffic in the case of failure, special attention must be paid to AGV recovery. A failed AGV should be taken out of the system for repair. A special *recovery vehicle* must be available to tow the AGV to the location at which it can be repaired. Note that it is not always necessary to

incorporate the secondary processes in the model, for example when load-bearers are not required.

### 2.3.2. Basic objects and inheritance structure

In an automated transportation network, the locations (nodes) in the network are connected by tracks, and there must be a path in both directions between each pair of nodes (not necessarily identical paths). Distances, and possibly capacities, are associated with these paths. The basic objects in the network topology are tracks and nodes. A node is a specific location in the network, for example a terminal, parking area, or dock, which is the starting or ending point of a track. In Figure 2.2 we see an example of an automated transportation network, in which a terminal and a dock within this terminal are highlighted. Notice the clear separation between the physical process and the control and information parts. The model contains a hierarchy of domains (system, terminal, dock) in which each domain consists of physical, information and control objects.

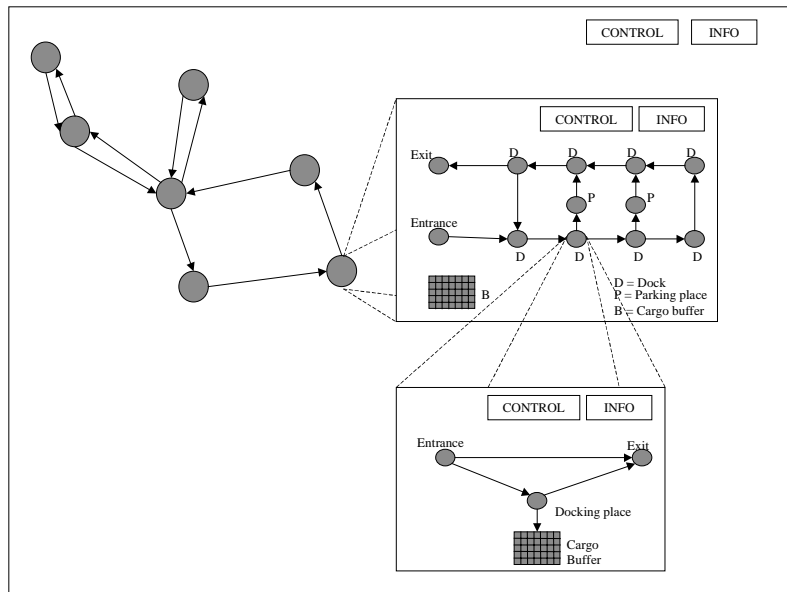


Figure 2.2. Example of an automated transportation network

In the previous section, we introduced the primary and secondary physical processes and all objects that are required for these processes. In this section, we describe the basic object classes, together with the attributes. From these object classes we can derive other classes by inheritance or aggregation. In this description we use the following logical structure. We already distinguished primary and secondary processes, which can both be separated into a transportation process and a logistic handling process. The transportation process concerns the transportation of cargo by



the AGVs, whereas the logistic handling process deals with the loading/unloading and storage of cargo, batteries and load-bearers. In these processes we distinguish three elements: a *flow unit*, a *handling unit* and a *storage unit*. In this section these units (objects) will be defined for the different processes. First this will be done for basic units, after which composite units will be considered. To avoid lengthy attribute lists, we note that as attributes, all objects have certain *references* to other objects and *status indicators*. There are *references* between objects to indicate which object is handled or stored by another object. For example, *cargo* contains a reference to its *position*, whether this is a specific cargo buffer or an AGV. A *cargo buffer* has references to the *cargo* currently stored in the buffer. The status indicators are different for the three elements. The status of a handling unit is *free/occupied* and *up/down*. A flow unit has a *current location*, and the status of a storage unit is its *occupancy*.

### ***Primary transportation process***

#### *Flow unit: vehicle*

We need two different vehicles in the object-model: AGVs transport cargo and recovery vehicles tow failed AGVs (see secondary transportation process). These objects have a lot in common. Therefore, we introduce a basic object class *vehicle*, from which the other two object classes will be derived by inheritance (sub-class). As attributes the vehicle object has length, speed, maximum speed, driving direction (forward, backward), up/down and the destination. An additional attribute is the energy consumption rate. The energy consumption depends on slopes, speed, weight and the acceleration and deceleration. The AGV object inherits the behavior and attributes of the vehicle object, but some additional functionality and attributes have to be included. An AGV is used to transport cargo and therefore the AGV has a certain capacity for cargo. The attribute ‘status’ indicates whether the AGV is loaded or empty. Since we only consider unit-load AGVs, the capacity of the AGV is always equal to 1 (see Chapter 1, Section 1.2.3). For the transportation process, the AGV is a flow unit, but the AGV is also a handling unit for cargo and load-bearers.

#### *Handling unit: track*

The AGVs are “handled” by the *tracks*. For example the tracks are tunnels in an underground transportation system or roads. Vehicles cannot pass each other on a track, but parallel tracks can be used for passing. We define a *track* as an object on which vehicles can drive. The track has a specific length, capacity, driving direction and speed limit. A track has only one predecessor and one successor. Several objects of the track system share a common behavior and can be derived by inheritance from the basic track object: junctions, branches and two-way tracks. All derived objects share the functionality of the basic object class *track*, but with additional functionality and/or attributes. A *junction* is a track with more than one predecessor, while a *branch* is a track with more than one successor. A *two-way track* is a track the driving direction of which can be changed; it can alternately serve traffic from two directions. The two-way track has two predecessors and two successors, one for

each direction. An important attribute of the two-way track is the driving direction. The two-way track control object determines this driving direction (cf. Section 2.4).

*Storage unit: parking place*

Vehicles that are temporarily not needed can be parked in a *parking place*. The parking place is derived by inheritance from the track object, but has specific control methods and attributes to account for its storage function. A vehicle needs permission to enter a parking place and when a vehicle receives a job it will leave the parking place and drive toward a dock or another terminal. A parking place can be dedicated to loaded or empty vehicles, or alternatively to recovery vehicles.

***Primary logistic handling process: cargo handling***

*Flow unit: cargo*

Cargo is transported by AGVs and is related to a transportation job (see Section 2.6). Cargo has a weight and volume. The corresponding transportation job determines the origin and destination of the cargo, together with the release time and due time.

*Handling unit: loading/unloading dock*

Cargo has to be loaded onto or unloaded from an AGV at a loading/unloading dock. Several docks were mentioned during the description of the physical process: a loading dock, an unloading dock and a battery dock. These objects have several characteristics in common and therefore we define a basic dock object. We define a *dock* as a location where a specific operation can be executed on an AGV. An attribute of the dock is the time required for a dock operation. A dock has a loading/unloading facility, which can be used for loading, unloading or both. An attribute of the dock is its functionality: loading, unloading or both. Other dock objects can be derived, the functionality of which alternates between loading and unloading, with a set-up time in between.

*Storage unit: cargo buffer*

Cargo that cannot be loaded directly, because no AGV and/or dock is available, or cargo that has to wait to leave the terminal, e.g. on a plane, has to be stored. We define a basic *buffer* object, which has a finite capacity for storing flow units, such as cargo, batteries or load-bearers. A buffer is dedicated to a specific class of moveable objects. The cargo buffer can only store cargo.

***Secondary transportation process: AGV failure management (optional)***

*Flow unit: recovery vehicle*

A recovery vehicle is used to tow failed AGVs. The recovery vehicle is derived by inheritance from the basic vehicle object. To be able to reach failed AGVs the recovery vehicle should be able to drive in two directions (forward and backward). Unlike the AGV, the recovery vehicle cannot move cargo and load-bearers. However, it can act as a handler for AGVs.

*Handling unit: track*

The recovery vehicles use the same infrastructure as the AGVs: the track system. We refer to the primary transportation process for more details on the track object.

*Storage unit: parking place*

A recovery vehicle that is temporarily not needed can be parked in a parking place. This is the same storage unit as the one used in the primary transportation process.

***Secondary logistic handling process: load-bearer management (optional)***

*Flow unit: load-bearer*

A load bearer can be used when cargoes of different dimensions have to be transported. A load-bearer might be required to facilitate the use of standard material handling equipment, such as the loading/unloading dock. AGVs can be used to transport the load-bearers

*Handling unit: loading/unloading dock*

Given the fact that a loading/unloading dock is designed to handle load-bearers, these load-bearers can be loaded and unloaded by the same dock.

*Storage unit: load-bearer buffer*

When a load-bearer is temporarily not needed it is stored in a load-bearer buffer.

***Secondary logistic handling process: battery management (optional)***

*Flow unit: battery*

A battery supplies a vehicle with energy. Attributes of the battery are its actual and maximum charge. This information is important in determining whether an AGV can still perform a given job or whether the battery should be changed or charged.

*Handling unit: battery dock and charge station*

The battery of an AGV can be charged inside or outside the AGV. When the battery is recharged inside the AGV, it must be possible to make the necessary connection to recharge the battery at the dock. Otherwise the battery dock must be able to supply a fully charged battery to replace the nearly empty battery. An additional handling unit is required if the battery is charged outside the AGV, so a location is needed where the batteries can be recharged. We define a *charge station* object, which has a specific capacity. This capacity indicates the number of batteries that can be recharged at the same time.

*Storage unit: battery buffer*

Recharged batteries, or batteries that have to wait for space in a charge station, can be stored in a battery buffer. This object is derived from the basic buffer object, dedicated to the storage of batteries.

### **2.3.3. Object hierarchy ("part of", aggregation)**

In the previous section we described the basic object classes and the object classes derived by inheritance. Compound objects, in which several objects from the previous section are combined, can be defined. These compound object classes are derived from other objects based on aggregation relationships, using the concept of modularity (see Figure 2.5). Below we describe the compound object classes, which can be related to the transportation process, to the logistic handling process or to both processes.

#### ***Transportation: roundabout***

A *roundabout* is a combination of tracks, junctions and branches, and therefore completely based on aggregation relationships. The capacity of a roundabout is limited, being related to the capacity of the underlying track objects.

#### ***Transportation: track system***

A *track system* is a combination of tracks and derived track objects: branches, junctions, two-way tracks, roundabouts and parking places. Together these objects constitute the infrastructure for the vehicles.

#### ***Transportation: parking area***

Not all AGVs are continuously needed in the system, because transportation flows may fluctuate. For the AGVs that are not needed for a while, a *parking area* can be included in the model. Such a parking area consists of a track system with a number of parking places, which determine the limited capacity of the parking area.

#### ***Logistic handling: dock station***

A *dock station* consists of a dock, tracks and a buffer. There should be at least one place (a track) for an AGV and also a storage location for a flow unit (cargo, battery or load-bearer). Extra places for AGVs or extra storage locations might be present for the pre-positioning of AGVs and flow units to reduce the time between subsequent dock operations, and to maximize dock utilization.

#### ***Transportation and logistic handling: terminal***

The *terminal* object consists of a track system (tracks, branches and junctions) to ensure that the AGVs can reach all destinations inside the terminal. Dock stations are present for loading and unloading of AGVs, and there is a cargo buffer for cargo storage. Optional objects in a terminal are parking places and a load-bearer buffer. Parking places may be part of the track system inside the terminal to increase the flexibility of the terminal. A load-bearer buffer is necessary when the corresponding secondary process is incorporated in the model.

#### ***Secondary logistic handling: battery station***

When AGVs receive their energy from batteries, rather than from other sources of electricity (e.g. charge-rails), it is necessary to charge them in a *battery station*. Such a battery station contains both a track system and battery dock stations. A battery

buffer and charge stations are required when the batteries are recharged outside the AGV. Parking places are optional.

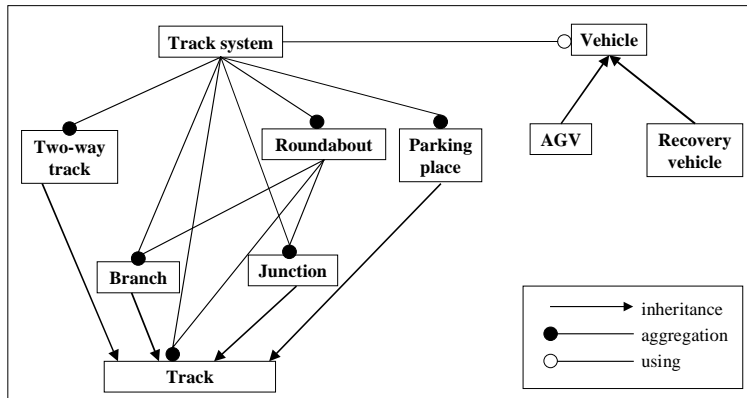
**System**

The total *system* consists of a number of locations and a track system, which connects these locations (cf. Figure 2.2). Terminals must be present in the system. A parking area and a two-way track are optional objects. When the secondary processes are modeled additional objects have to be present.

**2.3.4. Class diagrams**

Now that all required objects have been defined, we can construct class diagrams. In these diagrams we show the relationships between the different objects. We distinguish three possible relationships. First the *inheritance* relationship, indicating that an object class (sub-class) is derived from another class (super-class). This relationship is depicted as a directed arc, with the head pointing to the super-class. Second, an *aggregation* relationship, indicating that an object is part of another object. This relationship is represented by a connection with a dot at the end. The dot is connected with the sub-aggregate. Third, a *using* relationship, in which one object uses the services of another object. The *using* relationship is represented by a connection with an open dot at the end, indicating the client.

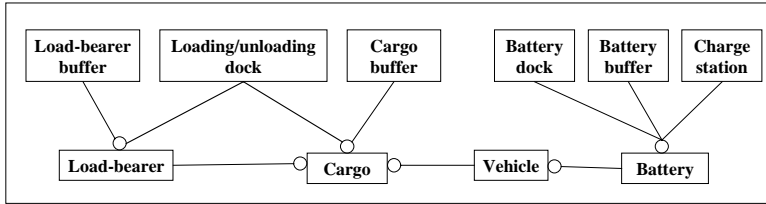
Figure 2.3 shows the transportation process. The track system consists of several objects, some of them derived by inheritance from the super-class track, and others based on aggregation relationships. The vehicles use (or are handled by) the track system to reach their destination, and can be stored in a parking place.



**Figure 2.3.** Class diagram of the objects that are part of the transportation process

Notice that the secondary process of AGV failure management is also included in Figure 2.3. The recovery vehicles use the same infrastructure as the AGVs. The logistic handling process is shown in Figure 2.4. These processes consist of ‘using’

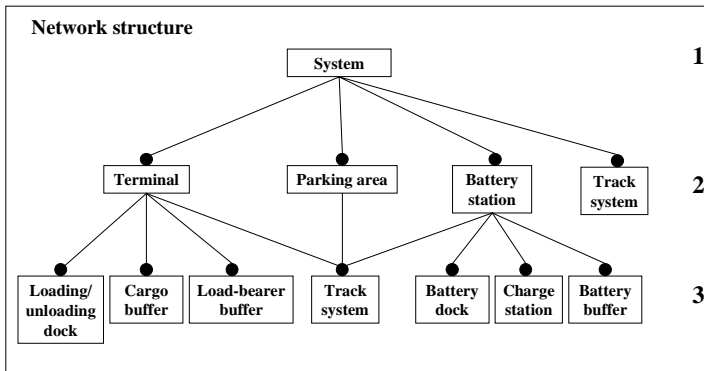
relationships. The flow items (cargo, battery, load-bearer) are handled by (or use) the handling and storage objects.



**Figure 2.4.** Class diagram of the objects that are part of the logistic handling process

The *using* relations between cargo and load-bearer and between vehicle and battery indicate the relationships between the primary and secondary processes in the logistic handling. Cargo might need a load bearer for the handling at the loading/unloading dock. A vehicle might need a battery to provide its energy. In this case this secondary process, battery management, is included in the model.

In Figure 2.5 we see the network structure and the hierarchical decomposition. We distinguish three different levels for logistic control and transportation management: system level, terminal level (2) and sub-terminal level (3).



**Figure 2.5.** Class diagram of the network structure and hierarchical decomposition

Notice that the track system appears at both lower levels (2 and 3). The track system at the 2<sup>nd</sup> level connects the terminals, parking areas and battery stations, while these objects contain their own internal track system.

## 2.4 Control structure

It is not sufficient to model physical elements alone, since for an operational system a logistic planning and control structure is required. Control objects take care of decisions on the activities of physical objects within the area of their control. Furthermore, control objects use information available to them to ensure efficient use of the physical objects and to achieve certain performance targets. Available information includes the data stored in the control objects own accompanying information object and data that can be accessed by an information request to another information object. We return to this information issue in Section 2.6.

As explained in Chapter 1, we allow for hierarchical decomposition of the control over several levels, for example over network and terminal levels. This allows for a *local control* concept, with the possibility of central coordination and information exchange between the controllers (managers). This means that the node managers are responsible for all control activities inside the nodes, but they can receive performance targets on their activities or even job definitions, such as *empty vehicle jobs*, from a higher level in the object hierarchy to ensure coordination between the nodes or levels. If local problems are solved locally and network problems centrally, we can construct a relatively simple and robust control structure without excessive information exchange.

Local control does *not* necessarily mean significant loss of performance, provided that appropriate information exchange between objects takes place. In the case of local control, the terminal manager bases its planning decisions and control activities on local terminal information only (embedded in the terminal information object). Insofar as other information is useful to optimize local decisions, this information can be obtained by communication with other information objects. We allow for local controllers that communicate with each other, either directly or indirectly via a common global controller. The local and global controllers communicate with each other in order to tune the effects of local decisions, and thus prevent negative effects of sub-optimization as much as possible. In Chapter 4 we will analyze the effect of central versus local decision making on the logistic performance. Now we consider the relation between the control objects and the physical objects in more detail. Basically, each control object is uniquely associated with a physical object of a certain aggregation level. The control activities and the aggregation level (cf. Figure 2.5) are represented in Table 2.1, and are related to the classification in primary/secondary processes and transportation/logistic handling processes.

**Table 2.1.** *Control activities and related objects*

<b>Primary process control</b>	<b>Control activity</b>	<b>Level</b>
<i>Transportation and logistics</i>	Local empty vehicle management	2
	Vehicle scheduling	2
<i>Transportation</i>	Global vehicle management	1
	Local vehicle control	2
	Two-way track control	2,3
	Routing control	2,3
	Traffic control	2,3
	Distance control	3
<i>Logistics</i>	Order acceptance	1 (2)
	Consolidation control	2
	Order release	2
	Task allocation to docks	2
	Dock control	3
<b>Secondary process control</b>		
<i>Transportation and logistics</i>	Battery management	1,2
	Load bearer control	1,2
	Failure management (AGVs, docks)	1,2,3

The control objects investigated in detail in the following chapters include vehicle management, two-way track control, battery management and failure management. Although it is also interesting to study the other control objects, we chose to specify only one variant (see Section 2.5 for the corresponding definitions). In the next two sections we describe the function of the primary (Section 2.4.1) and secondary process control objects (Section 2.4.2).

### **2.4.1. Primary process control**

Vehicle management can be considered at two hierarchical levels, namely at network level (how to allocate empty vehicles amongst terminals and parking areas) and at terminal level (how to handle a vehicle in a terminal). At terminal level, it is easier to react to unexpected events, such as the arrival of rush orders and the failure of docks or AGVs. All information required to make a good decision is available at this level. At network level, we can focus on balancing vehicle flows, without taking into account all detailed events at terminals. To this end, we distinguish between *global vehicle management* that covers the network level and *local vehicle management* that covers the terminal level. From a practical point of view such an approach is to be preferred, but it requires fine-tuning between the two hierarchical levels.



## ***Transportation and logistic handling control***

### *Local empty vehicle management*

When an empty AGV becomes available, it is necessary to decide whether the AGV will be used for an *empty vehicle job* (an empty trip) or for a *transportation job*. An empty vehicle job is an assignment of global vehicle management to send an empty vehicle to another terminal within a specified time window. A transportation job is an assignment to transport a load from origin to destination within a specified time window. If no jobs are present the AGV can be passed to local vehicle control. If the AGV can be used for a transportation job, the AGV is passed to vehicle scheduling.

### *Vehicle scheduling*

Vehicle scheduling is the assignment of an AGV to a loading or unloading job. An empty AGV can be assigned to a loading dock in combination with a transportation job, while a loaded AGV can be assigned to an unloading dock to start an unloading operation. Here the transportation and logistic handling process come together. If no job can be assigned to an AGV, the AGV receives a destination from local vehicle control.

## ***Transportation control***

### *Global vehicle management*

The function of the global vehicle manager is to ensure that the AGVs are at the right network node at the right time. This is required to ensure that all transportation jobs are carried out. Global vehicle management is the topic of Chapter 4.

### *Local vehicle control*

Local vehicle control selects a destination for a vehicle that cannot be assigned to a specific job, that is when there is no *empty vehicle job* or load job for an empty AGV, or no unload job (no dock available) for a loaded AGV. A loaded AGV has to be positioned somewhere in the terminal, e.g. a parking place, while an empty AGV can also be sent to an empty dock or a parking area.

### *Two-way track control*

The function of two-way track control is to determine the time at which the driving direction of the two-way track should be switched. Two-way track control is part of the track system (level 2 and 3) and is considered in Chapter 5.

### *Routing control*

Given the location of a vehicle, and the destination of this vehicle, a travel route from the current location to the destination must be determined. The routing of vehicles is a control object of the track system. At branches a vehicle must be sent in the right direction (corresponding successor).

### *Traffic control*

Traffic control is an object of the track system and has to prevent deadlocks and ensure that traffic circulates freely. A deadlock is a situation in which vehicles are

blocked by other vehicles in such a way that the situation cannot be resolved. Furthermore, in order to prevent the average speed from being significantly reduced the vehicles must not hinder each other too greatly.

#### *Distance control*

The function of distance control is to prevent the collision of a vehicle with another object. Therefore, distance control is incorporated in the vehicle object.

#### ***Logistic handling control***

##### *Order acceptance*

When a transportation request arrives in the system, order acceptance has to decide whether the transportation job is accepted or declined. The order acceptance procedure may also be part of terminal control, but with communication between the terminals.

##### *Consolidation control*

The function of consolidation control is to consolidate cargo into unit-loads. Such a unit-load will be translated to a transportation job for an AGV. Cargo that is consolidated to one unit-load should have the same destination. Breaking up these unit-loads at the destination terminal is also incorporated in this control object. We do not consider the case of several pick-up and delivery terminals. Consolidation takes place at a terminal.

##### *Order release*

By order release we mean the assignment of transportation jobs to load docks in the terminal of origin (level 2). It is necessary to determine *which* order is sent to a dock and *when* an order is to be sent to a dock for loading.

##### *Task allocation to docks*

Docks on a terminal are used for loading and unloading of AGVs. A dock can perform both loading and unloading operations at the same time. This has a disadvantage from a logistic point of view because the incoming and outgoing flows cannot be clearly separated. To separate these flows we can change the functionality of the loading/unloading dock periodically, from loading to unloading and vice-versa. The control object under consideration determines the timing at which the functionality of each dock should be switched. Changing the functionality of a dock has as an advantage over dedicated loading and unloading docks in that the dock capacity can be fitted to the expected workload. A drawback is that setup times may be involved.

##### *Dock control*

The docks load and unload the vehicles or change (charge) the battery of an AGV. Dock control is responsible for the transfer of cargo (or battery) from the dock to the AGV and vice versa.

## **2.4.2. Secondary process control**

### *Battery management*

The function of battery management is to ensure that batteries have sufficient charge to perform the current job of the AGV they are provisioning with energy, and that the AGV does not stop somewhere in the system with an empty battery. Battery management is required when AGVs have to change or charge their batteries at a dock in a battery station. A decision should be made on the timing and location of battery change/recharge. Battery management is one of the topics in Chapter 6.

### *Load-bearer control*

Load-bearer control has to arrange that load-bearers are in the right place at the right time. It must prevent situations arising in which transportation jobs cannot be loaded because there is no load-bearer available. In this thesis this aspect is not taken into account; we assume that there are always sufficient load-bearers present when needed.

### *Failure management*

The resources are subject to failures, particularly AGVs and docks. Given a limited failure handling capacity, failure management focuses on the most efficient recovery of equipment. Its purpose is to minimize downtime and to reduce the negative effects on system performance, for example by removing an AGV that is blocking traffic as quickly as possible. The decisions involved are the assignment of recovery capacity to failed equipment and, in the case of AGV failures, the reservation of tracks for the recovery vehicle to approach the AGV and the release of the tracks for regular traffic afterwards. Clearly, failure management requires coordination between several control objects. Failure management is described in detail in Chapter 6.

## **2.5 Design of the control objects**

In Section 2.4 we discussed the control structure and the functions of the control objects. Now we have to design the control objects. In Chapter 1 (Section 1.2.3) we decided to design several options for vehicle management and two-way track control. These topics will be discussed in Chapters 4 and 5. The secondary processes of failure management and battery management will be discussed in Chapter 6. Here we survey the control objects that are not discussed or varied during the analyses in the next chapters. Different implementations of the control objects are possible, from heuristics to optimization procedures, as long as these are real-time. We describe the control activities from a local control perspective. When it emerges that a local control concept cannot reach the performance targets, it is possible to integrate several of the control activities. The integrated object should then comply with all interface requirements of the individual control objects. For example, global vehicle management and order release could be integrated, such that the global

vehicle manager also determines the order release sequence in the terminals to ensure that the planning of the global vehicle manager is followed as closely as possible (see Chapter 4). We describe the basic behavior of the control objects. This behavior can be overruled in the case of integrated planning, when several decisions are taken simultaneously (coordinated).

### ***Transportation and logistic handling control***

#### *Local empty vehicle management*

Every time an AGV becomes idle (after unloading) or when an empty AGV arrives, the AGV can be assigned to a specific job. Depending on corresponding priorities, such as a latest departure time, the local empty vehicle manager can assign the AGV to an *empty vehicle job* or a *transportation job*. It may be possible to combine the jobs; a transportation job is selected which has the same destination as the *empty vehicle job*. In this way, two jobs are fulfilled at once. When no empty vehicle jobs and transportation jobs are available, the AGV receives a destination from local vehicle control. If a transportation job is selected, the AGV is handed over to vehicle scheduling.

#### *Vehicle scheduling*

A loaded AGV has to be assigned to an unloading dock. The nearest available unloading dock for a loaded AGV is selected because in this way the AGV is unloaded as quickly as possible. When no unloading dock is available, a destination will be assigned to the AGV by local vehicle control.

An empty AGV should be assigned to a load job. If multiple load jobs are available, the AGV is assigned to the dock with the highest priority order (see order release) that is released but not yet scheduled (no vehicle is assigned).

Vehicle scheduling is also called when an unloading dock becomes available or an order is released (new load job). Vehicle scheduling tries to select an AGV for the loading/unloading job. For both jobs the nearest AGV is selected; this matches with the other control objects and ensures that dock utilization is maximized.

### ***Transportation control***

#### *Local vehicle control*

AGVs that cannot be assigned to a specific job have to receive a destination. A loaded AGV will be sent to the parking place closest to the first available unloading dock. This maximizes dock utilization and minimizes the waiting times of the loaded AGVs. Information about start-times of the dock operations and the average unloading time is required for this rule (cf. Section 2.6).

An empty AGV can be positioned at an idle loading dock (no transportation job present), or at a parking place. Again, the parking place closest to the first available loading dock will be selected. If there are several idle loading docks, the AGV is sent to the nearest loading dock. This pre-positioning minimizes waiting times when a transportation job arrives. If no new transportation jobs are expected in the near future, the empty AGV might be sent to a parking area.

### *Routing control*

Routing control has to assure that the vehicles arrive at their destination. Routing can be done dynamically, in such a way that at every branch the currently fastest route to the destination is determined. Such a routing strategy can be important when there are a lot of alternative routes and the vehicles hinder each other on the routes. We implement routing control according to a static routing algorithm. On routes between terminals for much of the time there are no real alternatives. The routing of vehicles is implemented through a standard shortest path algorithm (Dijkstra, 1959). This shortest path algorithm is solved once for all origin-destination pairs. The results of these computations are stored in the layout. All branches have a destination list, indicating which direction should be followed, given the destination of a vehicle. Therefore, all vehicles travel along the shortest path from origin to destination; no routing flexibility is incorporated.

### *Traffic control*

TRACES (TRAffic Control Engineering System) is a traffic-control system which can handle high traffic densities and which was developed by Evers et al. (1999, 2000) and modified by Verbraeck et al. (1999). TRACES fulfils the tasks of managing the scarce infrastructure by providing routes to AGVs and by safeguarding potentially unsafe parts of the infrastructure (e.g. because of collision risk). Analogous to the control systems at the network level, these tasks are decentralized: the AGV executes its *script* that contains script statements, describing the route to take and the locations along the route where a conflict might arise. The AGV gets its script from the *script dispatcher* control object, which has a virtual map of the terminal. When executing its script, the AGV requests access to conflict locations, such as junctions or crossings, at local semaphores. If successful, the AGV receives a ticket, which it returns after leaving the conflict location. The scripts, which can be assigned by the script dispatcher, are based on a wide range of conditions, such as the traffic density in different areas, the destination of the AGV, information about failures, and the actual status of the AGVs battery. Furthermore, intelligence can be added to the scripts as well, so that the AGV can select the least dense route dynamically. Notice that routing control is incorporated in this concept. We model traffic control rather simply. As long as there is capacity on a track a vehicle can enter the track. On larger track configurations, such as roundabouts, we limit the number of vehicles: a semaphore is used to control the area. This semaphore is necessary to prevent deadlocks. The value of this semaphore is determined experimentally. Deadlocks also have to be prevented on terminals. A terminal semaphore is also used to reduce the traffic on the terminal, thereby increasing the average speed on the terminal by reducing the number of interactions.

### *Distance control*

Distance control should be included to ensure that vehicles always remain at a safe distance from other vehicles. This minimum distance depends on the speed of the vehicle and the communication and reaction times. If a vehicle breaks down, the other vehicles should still be able to stop, without colliding. By using such distance

control, vehicles should continuously check the intermediate distance with respect to other vehicles. This introduces a lot of events. We implement a simplified distance control, which is not implemented in the vehicle. It can be considered a separate control object installed at the entrance of bottleneck infrastructure or the exit of a certain track object. When it appears that safety distances between AGVs have serious impact on system performance, the model can check the intermediate distances at certain locations, for example the exit of a terminal and the entrance of a two-way track. If the intermediate distance is smaller than the required distance, the last AGV will have to wait until the required distance is reached. We control the intermediate distance at the entrance of the two-way track. Assume that 60 AGVs are waiting in front of the two-way track and have to enter this two-way track. When an intermediate distance of 2 seconds is required between AGVs, given the maximum speed and maximum deceleration, it will take two extra minutes for all AGVs to enter the two-way track. Therefore, AGVs at the other end also have to wait 2 more minutes before they can enter the two-way track. Neglecting this effect might have a large impact on system performance. Intermediate distances could also be controlled at other locations in the network. We did not implement this because the AGVs in the simulation model seldom leave nodes less than several seconds apart.

### ***Logistic handling control***

#### *Order acceptance*

In our model we assume that all transportation jobs are accepted. We do not have sufficient information about the revenues of the different transportation jobs. Furthermore, we want to determine resource requirements for a future system. This system should be able to handle the projected transportation demand. When information on actual revenues of transportation jobs is available one can estimate the chance that a transportation job with higher revenues arrives in the near future (cf. Yield management, Smith et al., 1992).

#### *Consolidation control*

On the terminal several rules can be used to consolidate cargo to unit-loads. Volume and weight in particular are important for the physical possibility of consolidation. Furthermore, the destination and due times are important data used to determine whether it is attractive to combine several cargo units. Here, we do not take consolidation into account, but rather we assume that all transportation jobs concern unit-loads.

#### *Order release*

Order release is the assignment of a load to a dock when a place at a dock is or becomes available. At each dock station, one or more loads can be placed in a cargo buffer, waiting to be loaded. When a load enters a terminal of the OLS system, it is assigned to the loading dock with the shortest queue of unprocessed transportation jobs. If all cargo buffers of dock stations are occupied, the load waits in the terminal cargo buffer until a buffer position at a dock becomes available. At that time, the

transportation job with highest priority is released. Transportation jobs are released according to a priority rule based on minimum latest departure time. Recall that the latest departure time is the time at which the cargo should leave the terminal of origin to arrive with a high probability of arriving on time at its destination terminal.

#### *Task allocation to docks*

Task allocation to docks is based on known and expected transportation jobs and related to the number of loading and unloading activities. Preferably the period during which the task allocation at a terminal is fixed ( $T$ ) should be not too short and should be attuned to the frequency at which the balance between load and unload jobs changes.

Now, suppose that we have to decide on the number of dedicated loading and unloading docks for the next period with duration of  $T$  at time  $t_0$ . The amount of *unload jobs* is partly known from the number of loaded vehicles on the way to terminal  $i$  with expected arrival time before  $t_0+T$ . Other information, for example from demand forecasting, can be used to compute the total expected number of *unload jobs*. Some of the *load jobs* in the next period are known, and information on additional jobs can be demanded from demand forecasting. However, not all *load jobs* have high priority, whereas *unload jobs* on transport to terminal  $i$  occupy AGVs, and should preferably be unloaded without delay. Therefore, we give a higher priority to *unload jobs* than to *load jobs*. The procedure is as follows:

1. Calculate the number of unloading docks required to process all (predicted) *unload jobs* without much delay, given maximum dock utilization  $\rho$ . The maximum utilization  $\rho$  is a control parameter that can be used to avoid excessive waiting times at the docks.
2. Calculate the number of loading docks required for all (predicted) *load jobs*.
3. If there are no excess docks, priority is given to *unload jobs*.
4. If the total number of docks required for all jobs is less than the number of docks available, allocate subsequently additional docks to the task for which the utilization is highest.

Using this rule, the maximum utilization for loading and unloading is minimized heuristically. Because order arrivals are random, it can be expected that delay in order processing will be reduced.

#### *Dock control*

When an empty AGV enters the dock station, the dock manager has to check whether the cargo is on the dock and the transfer can start. When cargo arrives at the dock, the dock manager also has to check whether an empty AGV is waiting to be loaded. The technical details for the transfer of the cargo, such as the information exchange between AGV and dock, are not modeled. After unloading, the *dock manager* will send the cargo to the terminal cargo buffer, and the AGV requests a new destination from the *local vehicle manager*.

### ***Secondary process: Load-bearer control***

Load-bearers should be present when cargo has to be loaded. Therefore, relocation of load-bearers might be required to ensure that enough load-bearers are available at the different terminals, especially if there are unbalanced transportation flows. We assume that there are always load-bearers available, and we do not explicitly model load-bearer control.

## **2.6 Information structure and information objects**

Information exchange between objects is required in a local control concept. First, we distinguish information that becomes available during the process, but cannot be influenced, such as transportation demand. Secondly, for the evaluation of the system, we need performance information as model output; this is only recorded and not used by other processes. Thirdly, we distinguish information that can be used to improve control decisions (see Figure 2.6). The purpose of the information structure is to ensure that the necessary information is at the right place at the right time, with a minimum of information exchange.

We distinguish two approaches to obtaining the required information from other objects: the pull approach and the push approach. When using the pull approach, information is requested from a specific object when it is needed. This approach prevents information exchange at moments when this information is not needed by other objects. It might not always be possible to use the pull approach. For example, when communication with AGVs is not possible at all times, the AGVs cannot be asked for the value of their attributes and the push approach should be used. In the push approach an object communicates all its attribute changes to the objects that require this information. The object that receives this information maintains the information in its local database. Information that requires a reaction or response should always be push, for example the completion of a job. A disadvantage of this approach might be the enormous amount of information exchange and high data storage requirements. For example, each time an attribute of an AGV changes, communication will take place and this information has to be stored locally until it is needed for a control decision.

Depending on the information that has to be exchanged, one can choose either the pull or push approach. In trying to minimize the information exchange in the system, we could use the following strategy. Information that is required very often, but which does not change frequently, can best be communicated via a push approach. Information that changes frequently, but is not required very often, might better be communicated by using a pull approach. Between two information requests, the value of an attribute might change often, but the values in between two requests will not be used. Information that changes infrequently and is also used infrequently, can best be communicated by a pull approach. For cases of frequent change and frequent



use, both options can be considered and the amount of information exchange can be measured in a simulation study. Note that, in addition to information exchange, the time available to make a decision is also relevant. If a decision has to be made almost instantaneously, a pull concept could be impossible because of the communication time required.

### 2.6.1. Information exchange and communication lines

Information objects gather information from a set of physical objects, such as all AGVs, and this information is made available to the corresponding control objects. Attributes can be seen as the information object at the lowest level. In case of a push approach, the information will be stored in the accompanying information object at the same level. Now we will specify the most important communication lines (see Figure 2.6) and the information that is exchanged on these lines. For the overview, control and information objects are only separated for the higher levels (network and terminal). The terminal control object is the aggregate of all control activities that take place within the terminal. The same holds for the other control and information objects in Figure 2.6.

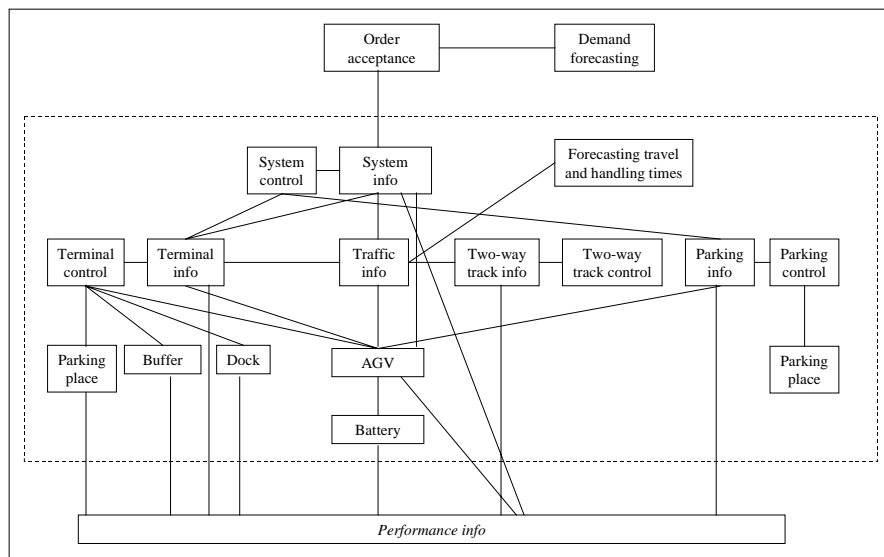


Figure 2.6. Information exchange between control and information objects

#### General control information

##### Forecasting travel times and handling times

Travel times and handling times play an important role in some of the control activities (especially vehicle management and task allocation to docks). Fluctuations in these times arise from fluctuations in workloads. Forecasts of travel times between terminals (system level) and on a terminal can be made. Actual travel times

and handling times are gathered in the model. Based on actual realizations the expected travel time and handling time can be estimated. The handling time is the time from the arrival at a dock at the terminal of origin until the unloading is finished at the destination terminal.

Note that the mean and standard deviations of the handling time ( $H_{ij}$ ) can be estimated using a standard exponential smoothing procedure (see e.g. Silver et al., 1998). The mean  $E[H_{ij}]$  is estimated by  $\hat{\mu}_{ij}^{(n)}$  :

$$\hat{\mu}_{ij}^{(n)} = \alpha \hat{\mu}_{ij}^{(n-1)} + (1-\alpha) \hat{H}_{ij}^{(n)}$$

where  $\hat{H}_{ij}^{(n)}$  is the  $n^{\text{th}}$  observation of the handling time for a transportation job from  $i$  to  $j$ ,  $\hat{\mu}_{ij}^{(n)}$  is the estimate for  $E[H_{ij}]$  after  $n$  observations and  $\alpha$  is a smoothing factor (e.g.,  $\alpha = 0.8$ ). The standard deviation  $\sigma[H_{ij}]$  is estimated by  $1.25 * MAD_{ij}^{(n)}$ , where  $MAD_{ij}^{(n)}$  denotes the mean absolute deviation after  $n$  observations. This quantity is also estimated by exponential smoothing:

$$MAD_{ij}^{(n)} = \alpha * MAD_{ij}^{(n-1)} + (1-\alpha) * \left| \hat{\mu}_{ij}^{(n)} - \hat{H}_{ij}^{(n)} \right|$$

These expected travel times and handling times are used in the planning procedure for the AGVs and in setting priorities for transportation jobs (cf. Chapter 4), but also to determine expected arrival times of AGVs at terminal entrances or two-way tracks. The latter information can be used to improve two-way track control (see Chapter 5).

From the expected travel times and handling times, the latest departure time for a transportation job is derived. The latest departure time is used as a priority rule for scheduling the transportation jobs (cf. Chapter 4). The latest departure time,  $t_{L,n}$ , is set such that the due time for job  $n$ ,  $t_{D,n}$ , is met with high probability:

$$t_{L,n} = t_{D,n} - E[H_{ij}] - k\sigma[H_{ij}]$$

where  $k$  is a safety factor. The term  $k\sigma[H_{ij}]$  allows for fluctuations in the handling time, caused by:

- random loading and unloading times;
- variation in travel times between locations in the network depending on local origin/destination (dock in terminal, place in parking area);
- unpredictable waiting time at the destination terminal, depending on workload.

We assume that  $H_{ij}$ , which is the sum of the loading time, unloading time, travel time and waiting time, is normally distributed. Hence, a reasonable value for  $k$  can be found from the table for the standard normal distribution. For example,  $k=2.33$

gives approximately a 99% probability that the transportation job will be handled before the due time, given that it leaves the terminal of origin at  $t_{L,n}$ . By including the variation in the handling time, we set priority for an order that has higher uncertainty in handling time, for example because the distance is larger or because the destination terminal occasionally faces congestion.

### *Demand forecasting*

Demand forecasting predicts the transportation jobs for a specific future time period. Demand forecasting can take place at system or terminal level. Information about future orders is useful for task allocation to docks and empty vehicle management. Possibly, transportation jobs are known for a certain time horizon. The terminal managers might forecast transportation jobs for each destination, and a possible priority class within a given time horizon, which is longer than the horizon of known jobs. Forecasts can be based on historical data using standard procedures (cf. Silver et al., 1998). We use the average transportation flows derived from historical data (see task allocation to docks).

### **Communication**

#### *Order acceptance to system info: transportation jobs (push)*

Key input to a model of a transportation network are transportation jobs. A transportation job corresponds to transporting one unit-load from origin to destination with a release time (arrival time at the terminal of origin) and a due time (the time at which the cargo should be available for further transport). Once a transportation job is announced, the information is available in the system information object.

#### *System info to terminal info: transportation jobs (push)*

Transportation jobs should be announced to the terminal of origin of the job. The terminal control objects can use this information, e.g. for task assignment to docks and order release. This information is needed frequently and does not change, so we chose a push approach.

#### *Traffic info to system info: AGV location (pull)*

The traffic system contains sensors (induction loops), where all passages of AGVs are recorded. At the moment that an AGV passes such a sensor, the information is recorded in the traffic information object. The sensors naturally have a push approach; otherwise several sensors have to be approached to locate an AGV. Upon request, *traffic information* can give the approximate location of an AGV to system information (pull). This information can be used to determine the approximate location of an AGV and the expected arrival time of an AGV at its destination or at other places on its route.

#### *Traffic info to two-way track info: expected arrival time of an AGV (pull)*

The traffic information object has information about the approximate locations of the AGVs. Once the destination of an AGV is known, the route is given (static

routing) and therefore also the two-way tracks on this route. The traffic information object can provide the two-way track control object with information about the expected arrival times of AGVs at the entrance of the two-way track. Either a push or pull approach can be used in this respect, depending on the frequency with which this information is used. Considering the decision frequency and the number of information updates, we chose to use a pull approach in our simulation model.

*Track system to system and terminal info: distances (pull)*

For several control activities at system and terminal levels, distances between locations are required, e.g. to find the nearest terminal or dock. This information can be obtained from the track system. Since this information does not change, the system information object requests this information from the track system only once. From the moment that the distance matrix is known, no further communication is required. The same holds for the terminal information object, but only distances on the terminal are of interest. Another option is to use travel times instead of distances. Travel times can change over time, but can be requested from traffic information (pull).

*Dock to terminal control: status and functionality (push)*

Terminal control objects need to know the status and functionality of the docks for several control activities, and also the starting time of the last dock operation and the average dock operation time. When this information is needed, they can obtain it from the dock objects. Either a push or a pull approach can be used; we chose a push approach. When the status of the dock changes, for example the dock becomes idle, terminal control is notified and can directly take action.

*AGV to terminal control: status (pull), presence (push)*

The terminal control objects need information about the status of the AGVs on the terminal. This information is requested from the AGVs. When an AGV enters a terminal, the responsibility for this AGV changes from the global vehicle manager to the local vehicle manager. The *terminal information object* will maintain a list of the AGVs that are within the terminal domain.

*AGV to system info: trip finished, status, destination (push)*

After finishing a particular trip (empty trip or a transportation job), an AGV communicates with *system information*. Realizations of travel times and handling times are recorded for all origin-destination pairs. The handling time is the time from the start of loading the vehicle until the unloading is finished. The AGV also communicates the battery charge that was required for a particular trip, using information from the battery. The required battery charge for the different routes is important information for the secondary process of battery management.

Control objects at system level, especially the global vehicle manager, need information about the status and destination of the AGVs. Because this information is needed frequently, we chose a push strategy. The system information object records and updates the information for all AGVs in the system.

*Parking place to terminal and parking control: status, functionality (pull)*

The terminal and parking control objects might need information about the status, i.e. the number of available places, and the functionality, loaded or empty, of the parking places. This information is requested whenever necessary. Because this information is not needed very frequently, and also the information does not change frequently, we chose a pull approach.

*System control to terminal and parking info: empty car jobs (push)*

When there is central coordination the *global vehicle manager* (part of system control) can send (push) empty vehicle jobs to the terminal and parking information object. Terminal control and parking control will take this information into account in local vehicle management.

*All information objects report to performance measurement: information for evaluation (push and pull)*

For all objects in the model, performance information can be recorded, for example the utilization of docks and AGVs or the throughput time of transportation jobs. These figures might be requested, for example every half-hour (pull), or can be submitted after finishing a specific event, such as the fulfillment of a transportation job (push). These figures can be aggregated over all objects in a certain class (AGVs), or for a certain object (terminal).

## **2.6.2. Performance information**

The model should provide performance information that can be used to evaluate several alternatives with respect to system layout and control methods. A lot of information can be extracted from the model. Examples of performance information are shown in Table 2.2.

**Table 2.2.** *Performance information*

<b>Objects</b>	<b>Performance information</b>
System	number of orders on time, throughput time distribution failure registration (docks and AGVs)
Terminal	average dock utilization
Parking area	occupation
Battery station	number of battery changes, average lost charge
Two-way track	average waiting times, convoy lengths
Dock	capacity utilization, failures
Buffer	capacity utilization, maximum occupation
Vehicle	driven distance (loaded and empty)
Cargo	on-time, throughput time

At system level, aggregated figures are available on the transportation jobs. After completion of a transportation job, we record whether the job was late, if late how late it was and the throughput time, i.e. the time the job was completed minus the release time. From these figures, the number of late orders and the throughput time distribution can be determined per origin-destination pair, and can be aggregated for the total system. To evaluate system performance, the most important figure from a customer's point of view is the service level. This level indicates the percentage of transportation jobs that is delivered on time at its destination. From the point of view of cost, minimizing the empty kilometers might be attractive. Other important information relates to the utilization of the resources. Low utilization throughout the entire day may indicate that the number of docks or AGVs can be reduced. On the other hand, high utilization rates may indicate a shortage of resources. If failures are included in the model, these are recorded at system level. The number of failures, the location of the failures and the duration of the failure are recorded. Depending on the objectives of the research, one or more of these performance measures can be used.

## **2.7 Summary**

In this chapter we described an *object structure* for automated transportation networks, in which we distinguished physical, control and information objects. The processes in such a transportation network were divided into primary and secondary processes, and subdivided into transportation and logistic handling processes. For the different processes we distinguished flow, handling and storage units. Based on this classification we designed an *object model* for automated transportation networks. We described the function of the physical, information and control objects and the relations between these objects. Furthermore, we designed the control objects that are not the subject of further research in this thesis, and described performance measures that can be used in the evaluation of several alternatives with respect to system layout and control methods.



# Chapter 3

## Case description OLS: the role of simulation

### 3.1 Introduction

In June 1994, the first discussions started about the possibility of using underground freight transportation between Amsterdam Airport Schiphol and the flower auction market at Aalsmeer. In this chapter we describe the OLS-project, which is used as a case study in this thesis. Our aim is to discuss the position of our mainly operations research and simulation based research within the project. The main goal of the simulation group within the OLS-project was to develop and evaluate logistic control rules and to determine the logistic performance of the system in relation to layouts and capacity. We first describe the project history in Section 3.2. Next, the activities of the transportation and control technology group within the OLS-project, as part of the pre-design phase, are described in Section 3.3. The results of this research, for example terminal, dock, AGV and control system design, constitute input for our models. In Section 3.4 we describe the relevant estimates with respect to transportation flows. As one of the project goals is system dimensioning, in Section 3.5 we show to what extent analytical models can be used for this purpose. We develop a model that provides a rough estimate of the resource requirements and apply it to one of the proposed system layouts. As we will see, analytical models are only partially useful for the estimation of resource requirements, and therefore we also use simulation to determine more accurate resource requirements. Simulation is applied to all the system layouts under consideration. In Section 3.6 we summarize our contribution to the overall project and its relation to contributions from other



groups. Furthermore, we summarize the figures that will be used as input for the simulation model in subsequent chapters.

## **3.2 Project structure**

Three phases of the project have been finished since the start of the project: a feasibility study, a definition study and the pre-design phase. Our research was part of the pre-design phase. In this section we describe some results and conclusions from these three phases.

### ***3.2.1. Feasibility study***

In 1995, the managers of Amsterdam Airport Schiphol and the flower auction market at Aalsmeer initiated a feasibility study to investigate the possibility of using underground freight transportation. The Ministry of Transport, Public Works and Water Management, and the Holland International Distribution Council (HIDC) supported this initiative. A feasibility study (Van de Geijn and Katgerman, 1996) was designed to determine whether an underground logistic system between Schiphol Airport, the flower auction market, and a rail terminal could be justified on spatial, logistical, technical, organizational and economic grounds. The underground logistic system was intended to transport time-critical products, such as flowers, dairy produce, newspapers and spare-parts. Speed, reliability and timeliness are very important to such a system. The conclusions of the feasibility study were positive, but further research was required to determine the system layout, the diameter of underground tubes, the operating costs and the preconditions to be fulfilled in order to generate an OLS associated, high-quality rail product, which does not yet exist.

### ***3.2.2. Definition study***

Following on the feasibility study, a definition study was performed on the initiative of Amsterdam Airport Schiphol, the flower auction market at Aalsmeer, NS Cargo, Railned, the Ministry of Transport, Public Works and Water Management, the Holland International Distribution Council, the Air Transport Association Netherlands (ATAN) and the Center for Transportation Technology (CTT). The main conclusions of this definition study were (CTT, 1997):

- The OLS contributes to the accessibility of Amsterdam Airport Schiphol and the flower auction market at Aalsmeer.
- The OLS relieves the roads in the neighborhood of Schiphol and Aalsmeer, and as a result reduces the burden on the environment.
- For Schiphol and Aalsmeer, the OLS is the missing link with rail transportation.
- The transition from road transportation to the OLS and rail transportation is feasible on economic grounds, given the forecasted growth in transportation flows.

- More research is required to corroborate the choice between a tube diameter of 3.5 or 5 meter.
- More research is needed to compare various system designs.
- More research is required on the technical and logistical control of AGVs, the design of loading/unloading equipment, and on drilling in soft soil.

### **3.2.3. Pre-design phase**

The results of the definition study were sufficiently positive for the pre-design of the system to be started. The goal of this pre-design phase was to refine the results of the definition study and to prepare plans for implementation. Some objectives of this phase were (Van der Heijden et al., 2000):

- to obtain adequate information for a go/no-go decision for the next phase, in which the first part of the system will be constructed.
- to deliver an optimized system design to the future owner of the infrastructure.
- to deliver tested AGVs and loading/unloading technology to the system operating company.
- to design new control systems and hand these over to the operating company.
- to test the prototype AGVs and loading/unloading docks, with their control systems, at a test site.

Several research disciplines cooperated intensively to perform an integral assessment of this new transportation system, to optimize the system design, and to explore its technological and economical viability. Among these research disciplines are underground construction technology, AGV technology, automated docking and warehousing technology, information and communication technology, control technology, logistics, simulation, and economics. An important aspect was to continuously involve a user-group in the definition of the research questions and the evaluation of the research results. In the pre-design phase, more than a hundred researchers and developers at universities and research institutes, as well as customers and business partners were directly involved.

Three different project groups were installed. One project group focused on the management and exploitation of the OLS, another project group on underground construction, and still another group on the transportation and control system of the OLS. In the next section, we describe the activities of the project group on transportation and control technology. For the results of the other two project groups refer to the reports from Schiphol Project Consult (2000) and Centrum Ondergronds Bouwen<sup>1</sup> (2000).

During the pre-design phase, several assumptions and specifications changed owing to additional insight obtained from the research. Examples of changes are the system design, transportation flows, and AGV specifications such as speed and length, and

---

<sup>1</sup> This is the Dutch name for “Center for underground construction”

docking times. Where relevant, we indicate which specific assumptions were used for which part of our research.

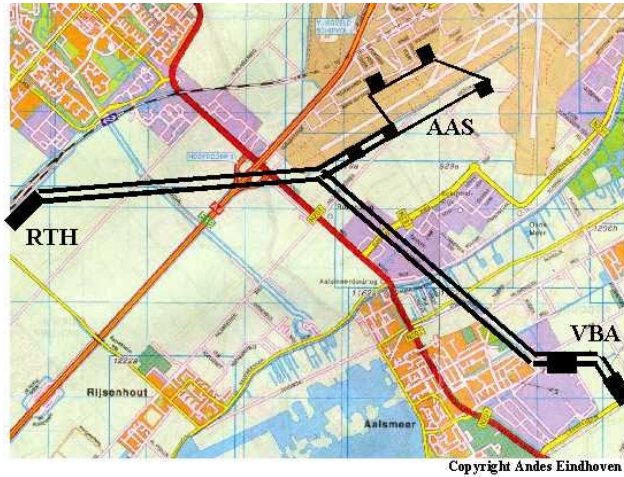
At present (March 2001), some details relating to questions that arose in the pre-design phase remain to be filled in. The AGVs and docks are still being tested at the Test Site to decide about a final AGV design. Furthermore, the internal processes at Schiphol Airport are being investigated to connect these processes to the OLS in the best possible way, thereby considerably reducing the internal transportation at Schiphol Airport. Simulation studies still play an important role in this part of the project. By the end of 2001 a decision should have been made about the implementation of the OLS.

### **3.3 Sub-project transportation and control technology**

In this section, we describe some aspects of the research done within the transportation and control technology group. The activities regarding the design of logistic planning and control rules and the network simulation were concentrated within this group. Here we give a brief summary of the main activities. In the subsequent sections we distinguish system design, functional requirements, vehicles, terminal and dock design, operating and information system, simulation and the Connekt Test Site. For more details and results refer to Rijsenbrij et al. (2000).

#### ***3.3.1. System design***

The OLS-system consists of three main locations: a Rail Terminal near Hoofddorp or the Zwanenburg runway (RTH or RTZ, respectively), the flower auction market at Aalsmeer (VBA = "Verenigde Bloemenveiling Aalsmeer") and Amsterdam Airport Schiphol (AAS). One of the layouts that resulted from the definition study, and that was used for further research within the simulation group, is the Stommeer layout (see Figure 3.1). RTH, consisting of one terminal, is located in the west, VBA, consisting of two terminals, in the southeast, and AAS, consisting of five terminals, in the north. The required number of terminals at each location, based on costs and expected transportation flows (CTT, 1997) was determined in the definition study. AGVs that are not needed for a while, or for which there is no space at the terminals, are dispatched to a parking area that is located just south of Schiphol Airport, close to the central intersection. This central parking area is not shown on the map.



**Figure 3.1.** *Stommeer layout*

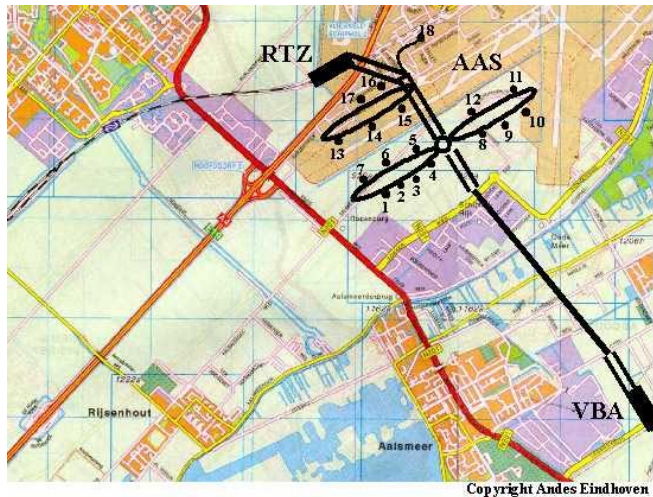
The plan requires that all terminals and the parking area be built on or just below the surface, while the tube system is underground, about 15 meters below the surface, so slopes must be constructed between each terminal entrance/exit and the tube system. The slopes will have to be approximately 150 meters long. A tube diameter of 5 meters was selected to facilitate transport of complete 10-foot main-deck aircraft pallets. A large part of the system has two-way traffic, but the five terminals at Schiphol Airport are connected by a loop where only one-way traffic (counter clockwise) is possible. To give an idea of the scale, the distance between RTH and VBA is approximately 10 kilometers; the loop at Schiphol Airport is about 5 kilometers. The Stommeer layout is considered in the experiments in Chapter 4, but after financial studies and simulation experiments, it turned out that this layout would be too expensive, especially owing to the long tubes that have to be bored between RTH and VBA. These routes cannot be constructed at surface level because of canals, roads and other barriers. The costs of underground construction for a tube with a diameter of 5 meters is about \$20.000 per meter for the concrete alone, without any of the necessary facilities inside the tube. Considering a total tube length of 20 kilometers between RTH and VBA, this results in investment costs of \$400 million. To reduce the investment costs considerably, two-way tubes (only one track for two directions) were introduced. The layout with two-way tubes is shown in Figure 3.2, in which the distance between RTH and VBA is about 12 kilometers, instead of the 10 kilometers of the previous layout. This is due to a change in the layout at Schiphol Airport.



**Figure 3.2.** OLS layout with two-way tubes

Three two-way tubes are present in the layout, one between RTH and AAS (length 2200 meter), one between AAS and VBA (2700m) and one at Schiphol Airport (750m). Further investment reductions were obtained by reducing the number of terminals from 5 to 3 at AAS, and from 2 to 1 at VBA. The parking area (not on the map) is located at Schiphol Airport, near the left terminal (AAS1). The layout with two-way tubes is used in the experiments described in Chapter 6.

The reason for designing yet another layout was the request of Schiphol Airport to adjust the design, such that the system can also be used for internal transportation at Schiphol Airport (see Figure 3.3). The business parties at Schiphol want to reduce the number of freight trips on the airport; at present most trips at Schiphol carry less than a truckload. At Schiphol Airport 1<sup>st</sup> and 2<sup>nd</sup> line terminals are distinguished. The 1<sup>st</sup> line terminals are located at the loading and unloading platforms of aircraft. Terminals 5, 6, 7, 12, 14, 15 and 18 are 1<sup>st</sup> line terminals, the remaining ones are 2<sup>nd</sup> line terminals. At present, trucks deliver to 1<sup>st</sup> line and 2<sup>nd</sup> line terminals, and the size of their deliveries varies from 1 box to a full truckload. Once the OLS is operational, trucks should deliver and pick-up cargo only at the 2<sup>nd</sup> line terminals. In the future, the OLS will take care of the transportation between 2<sup>nd</sup> line and 1<sup>st</sup> line terminals. Possibly, some of the small 2<sup>nd</sup> line cargo handling companies will not have their own connection to the OLS. For these companies, a cargo consolidation center can be constructed, where they can deliver and pick up their loads. This cargo consolidation center will have its own OLS terminal. In this way, the truck movements around Schiphol Airport can be reduced. Note that freight from several deliveries, but with the same origin and destination terminals within the OLS system, can be consolidated to form one transportation unit, thereby reducing the number of trips.



**Figure 3.3.** OLS with internal transportation system at Schiphol Airport

The number of terminals at Schiphol increases significantly in comparison with the previous layout, from 3 to 18, but the size of the terminals decreases. The AAS terminals are located on three loops at Schiphol Airport, with one isolated terminal. Terminal investment costs increase, but the costs of the tube network are decreased by moving the rail terminal from Hoofddorp to Schiphol Airport. The two-way tube between the rail terminal and Schiphol Airport is removed, while the two-way tube between VBA and AAS remains. Notice that the distance between RTZ (renamed from RTH because of the location change) and VBA is almost halved, from more than 10 to about 5 kilometers. The loops at Schiphol Airport are between 2 and 3 kilometers long. The parking area is not represented in the figure, but is located on the right loop at Schiphol (East). This layout is used in Chapters 5 and 6.

### **3.3.2. Functional requirements**

It is important for a successful implementation of the OLS that the system complies with the expectations and demands of its users. The demands should be clear before the system is designed. These demands are related to (Rijsenbrij et al., 2000):

- Transportation conditions: in particular temperature and humidity.
- Reliability: with regard to throughput time requirements and service levels.
- Availability: failures should have limited impact.
- Accessibility: all users should be able to deliver their goods to the system.
- Spatial planning: existing infrastructure should be taken into account.
- The environment: the negative environmental effects should be limited.

Simulation studies are playing an important role in assessing both the reliability and the availability.

The OLS has to transport several cargo types: main-deck aircraft pallets, Euro-pallets, industry pallets, Danish carts and flower auction market carts. All these cargo types have different weights and dimensions. We define a transportation unit (TU) as an amount of cargo, which is handled by the OLS as 1 unit. The loading capacity of an AGV can be expressed in TU. In the OLS-case a TU has the same dimensions as a 10 foot main-deck aircraft pallet:  $L \times W \times H = 3.18 \times 2.44 \times 3.00$  meter (Rijsenbrij et al., 2000). Given these dimensions, a TU can consist of:

- 1 main-deck aircraft pallet
- 6 industry pallets
- 6 Euro pallets
- 8 Danish carts
- 4 flower auction market carts

The maximum weight of a TU is 3500 kilo. Special load-bearers could be required to make possible the standardization of the handling of these different cargo types, for example for loading and unloading of AGVs and to secure the cargo.

### **3.3.3. Vehicles**

Automated Guided Vehicles (AGV) transport the cargo between the terminals of the OLS. Three prototypes were designed and built, one of which is shown in Figure 3.4. The prototypes are based on different characteristics, but with some general specifications, which are used in the simulations (Rijsenbrij et al., 2000):

- Normal speed: 6 m/s
- Maximum speed on terminal: 2 m/s
- Speed on slopes up: 3 m/s
- Speed on slopes down: 6 m/s
- Capacity of an AGV: 1 TU
- Maximum weight of one TU: 3500 kilogram
- Maximum acceleration/deceleration in a normal situation:  $1 \text{ m/s}^2$
- Maximum acceleration/deceleration in case of an emergency stop:  $2 \text{ m/s}^2$
- AGVs drive on rubber wheels in the terminals
- AGVs are free-ranging in the terminals
- AGVs are driven by an electric motor
- At the terminals electricity is supplied by batteries

The three prototypes are used to test differences between two and four wheel steering, rubber wheels and steel wheels on rails, rail guidance, electrical guidance using transponders or a magnetic grid, and self-guidance of the vehicles. For more information on these vehicle designs refer to Rijsenbrij et al. (2000). A special recovery vehicle will be present to tow failed AGVs. This recovery vehicle has not yet been designed. The AGVs are controlled according to a local control principle. All AGVs control their own speed and distance with respect to other AGVs or obstacles (Verbraeck et al., 1998a).



**Figure 3.4.** One of the prototype vehicles while docking: the Lödige vehicle (Rijsenbrij et al., 2000)

To prevent collisions, a specific distance should be maintained between two AGVs. The safest solution is to use the *brick-wall principle* (Verbraeck et al., 1998a): an AGV should be able to stop without colliding when the AGV in front stops instantaneously. In the OLS-case a minimum intermediate time between two AGVs (front-front) of 3.5 seconds is used (Rijsenbrij et al., 2000). This can be subdivided into the time required for breaking, the mechanical reaction time ( $t_{mech}=0.2$  seconds), the control reaction time ( $t_{ctrl}=0.5$  seconds) and a safety margin of 2 meters ( $d_{safety}$ ). The length of an AGV ( $l_{agv}$ ) is 6 meters, which accounts for another second. In case of a speed ( $v$ ) of 6 m/s and a maximum deceleration ( $a$ ) of  $-2 \text{ m/s}^2$ , the required breaking distance is 9 meters. This can be easily computed from the formula  $d_{brake}=v \cdot t_{brake} + \frac{1}{2} \cdot a \cdot t_{brake}^2$  ( $t_{brake} = 3$  seconds). The resulting intermediate distance is:  $d_{intermediate}= d_{brake} + l_{agv} + (t_{mech} + t_{ctrl}) \cdot v + d_{safety} = 21.2$  meters (front to front), which equals an intermediate time of 3.53 seconds at a speed of 6 m/s.

### **3.3.4. Terminal and dock design**

Terminals have to satisfy several requirements, some of them mutually conflicting (Van der Heijden et al., 2000). These include, speed of AGVs versus safety and energy use, speed of docking operations versus reliability, and traffic density versus energy use and safety. Many criteria have had to be taken into account, such as area of the terminal ( $\text{m}^2$ ), facilities for efficient transshipment to trains, resource utilization (docks, parking places), duration of an AGV visit, and the number of loading/unloading operations per hour. In an iterative process with experts on automated transportation and transshipment technology, some alternative terminal configurations have been designed (Verbraeck et al., 1998b).

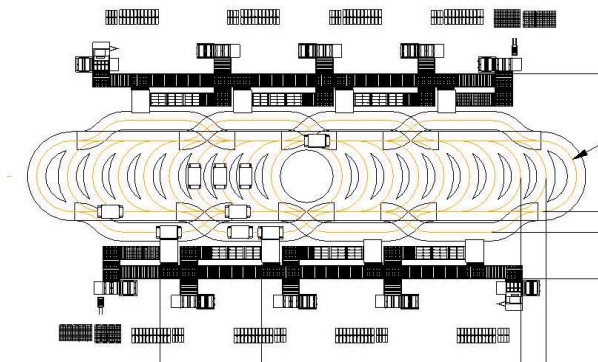
Simulation experiments showed that terminal concepts that seemed attractive from a spatial perspective, perform poorly in terms of transshipment capacity, simply



because AGVs cannot reach the docks in time. The conclusions from these terminal simulations were that (Verbraeck et al., 1998b):

1. The performance of the terminal benefits from limiting the number of AGVs in the terminal. Limiting the number of AGVs reduces congestion and as a result terminal throughput times decrease. This limit is called the terminal semaphore, because every AGV has to request a ticket for the terminal and a limited number of tickets is available. If the limit has been reached, AGVs have to wait outside the terminal until other AGVs leave the terminal.
2. Parking locations should not interfere with passing traffic.
3. Instead of minimizing the length of tracks there should be many tracks in the terminal to facilitate the distribution of the vehicles over the available space. This decreases the number of vehicle interactions. As a result, the average AGV speed in the terminal increases. Note that this is different from the design of train networks, where one tries to minimize total track length.

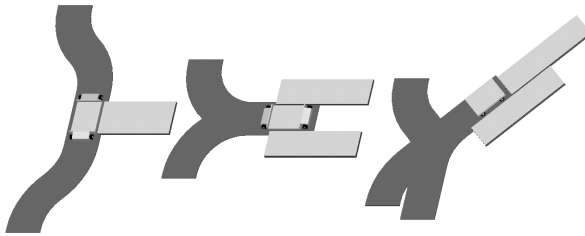
The preferred terminal layout is shown in Figure 3.5. Each dock has two places for AGVs, one for an AGV that can be loaded or unloaded and another place for an AGV that can already be positioned at the dock to minimize the time between subsequent dock operations.



**Figure 3.5.** Terminal layout (Rijsenbrij et al., 2000)

AGVs load and unload at the docks. Three different dock layouts were considered (see Figure 3.6). The advantage of the left dock layout is that the AGV always drives forward, which could simplify the AGV design. Cargo is transferred from one side. A disadvantage is the required space. For the dock in the middle the AGV must be able to drive in two directions. An advantage is that the AGV can be loaded and unloaded from two sides. The dock at the right has the advantage that the AGV can be loaded or unloaded from the front and from the side. After loading/unloading the AGV drives a few meters backward and drives forward again to enter the main stream. Driving back and forth takes some time, which is a disadvantage of this dock design. In a detailed simulation study, the left dock emerged as the dock with the best performance (Verbraeck et al., 1998a). This dock is used in all network

simulations. In the definition phase, the loading/unloading time was assumed to be 2 minutes, while this time was reduced to 1 minute during the pre-design phase. In the simulations, we assume that the loading and unloading times are gamma distributed, with an average of 2 minutes and a standard deviation of 15 seconds (later decreased to 1 minute and 6 seconds, respectively). The standard deviation is introduced because fluctuations in loading/unloading times may arise from cargoes of different type, size and/or weight.

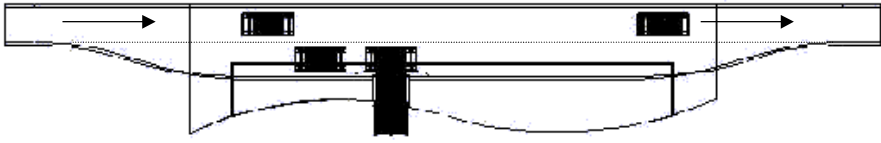


**Figure 3.6.** Three different dock layouts (Rijsenbrij et al., 2000)

For the loading/unloading operation, a choice had to be made between rolling and lifting. An advantage of rolling conveyor technology is that it is proven technology, and no lifting device is required. Lifting requires expensive construction and is riskier in relation to possible load/vehicle damage in case of failures. An advantage of lifting technology is that no roller conveyor is required within the AGV. The final choice with respect to rolling or lifting has not been made yet.

At the terminal entrance there are two line-up tracks, one for loaded and one for empty vehicles (not shown in Figure 3.5). Queuing in front of the terminal entrance may occur if an arriving vehicle cannot be assigned to a free destination in the terminal (dock, parking place), or if the maximum number of vehicles on the terminal has been reached (terminal semaphore). These line-up tracks operate according to the First-In, First-Out (FIFO) principle. Separate queues for loaded and empty vehicles offer flexibility to set priorities for either job type. Queuing may have an impact on the logistic performance, because loaded vehicles cannot overtake in the queue, whereas, given the different job priorities, this may be desirable. Loaded and empty AGVs are parked in separate parking places, to ensure that empty AGVs do not block loaded AGVs.

At Schiphol Airport space is limited and very expensive. Therefore, small terminals were designed with only 1 or 2 docks, no parking places and no line-up track at the entrance. A drawback is that AGVs that wait at the entrance of a terminal block all other traffic, because all AGVs drive through the terminal. There are only two waiting places next to the main track (see Figure 3.7), one at the dock and one place behind the AGV at the dock. The space at the right side of the dock is required for acceleration.



**Figure 3.7.** *Small terminal at Schiphol Airport (Rijsenbrij et al., 2000)*

The rail terminal at Hoofddorp is shown in Figure 3.5, but the design of the rail terminal near the Zwanenburg runway had to be different. The trains will be stationed in the middle of this terminal, while the docks are at the outside, with 3 places for AGVs at each dock. A bridge or tunnel is required for vehicles to drive to the other side of the terminal. Because the trains are in the middle of the terminal, there is no space for parking places or shortcuts. The terminal is very long and one trip around the terminal is about 1 kilometer. To reduce the impact of these long distances, the speed of vehicles on this terminal is increased to 4 m/s.

In all the system designs presented (see Section 3.3.1) there is 1 central parking area in the system. This parking area is used for AGVs that are not needed for a while, for example during the night. The parking area contains several parking lanes in which AGVs line-up. The capacity of the parking area should be sufficient to accommodate all AGVs. Maintenance facilities can also be located at or near the parking area.

### **3.3.5. Operating and information system**

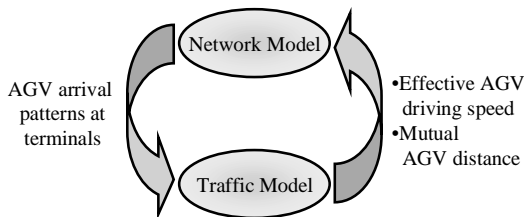
Another important feature of the OLS-system is the operating and information system. The most important tasks of such a system are (Rijsenbrij et al., 2000):

- Order acceptance and order processing.
- The planning of transportation jobs, such that the available resources are efficiently used.
- Operating the resources to ensure that these resources act at the right time at the right place.
- Supply of information on the actual state of the system, the history, the performance characteristics and other relevant information.

Some of the planning and control rules needed for the operating and information system are considered in this thesis, especially the higher levels concerned with transportation planning in the system as a whole. These rules were evaluated by using a simulation model, which is described in more detail in the next section. For more information on the lower levels of, for example, AGV and dock control, refer to Rijsenbrij et al. (2000).

### 3.3.6. Simulation

Simulation models were used continuously in the pre-design phase to act as a common structure, to bring together different research groups and to ease communication. *Two statistically linked simulation models* were constructed, see Figure 3.8: a network model, constructed from a top-down approach, and a traffic model, constructed from a bottom-up approach. These two models meet at the terminal level.



**Figure 3.8.** *Two linked simulation models*

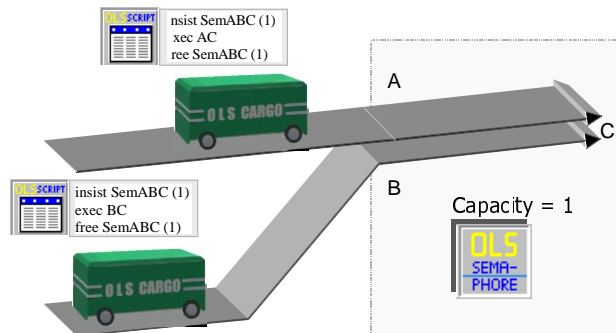
The network model mainly focuses on logistic network control, choice of a network design and logistic performance measurement. The traffic model mainly focuses on traffic control, terminal and dock design, and detailed AGV characteristics. These two models are linked by exchanging key AGV and order characteristics: AGV and order information flows from the network model to the traffic model, while information on the effective driving speed of AGVs and loading/unloading times flows in the other direction. From the traffic model it was determined that the average speed on the terminal depends on the number of vehicles on the terminal ( $N$ ). The average speed on the terminal shown in Figure 3.5 is  $2-0.024N$  m/s (Saanen et al., 1999). This relationship is used in the network model. The network model was described in detail in Chapter 2. The objects are the building blocks of an object-oriented simulation library, implemented in eM-Plant (Tecnomatix, 2000). Some OLS specific building blocks are included in the library, such as OLS-terminals. Different layouts or control structures can be assembled quickly from the tested library components. Let us briefly summarize the main contributions of the network model simulations to the overall project. The network model was used to design and evaluate methods for the higher-level control activities (cf. Chapters 4 and 5) and to determine the resource requirements for the three layouts in order to achieve an acceptable service level. Furthermore, the implications of secondary processes, battery management and failure management, and primary processes with restricted storage capacities were studied (cf. Chapter 6). The traffic model and its relations with the network model are described in greater detail below.

#### *The traffic model*

Two main challenges arose in relation to the terminal design. The first challenge was to design a traffic control system that is able to control dense traffic in a safe and efficient way. In order to find the best terminal design, a number of alternatives

was developed and evaluated by means of simulation. The second challenge was to support terminal design by using simulation, taking into account a number of conflicting requirements. For instance, spacious terminals provide AGVs the opportunity to maneuver without hindering each other, which is beneficial to the attainment of short throughput times. However, the available space is very limited because of existing infrastructure and the high ground prices in the region.

The existing AGV traffic control technology at container terminals is considered to be reliable, but too conservative in its track claiming and safety processes to allow for more flexible, higher speed traffic handling. Therefore, the TRACES-concept (TRAffic Control Engineering System) was adopted. This is an intelligent AGV traffic control framework developed by Evers et al. (1999). TRACES fulfils the tasks of managing the scarce infrastructure by providing routes to AGVs and by guarding potentially unsafe parts of the infrastructure (e.g. because of collision risk). Analogous to the control systems at the network level, these tasks are decentralized: the AGV executes its script that contains script statements, describing the route to take and the locations along the route where a conflict might arise. The AGV gets its script from the *script dispatcher control object*, which has a virtual map of the terminal. When executing its script, the AGV requests access to conflict locations, such as junctions or crossings, at local semaphores. If successful, the AGV receives a ticket, which it returns after leaving the conflict location. This mechanism is shown in Figure 3.9. The scripts can be assigned by the *script dispatcher* on the basis of a wide range of conditions, such as the traffic density in different areas, the destination of the AGV, information about failures, and the actual status of the AGV's battery. Furthermore, intelligence can be added to the scripts as well, so that the AGV can dynamically select the route with the least dense traffic. The implementation of TRACES in the terminal simulations and at a test site has shown that it provides a safe mechanism to route AGVs through a complex infrastructure.



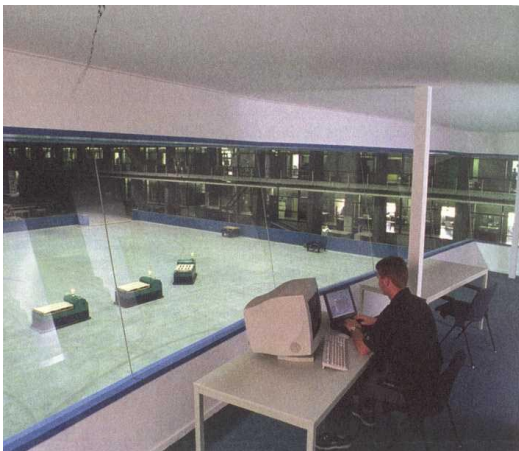
**Figure 3.9.** Basic principle of TRACES (source figure: Van der Heijden et al., 2000)

The network and traffic models can also be linked to form one simulation model, for example by exchanging information by using dynamic data exchange (DDE). Terminals modeled with the traffic model can then replace the terminals in the

network model, while running on different computers (distributed computing). The network model controls the AGVs outside the terminals and can give empty vehicle jobs to the terminal managers. Once an AGV enters a terminal in the network model, it is handed over to a terminal modeled by the traffic model. In this way the terminals from the network model can be validated by using the results from terminals with detailed vehicle behavior. One of the problems is that the models have to be synchronized.

### **3.3.7. Connekt Test Site**

Because of the scale of the OLS system and the novelty of the technology, it was decided to test the equipment and the control framework under laboratory conditions. To this end, a Test Site has been constructed at Delft Hydraulics|WL (see Figure 3.10). At this Test Site, ten 1:3 scale AGVs were built, as well as three prototype vehicles (1:1). Furthermore, twelve 1:3 scale docks and 3 prototype docks (1:1) were built to enable the transshipment processes between AGVs and other modes of transportation. At the Test Site, the traffic simulation model is used as a real-time control system.



**Figure 3.10.** *Connekt Test Site at Delft Hydraulics|WL (Rijsenbrij et al., 2000)*

Since only a small part of the OLS-system could be implemented at the Test Site, the other part of the system, including physical equipment, had to be simulated. Therefore, the Test Site control system was linked to the simulation. When the simulated system runs and a simulated AGV enters the terminal that has been implemented at the Test Site, a real AGV takes the place of the simulated AGV. Control methods that have been developed using the network simulation model, for example two-way track control (Chapter 5), can also be tested on their correctness.

### 3.4 Model input: transportation flows

Important input for the simulation model is the estimated transportation flows. The OLS system should be able to transport the projected transportation flows of 2020. These projections changed during the project owing to new information and changing insights. In this section, we describe the various estimates that were used during the project and we discuss which data have been used for which part of the research.

The estimated transportation volumes for 2020 are translated to transportation units (TU) and were initially estimated by the Netherlands Economic Institute (Table 3.1, CTT 1997). We clearly see an imbalance in these transportation flows; especially the route from VBA to RTH is very busy. These transportation flows are used in the experiments in Chapter 4.

**Table 3.1.** *Estimates in 1997 of the transportation flows in the year 2020 (x1000 TU)*

from ↓	to →	AAS	RTH	VBA
AAS		-	130	50
RTH		194	-	177
VBA		10	350	-

The project team adjusted the estimates of transportation flows during the project, which resulted in the new transportation flows as shown in Table 3.2. The total flow increased by about 9%. These transportation flows are the input for the simulations in Section 6.2 and 6.3.

**Table 3.2.** *Estimates in 1998 of transportation flows in the year 2020 (x1000 TU)*

from ↓	to →	AAS	RTH	VBA
AAS		-	122	67
RTH		182	-	197
VBA		10	414	-

New changes in transportation flows occurred at the moment the rail terminal was planned near Schiphol Airport (cf. Figure 3.3). Transportation flows between the 1<sup>st</sup> and 2<sup>nd</sup> line terminals at Schiphol Airport had to be added. The data for these flows were obtained from the parties involved at Schiphol Airport. The term AAS still indicates all terminals at Schiphol Airport, both 1<sup>st</sup> line and 2<sup>nd</sup> line. The transportation flows were not only estimated for 2020, but also for 2005, 2010 and 2015 (Table 3.3). From these flows a growth path can be computed, for example to assess the increasing resource requirements (Van Harten et al., 2000). These transportation flows are input in Chapter 5 and for the evaluation of storage capacity restrictions in Section 6.4.

**Table 3.3.** Estimates in 1999 of the transportation flows in 2005, 2010, 2015 and 2020 (x 1000 TU, Rijsenbrij et al., 2000)

From	To	2005	2010	2015	2020
AAS	VBA	28	37	50	67
VBA	AAS	7	8	9	10
VBA	RTZ	281	319	364	415
RTZ	VBA	135	153	174	199
AAS	RTZ	54	69	88	114
RTZ	AAS	80	101	130	165
1 <sup>st</sup> line	2 <sup>nd</sup> line	112	143	183	234
2 <sup>nd</sup> line	1 <sup>st</sup> line	92	117	149	191

The number of TU in an average month is 8.33% of the total in one year. The demand in one week is an equal part of the monthly demand (23%). The distribution of the demand over the days in the week and the hours on a day is represented in percentages in Table 3.4 and Table 3.5.

**Table 3.4.** Distribution of TU over days in the week in terms of percentage

From	To	Mo	Tu	We	Th	Fr	Sa	Su
AAS	VBA	30	15	15	20	10	5	5
VBA	AAS	14	28	14	10	21	9	4
VBA	RTH/RTZ	14	28	14	10	24	10	0
RTH/RTZ	VBA	30	15	10	25	5	0	15
AAS	RTH/RTZ	28	21	14	14	14	5	4
RTH/RTZ	AAS	10	10	14	24	25	10	7
1 <sup>st</sup> line	2 <sup>nd</sup> line	19.8	14.8	15.8	17.0	15.1	6.6	10.8
2 <sup>nd</sup> line	1 <sup>st</sup> line	12.6	12.4	13.9	14.7	19.2	14.7	12.4

We see that Tuesday is the busiest day for transport leaving the flower auction. As the major flow is from VBA to RTH/RTZ, Tuesday is the peak day for the system as a whole. For other routes and locations the peak might be on another day. For example, at Schiphol it is busier on Monday and Friday.

**Table 3.5.** Distribution of TU over hours on a day in terms of percentage

From	To	0	1	2	3	4	5	6	7	8	9	10	11	12
AAS	VBA	12	13	3	0	6	4	15	9	7	3	0	1	3
VBA	AAS	0	0	0	0	0	0	0	6	6	4	3	11	5
VBA	RTH/RTZ	0	0	0	0	0	0	0	3	4	5	7	9	10
RTH/RTZ	VBA	4	4	2	2	2	2	5	5	5	5	5	5	5
AAS	RTH/RTZ	2	3	5	6	8	6	7	5	4	4	4	3	2
RTH/RTZ	AAS	1	0	1	2	4	5	7	8	7	5	4	3	4
1 <sup>st</sup> line	2 <sup>nd</sup> line	3.5	1.5	4	3	9	6.5	6	4.5	2.5	3	3.5	3.5	1.5
2 <sup>nd</sup> line	1 <sup>st</sup> line	2.5	1.5	1	1.5	2.5	3.5	3.5	3	2	5	3.5	2	2



From	To	13	14	15	16	17	18	19	20	21	22	23
AAS	VBA	4	0	1	4	0	0	0	4	10	1	0
VBA	AAS	16	16	5	1	0	1	16	4	6	0	0
VBA	RTH/RTZ	7	6	5	6	9	9	6	4	1	0	0
RTH/RTZ	VBA	5	5	5	5	5	4	4	4	4	4	4
AAS	RTH/RTZ	2	3	4	5	6	5	5	4	3	3	1
RTH/RTZ	AAS	3	5	3	2	3	5	7	7	6	4	4
1 <sup>st</sup> line	2 <sup>nd</sup> line	1	3	1.5	2.5	3	6	8	4	2.5	8	8.5
2 <sup>nd</sup> line	1 <sup>st</sup> line	6.5	4.5	10.5	11.5	6.5	4	2.5	2.5	9.5	4	4.5

The transportation flows are also distributed very unevenly over the day. On most routes it is rather quiet during the night and early morning. There are two peaks on the busiest route, from VBA to RTH/RTZ, one around noon and another around 18:00. It turns out that these peaks determine the system performance to a large extent. All routes have their own peaks, mostly at different times of the day. Given the transportation flows on the different routes, the demand distribution in an average month or week can be easily derived from these figures.

The transportation flows are not uniformly distributed over the 1<sup>st</sup> and 2<sup>nd</sup> line terminals. These flows have to be separated into four different flows. The distribution of the internal transportation flows at Schiphol over 1<sup>st</sup> and 2<sup>nd</sup> line terminals, and the distribution of the transportation flows between Schiphol and RTZ, and Schiphol and VBA. These distributions are represented in the following tables, and were obtained from interviews with the parties involved. When terminals are not mentioned, the corresponding percentages are equal to 0.

**Table 3.6.** *Distribution of internal transportation flows at Schiphol over 1<sup>st</sup> line terminals in 2020 in terms of percentage*

1 <sup>st</sup> line Terminal	Export 2 <sup>nd</sup> -> 1 <sup>st</sup> line	Import 1 <sup>st</sup> -> 2 <sup>nd</sup> line
5	0.0%	58.1%
6	24.3%	0.0%
7	24.3%	0.0%
12	20.9%	15.2%
14	4.8%	8.0%
15	25.7%	18.7%

**Table 3.7.** *Distribution of the internal transportation flows at Schiphol over the 2<sup>nd</sup> line terminals in 2020 in terms of percentage*

2 <sup>nd</sup> line Terminal	Export 2 <sup>nd</sup> -> 1 <sup>st</sup> line	Import 1 <sup>st</sup> -> 2 <sup>nd</sup> line
1	2.6%	5.0%
2	12.9%	19.3%
3	12.9%	19.3%
4	12.9%	19.3%
8	20.7%	22.8%
9	38.0%	14.3%

The internal transportation flows at Schiphol from a 1<sup>st</sup> line terminal to a 2<sup>nd</sup> line terminal can now be computed as follows. The total volume (Table 3.3) has to be multiplied by the percentage of the export flow at the 1<sup>st</sup> line terminal and the percentage import at the 2<sup>nd</sup> line terminal. We assume that these distributions are independent. The flow from 2<sup>nd</sup> line to 1<sup>st</sup> line terminals can be computed similarly. Table 3.8 shows the percentage of the flow to other locations.

**Table 3.8.** *Distribution of transportation flows between AAS and VBA, and AAS and RTZ over the AAS terminals in 2020 in terms of percentage*

Terminal	AAS-VBA	VBA-AAS	AAS-RTZ	RTZ-AAS
5	0.0%	0.0%	32.0%	0.0%
6	0.0%	37.5%	0.0%	18.25%
7	0.0%	37.5%	0.0%	18.25%
10	0.0%	0.0%	2.2%	2.3%
11	0.0%	0.0%	8.9%	6.4%
12	40.0%	25.0%	23.1%	22.3%
14	20.0%	0.0%	5.3%	5.1%
15	40.0%	0.0%	28.5%	27.4%

Note that not all terminals at Schiphol will be used in 2020. Some of the terminals will be used until 2010 and then replaced by other terminals that have not yet been built.

For the transportation flows we use the following arrival processes in our simulations:

- Poisson arrivals at AAS and VBA, with the average number of arrivals depending on the time of day.
- Batch arrivals at RTH/RTZ (train station), with an average number of arrivals dependent on the time of day. At first we assume that the number of trains per hour is uniformly distributed between 2 and 4, with fixed inter-arrival times (Chapter 4 and 6). Later on we chose 6 train arrivals per hour, also with fixed inter-arrival times (in the case of a rail terminal at Schiphol Airport, Chapter 5 and Section 6.4). The number of loads per train is Poisson distributed.

All transportation jobs that are generated in the system have to be executed, even when they are late, because the order acceptance procedure has already been passed. The dimensioning and performance measurement will be done for a peak day in an average month (Tuesday). We assume that transportation jobs are known 30 minutes before arrival at the terminal of origin. This is a reasonable assumption for our application because plane and train arrivals are known a long time before the actual time of arrival. This information can be used in the different planning methods. In some experiments this information horizon is an experimental factor.

Key performance measures with respect to transportation jobs are:

- the throughput time: the time from order arrival at the terminal of origin until the job is unloaded at the destination;
- the service level: the fraction of orders that is delivered before the due time;
- the lateness: the difference between actual job completion time and due time.

#### *Throughput time requirements*

With respect to throughput time demands, four priority classes are distinguished for each route. These different classes are introduced to take into account the fact that only a few jobs will be rush orders, while other jobs have more time to be transported. Different prices could be attached to these priority classes. After interviews and discussions with the parties involved, it appeared that no real information was available on this subject. We decided to use the throughput time requirements shown in Table 3.9. Each entry indicates which percentage of the transportation jobs has to be delivered within the specified time. For example, 20% of the transportation jobs on the route from RTH to VBA has to be delivered at its destination within 75 minutes.

**Table 3.9.** *Throughput time requirements with the rail terminal at Hoofddorp*

Time (min)	RTH ↔ VBA	from and to AAS
45	0%	10%
60	10%	20%
75	20%	0%
90	20%	20%
120	50%	50%

The figures in Table 3.9 were used for the experiments with the rail terminal at Hoofddorp (Figure 3.1 and Figure 3.2). When the rail terminal is located at Schiphol, the distances from and to the rail terminal change (Figure 3.3), and therefore we adjust the throughput time requirements (see Table 3.10). The throughput time demands on the routes between AAS and VBA stay the same, because these distances have not been decreased; for some terminals the distance even increased. Furthermore, we have to define throughput time demands for the internal transportation at Schiphol (1<sup>st</sup> line to 2<sup>nd</sup> line and vice versa).

**Table 3.10.** *Throughput time requirements with the rail terminal at Schiphol*

Time (min)	RTZ↔VBA	RTZ↔AAS	VBA↔AAS	1 <sup>st</sup> line↔2 <sup>nd</sup> line
30	0%	10%	0%	20%
45	10%	20%	10%	30%
60	20%	20%	20%	50%
75	20%	0%	20%	0%
90	0%	0%	0%	0%
105	50%	50%	0%	0%
120	0%	0%	50%	0%

### 3.5 Optimization versus simulation: estimation of the resource requirements

One of the issues in the pre-design phase of the project was the estimation of resource capacities required to handle the projected cargo within the throughput time limits as specified in the previous section. Although the simulation model may be used, one may wonder whether it is possible to develop an analytical model for this purpose. In this section, we show that such a model can be constructed for some cases. However, it appears that some system layouts (those including one or more two-way tubes) are difficult to analyze analytically. For those cases, simulation is the only feasible alternative.

We first present an analytical model to estimate capacity requirements (AGVs, docks) for the Stommeer layout (see Figure 3.1). The ideas of Maxwell and Muckstadt (1982) are used, but in addition time is taken into account. Obviously, the opportunities for peak shaving influence capacity requirements, because the workload fluctuates strongly over time (see Table 3.4 and Table 3.5). If the dispatch time window for the jobs increases, the effective peak demand decreases and fewer AGVs and docks are required to handle the cargo flows. This aspect is part of our model.

We compare the estimates from this model with the simulation results. For the other system designs we only present the resource requirements derived from the simulation experiments. In order to be able to determine the resource requirements, we have to know the maximum transportation flows in a given time interval in a more-or-less steady state. Once these steady state flows are known, there should be sufficient resources (AGVs and docks) to handle these flows.

#### 3.5.1. An analytical model

We present an analytical model for a network with deterministic travel times (cf. Section 1.3.2). In this model the transportation flows are spread over the time intervals, termed peak shaving, to level the workload in the different time intervals. The possibilities for peak shaving are limited by the throughput time requirements. Furthermore, the empty vehicle flows, necessary to reposition the vehicles because of the imbalance in transportation flows, are estimated. Time is modeled in discrete time intervals,  $t \in \{0, \dots, T\}$ . First we assume that all transportation jobs that are started in time interval  $t$  will also be delivered within that same time interval, later we leave this assumption. To keep the model solvable, the number of time intervals should not be too large. A time interval of 1 hour seems reasonable for our application. First, we introduce some notation:

$M$  = set of terminals, indexed by  $i$  and  $j$   
 $\tau_{ij}^l$  = handling times, the total time required for handling one transportation job, i.e. the loading time, driving time to the exit of terminal  $i$ , driving time to terminal  $j$ , driving time from terminal entrance to a dock, and the unloading time (in minutes).

- $\tau_{ij}^e$  = empty driving time, that is the driving time to the exit of terminal  $i$ , the driving time to terminal  $j$ , and the driving time on terminal  $j$  toward a dock (in minutes).
- $f_{ijt}$  = transportation flows, the number of TU that has to be transported from terminal  $i$  to terminal  $j$ , with order release time at terminal  $i$  in interval  $t$ .
- $a_{ijts}$  = shift parameters,  $s \in \{0,1,2,\dots\}$ , the percentage of transportation jobs from terminal  $i$  to terminal  $j$ , that arrives in time interval  $t$  at terminal  $i$  and has to be dispatched within  $s$  time intervals after arrival. This is a cumulative percentage, i.e.  $a_{ijts} \geq a_{ijts-1}$ . For simplicity we assume that the  $a_{ijts}$  are independent of  $t$ , as is the case in the OLS, see Table 3.12. For example,  $a_{ijl}$  is the percentage of the transportation jobs that arrives in time interval  $t$  and that has to be dispatched before the end of time interval  $t+l$ .

*Decision variables:*

- $x_{ijt}$  = the number of transportation jobs from terminal  $i$  to terminal  $j$  started in time interval  $t$ .
- $y_{ijt}$  = the number of empty trips between terminal  $i$  and terminal  $j$  started in time interval  $t$ .
- $\alpha_{ijts}$  = percentage of transportation jobs that arrived in time interval  $t$ , and is dispatched in time interval  $t+s$ .

The required number of AGVs  $N(t)$  can be estimated from the total driving time between terminals plus the total time spend at terminals. We have to look at time interval  $t^*$ , in which the maximum amount of AGV time is required. The required number of AGVs is:

$$N(t) = \frac{\sum_i \sum_j (x_{ijt} \cdot \tau_{ij}^l + y_{ijt} \cdot \tau_{ij}^e) + \sum_i \lambda_i \cdot W_q^i}{60} \quad (1)$$

and  $t^* = \arg \max N(t)$ . Here  $\lambda_i$  denotes the dock operations intensity at terminal  $i$  and  $W_q^i$  the waiting time at terminal  $i$  for one dock operation. The time spent on a terminal is subdivided in the time for driving, the loading and unloading time and the waiting time. The driving time on the terminal and the loading and unloading time are included in the handling time ( $\tau_{ij}^l$ ) and the empty driving time ( $\tau_{ij}^e$ ). Waiting times at docks occur owing to the limited dock capacity. Once the number of dock operations in time interval  $t$  at terminal  $i$  is known ( $\lambda_i = \sum_j x_{ijt} + \sum_j x_{jit}$ ), the expected waiting time for one dock operation ( $W_q^i$ ) can be approximated using a standard multi-server queuing model.

*Model:*

In our model we ensure that the transportation flows are distributed over the time intervals in order to minimize the maximum workload in an interval for AGV

driving time plus the docking time. Hence, waiting time is introduced as a correction afterwards. The objective function is:

$$\text{Min} \left\{ \text{Max}_t \sum_i \sum_j (x_{ijt} \cdot \tau_{ij}^l + y_{ijt} \cdot \tau_{ij}^e) \right\} \quad (2)$$

Subject to:

$$\sum_s \alpha_{ijts} = 1 \quad \forall i, j, t \quad (3)$$

$$\sum_{k \leq s} \alpha_{ijtk} \geq a_{ijs} \quad \forall i, j, t, s \quad (4)$$

$$x_{ijt} = \sum_{k \geq 0, k < t} \alpha_{ij,t-k,k} \cdot f_{ij,t-k} \quad \forall i, j, t \quad (5)$$

$$\sum_j (x_{ijt} + y_{ijt}) = \sum_j (x_{jit} + y_{jit}) \quad \forall i, t \quad (6)$$

$$x_{ijt} \geq 0, \quad \text{integer}, \quad \forall i, j, t \quad (7)$$

$$y_{ijt} \geq 0, \quad \text{integer}, \quad \forall i, j, t \quad (8)$$

Constraint (3) indicates that the sum of the fractions should be equal to 1. Postponing transportation jobs to later time intervals is limited by the throughput time requirements (4). The number of loaded trips in one time interval is equal to the transportation jobs that are postponed from the previous time intervals plus the percentage of transportation jobs that arrive in the current time interval and are also dispatched in the current time interval (constraint 5). The first three constraints spread the total workload over the different time periods (peak shaving). Constraint (6) includes the conservation of flow constraints. All flows have to be nonnegative integers (constraint 7 and 8). In this formulation we assign handling times to the time interval in which the job is started. Note that this formulation does not necessarily minimize the empty trips. The objective is to minimize the maximum workload in a time interval. In quiet time intervals unnecessary empty trips may be planned, without affecting the objective function. Additional constraints can be added to also minimize empty trips outside the peak periods. However, this is not the focus of our model.

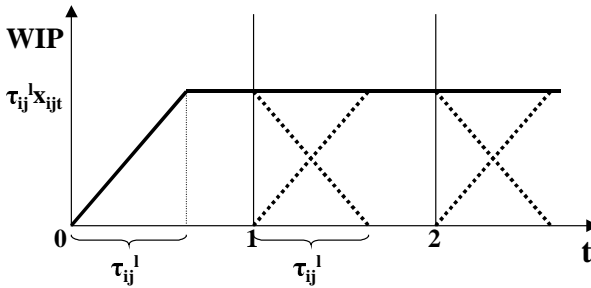
To solve the problem the objective function is transformed to constraint (9), with an objective function to minimize  $C$ :

$$\sum_i \sum_j (x_{ijt} \cdot \tau_{ij}^l + y_{ijt} \cdot \tau_{ij}^e) \leq C \quad \forall t \quad (9)$$

The objective is to minimize the maximum workload in a specific time interval, waiting times excluded. The number of decision variables in the mixed integer problem increases rapidly with the number of terminals (quadratically). For a rough estimate it is probably sufficient to solve the linear programming relaxation, where the integer requirements in (7) and (8) are omitted. For small problems the mixed integer problem can be solved as well.

*A modified approach with improved workload dynamics*

With an alternative approach we can take into account the fact that jobs started in one time period finish in another time period. The objective is to minimize the maximum work in progress in a particular time period. To simplify the computations, we assume that the length of one time interval is longer than the maximum handling time ( $\tau_{ij}^l$ ). This means that transportation jobs that are started in time interval  $t$  arrive at their destination in time interval  $t$  or in time interval  $t+1$ . Furthermore, we express the handling time and travel time in time intervals, i.e.  $0 < \tau_{ij}^l < 1$ . This can easily be generalized; for our application this simplification is not a problem and it also makes the equations easier to read. The transportation jobs are equally distributed over 1 time interval. The work in progress for one route from  $i$  to  $j$  looks like Figure 3.11.



**Figure 3.11.** Work in progress, on the route from  $i$  to  $j$ , as a function of time

Work in progress increases until the first AGV arrives at its destination, i.e. until  $\tau_{ij}^l$ . The work in progress is at that time equal to  $\tau_{ij}^l x_{ijt}$ . Till the end of the time interval the work in progress is constant, because at the time new transportation jobs depart other jobs arrive at the destination. In the next time interval jobs started in the previous time interval still have to arrive at their destination, also for a time equal to  $\tau_{ij}^l$ . The total work in progress on one route in a particular time interval is equal to the surface below the function(s) in Figure 3.11. When the  $x_{ijt}$  are equal for all  $t$ , the work in progress would be fixed at  $\tau_{ij}^l x_{ijt}$  (cf. Figure 3.11).

The objective function is:

$$\min \left\{ \max_t \left\{ \sum_{i=1}^M \sum_{j=1}^M \left\{ \left(1 - \frac{1}{2} \tau_{ij}^l\right) \tau_{ij}^l x_{ijt} + \left(1 - \frac{1}{2} \tau_{ij}^e\right) \tau_{ij}^e y_{ijt} + \frac{1}{2} \left(\tau_{ij}^l\right)^2 x_{ij,t-1} + \frac{1}{2} \left(\tau_{ij}^e\right)^2 y_{ij,t-1} \right\} \right\} \right\} \quad (10)$$

With flow conservation constraints:

$$\sum_{j=1}^M \left\{ \tau_{ji}^l x_{jit} + \tau_{ji}^e y_{jit} + \left(1 - \tau_{ji}^l\right) x_{jit} + \left(1 - \tau_{ji}^e\right) y_{jit} \right\} = \sum_{j=1}^M \left\{ x_{ijt} + y_{ijt} \right\} \quad \forall i, t \quad (11)$$

The other restrictions, (3)-(5), (7) and (8) still apply. Owing to these flow conservation constraints, we can no longer solve the problem as a mixed integer problem (cf. constraint 6); we have to solve a linear programming relaxation. But in any case the model is only used to obtain a rough estimate of the required resources. Furthermore, experiments with the formulation in equations (2)-(8) show that the integer solution is within 1 percent of the linear programming relaxation. The formulation in equations (10) and (11) can be extended to smaller time intervals,  $\tau_{ij}^l > 1$ , using a similar approach. The driving time can be longer than 1 time interval, which leads to additional terms in equation (10) and (11). Because we are only searching for a rough estimate, we do not discuss this extension here.

#### *Determining the $a_{ijs}$*

The  $a_{ijs}$  are determined using the throughput time requirements (see Section 3.4). We assume that the length of a time interval is one hour. When a specific transportation job arrives, which has a required throughput time of 60 minutes and required handling time ( $\tau_{ij}^l$ ) of 30 minutes, the time window for starting this transportation job is 30 minutes. When the job arrives in the first half-hour, the job has to be transported within that same hour, while the job can be postponed to the next hour if it arrives in the second half-hour. For example, given a total number of 140 transportation jobs on a particular route, 70 jobs cannot be postponed to the next hour and 70 jobs can be postponed:  $a_0=0.5$  and  $a_1=1$ . These computations can be executed for all routes. In the OLS-case, the transportation jobs always have to be transported within the next two hours after the hour of arrival ( $s \leq 2$ ). The larger  $a_{ij0}$ , the less possibilities for peak shaving.

#### *Determining the required number of docks from the maximum dock utilization*

The  $x_{ij}$ , computed with one of the two formulations presented, can now be used in a queuing model to determine the required number of docks at each terminal. For these computations, we assume that the AGVs and transportation jobs arrive at terminal  $i$  according to a Poisson distribution (simplification of the OLS-case) with mean arrival rate  $\lambda_i = \max_i \left\{ \sum_j x_{ijt} + \sum_j x_{jit} \right\}$ , the maximum number of dock operations in one time interval. For simplicity, we assume that the loading and unloading times are also exponentially distributed, with  $\mu_i$  the possible number of



dock operations per dock per time interval at terminal  $i$ . For such an  $M/M/s$  queue, in which  $s$  represents the number of docks, the utilization rate is defined as:

$$\rho_i = \frac{\lambda_i}{s_i \mu_i}$$

To ensure that the steady-state probabilities exist this utilization rate should be smaller than 1. Given a maximum dock utilization  $\rho$  (e.g. 0.8), we can compute the minimum required number of docks at terminal  $i$ ,  $N_{d,i}$ .

$$N_{d,i} = \left\lceil \frac{\lambda_i}{\rho_i \mu_i} \right\rceil, \text{ where } \lceil x \rceil \text{ denotes the smallest integer that equals or exceeds } x.$$

Given the number of docks,  $s_i$ , the expected waiting time per dock operation follows immediately. The average time an AGV waits in line at terminal  $i$  is (Winston, 1991):

$$W_q^i = \frac{L_q^i}{\lambda_i}$$

$$\text{with } L_q^i = \frac{P(m \geq s_i) \rho_i}{1 - \rho_i}, \quad P(m \geq s_i) = \frac{(s_i \rho_i)^{s_i} \pi_0}{s_i! (1 - \rho_i)}$$

$$\pi_0 = \frac{1}{\sum_{k=0}^{k=s_i-1} \frac{(s_i \rho_i)^k}{k!} + \frac{(s_i \rho_i)^{s_i}}{s_i! (1 - \rho_i)}}$$

These waiting times can also be computed for time interval  $t^*$ , and used in the computation of the required number of AGVs (equation 1). If desired the number of docks can be determined using the expected waiting time for a dock operation as a constraint, see the discussion following Table 3.13.

### **3.5.2. Example: Stommeer layout**

We apply the model with equations (2)-(8) from the previous section to the Stommeer layout (Figure 3.1), with the transportation flows from Table 3.1 and corresponding throughput time requirements. Because the transportation flows are defined per location (VBA, RTH and AAS), we use these locations in the analytical model. The average empty travel times between the locations, given an AGV speed of 6 m/s, are shown in Table 3.11. We assume that the average dock operation time is equal for loading and unloading and for all terminals, and equals 2 minutes. The driving time on the terminal is equal to 0.8 minutes for VBA, 0.4 minutes for AAS and 1.2 minutes for RTH.

**Table 3.11.** Average travel time between the locations in minutes (6 m/s)

	VBA	RTH	AAS
VBA	0	26.2	23.2
RTH	26.2	0	20.1
AAS	23.5	21.4	0

In Table 3.12 we show the shift parameters  $a_{ijs}$ , derived from the throughput time requirements. Notice that only a small percentage concerns rush orders, which have to be transported almost immediately.

**Table 3.12.** The shift parameters for the different routes ( $a_{ijs}$ )

route	$a_{ij0}$	$a_{ij1}$	$a_{ij2}$
VBA-RTH	0.11	0.76	1
RTH-VBA	0.11	0.76	1
VBA-AAS	0.16	0.72	1
AAS-VBA	0.16	0.72	1
RTH-AAS	0.15	0.70	1
AAS-RTH	0.15	0.70	1

The length of a time interval is 1 hour;  $T$  is equal to 24. We will address the peak day (Tuesday) in an average week. The model is terminated after finding the optimal solution or when no improvements in the objective function have been found in 30 seconds. The model was implemented in AIMMS (using the XA solver).

### Docks

The model of equations (2)-(8) results in  $\lambda_1=180$ ,  $\lambda_2=230$ , and  $\lambda_3=90$ . We assume that the number of dock operations,  $\lambda_i$ , is equally distributed over the number of terminals of location  $i$ , so a different  $M/M/s$  queue for each terminal (2 at VBA and 5 at AAS). In Table 3.13 we show the mean waiting times in minutes and utilization rates for the different locations and different number of docks ( $s$ ).

**Table 3.13.** Expected waiting times and the dock utilization rates (between parenthesis)

	s=1	s=2	s=3	s=4	s=5	s=6	s=8	s=9	s=10	s=11
VBA				1.42 (0.75)	0.27 (0.6)	0.07 (0.5)				
RTH							23.6 (0.96)	1.41 (0.85)	0.39 (0.77)	0.15 (0.7)
AAS	9.12 (0.6)	0.45 (0.3)	0.05 (0.2)							

The dock utilization rate ( $\rho$ ) can be misleading when used in determining the required number of docks. One could assume that 1 dock per AAS terminal is sufficient, but the expected waiting time for a dock operation in the peak hour is more than 9 minutes. Because we do not want the docks to be a bottleneck, we chose 2 docks per AAS terminal, 5 docks per VBA terminal and 10 docks at RTH.

### AGVs

In determining the required number of AGVs we find  $\lambda_1^* = 180$ ,  $\lambda_2^* = 230$ , and  $\lambda_3^* = 50$ , also equally distributed over the terminals at one location. Notice that in this particular time interval the AAS terminals are not at their peak capacity. After filling in equation (1), the required number of AGVs is 165. The second approach, using equations (10) and (11) instead of (2) and (6), results in a required number of AGVs of 163. Recall that this is the solution of the linear programming relaxation.

It is interesting to look at the contribution of each activity to the total number of required AGVs. The different contributions give an indication of the influence of the speed of the AGV and the loading and unloading times of docks. An increase in AGV speed could reduce the number of required AGVs considerably when there are long distances between the terminals. At a speed of 10 m/s the estimated number of AGVs is 108, at 15 m/s the estimate is 81 AGVs. In these experiments we used the same intermediate distance between AGVs (3.5 seconds). When the intermediate distance increases the reduction in AGVs is smaller. The estimate for the required number of AGVs is more-or-less a lower bound. In reality the transportation jobs will not be spread uniformly over the time interval and the peak shaving will also not be perfect. Comparison with simulation results in the next section shows that this leads to significant effects.

### 3.5.3. Resource requirements for the 3 different layouts

Extending the analytical model to systems with two-way tracks is difficult. The travel times are no longer deterministic, and long waiting times occur at the two-way tracks. Especially these waiting times at the two-way tracks, and resulting waiting times at terminals due to convoy arrivals are difficult to estimate. For these kinds of systems simulation has proven its value. Simulation experiments can determine the resource requirements rather easily. The sensitivity of the system to parameters such as AGV speed and docking speed might be even greater with two-way track sections. Other parameters, such as the intermediate distance between AGVs and the acceleration and deceleration of AGVs also influence the resource requirements. The results of the simulation study are shown in Table 3.14.

**Table 3.14.** Resource requirements from the simulation model

Layout	AGVs	docks		docks AAS
		RTH/RTZ	VBA	
Stommeer	185	10	2x5	5x2
Two-way tubes	360	8	8	3x2
Internal transport Schiphol	250	8	8	18x2

The required number of docks for the Stommeer layout is equal to the estimates of the analytical model. The required number of AGVs in the simulation is about 12% more than estimated by the analytical method. This is not really surprising, because the transportation jobs are not equally distributed over a time interval due to the

stochasticity and batch train arrivals, and the peak shaving will not be perfect. Another reason is that the planning of the AGVs will not be optimal, partly due to incomplete information and stochasticity in the travel times. We also see that the introduction of two-way tracks, plus some other modifications, significantly increases the required number of AGVs. Most of these additional AGVs are required because of the waiting times at the two-way tracks and, as a result of the convoys created, the waiting times at the terminals. Moving the rail terminal to Schiphol clearly reduces the required number of AGVs, because of shorter distances, but the number of terminals at Schiphol increases considerably.

### 3.6 Conclusions

From the previous section we can conclude that analytical methods are only useful for particular sub-problems, such as the determination of the required number of AGVs, and then only in case of nearly deterministic travel times. As was discussed in Chapter 1, a simulation model seems the only way to incorporate all features of the model.

The simulation group appeared to be important in the pre-design phase of the OLS-project. The simulation group received a lot of input from the other research areas, as described in Section 3.3. The simulation model could determine the logistical consequences of the technological ideas from several other groups. An example is the question of whether the use of two-way tubes is feasible from a logistic point of view. Simulation experiments showed that two-way tubes could be used, but with serious consequences on the logistic performance and resource requirements. Furthermore, simulation results were important input for the energy study (Van der Heide, 1999) and cost analyses of the different system layouts.

In the experiments in the following chapters the AGV specifications, as described in Section 3.3.3, will be used, with a required time between two AGVs of 3.53 seconds and with the speed on the terminal equal to  $(2-0.024N)$  m/s, as was derived in the traffic simulation model. The Stommeer layout from Figure 3.1 is used for the experiments in Chapter 4, with terminals similar to the one shown in Figure 3.5 and an average docking time equal to 2 minutes. These experiments are based on the transportation flows in Table 3.1. The layout with two-way tubes (Figure 3.2) is used in Chapter 6, with terminals similar to the one shown in Figure 3.5 and an average docking time equal to 1 minute. The transportation flows from Table 3.2 are used for these experiments. The layout with the rail terminal at Schiphol (Figure 3.3) is used in Chapter 5 and in Section 6.4, with the small Schiphol terminals as shown in Figure 3.7, the VBA terminal similar to Figure 3.5, and a rail terminal without shortcuts and parking places. The average docking time is 1 minute and the transportation flows are shown in Table 3.3. This layout was also used to compute a growth scenario for the years 2005-2020, see Van Harten et al. (2000).



# Chapter 4

## Control of the primary process: vehicle management<sup>2</sup>

### 4.1 Introduction

In this chapter we discuss various options for empty vehicle management, i.e. the pre-positioning of empty vehicles in the network to anticipate known and predicted demand. The need for empty vehicle management is inevitable because transportation flows will generally be unbalanced over time; the number of vehicles arriving in a time interval at a terminal differs from the number of vehicles to be loaded and dispatched in the same interval and from the same terminal. Empty vehicle management is especially critical in situations where due times are tight and resource utilization (vehicles, docks) is high. The scheduling of empty AGVs is closely related to the scheduling of transportation jobs. Transportation jobs require empty AGVs and finished transportation jobs result in empty AGVs. Therefore we will study the planning of both empty and loaded vehicle travel. We consider several planning options in which the required information and the level of coordination are varied. We are especially interested in the following subjects:

- How do look-ahead policies compare to simple myopic planning rules?
- What is the value of information about future transportation jobs?
- To what extent can central coordination contribute to a more efficient system compared to decentralized planning and control?

---

<sup>2</sup> This chapter is partly based on Van der Heijden et al. (2001a).

These issues will be discussed for general transportation networks. The OLS project serves as a test case. We will use the model developed in Chapter 2 to test and evaluate the various options for empty vehicle management. All other control activities besides vehicle management, such as task allocation to docks and vehicle control, will be controlled according to the descriptions in Chapter 2.

The remaining part of this chapter is organized as follows. In the next section, we discuss an integer linear programming and a dynamic programming formulation for the vehicle management problem. An overview of related literature is given in Section 4.3. Various options for empty vehicle management without central coordination are introduced in Section 4.4. In Section 4.5 we present a serial scheduling approach to solve the vehicle management problem. A large part of this chapter is devoted to a logistic queuing network approach, which is described in Section 4.6. Note that this is a queuing network as defined by Powell and Carvalho (1998a), which is not exactly the same as other queuing networks encountered in the literature. The simulation study with numerical results based on the OLS-case, is the subject of Section 4.7. We summarize our main conclusions in Section 4.8.

## 4.2 Model and assumptions

In this section we describe an integer linear programming and a dynamic programming formulation for the vehicle management problem. First, we present a discrete model as proposed by Powell and Carvalho (1998a). We assume that time is divided into a set of discrete time periods  $t = (1, \dots, T)$ .  $T$  is the planning horizon in total number of time periods. All time-related activities, such as travel times and loading/unloading times have to be transformed into discrete time periods. First we present the notation for the problem.

*Network variables:*

- $M$  is the set of terminals and parking areas, indexed by  $i$  and  $j$  (for simplicity we will mainly talk about terminals in the remainder of this chapter, but parking areas are included)
- $\tau_{ij}^e$  is the expected empty travel time from terminal  $i \in M$  to terminal  $j \in M$ ,  $\tau_{ii}^e = 1$ ,  $\tau_{ij}^e > 0$ ; all travel times are positive
- $\tau_{ij}^l$  is the expected handling time from terminal  $i \in M$  to terminal  $j \in M$ ,  $\tau_{ij}^l > 0$  (from the start of the loading operation until the end of the unloading operation, travel time included)
- $N$  is the set of nodes  $(i, t)$  in the dynamic network,  $i \in M$  and  $t \leq T$

The expected travel times and handling times will be estimated using the methods described in Section 2.6.

*Parameters:*

- $r_{lt}$  is the revenue generated by choosing time  $t$  to start transportation job  $l$
- $c_{ijt}$  is the cost of repositioning one empty vehicle from terminal  $i$  to terminal  $j$ , starting in time period  $t$ .
- $T_l$  is a discrete interval, which contains all the feasible time periods for starting transportation job  $l \in L$ , otherwise known as the departure time window.
- $T_l^s$  is the first time period of the departure time window  $T_l$  of transportation job  $l$ .

*Activity variables:*

- $L$  is the set of transportation jobs  $l$  available within the planning horizon  $T$ , i.e. the release time of order  $l$  is contained in the planning horizon.
- $L_{ijt}$  is the set of transportation jobs  $l \in L$  with origin  $i$  and destination  $j$ , having  $t$  as a feasible departure time.
- $RV_{it}$  is the net inflow ( $RV_{it} > 0$ ) or outflow ( $RV_{it} < 0$ ) of vehicles at terminal  $i$  at time  $t$ .

*Decision variables:*

- $x_{lt} = 1$  if transportation job  $l$  is started at time  $t$ ,  $x_{lt}$  only exists for  $t \geq T_l^s$ .
- $y_{ijt}$  is the number of empty vehicles being repositioned from terminal  $i$  to terminal  $j$  at time  $t$ .  $y_{iit}$  is the number of vehicles, which is kept in inventory at terminal  $i$  from time period  $t$  to time period  $t + 1$ .

#### 4.2.1. Integer linear programming formulation

The problem can be formulated as an integer linear program ( $P_I$ ), for the periods  $t = 1, \dots, T$ . Our problem consists of maximizing profit by assigning one vehicle to each transportation job, and enforcing conservation of flow at each node.

$$\max_{x, y} F(x, y) = \sum_{t=1}^T \sum_{i \in M} \sum_{j \in M} \left( \sum_{l \in L_{ijt}} r_{lt} x_{lt} - c_{ijt} y_{ijt} \right) \quad (1)$$

*Subject to:*

$$\sum_{t \geq T_l^s} x_{lt} = 1 \quad \forall l \in L \quad (2)$$

$$\sum_{j \in M} \left( \sum_{l \in L_{ijt}} x_{lt} + y_{ijt} \right) - \sum_{j \in M} \left( \sum_{l \in L_{jii}} x_{l, t - \tau_{ji}^l} + y_{j, i, t - \tau_{ji}^e} \right) = RV_{i, t} \quad \forall (i, t) \in N \quad (3)$$

$$x_{lt} \in \{0, 1\} \quad \forall l \in L \quad \forall t \in [1, \dots, T] \quad (4)$$

$$y_{ijt} \geq 0 \text{ integer} \quad \forall i, j \in M \quad \forall t \in [1, \dots, T] \quad (5)$$

The objective function (1) maximizes the revenues, which can be separated into the positive revenue of a loaded move ( $r_{lt}$ ) and the negative revenue (costs) of an empty move ( $c_{ijt}$ ). We need to sum these terms over all the possible movements and all the



time periods. The first constraint (2) deals with the fact that all the transportation jobs have to be transported. This means that the variable  $x_{lt}$  has to be equal to 1 only once, because the start of transportation job  $l$  can only take place in one time period. To start transportation job  $l$ , it has to be available at the terminal of origin. Therefore, the time period  $t$  has to be greater than or equal to the time period of the arrival of order  $l$  at the terminal ( $T_i^s$ ). Because all transportation jobs have to be transported, there is no latest starting time. Nevertheless, the revenue function can be related to the due time of the transportation job. Constraint (3) has to deal with the flow of AGVs in the system. Flow conservation constraints must be present. However, we can also have a flow into terminal  $i$  in time period  $t$  as a result of the initial system state, in which AGVs are driving between terminals and arrive at a terminal during some future time period. This flow is called  $RV_{ii}$ . Since the first part of the right hand side of this equation only refers to the present planning horizon  $t - \tau_{ij}^l \geq 1$  and  $t - \tau_{ij}^e \geq 1$ . For an arbitrary period  $t$ , we see that at terminal  $i$ , the incoming flow is equal to the outgoing flow. The last two constraints are integrality constraints.

The model focuses on revenue maximization, whereas we are mainly interested in minimizing lateness. The formulation of the model can easily be adjusted to another objective. To this end, we use the following definitions:

$d_l$  = the due time of job  $l$  (input data)

$a_{lt}$  = the lateness of job  $l$  if it is started at time  $t$

The lateness of each job, depending on the starting time  $t$ , can be calculated as

$$a_{lt} = \text{Max}\{t + \tau_{ij}^l - d_l, 0\}$$

Hence the minimization of the total lateness can be formulated as

$$\min_x F(x) = \sum_{t=1}^T \sum_{i \in M} \sum_{j \in M} \left( \sum_{l \in L_{ij}} a_{lt} x_{lt} \right)$$

Alternatively, we can formulate the objective to minimize the maximum lateness as

$$\min_x F(x) = \max_{l \in L, 1 \leq t \leq T} a_{lt} x_{lt}$$

To obtain the optimal solution of the problem, a branch-and-bound approach can be used, possibly based on the linear programming relaxation (if the objective is revenue maximization or total lateness minimization). What makes this problem so complex is not only the presence of the integer variables, but also the fact that the problem has to be solved over an extended planning horizon. Because of the magnitude of the problem, problems like this also result in an intractably large integer (linear) program, which takes little advantage of the dynamic structure of the system. Maximizing the total revenue can be equal to minimizing the total lateness

if all  $c_{ijt}$  are equal to 0. The revenue function should penalize lateness and could look like  $r_{it} = R - a_{it}$ , with  $R$  the revenue for delivering a transportation job on time (cf. Section 4.6.5).

#### 4.2.2. Dynamic programming formulation

Besides formulating the problem as an integer linear program, we can also formulate it as a dynamic programming problem (DP). Here we give a possible formulation. The state consists of:  $R_k^i, e_l$ . Here  $i, j \in M$  refer to the terminals,  $k \in \{1, \dots, \tau\}$ , indicates a future time period that can be influenced by the decision at time  $t$ , with  $\tau = \max\{\tau_{ij}\}$ , an upper bound on the travel time between two terminals. Furthermore,  $l \in \bar{L}_t$ , the set of jobs that is available for transport at time  $t$ , but that has not yet been moved.  $\bar{L}_{it}$  is the subset of  $\bar{L}_t$ , only including those jobs with terminal  $i$  as origin. The interpretation of the state variables is as follows:

- $R_k^i$  = the number of empty vehicles that comes available at terminal  $i$ ,  $k$  periods ahead if no further action is taken.
- $e_l$  = the earliness of job  $l$ , from terminal  $i$  to terminal  $j$ , if it is executed in the current time interval  $t$ :  $d_l - (t + \tau_{ij}^l)$ .

Note that  $R_0^i$  at decision moment  $t$  is equal to  $V_{it}$  defined in Section 4.6.1 and  $\sum_{i,k} R_k^i$  is equal to the total number of vehicles. As a shorthand notation for the state we use  $\{R, e\}$ .

*Decision variables at decision moment  $t$ :*

$x_{it} = \{0, 1\}$ , indicating whether job  $l$ , corresponding with  $e_l$ , is scheduled in the current decision period.

$y_{ijt}$  = number of empty vehicles moved from terminal  $i$  to terminal  $j$ .

All vehicles need to be assigned to a job, possibly a job on the current terminal:  $y_{iit} > 0$ . Thus, by definition:

$$R_0^i = \sum_{l \in \bar{L}_{it}} x_{lt} + \sum_j y_{ijt}, \quad \forall i$$

The direct costs, given state  $\{R, e\}$  and decision  $\{x, y\}$ , can be separated into:

- 1) Direct costs for lateness:  $C_1 = \sum_{l \in \bar{L}_t} I_{\{e_l < 0\}}$ , with  $I$  the indicator function.
- 2) Direct costs for empty moves from terminal  $i$  to terminal  $j$ , with  $i \neq j$ :  
 $C_2 = \sum_{i, j, i \neq j} y_{ijt}$ . If there are non-capacitated terminals we associate no costs with waiting empty at a terminal.
- 3) Direct rewards from the execution of transportation jobs:  $W_3 = \sum_{l \in \bar{L}_t} x_{lt}$

It is possible to consider more general cost functions, for example:

$$C_2 = \sum_{i,j} c_{ijt} y_{ijt} \quad \text{and} \quad W_3 = \sum_{l \in \bar{L}_t} r_{lt} x_{lt} ,$$

with  $c_{ijt}$  the costs of an empty move from terminal  $i$  to terminal  $j$  at time  $t$ , with  $t$  a decision moment, and  $r_{lt}$  the revenue of executing transportation job  $l$  at time  $t$ . These functions are comparable to the ones used in Section 4.6.1. In this formulation direct costs are associated with lateness, which in equation (1) can be incorporated in the revenue function  $r_{lt}$ . If desired, other cost components can be introduced, for example for exceeding the terminal capacity. The structure of equations (2) and (3) could also be generalized, but we consider the simple form as given.

The state transition from  $\{R, e\}$  at time  $t$  to  $\{\hat{R}, \hat{e}\}$  at time  $t+1$ , given decision  $\{x, y\}$ , looks like:

- a)  $\hat{R}_0^i = R_1^i + \sum_{l \in \bar{L}_t, \tau_{ji}^l=1} x_{lt} + \sum_{j, \tau_{ji}^l=1} y_{jlt} - \sum_{l \in \bar{L}_t} x_{lt} - \sum_j y_{jlt}$ ; the vehicles that arrive due to previous decision moments, plus the vehicles that arrive due to current decisions at other terminals, minus the vehicles that leave the terminal given decision  $\{x, y\}$ . Recall that  $\tau_{ij}^e > 0$  and  $\tau_{ii}^e = 1$ .
- b)  $\hat{R}_k^i = R_{k+1}^i + \sum_{l \in \bar{L}_t, \tau_{ji}^l=k+1} x_{lt} + \sum_{j, \tau_{ji}^l=k+1} y_{jlt}$ , for  $k < \tau$ , where we define  $R_\tau^i = 0$ .
- c) In  $\hat{e}$  we omit all elements from  $e$  for which  $x_{lt} = 1$ .
- d)  $\hat{e}_l = e_l - 1 \quad \forall l \in \bar{L}_t$  with  $x_{lt} = 0$ ; the jobs that were not planned get closer to their due times (possible lateness increases).
- e) The set  $\bar{L}_{t+1}$  is constructed from the jobs of set  $\bar{L}_t$  with  $x_{lt}$  equal to 0, plus the jobs that are released at the start of time interval  $t+1$ .
- f) For each job that is released at time  $t+1$  we create (iteratively) a new  $l'$ , and we define  $\hat{e}_{l'} = d_{l'} - (t+1 + \tau_{ij}^l)$ .

We define the value function  $G_t\{R, e\}$  as the minimum costs over  $[t, T]$ , given state  $\{R, e\}$  at the beginning of time  $t$ . The overall DP objective is to minimize  $G_1$  given the initial state  $\{R, e\}$ . The recursion relation then looks as follows:

$$G_t\{R, e\} = \max_{x, y} \left\{ -C_1(e) - C_2(y) + W_3(x) + G_{t+1}\{\hat{R}, \hat{e}\} \right\} \quad (6)$$

Before we can solve the DP, we have to define the end values for state  $\{R, e\}$  at time  $T$ . First, we consider the *lateness costs*. Let us introduce the notation:

$n_0^i = x\{e_i > 0\}$ , the number of jobs already late at terminal  $i$  at time  $T$

$n_k^i = x\{e_i = k - 1\}$ , the number of jobs that become late in period  $T+k$  at terminal  $i$ .

Then it is clear, that *at least* the following lateness costs are generated at terminal  $i$  at time  $T$ :

$$\text{for } k=0: n_0^i \stackrel{\text{def}}{=} S_0^i$$

$$\text{for } k=1: n_0^i + \max(n_1^i - R_0^i, 0) \stackrel{\text{def}}{=} S_1^i$$

$$\text{for } k=m: n_0^i + \max\left(\sum_{p=1}^m n_p^i - \sum_{p=0}^{m-1} R_p^i, 0\right) \stackrel{\text{def}}{=} S_m^i$$

The lateness costs consist of the transportation jobs that are already late at time  $T$  plus the jobs that cannot be transported in time by the number of vehicles that arrives in future time periods. Hence, the lateness costs are at least:

$$\bar{C}_1 = \sum_i S_{K(i)}^i$$

with  $K(i)$  the minimum travel time to terminal  $i$ , from a terminal  $j \neq i$ . After time  $K(i)$ , new vehicles may arrive from other terminals. Next, let us consider the transportation rewards. It is logical to use the minimum of the number of transportation jobs and the number of available vehicles per terminal as an end value:

$$\bar{W}_3 = \sum_i \min(|\bar{L}_{iT}|, |R^i|)$$

where  $|\cdot|$  denotes the sum of the vector entries, and  $R^i$  the sub-vector of  $R$  related to terminal  $i$ . It might be possible to define an end value for the costs of empty moves, based upon the remaining jobs in  $\bar{L}_{iT}$  and the distribution of the vehicles over the terminals,  $R$ , but it is not directly clear how to define this end value. Altogether, this leads us to the following end values:

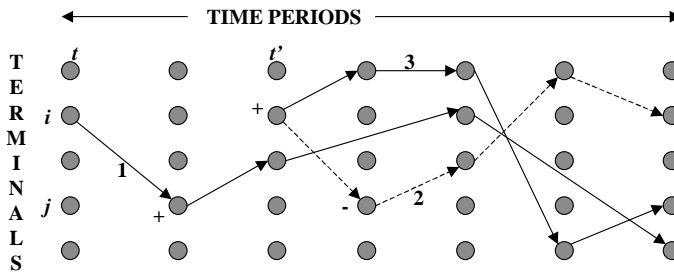
$$G_T\{R, e\} = -\bar{C}_1 - \bar{C}_2 + \bar{W}_3$$

The DP formulation is now complete. However, the ‘‘curse of dimensionality’’ hits us. The state space is far too large. There is no other way to solve the problem than by introducing approximations for  $G_T\{R, e\}$ . We do not present an approximation for  $G_T\{R, e\}$  here, but in Section 4.6 we show an approximation of the value function of a recursive model formulation given by Powell and Carvalho (1998a).

### **4.2.3. The difficulties of vehicle management and the available information**

The difficulties of the vehicle management problem can be illustrated by looking at the effect of a perturbation of the vehicle supply. The effect of increasing the supply

of vehicles by one unit at a node in the network is described by a flow-augmenting path. This can be illustrated with a simple situation in which the flows do not interfere with each other. Each node represents a terminal at a specific time interval.



**Figure 4.1.** Example of flow augmenting paths.

In Figure 4.1 we have an additional unit of capacity (AGV) at terminal  $i$  in time period  $t$ . By assigning this extra AGV to a transportation job  $l$ , which was otherwise assigned to an AGV in time period  $t'$ , we get an extra AGV at node  $(j, t+1)$ . This results in an increase in flow on path **1**. But this also results in having one AGV less at node  $(j, t'+1)$ , which results in a decrease of the flow on path **2**. Since transportation job  $l$  is already planned, the AGV that was used for transportation job  $l$  at time  $t'$  can now be used to serve another transportation job (increase of flow on path **3**). In a large network, the effect of a change in vehicle flows early in the time horizon can result in an unpredictable impact on downstream activities. This is the reason to design heuristics because it is almost impossible to accurately estimate these downstream impacts.

To solve the vehicle management problem, we may use the following relevant information:

- 1) order information (for known and forecasted orders):
  - a. the origin and the destination
  - b. the arrival time and the due time
  - c. the current status (in progress, waiting to be processed or still to arrive)
- 2) vehicle information:
  - a. the current status (driving loaded, driving empty and assigned to an order, empty and unassigned)
  - b. the current (approximate) location and destination
  - c. the expected finishing time of the current activity
- 3) dock information:
  - a. the current functionality and status (assigned loading or unloading task; free or occupied)
  - b. the expected finishing time of the current activity

### 4.3 Literature

The issue of empty vehicle management is certainly not new in the literature. This problem arises in various settings, such as empty railcar distribution, empty truck repositioning and empty container distribution. One of the first papers in this area is by White and Bomberault (1969), who model the allocation of empty freight cars in a railroad system as a transshipment problem with deterministic demand. Since then, many model extensions and practical applications have been discussed in the literature.

In a railroad environment, Jordan and Turnquist (1983) formulated a dynamic model for empty car allocation with stochastic demand and stochastic travel times. A recent contribution has been made by Holmberg et al. (1998), who develop an optimization model for a situation with deterministic supply and demand, pre-specified train schedules, a heterogeneous fleet and limited train capacity. Spieckermann and Voss (1995) present an interesting case study, comparing a heuristic approach to a network flow model for a German railcar rental company.

In the field of empty truck distribution, Powell in particular has done a lot of work. Powell et al. (1988) describe the basic problem setting, where a practical solution is given for empty truck repositioning for a full truckload carrier in the United States under uncertain demand. Amongst others, Cheung and Powell (1994), Powell (1996) and Powell and Carvalho (1998a) presented theoretical improvements and model extensions. A different setting is described by Du and Hall (1997), who consider a hub-and-spoke network with random demand, where empty car stocks are controlled by a decentralized (S-1, S) inventory policy. However, the basic problem in their paper is fleet sizing.

A third related area is empty container allocation. Crainic et al. (1993) construct a hierarchic model, consisting of a strategic level (e.g. customer assignment), a tactical level (e.g. empty container balancing based on known orders and forecasts) and an operational level (e.g. freight routing). However, they do not describe solution procedures for the models presented. A recent contribution has been made by Cheung and Chen (1998), who constructed a single-commodity stochastic network flow model for the distribution of empty containers under random demand and transportation capacities.

Finally, related literature includes the scheduling of AGVs in a production or warehousing environment, see e.g. Akturk and Yilmaz (1996). De Koster and Van der Meer (1998) compare centralized and decentralized control systems for internal transport. In a case study they show that a centralized control system can lead to a significant reduction in the required number of vehicles. For a literature review of earlier work in the area of empty car management, we refer to Dejax and Crainic (1987). They present an overview and classification of empty car allocation models. We can relate our model to the classification scheme of Dejax and Crainic (1987) as follows: We have an operational model for empty car dispatching with a single

transportation mode and a homogeneous fleet in a dynamic setting with stochastic demand.

If we compare our model with the available literature, we see that the model discussed by Powell (1996, 1998a) shows most similarities. However, there are some important differences:

- 1) We face capacity restrictions at each network node, arising from a limited number of loading and unloading docks.
- 2) The time horizon is very short compared to applications in rail car distribution, truck distribution or container distribution. While in existing applications the planning horizon covers at least several days, we have to plan for at most a few hours. Orders are known only a short period in advance and should be processed very rapidly. This implies that a relatively high planning frequency and hence fast planning methods are required. From a computational point of view, a heuristic approach seems to be more suitable in this situation than time-consuming optimization models.
- 3) We focus on attaining customer service levels instead of cost minimization.
- 4) We allow an asymmetric track network, which is quite uncommon in other applications.
- 5) The presence of terminals requiring operational control rules for order release and scheduling demands a proper planning decomposition.

To our knowledge, empty vehicle management in such a setting has not been discussed before in the literature.

#### **4.4 Heuristics for Empty Vehicle Management**

In this section, we present five options for empty vehicle management (EVM). We focus on global empty vehicle management and explain the connection to the *local empty vehicle manager* per option. The variants differ with respect to the amount of information used and the level of coordination; see Table 4.1 for an overview. The names mentioned in the table primarily refer to the operation of the *global empty vehicle manager*. The first two variants are merely on-line dispatching rules that react to each order arrival, while the last three variants are capacity planning procedures that rebalance vehicle flows periodically.

**Table 4.1.** *Variants for Empty Vehicle Management (EVM)*

<b>Variant for EVM</b>	<b>Central information</b>	<b>Coordination</b>
EVM1: FCFS, myopic	1. release time and route for <ul style="list-style-type: none"> <li>• transportation jobs present</li> </ul> 2. vehicle status, location and time ready	local
EVM2: FCFS, look-ahead	1. release time and route for <ul style="list-style-type: none"> <li>• transportation jobs present</li> <li>• future transportation jobs known</li> </ul> 2. vehicle status, location and time ready	local
EVM3: Hierarchical coordination	1. release time, latest departure time and route for <ul style="list-style-type: none"> <li>• transportation jobs present</li> <li>• future transportation jobs known</li> </ul> 2. vehicle status, location and time ready	hierarchical
EVM4: Integrated planning	1. release time, latest departure time and route for <ul style="list-style-type: none"> <li>• transportation jobs present</li> <li>• future transportation jobs known</li> </ul> 2. vehicle status, location and time ready	central
EVM5: Logistic Queuing Network	1. release time, latest departure time and route for <ul style="list-style-type: none"> <li>• transportation jobs present</li> <li>• future transportation jobs known</li> </ul> 2. vehicle status, location and time ready	central

In this section we discuss the first three heuristics. The central coordinated heuristics are discussed in later sections, the integrated planning of EVM4 in Section 4.5 and the Logistic Queuing Network approach in Section 4.6.

#### **4.4.1. Local coordination using dispatching rules (EVM1 and EVM2)**

The simplest variant of global empty vehicle management (EVM1) is to dispatch available vehicles to terminals on a First-Come First-Served (FCFS) basis. Vehicle requests arrive at the *global empty vehicle manager* at times  $T_i^s$ . At any point in time, the set of available vehicles is defined as:

- empty vehicles that have not yet been dispatched to a terminal, i.e. vehicles that are waiting in or driving to a (local or central) parking location, or vehicles that become empty;
- loaded vehicles that can be assigned to the next transportation job, to be processed when the current job is finished.

Hence, every vehicle (loaded or empty) has as attribute the terminal where the next transportation job should be picked up. Available vehicles are vehicles without a next pickup location. Note that a vehicle can already receive a next pickup location when it starts loading at a dock. The time at which this vehicle can pickup the next



transportation job can be estimated from the expected handling time of the current job (cf. Section 2.6).

The procedure for global empty vehicle management is as follows. Each time when a load  $l$  arrives at a terminal  $i$  (at time  $T_l^s$ ), the *global empty vehicle manager* receives a vehicle request. If there are still vehicles available, the vehicle that is nearest to terminal  $i$  is dispatched. This is the vehicle with the earliest expected arrival time at terminal  $i$ . Computation of this time depends on the current status and location of the vehicle. If no vehicle is available, the request is added to a backorder list. This backorder list is sorted according to order arrival. As soon as a vehicle becomes available, the first request from the backorder list is satisfied.

*Local empty vehicle management* handles empty vehicles *within* the terminal. The following two events require a decision:

- 1) An empty vehicle arrives at the entrance of terminal  $i$ , having  $i$  as next pickup location. In the case of a local control concept, the decision at which dock to pick up a transportation job has not been taken yet. *Vehicle scheduling* has to decide upon the loading dock to be assigned. If all loading docks are occupied or no unassigned load jobs are present, the empty vehicle is sent to a parking place by *local vehicle control*. If no parking place is available, the empty vehicle waits in front of the terminal until space becomes available.
- 2) A vehicle becomes empty after it has unloaded at some dock in terminal  $i$ .
  - a. If it has no next pickup location, it is directed to the nearest parking area by *local vehicle control*. In periods of heavy traffic, this situation is not likely to occur because the backorder list will not be empty and then it is impossible for a vehicle to have no next pickup location.
  - b. If it has a next pickup terminal  $j \neq i$ , the vehicle is dispatched to terminal  $j$  (this has already been decided by the *global empty vehicle manager*). If a transportation job with the same destination is available at the terminal and a loading dock is available too, the vehicle transports this order. In this way, empty vehicle dispatch from the terminal and order processing can be combined without significant delay.
  - c. If it has terminal  $i$  as next pickup location, a load is assigned to it by *vehicle scheduling*, if possible. If the vehicle cannot be assigned to a loading dock, it is sent to a parking place by *local vehicle control*. If the parking places are full, the vehicle is dispatched to the nearest parking area with notification to the *global empty vehicle manager*. As the vehicle was assigned to pick up a transportation job, the *global empty vehicle manager* immediately dispatches the nearest available vehicle to terminal  $i$  as replacement. If no other vehicle is available, the request is inserted at the top of the backorder list.

Note that a simple FCFS rule may not be as bad as it seems, because transportation jobs are released in each terminal according to their latest departure time (cf. Section

2.6) and because empty vehicle dispatch is combined with transportation jobs whenever possible.

#### *Look-ahead variant (EVM2)*

If transportation jobs are known some time in advance (for example 30 minutes), we may use this information to improve the planning procedure. When transportation job  $l$  becomes known in the system, a vehicle request is sent to the *global empty vehicle manager*. Similarly to the first variant (EVM1), the vehicle requests are satisfied FCFS and if multiple vehicles are available, the vehicle  $m$  with the earliest expected arrival time at terminal  $i$  is selected. However, if vehicle  $m$  is dispatched immediately, it may arrive too early on the terminal and cause congestion. For vehicles from a parking area, the empty vehicle manager can prevent this by delaying their departure. This is never useful for EVM1, as vehicles cannot arrive too early. Therefore, the *global empty vehicle manager* dispatches vehicle  $m$  to terminal  $i$  as follows:

- if vehicle  $m$  is loaded and has no next pickup location, assign terminal  $i$  as next pickup location;
- if vehicle  $m$  is empty and waiting in or driving to a parking area, reserve the vehicle and dispatch it at its earliest dispatch time, which guarantees that the vehicle will not arrive too early (and cause excessive queues at the terminal entrance); if dispatch is delayed, the vehicle waits in the parking area.

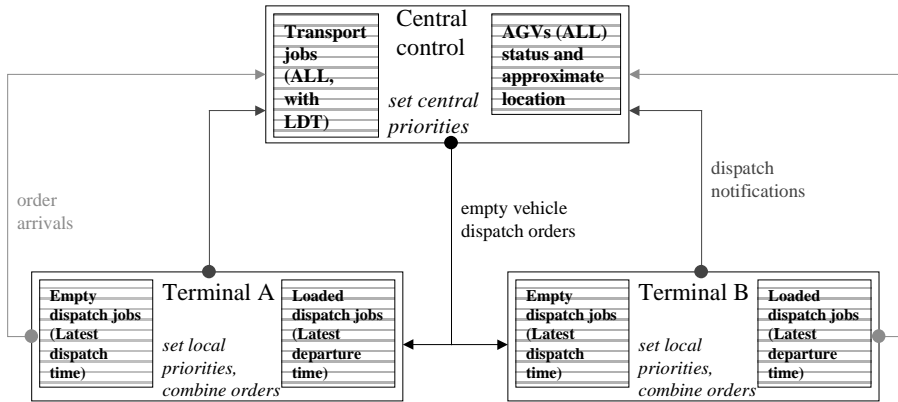
Note that the earliest dispatch time is introduced here, because early empty vehicle arrival is only possible in the case of prior information about transportation jobs. Under myopic planning, a load will always be present for every vehicle that arrives at a terminal, because the transportation job for the vehicle has already been released. Furthermore, note that *local empty vehicle management* is equivalent to the procedure from the previous section.

#### **4.4.2. Hierarchical coordination (EVM3)**

In the first two options, the *global empty vehicle manager* does not take into account priorities when dispatching vehicles to terminals and also, the vehicles cannot be fully locally controlled in a terminal, because some vehicles may already have a next pickup location. In this section, we discuss improvements on both issues. That is, we improve coordination between terminals. We neglect finite handling capacities caused by a finite number of docks.

We assume that the *global empty vehicle manager* has full control of vehicles outside the terminals, i.e. the destination of each individual empty vehicle may be changed at any point in time. Within a terminal however, the *global empty vehicle manager* only controls the *number* of vehicles. This fits into the local control concept, even though complete scheduling is possible (cf. Section 4.5). A *global empty vehicle manager* may make a request for empty vehicles to be sent to another terminal, but the local authority may decide *which* vehicles and *when* to dispatch.

The *global empty vehicle manager* sets priorities to empty vehicle requests, using the latest dispatch time  $sL_m$ . That is, the latest time that an empty vehicle has to be dispatched from a specific terminal or parking area, in order to pick up transportation job  $l$  at terminal  $j$  before the latest departure time  $tL_l$ . The coordination between global and local empty vehicle management is shown in Figure 4.2.



**Figure 4.2.** Vehicle management through hierarchical coordination

The *global empty vehicle manager* periodically plans empty vehicle redistribution between terminals and parking areas (e.g. every 10 minutes). Two lists are available for planning purposes:

- a list of all known transportation jobs at all terminals (both present and underway) that have not been loaded yet;
- a list of all vehicles with status (loaded or empty) and approximate location.

The coordinated empty vehicle planning is as follows. First, the list of known transportation jobs is sorted in increasing order of latest departure times  $tL_l$ . This list is processed sequentially. Iteratively we assign the still-available vehicle, which can start transporting the job at the earliest point in time, to the transportation job at the top of the list. The planning procedure continues until the transportation job list is fully assigned or all vehicles are assigned. Depending on the location of the vehicles, empty vehicle requests are issued. Assume that job  $l$  has to depart from terminal  $i$  at time  $tL_l$  and vehicle  $m$ , which is assigned to this job, is currently at terminal  $j$ . Then terminal  $j$  will receive an empty vehicle request to send a vehicle (not necessarily vehicle  $m$ ) to terminal  $i$  before time  $sL_m$ , which is equal to  $tL_l$  minus the expected driving time from terminal  $j$  to terminal  $i$ .

A *local empty vehicle manager* assigns tasks to empty vehicles within a terminal. An empty vehicle becomes available in a terminal at the following events:

- 1) An empty vehicle enters the terminal.

2) A vehicle becomes empty after it has unloaded at a dock.

To assign a task to an empty vehicle that becomes available, the *local empty vehicle manager* uses the following information:

- A. a list of requests to dispatch empty vehicles of its own terminal, sorted on latest dispatch time  $\underline{sL}_m, m=1, 2, \dots$
- B. a list of known transportation jobs at its own terminal, sorted on latest departure time  $\underline{tL}_l, l=1, 2, \dots$

The underlined variables refer to the respective sorted lists. If  $\underline{sL}_j < \underline{tL}_l$ , or if the list of known transportation jobs is empty, the *local empty vehicle manager* dispatches the vehicle according to the first destination on list A. If  $\underline{tL}_l \leq \underline{sL}_j$ , or if the list of empty vehicles to be dispatched is empty, the *local empty vehicle manager* assigns the empty vehicle to the first transportation job on list B. If both lists A and B are empty, the *local vehicle control* sends the vehicle to a parking place if it is needed in the near future and if the parking place is not full. Otherwise the vehicle is sent to the nearest parking area and notification of this is sent to the *global empty vehicle manager*.

The *local empty vehicle manager* tries to combine jobs whenever possible. That is, if an empty vehicle request to terminal  $j$  has highest priority ( $\underline{sL}_j < \underline{tL}_l$ ), a transportation job for the same terminal  $j$  with smallest latest departure time  $\underline{tL}_l$  is sought. If such a transportation job is available on the list and a loading dock is available too, this job is scheduled (and removed from the list). Otherwise the vehicle is dispatched empty to terminal  $j$ . Vice versa, if a transportation job for some terminal  $j$  has highest priority ( $\underline{tL}_l \leq \underline{sL}_j$ ), the list of empty vehicle requests is searched for a request with smallest latest dispatch time with the same destination  $j$ . If such a job is found, it is removed from the list and the *global empty vehicle manager* is informed that the request has been fulfilled. In response, the *global empty vehicle manager* may provide another empty vehicle job to the terminal, but this job will have less priority.

This procedure has advantages over the simple rules from Section 4.4.1 in the sense that priorities are used by both the *global* and the *local empty vehicle manager*, while the two decision layers still have responsibilities that are properly separated.

## 4.5 Integrated planning (EVM4)

In the previous variant, the global and local empty vehicle managers seem to be well coordinated. A drawback, however, is the fact that only the next assignment of each vehicle is taken into account, without considering its impact on the distribution of vehicles throughout the transportation system later on (cf. Figure 4.1). For example, the *global empty vehicle manager* may dispatch empty vehicles to terminal 1,

knowing that they will be used to transport cargo to terminal 2. Therefore these vehicles can be used for transportation jobs from terminal 2 to another destination later on. If the *global empty vehicle manager* ignores this information, as in EVM3, the result can be that empty vehicles will be dispatched to terminal 2 that appear not to be needed later. This may be overcome by making an integrated planning procedure in which all transportation jobs and empty vehicle trips are considered. This integrated planning is the main difference between EVM4 and EVM3. A requirement for integrated planning is that all necessary information about transportation jobs, vehicles and docks is centrally available.

#### *Relation between global and local empty vehicle manager*

Based on information about the system status and all transportation jobs, the *global empty vehicle manager* optimizes a transportation schedule for all transportation jobs and trips. This means that the *global empty vehicle manager* virtually assigns a sequence of jobs to each vehicle. When planning more jobs ahead, as in this integrated approach, some jobs may seem inefficient locally, but lead to a better schedule globally. Therefore, based on the transportation schedule, the *global empty vehicle manager* provides the *local empty vehicle managers* with a sequence of transportation jobs for each terminal and a list of empty vehicle jobs with dispatch times. The *local empty vehicle managers* follow this schedule in the sense that transportation jobs are handled in the prescribed sequence and empty vehicles are dispatched in the prescribed time window, or as soon as possible afterwards. Note that the dispatch windows of empty vehicles are required to avoid a situation in which an available empty vehicle, waiting for a transportation job, is used for another empty vehicle job. Recall that the *global empty vehicle manager* as described in the previous section (EVM3) simply provides a list of empty vehicle dispatch jobs with *latest dispatch time* to the terminals.

Given the order prescribed by the *global empty vehicle manager*, the assignment of docks, the scheduling of unload jobs and the exact timing of transportation jobs, is left to the *local vehicle scheduler*. The *local empty vehicle manager* is also responsible for the assignment of particular empty vehicles to particular jobs. Although the latter seems to be a marginal authority, this element of the control structure is still important. It gives the *local empty vehicle managers* the authority to deal with local circumstances, such as vehicle positioning related to terminal layout and reacting to local disturbances (dock and vehicle failures). Assigning all authority to the network level would lead to an unwieldy control system with excessive information exchange (e.g. all local disturbances should be known at network level).

#### *The integrated approach for the global empty vehicle manager*

The idea for the integrated planning approach is as follows. Periodically the *global empty vehicle manager* constructs an integral schedule. This can be considered as an off-line problem, since new transportation jobs that become known later on are neglected. New transportation jobs will only be incorporated when the *global empty*

*vehicle manager* makes a new plan. Therefore, for such an approach to be fruitful, the planning period for the *global empty vehicle manager* should not be too long, since otherwise too many new transportation jobs are missed. However it should not be too short either since otherwise the efficiency of the global plan is not obtained. Of course, what too short and too long means may vary from case to case. We will deal with this issue in Section 4.7.2 when we discuss the numerical experiments.

For the integrated planning by the *global empty vehicle manager* EVM4 we apply a serial scheduling method, which uses the priorities of jobs to sequentially build a schedule, cf. Kolisch (1996), who describes serial and parallel scheduling methods for resource-constrained project scheduling. The objective is to optimize due-time performance, i.e. to minimize the maximum lateness of the jobs. Next, given a fixed system status, we describe this method to optimize a schedule. As mentioned above, the information is continuously subject to changes due to the dynamics of the system. This can be handled by calling the serial scheduling method either periodically (with a fixed time interval) or event triggered (at each order arrival). As the second option would lead to excessive planning time because of the high order arrival rate, we chose scheduling with a fixed time interval.

#### *The serial scheduling method*

Input to the scheduling is the current status of the system and a list of transportation jobs. The status of the system is defined by the current time, the expected travel times for all origin-destination pairs in the system, the destination and the expected arrival time of every vehicle and the status of a vehicle (loaded or empty). The list of transportation jobs contains all transportation jobs  $l$  known to the system ( $tK_l \leq t_0$ ), including transportation jobs in progress.  $tK_l$  is the time at which transportation job  $l$  is known in the system. Recall that a transportation job in progress may be on a loading dock waiting to be handled, being transported on a vehicle, or at a dock, being unloaded.

We use a serial scheduling method, based on deterministic information. That is, we treat all expected handling, travel and arrival times as deterministic variables. Recall that these times are actually random variables, so the assumption of deterministic information serves as an approximation. This assumption is justified by the fact that the variations in the on-line, time-dependent forecasts for these random variables are relatively small. Note however, that the values of e.g. mean handling times, may vary, depending on workload. The following steps are taken in our serial scheduling method:

- 1) *Initialization.* From the status of the system, compute *initial empty vehicle profiles* for every parking area and terminal. An initial empty vehicle profile describes the number of available empty vehicles for any  $t \geq t_0$ , given that no further jobs will be started. Furthermore, we compose a sorted list of jobs to be scheduled, containing all jobs  $l$  that still have to be started, sorted by increasing latest departure time  $tL_l$ .

- 2) Let job  $l$  be the first job on the sorted list. Evaluate all possible assignments of empty vehicles to job  $l$ , and assign the empty vehicle that serves job  $l$  as early as possible. The resulting assignment leads to an *earliest serving moment*  $tE_l$  and a slack  $tS_l = tL_l - tE_l$  for job  $l$ .
- 3) Determine whether any of the other jobs on the list can be *combined* with job  $l$ . By our definition, a job  $k$  can be combined with job  $l$  only if the destination of job  $k$  equals the origin of job  $l$ . If such a combination is possible, it prevents unnecessary empty vehicle movements, but it may also cause a delay compared to the earliest serving moment  $tE_l$ . Because job  $l$  has highest priority, combined transportation is controlled by two parameters, namely the *maximum delay* in serving time and the *minimum slack* that must remain for job  $l$ ; that is the time gap between the scheduled starting time and the latest departure time of job  $l$ . To combine transportation, determine the jobs that can be combined with job  $l$ , such that the minimum slack and maximum delay are not violated. Amongst all these jobs, accept the job  $k$  that leads to the smallest delay for job  $l$ . Note that job  $k$ , and afterwards job  $l$ , may use a different empty vehicle from the one assigned to job  $l$  in step 2. In that case the empty vehicle assignment of step 2 is cancelled.
- 4) Remove job  $l$  and, in the case of a combined transportation, remove job  $k$  from the list of jobs to be scheduled. Update the empty vehicle profiles based on these scheduled jobs.
- 5) Repeat steps 2 to 4 until the list of jobs to be scheduled is empty.

We notice that the transportation jobs govern the scheduling process for EVM4. That is, unlike the planning options as described in Sections 4.4.1 and 4.4.2, in the integrated planning approach a vehicle may wait empty for a transportation job to be processed instead of starting with the most urgent transportation job amongst those orders that can start immediately.

Clearly, this sequential scheduling approach may improve the performance of empty vehicle management in terms of due time performance. The efficiency of the method may depend on two parameters, maximum delay and minimum slack, so it may be worthwhile to tune it to a specific case. Note that larger values for the maximum delay and smaller values for the minimum slack leave more room for combined transportation. On the other hand, a larger maximum delay may postpone more jobs and a smaller minimum slack may result in more jobs that are scheduled close to their latest departure time. Such decisions may appear efficient for the integrated off-line planning. However, the efficiency in a real-time, dynamic environment will depend on the length of the planning period and the dynamics and stochasticity of the environment.

Note that the scheduling method neglects two finite capacities, namely the capacity of the docks and the parking capacity of the terminals. Thus it assumes that (1) every job can be handled immediately at the dock (at the assigned time) and (2) that every empty vehicle can stay at the terminal where it becomes available. The first

assumption may be critical at some moments. In the Stommeer layout of the OLS-case, the availability of the vehicles is the bottleneck rather than the capacity of the docks. For other cases, we believe that it is possible to extend our method towards finite docking capacities. The second assumption is less critical if the terminal has reasonable parking space. Besides, we assume that each terminal has queuing space in front of the entrance (see Chapter 3). As we shall see in Section 4.7, the simulation results suggest that these limiting assumptions are not too harmful in the case under consideration. This could change, however, if terminals have very little parking space. Then refinements are possible by introducing dock and parking profiles.

The flexibility of the serial scheduling approach is a clear advantage. Several schedules based on different performance criteria, priority rules and parameter settings (maximum delay, minimum slack) can quickly be generated. We can also use a given maximum computation time as restriction of the number of schedules to generate.

## **4.6 Logistics Queuing Network (EVM5)**

Powell and Carvalho (1998a) formulate the vehicle management problem as a dynamic network with three types of arcs. First, revenue generating arcs that correspond to vehicles moving cargo to satisfy demand. Second, empty repositioning arcs over which vehicles are moved from one location to the next (possibly with costs). And third, inventory arcs, which represent the vehicles that are kept at a terminal from one time period to the next. Powell and Carvalho (1998a) use a Logistic Queuing Network (LQN) to solve the vehicle management problem. Note that their logistic queuing network is not exactly the same as other queuing networks encountered in the literature. We will use the terminology presented by Powell and Carvalho (1998a). The objective is to maximize the revenues by assigning a vehicle to each transportation job. The LQN approach starts with a classical linear programming formulation, based on a discrete time setting, as described in Section 4.2, in which the parking area is modeled as a terminal without transportation jobs. This model is reformulated as a deterministic dynamic programming model. Next, the reward function in the dynamic programming recourse is approximated by a linear function, depending on the expected number of vehicles at each node at each time, given the vehicle dispatch decisions already taken. Given this linear approximation, vehicle dispatches can be determined for each terminal and time period using a simple single sort. Hence, the LQN approach decomposes vehicle management decisions by time and space. For details we refer to Section 4.6.1.

The challenge is how to estimate the slope of the approximating reward function, representing the marginal value of an additional vehicle at a certain location in a



certain time period. To this end, Powell and Carvalho (1998a) propose an iterative procedure and suggest limiting the number of empty vehicles dispatched to each terminal by an upper bound, that has to be estimated iteratively as well. Each iteration of the algorithm involves a simulation of the dispatching process, after which the gradients (slopes of the approximating reward function) and control variables (upper bounds on empty vehicle flows) are updated to improve the solution (cf. Section 4.6.2 and 4.6.3, respectively). In this way, the LQN approach is applied to solve fleet management problems. In their paper, Powell and Carvalho (1998a) show results that are within a few percent of the optimal solution of the linear programming relaxation. They do not apply the algorithm in a rolling horizon environment.

The LQN approach has some nice features:

- All transportation jobs are dealt with individually. This is very useful because each transportation job has specific values for its release time and latest departure time. So different priorities can be included in the model.
- Calculation times mentioned by Powell and Carvalho (1998a) are acceptable. The time-space decomposition reduces the calculation times enormously, which is required for a real-time approach.
- The method can handle a wide variety of operational issues that may arise in specific applications. Due to the time-space decomposition there is a lot of flexibility.

The development of EVM5 will largely be based on Powell and Carvalho (1998a). We have to make some adjustments to cope with the specific characteristics of our model. Before describing the LQN algorithm in more detail, we first discuss these differences.

Powell and Carvalho (1998a) make the following assumptions:

- Transportation jobs of which the latest departure time has been passed are lost (as is the corresponding revenue) and will not be planned.
- The travel time between two nodes is deterministic and fixed.

For automated transportation networks we use different assumptions. We assume that all transportation jobs have to be transported. In consequence the revenues for a particular transportation job always have to be positive, even when the job is delivered late (after the due time) there still has to be some revenue left. Another difference is that owing to the relatively short driving times, the loading and unloading of vehicles takes a significant amount of time. Therefore, the travel times for empty and loaded vehicles are different. Loaded driving times include loading, unloading and waiting times in between these activities. These differences in travel times may be significant. The algorithm had to be adjusted slightly, because it appeared to be sensitive to the order in which the list of terminals is processed. The reason is that a lot of transportation jobs have the same revenues. Furthermore,

unlike Powell and Carvalho (1998a), we apply the model in a dynamic context (simulation).

#### 4.6.1. Reformulating the ILP model

The main difference between the LQN approach and that of the normal solution methods for an integer linear programming formulation, as described in Section 4.2, is that we do not seek a global solution to the entire problem. Instead, we solve a sequence of smaller sub-problems. Each sub-problem assigns AGVs to transportation jobs and empty vehicle jobs at a single terminal in a given time period. The challenge of this decomposition is to make the local decision strategy smart enough to approximate the global optimum. We will now change the integer linear programming model in such a way, that we can use the LQN approach to find a solution. The new formulation results from:

1. Writing the objective function in a recursive form.
2. Approximating the future value function at each time period as a linear function of the number of vehicles available at each node.
3. Constraining the unbounded decision variables (in this case the  $y_{ijt}$ ). This is necessary to prevent a terminal from being flooded with empty AGVs at a specific point in time, because when a linear approximation is used the marginal value of additional AGVs does not decrease with the number of AGVs dispatched.

Before describing these steps, we first introduce some notation, in addition to the notation defined in Section 4.2. In the equations we use the following vector notations:

$$x_t = \{x_{lt} \mid l \in \bar{L}_t\}, x = \{x_t \mid t \in [1, \dots, T]\}$$

$$y_t = \{y_{ijt} \mid i, j \in M\}, y = \{y_t \mid t \in [1, \dots, T]\}$$

*Activity variables:*

- $L_{it} = \bigcup_{j \in M} L_{ijt}$ , i.e. the set of all transportation jobs  $l \in L$  with origin  $i$  having  $t$  as feasible departure time.
- $\bar{L}_{ijt}$  is the set of transportation jobs  $l$  with origin  $i$  and destination  $j$ , which are available to move at time  $t$  and have not been moved at a time prior to time  $t$  in a given solution.
- $\bar{L}_{it} = \bigcup_{j \in M} \bar{L}_{ijt}$ , i.e. the set of all transportation jobs  $l \in L$  with origin  $i$ , which are available to move in time period  $t$  and have not been moved in a time period prior to time period  $t$ , given the present values of the decision variables.
- $V_{it}$  is the total number of empty vehicles at node  $(i, t)$  waiting to be assigned to a transportation job or empty move,  $V_t = \{V_{it} \mid i \in M\}$ .
- $L_{it}^0 = \{l \in L_t \mid t = T_l^s\}$ , i.e. the set of transportation jobs  $l$  with origin  $i$ , where  $t$  is the time period  $T_l^s$ , which is the beginning of the time window  $T_l$ .

- $L_{it}^s = \{l \in \bar{L}_{ijt} \mid x_{lt} = 1\}$ , i.e. set of transportation jobs  $l$  with origin  $i$ , where  $t$  is the time period that transportation job  $l$  is started to be transported.
- $\bar{L}_t = \bigcup_{t' \geq t} \bar{L}_{it'}$ , i.e. all the transportation jobs, which are still in the system from time period  $t$  to time period  $T$ , given the present values of the decision variables.

(i) *Recursive model formulation*

We start with writing the objective function  $F(x,y)$  in a recursive form.

$$\begin{aligned}
 F(x, y) &= \sum_{t=1}^T \sum_{i \in M} \sum_{j \in M} \left( \sum_{l \in L_{ijt}} r_{lt} x_{lt} - c_{ijt} y_{ijt} \right) \\
 &= \sum_{t=1}^T \sum_{i \in M} g_{it}(x_t, y_t, V_{it}, L_{it})
 \end{aligned} \tag{7}$$

in which  $g_{it}(x_t, y_t, V_{it}, L_{it}) = \sum_{l \in L_{it}} r_{lt} x_{lt} - \sum_{j \in M} c_{ijt} y_{ijt}$

We now define  $G_t$  as:

$$G_t(V_t, \bar{L}_t) = \sum_{k=1}^T \sum_{i \in M} g_{ik}(x_k, y_k, V_{ik}, \bar{L}_{ik}) \quad \forall t \in [1, \dots, T] \tag{8}$$

Using this we can define  $G_t$  recursively as (cf. DP formulation, Section 4.2.2):

$$G_t(V_t, \bar{L}_t) = \sum_{i \in M} g_{it}(x_t, y_t, V_{it}, \bar{L}_{it}) + G_{t+1}(V_{t+1}, \bar{L}_{t+1}) \quad \forall t \in [1, \dots, T] \tag{9}$$

$$\text{We notice that: } F(x, y) = G_1(V_1, \bar{L}_1) \tag{10}$$

The objective function  $G_t$ , consists of two parts. The first part contains  $g_{it}$ , which is the contribution of the decisions taken at terminal  $i$  in time period  $t$ . The second part is  $G_{t+1}$ , which is the contribution of the decisions taken from time period  $t+1$  to the end of the planning horizon.

(ii) *Approximating the value function*

The value function  $G_t(V_t, \bar{L}_t)$  is intractably complex, and therefore we replace the objective function by a linear approximation. The decision to move capacity to a certain terminal depends on the value of the revenue generated by the additional capacity at this terminal. We therefore have to calculate the marginal value of additional capacity at each location, in each time period. By approximating  $G_{t+1}$  in time period  $t$ , we actually are approximating the future. This is very complex and we therefore approximate the future by a function that can be influenced in time period  $t$ . All loaded moves that start in time period  $t$  will arrive in time period  $t + \tau_{ij}^l$  and all

empty moves that start in time period  $t$  will arrive in time period  $t + \tau_{ij}^e$ . All moves in time period  $t$  affect time period  $t + \tau_{ij}$  ( $\tau_{ij} = \tau_{ij}^e$  or  $\tau_{ij}^l$ , abbreviated for convenience). We therefore adopt the following linear approximation for the future value function  $G_{t+i}$ :

$$\hat{G}_{t+i} = \xi_{t+\tau} V_{t+\tau} = \sum_{j \in M} \xi_{j,t+\tau} V_{j,t+\tau} \quad (11)$$

The value  $\xi_{it}$  is an estimate of the slope of the value function  $G_t$ , with respect to the supply of vehicles  $V_{it}$  and is a measure of how desirable it is to have one additional AGV at node  $(i,t)$ . We refer to  $\xi_{it}$  as the supply difference. The vector  $\xi_t$  gives the marginal value of additional AGV capacity at each location in time period  $t$ . We use a linear approximation of the value function  $G_{t+i}$  because it is the simplest to estimate and use. The original formulation of Powell and Carvalho (1998a) also includes a task gradient, which is incorporated because the supply of vehicles available at a node perturbs the queue of transportation jobs and therefore the attainable revenues in the future. Powell and Carvalho (1998a) neglect this task gradient in the description of the algorithm. This might not be too harmful in their case, because not all transportation jobs have to be transported and they only look at the static problem (no rolling horizon). In our case all transportation jobs have to be transported, and therefore an estimate of the remaining transportation jobs may be important. Two plans with the same total revenue can have a different set of remaining jobs, with corresponding different realizable revenues. We discuss the inclusion of a task gradient in Section 4.6.5.

Substituting  $\hat{G}_{t+i}$  for  $G_{t+i}$  into equation (9) gives  $\hat{G}_t$ , which is an approximation of  $G_t$ :

$$\begin{aligned} \hat{G}_t(V_t, \bar{L}_t) &= \sum_{i \in M} g_{it}(x, y, V_{it}, \bar{L}_{it}) + \xi_{t+\tau} V_{t+\tau} \\ &= \sum_{i \in M} \sum_{j \in M} \left( \sum_{l \in L_{ij}} (r_{lt} + \xi_{j,t+\tau_{ij}^e}) x_{ijt} + (-c_{ijt} + \xi_{j,t+\tau_{ij}^e}) y_{ijt} \right) \end{aligned} \quad (12)$$

Assume there is an empty AGV waiting at terminal  $i$  at time  $t$ . If we send this AGV to terminal  $j$ , it will arrive in time period  $t + \tau_{ij}^e$ . The value of this assignment consists of the negative revenue  $-c_{ijt}$  of the empty move from  $i$  to  $j$  and  $\xi_{j,t+\tau_{ij}^e}$ , representing the value of one additional AGV at terminal  $j$  in time period  $t + \tau_{ij}^e$ . If we assign the same AGV to a transportation job  $l$ , going to node  $(j, t + \tau_{ij}^l)$ , then the value of this assignment is the net revenue from the transportation job, which is  $r_{lt}$ , and the value of an additional AGV at the destination of transportation job  $l$  at time  $t + \tau_{ij}^l$ :  $\xi_{j,t+\tau_{ij}^l}$ .

(iii) *Constraining the unbounded decision variables.*

The approximation for the value function uncouples the decisions for the different terminals. Each vehicle arriving at terminal  $j$  in time period  $t'$  receives the same marginal value  $\xi_{j,t'}$  upon arrival, irrespective the terminal of departure. If this value is high, we run the risk of flooding the terminal with capacity and as a result starve other terminals with a low value of  $\xi_{j,t'}$ . We therefore introduce an additional decision variable  $u_{ijt}$ . This upper bound has to coordinate the flow of empty vehicles in our system, which leads to an additional constraint in the local problem.

For each terminal and specific time period we now have to solve  $P_2$ :

$$\max_{x_t, y_t} \sum_{j \in M} \left( \sum_{l \in L_{ij}} (r_{lt} + \xi_{j,t+\tau_{ij}^l}) x_{lt} + (-c_{ijt} + \xi_{j,t+\tau_{ij}^e}) y_{ijt} \right) \quad (13)$$

*Subject to:*

$$x_{lt} \leq 1 \quad \forall l \in \bar{L}_{it} \quad (14)$$

$$y_{ijt} \leq u_{ijt} \quad \forall j \in M, \quad j \neq i \quad (15)$$

$$\sum_{l \in L_{iq}} x_{lt} + \sum_{j \in M} y_{ijt} = V_{it} \quad (16)$$

$$x_{lt} = \{0,1\} \quad \forall l \in L \quad (17)$$

$$y_{ijt} \geq 0 \quad \text{integer} \quad \forall j \in M \quad (18)$$

We can successively solve the local problems at time period  $t=1, \dots, T$ . To this end, the set of available transportation jobs and the vector of the available vehicles in the next time period have to be determined using the information from period  $t$ . These two values are determined by using equations (19) and (20).

$$\bar{L}_{i,t+1} = \{ \bar{L}_{i,t} \setminus L_{i,t}^s \} \cup L_{i,t+1}^0 \quad (19)$$

$$V_{i,t+1} = RV_{i,t+1} + \sum_{k \in M} \left( \sum_{l \in \bar{L}_{k,i,t+1-\tau_{ki}^l}} x_{l,t+1-\tau_{ki}^l} + y_{k,i,t+1-\tau_{ki}^e} \right) \quad (20)$$

The last term of equation (20) does not include flows from before time period  $t = 1$ . These are included in the term  $RV_{i,t+1}$ , which contains the starting inventory of empty vehicles at terminal  $i$  in time period  $t+1$ , which is determined by using the initial state of the system. So for the indices we have the condition that  $t + 1 - \tau_{ki}^l > 0$  and  $t + 1 - \tau_{ki}^e > 0$ . Equation (19) states that the set of transportation jobs that is still available for transportation in time period  $t+1$  at terminal  $i$  is equal to:

- the set of available transportation jobs in time period  $t$ , except the transportation jobs for which we decided to start the transportation in time period  $t$ .

- the transportation jobs that are available for transportation for the first time in time period  $t + 1$ .

Given the simplicity of the sub-problem, we can think of the optimal solution  $(x_t, y_t)$  as being a function of the control variables  $\xi_{t+\tau}$  ( $= \xi_{t+\tau_j^i}$  or  $\xi_{t+\tau_j^e}$ ) and  $u_t$ . We can now state our total problem, concerning all terminals and all time periods, as one of optimizing the control variables  $\xi_{t+\tau}$  and  $u_t$  for all terminals  $i$  and all times  $t$ , as follows,

$$\max_{\xi, u} \sum_{t=1}^T \sum_{i \in M} \left( \sum_{l \in \bar{L}_i} r_{il} x_{il}(V_{it}, \xi_{t+\tau_j^i}, u_{it}, \bar{L}_{it}) - \sum_{j \in M} c_{ijt} y_{ijt}(V_{it}, \xi_{t+\tau_j^e}, u_{it}, \bar{L}_{it}) \right) \quad (21)$$

subject to (14)-(20). We denote this problem as  $P_3$ .

The restriction that all transportation jobs have to be planned,  $\sum_t x_t = |L|$ , is not included in the algorithm. But when the planning horizon ( $T$ ) is long enough, the number of iterations is not too small and the revenue of a transportation job ( $r_{it}$ ) is positive at all times  $t > T_l^s$ , then all transportation jobs should be transported because they all add revenue to the objective function.

#### 4.6.2. Calculating the gradients

The challenge at this point is to devise a strategy to choose  $\xi$  and  $u$ , such that the solution of the control problem closely matches the global optimization formulation. The approximation strategy will now be explained. We want to estimate  $\xi_t$ , which is the slope of  $G_t$ . Let  $v_{it}^+$  and  $v_{it}^-$  represent, respectively, the forward and backward differences of  $G_t$  with respect to  $V_{it}$ , i.e. the increase (decrease) in  $G_t$  if  $V_{it}$  increases (decreases) by one unit. We approximate these values by  $\hat{v}_{it}^+$  and  $\hat{v}_{it}^-$ , defined as:

$$\hat{v}_{it}^+ = \hat{G}_t(V_{it} + 1, \bar{L}_t) - \hat{G}_t(V_{it}, \bar{L}_t) \quad (22)$$

$$\hat{v}_{it}^- = \hat{G}_t(V_{it}, \bar{L}_t) - \hat{G}_t(V_{it} - 1, \bar{L}_t) \quad (23)$$

The forward difference will be used in the approximation of  $\xi_{it}$ , while both differences are required for calculating the upper bounds  $u_{ij}$ . Furthermore, we need the following definitions:

$$X_{it}^+ = x_{it}(V_{it} + 1, \xi_{t+\tau_{ij}^l}, u_{it}, \bar{L}_{it}) - x_{it}(V_{it}, \xi_{t+\tau_{ij}^l}, u_{it}, \bar{L}_{it}) \quad (24)$$

$$X_{it}^- = x_{it}(V_{it}, \xi_{t+\tau_{ij}^l}, u_{it}, \bar{L}_{it}) - x_{it}(V_{it} - 1, \xi_{t+\tau_{ij}^l}, u_{it}, \bar{L}_{it}) \quad (25)$$

$Y_{ijt}^+$  and  $Y_{ijt}^-$  are defined analogously. For equations (24) and (25) we assume that  $l \in \bar{L}_{it}$ . The variables  $X_{it}^+$ ,  $X_{it}^-$ ,  $Y_{ijt}^+$  and  $Y_{ijt}^-$  are  $\{0,1\}$  variables. Note that  $x_{it}$  and  $y_{ijt}$  are monotone and non-decreasing in  $V_{it}$ . We look at the consequences of increasing or decreasing values for  $V_{it}$ , which is the total number of vehicles at terminal  $i$  in time period  $t$ , waiting to be assigned to a transportation job or moved empty.  $X_{it}^+$  equals 1 if transportation job  $l$  is not transported when the number of vehicles is  $V_{it}$ , while transportation job  $l$  would have been transported if the number of vehicles had been  $V_{it} + 1$ .

By computing the differences of equation (12), we can produce the recursion relations for  $\hat{v}_{it}^+$  and  $\hat{v}_{it}^-$ , which are determined in the forward pass:

$$\hat{v}_{it}^+ = \sum_{j \in M} \sum_{l \in \bar{L}_{ij}} \hat{X}_{it}^+(r_{it} + \xi_{j,t+\tau_{ij}^l}) + \sum_{j \in M} \hat{Y}_{ijt}^+(-c_{ijt} + \xi_{j,t+\tau_{ij}^e}) \quad (26)$$

$$\hat{v}_{it}^- = \sum_{j \in M} \sum_{l \in \bar{L}_{ij}} \hat{X}_{it}^-(r_{it} + \xi_{j,t+\tau_{ij}^l}) + \sum_{j \in M} \hat{Y}_{ijt}^-(-c_{ijt} + \xi_{j,t+\tau_{ij}^e}) \quad (27)$$

The differences for all time periods  $t > T$  are assumed to be equal to zero, consequently there is no revenue for transportation jobs that could still be transported after time period  $T$ .

#### 4.6.3. Calculating the upper bounds

The upper bounds on empty vehicle trips play an important role in the algorithm. Increasing an upper bound  $u_{ijt}$  can have the effect of increasing the flow of empties from terminal  $i$  starting in time period  $t$  and arriving at terminal  $j$  in time period  $t + \tau_{ij}^e$  ( $i \neq j$ ). We introduce  $\eta_{ijt}^+$  and  $\eta_{ijt}^-$ , indicating the expected revenues of an increase or decrease, respectively, of  $u_{ijt}$ , to decide if we have to increase or decrease the upper bound on the empty flows. The decision to increase the upper bound should depend on the increase in the objective function of  $P_3$ . We have:

$$\eta_{ijt}^+ = \begin{cases} \hat{G}_i(x, y_{ijt} + 1) - \hat{G}_i(x, y_{ijt}) & \text{if } y_{ijt}(u_{ijt} + 1) - y_{ijt}(u_{ijt}) = 1 \\ 0 & \text{otherwise} \end{cases} \quad (28)$$

where we suppress the dependency of  $\hat{G}_i$  and  $y_{ijt}$  on other variables for ease of notation. We now have to determine the forward difference of  $\hat{G}_i$  with respect to

$y_{ijt}$ . The condition  $y_{ijt}(u_{ijt} + 1) - y_{ijt}(u_{ijt}) = 1$  can be seen as the condition in which if the upper bound is increased by 1 this indeed corresponds with an additional empty move. In order for this to happen, the empty move has to be more valuable than the task that would have capacity diverted to it. This can be presented as follows:

$$-c_{ijt} + \bar{v}_{j,t+\tau_{ij}^e}^+ > \bar{v}_{it}^- \quad (29)$$

This results in the following equation:

$$\eta_{ijt}^+ = \begin{cases} -c_{ijt} + \bar{v}_{j,t+\tau_{ij}^e}^+ - \bar{v}_{it}^- & \text{if } -c_{ijt} + \bar{v}_{j,t+\tau_{ij}^e}^+ - \bar{v}_{it}^- > 0 \\ 0 & \text{otherwise} \end{cases} \quad (30)$$

A similar derivation for  $\eta_{ijt}^-$  leads to

$$\eta_{ijt}^- = \begin{cases} c_{ijt} - \bar{v}_{j,t+\tau_{ij}^e}^- + \bar{v}_{it}^+ & \text{if } c_{ijt} - \bar{v}_{j,t+\tau_{ij}^e}^- + \bar{v}_{it}^+ > 0 \\ 0 & \text{otherwise} \end{cases} \quad (31)$$

When adjusting the upper bounds, we search for the largest increase in the objective function, i.e.  $\max_{i,j,t} \{\eta_{ijt}^+, \eta_{ijt}^-\}$ . Depending on whether the maximum is reached for  $\eta_{ijt}^+$

or  $\eta_{ijt}^-$ , the upper bound  $u_{ijt}$  is increased or decreased. One can also perform several upper bound adjustments at once, taking the adjustments that add most revenue to the objective function.

### Smoothing

The LQN algorithm is an iterative approach. In each iteration  $n$  we obtain a new set of flows  $(x_t^n, y_t^n)$ , new vehicle inventories  $V_t^n$ , and new differences  $\hat{v}_{it}^{(n)+}$  and  $\hat{v}_{it}^{(n)-}$ . As pointed out before, even a small change in an early time period can be magnified into larger changes in later time periods. As a result, the estimates  $\hat{v}_{it}^{(n)+}$  and  $\hat{v}_{it}^{(n)-}$  in iteration  $n$  might fluctuate considerably. For this reason, we smooth the estimates by using:

$$\bar{v}_{it}^{(n+1)+} = \gamma \hat{v}_{it}^{(n+1)+} + (1 - \gamma) \bar{v}_{it}^{(n)+} \quad (32)$$

$$\bar{v}_{it}^{(n+1)-} = \gamma \hat{v}_{it}^{(n+1)-} + (1 - \gamma) \bar{v}_{it}^{(n)-} \quad (33)$$

where  $\gamma$  is the smoothing factor ( $0 \leq \gamma \leq 1$ ), which can be chosen experimentally. If we want to reduce the instability, a low value for  $\gamma$  seems most appropriate. We will use in iteration  $n$  of the algorithm  $\xi_{t+\tau}^n = \bar{v}_{t+\tau}^{(n)+}$  for the linear approximation in (11).



#### 4.6.4. Summary of the LQN algorithm

The LQN algorithm contains the introduced variables, but they all have an extra index for the iteration number  $n$ . The idea behind this iterative procedure is the following. If we are in time period  $t$  and we need information about values of time period  $t'$ , with  $t' > t$ , we use the values determined in the previous iteration. The basic steps in the LQN algorithm are shown in Figure 4.3.

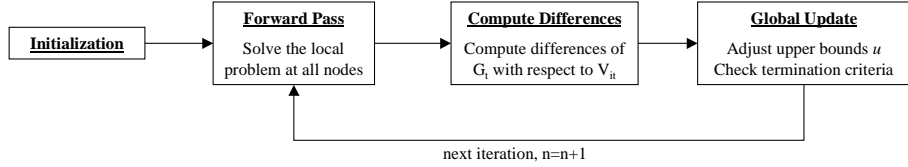


Figure 4.3. Basic steps in the LQN algorithm.

##### STEP 1 Initialization:

- Set  $\bar{v}_{it}^{(0)+}$ ,  $\bar{v}_{it}^{(0)-}$  and  $u^0 = 0$ , so we start with  $\xi_{it} = 0$ .
- Set  $n = 0$ .

##### STEP 2 Forward pass:

- For  $t = 1, \dots, T$ :
  - Find  $x_t^{n+1}(V_t^n, \bar{v}_{t+\tau}^n, u_t^n, \bar{L}_t^n)$  and  $y_t^{n+1}(V_t^n, \bar{v}_{t+\tau}^n, u_t^n, \bar{L}_t^n)$  by solving  $P_2$ .
  - Find  $V_{t+1}^{n+1}$  and  $\bar{L}_{t+1}^{n+1}$  given  $x_t^{n+1}$  and  $y_t^{n+1}$  using (19) and (20).

##### STEP 3 Computing differences:

- For  $t = T, T-1, \dots, 1$ :
  - Find  $X_t^{(n+1)+}$ ,  $Y_t^{(n+1)+}$  and  $X_t^{(n+1)-}$ ,  $Y_t^{(n+1)-}$ .
  - Calculate  $\hat{v}_t^{(n+1)+}$  and  $\hat{v}_t^{(n+1)-}$  and the smoothed differences  $\bar{v}_t^{(n+1)+}$  and  $\bar{v}_t^{(n+1)-}$ .

##### STEP 4 Global update:

- Update the control variables  $u_{ijt}^{n+1}$ .
- Check termination criteria; if not satisfied:
  - $n = n + 1$ , return to STEP2.

##### The forward pass

The forward pass takes as input the marginal vehicle values  $\bar{v}_{t+\tau}^n$  and bounds  $u_{ijt}^n$  on the empty moves to calculate the decision variables  $x_t^{n+1}(V_t^n, \bar{v}_{t+\tau}^n, u_t^n, \bar{L}_t^n)$  and  $y_t^{n+1}(V_t^n, \bar{v}_{t+\tau}^n, u_t^n, \bar{L}_t^n)$  by solving  $P_2$ . This is done by solving a simple sort for each terminal  $i$  and each time period  $t$ . We rank all the options of moving loaded or empty, we call this the ranking list. We rank the available transportation jobs  $l \in \bar{L}_{it}$

according to the values  $r_{it} + \bar{v}_{j,t+\tau_{ij}}^n$  and the possible empty moves according to the positive values  $-c_{ijt} + \bar{v}_{j,t+\tau_{ij}}^n$ , with a maximum of  $u_{ijt}^n$  terms. Given  $V_{it}$  units of capacity (AGVs), we assign the  $V_{it}$  highest values in the sorted list. In the case of a tie, i.e. two options with the same revenue (which was not discussed by Powell and Carvalho), we rank these options randomly. When the options are not ranked randomly a specific terminal might be favored, for example the terminal on top of the terminal list.

### *Computing the differences*

At the end of an iteration, marginal vehicle values  $\bar{v}_{t+\tau}^{n+1}$  for the next iteration have to be computed. Powell and Carvalho (1998a) present several methods for updating the marginal values. When no task gradient is present the marginal values can be updated by using  $\hat{v}_{t+\tau}^{n+1}$  computed in the current iteration and  $\bar{v}_{t+\tau}^n$  from the previous iteration. Another option is to use the new marginal values from iteration  $n+1$  to solve the local problem at node  $(i,t)$  again and compare it to the current solution at node  $(i,t)$ . This has to be done in a backward pass, starting at the end of the planning horizon and working backward. A backward calculation is required because the marginal vehicle values depend on future values at other terminals, but not on past values. The marginal vehicle values for  $t > T$  are all assumed to be equal to zero. We use the first method for updating the marginal values, because Powell and Carvalho (1998a) show that the results for this method are better than the results for the other approaches.

### *Upper bound adjustment*

During the global update one of the  $u_{ijt}$  has to be adjusted. The adjustment with the largest expected positive contribution to the objective function is executed. To keep the algorithm stable only one upper bound adjustment is performed per iteration. If several options have the same revenues, one of the options is selected randomly. When the selection is not done randomly one terminal might be favored above another.

### *Termination criteria*

As termination criteria we can use:

- the number of iterations. The algorithm terminates when the number of iterations has reached some pre-specified maximum value.
- the relative improvement in the objective function. The algorithm terminates when the improvement in the last iteration is smaller than some small value  $\epsilon$ .
- the total improvement of the objective function in  $n$  successive iterations. When the improvement in  $n$  successive iterations is less than some small  $\epsilon'$ , we stop.

The objective function does not necessarily converge to a specific solution (Carvalho and Powell, 1998). Because of this disadvantage, we chose the last option and we store the best intermediate solution found.

#### 4.6.5. Application of the LQN algorithm

Because we want to implement the LQN algorithm in a rolling horizon environment, we have to solve the problem repeatedly. When making a new plan we use updated information about estimated arrival times of AGVs at the terminal of destination, demands and inventories. The time between two consecutive planning moments is called the planning interval. This planning interval obviously should not exceed  $T$  and should also be less than the information horizon to avoid vehicles having to wait when jobs are available.

We need to define the revenue function  $r_{it}$  and the cost function  $c_{ijt}$  for our model. We have to take into account that all transportation jobs have to be handled and that the revenue of a job, if transported on time (before the latest departure time), is independent of the time at which it is handled. However, once the latest departure time (LDT) is passed, revenues should decrease. In order to ensure that all jobs are served (even when they are late), the revenues should be strictly positive for any realistic lateness. One logical function is represented in Figure 4.4. We will examine various shapes of the revenue function in Section 4.7.2.

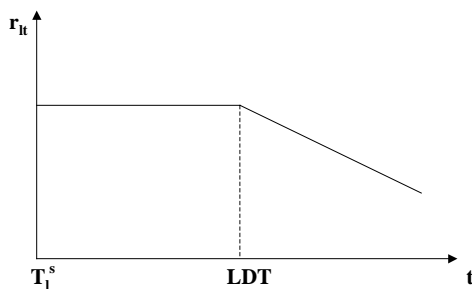


Figure 4.4. Revenue function for transportation job 1

A cost function for empty moves can be included in the model. It is natural to relate costs to distance and revenues to a particular job. Considering that lateness minimization is our main focus, we may as well set  $c_{ijt}=0$  for all  $i, j, t$ . If we want to minimize empty kilometers as a secondary goal, we can specify positive costs for empty moves. The travel time of an empty trip can be used as the cost for that particular trip, i.e.  $c_{ijt} = \tau_{ij}^e$ , but the relative costs of empty trips compared to the revenue of loaded trips might be important. The most appropriate cost function could be determined experimentally. Relating the empty travel costs to travel time might reduce the empty kilometers because an empty AGV will be sent from the nearest terminal with an AGV available. Nevertheless, such a cost function can make remote areas in the network less attractive. Furthermore, empty AGVs should implicitly already be drawn from the nearest terminals during peak periods, because the less empty kilometers, the more time for loaded kilometers and the more revenue can be generated. However, if there is a surplus of vehicles it can be expected that empty kilometers will not be minimized (cf. Section 4.7.3).

#### 4.6.6. Modifications of the LQN algorithm

Including a task gradient (Powell and Carvalho, 1998a)

The linear approximation adopted in the basic approach is a simplification. Instead of equation (11) we could use a better approximation which includes a task gradient.

$$\tilde{G}_t = \xi_t V_t + \mu_t L_t \quad (34)$$

where  $\mu_t$  is an estimate of the slope with respect to the number of transportation jobs waiting to be served, given by  $L_t$ . In planning transportation jobs, we should take into account the revenues that can be attained from the remaining jobs and the effects of a perturbation caused by moving task  $l$  from time period  $t'$  to time period  $t$  with  $t' > t$ . Powell and Carvalho (1998a) give an estimate of the task gradient, which depends on the supply of vehicles in future time periods, because the supply of vehicles determines when transportation jobs can be transported. An estimate of the impact of moving transportation job  $l$  from time  $t'$  to time  $t$ ,  $t' > t$ , is given by  $\tilde{\mu}_{l,t}$ .

$$\tilde{\mu}_{l,t+1} = \sum_{t' > t+1} \left( r_{l,t'} - v_{it'}^+ + v_{j,t'+\tau_{ij}^l}^- \right) x_{l,t'} \quad (35)$$

The perturbation results in a decrease in the objective function of  $\tilde{\mu}_{l,t}$ , i.e. the revenue of transporting load  $l$  at time  $t'$  plus the marginal value of an additional vehicle at terminal  $j$  at time  $t' + \tau_{ij}^l$  minus the marginal value of an additional vehicle at terminal  $i$  at time  $t'$ . Because transportation job  $l$  is not transported at time  $t'$ , the vehicle can be used for another transportation job. This provides a new approximation for  $v_{it'}^+$ , denoted by  $\tilde{v}_{it'}^+$ :

$$\tilde{v}_{it}^+ = \hat{v}_{it}^+ - \sum_{l \in L_{it}} X_{li}^+ \tilde{\mu}_{l,t+1} \quad (36)$$

The smoothed gradients now have to be determined in a backward pass and are computed as:

$$\bar{v}_{it}^{(n+1)} = \gamma \tilde{v}_{it}^{(n+1)} + (1 - \gamma) \bar{v}_{it}^{(n)} \quad (37)$$

*LAMA method (Carvalho and Powell, 1998)*

In another paper, Carvalho and Powell (1998) mention some drawbacks of the basic model from Powell and Carvalho (1998a). The main problem is the instability of the objective function; the value of the objective function usually fluctuates significantly between successive iterations and may even fail to converge. The smoothing procedure specified by equations (32) and (33) already reduces the effects of this instability. Another option is to adjust only one  $v_{it}$  or  $u_{ijt}$  per iteration, i.e. in all odd-numbered iterations one  $v_{it}$  is adjusted and in all even-numbered

iterations only one  $u_{ijt}$  is adjusted. The method is called the LAMA (linear approximation, multiplier adjustment) method. All adjustments are selected according to the addition to the objective function. The first 50 iterations use the basic approach to find a starting solution; afterwards the solution can be refined with the LAMA method. In this way, we can direct the solution towards the optimum and the changes from one iteration to the next are under better control although, as was mentioned before, adjusting only one  $v_{it}$  or  $u_{ijt}$  may already have a lot of impact in later time periods. Carvalho and Powell (1998) mention improvements of on average 0.7 percent by using the LAMA method.

#### **4.6.7. Drawbacks of the LQN algorithm with possible solutions**

During some preliminary experiments, we encountered several disadvantages of the LQN algorithm. Some of them were also indicated by Powell and Carvalho and some are due to our specific model variant. We propose modifications to the LQN algorithm to eliminate these disadvantages. Numerical experiments will evaluate the effects of these modifications (see Section 4.7.2).

##### *Taking the LDT into account*

One disadvantage of the LQN algorithm is that the latest departure time (LDT) is not taken into account when jobs are planned before their LDT, because the revenues are the same (see Figure 4.4). As a consequence, some jobs may be planned very close to their LDT whereas others with the same origin and destination have a lot of slack. In a dynamic environment, this could mean that a job with much slack has already been started by the next planning moment, whereas a job that is planned close to the LDT has not yet been started and thus is pushed behind in the plan by a rush order that has arrived in between. To overcome this, we can rank the transportation jobs according to their LDT. One possibility is to rank the transportation jobs according to the LDT in the forward pass. Another possibility is to rearrange the plan after every iteration, i.e. jobs with the same origin and destination are sorted on LDT.

##### *Dynamic planning horizon ( $T$ )*

Increasing the time horizon  $T$ , while keeping the length of a time interval constant, results in more time intervals, which increases the solution space. The same holds for decreasing the length of the time intervals at a constant  $T$ . The larger the solution space, the more difficult it is to find a good solution (cf. Powell and Carvalho, 1998a). A short planning horizon is sufficient in quiet periods, while a longer planning horizon might be required in peak periods. To take this into account we can introduce a dynamic planning horizon. The planning horizon starts, for example, at 30 minutes and is increased every  $n$  iterations until all transportation jobs have been planned. At the same time this approach might result in peak-shaving behavior, because as many transportation jobs as possible are planned in the first time intervals.

### *Using an initial solution*

The continuous inflow of new transportation jobs requires frequent re-planning. In such a dynamic context, we can use the results of the previous planning round. The idea is that the  $u_{ijt}$  and the  $v_{it}$  from the previous plan are a good starting solution for the new plan. Owing to this starting solution the computation times may decrease significantly. Using a starting solution might result in another outcome, because the value of the objective function usually fluctuates significantly between successive iterations and may not even converge, as stated before. At present it is not clear whether the final solution will be improved by using an initial solution.

### *Using expected handling times from the simulation*

Travel times and handling times are time-dependent, i.e. they depend on the workload in the system, which varies considerably during the day. Improvements might be obtained when the actual (estimated) handling times with the standard deviation are used in the planning process (see Chapter 2). More accurate information from the simulation is taken into account, possibly resulting in a plan that can be closely executed.

## **4.7 Numerical investigation**

The algorithms, as described in the previous sections, were implemented as separate building blocks in the simulation library. For empty vehicle manager EVM4, we have coded the serial scheduling method as an external routine in C++, which is periodically called by the eM-Plant simulation model. The LQN approach was coded as an external routine in Delphi and also periodically called by the eM-Plant simulation model. In the remainder of this section, we first present the experimental setting (Section 4.7.1) and then we discuss some initial experiments on planning parameters (Section 4.7.2). The numerical results of our simulation experiments are considered in Section 4.7.3.

### **4.7.1. Experimental setting**

For all our experiments, we used the Stommeer layout (Figure 3.1) and corresponding input figures as described in Chapter 3. The transportation network is further specified by the following assumptions:

- 1) Docks are dedicated to loading or unloading, and the functionality can be changed periodically without set-up time.
- 2) The time required for loading and unloading has a mean of 2 minutes and a standard deviation of 15 seconds. In the simulation experiments, these values were sampled from a gamma distribution.
- 3) Energy provisioning and equipment failures are not taken into account.

In this section, we focus on the key performance characteristic, namely the service level. Recall that the service level is defined as the fraction of orders that is delivered before the due time (cf. Chapter 3). All numerical results presented in this chapter are based on simulation runs of 30 consecutive days. The CPU requirements are in the range 1-2 hours on a Pentium II-266 MHz PC with 128 MB RAM.

As there are very few transportation jobs to be served around midnight, the statistics gathered per day are treated as independent observations. Formally the observations are dependent, but our experiments showed that the correlation between successive days is negligible. Therefore, we can give an indication of the accuracy of the results corresponding to a run length of 30 days. We found that the confidence intervals of the service levels were quite small in the range that is of interest to us, a service level of about 99%. Then the 0.95 confidence interval length for the service level is in the range 0.1–0.25%. When the service levels are smaller than 98%, the confidence intervals become larger, even larger than 1%, but the required system performance should be around 99%.

In the numerical study we examined the following effects:

- 1) The performance of the five options, EVM1-EVM5.
- 2) The impact of the period for which transportation jobs are known in advance (value of information).

Prior to the simulation study, a number of experiments was performed in order to determine the following model parameters:

- a) Resource capacities
- b) The choice of the planning frequency of task allocation to docks
- c) Parameter settings for EVM4
- d) Parameter settings for EVM5
- e) The choice of the planning frequency when empty vehicle management is coordinated at network level (EVM3, EVM4 and EVM5)

The results are presented in the following section.

#### **4.7.2. Initial experiments**

##### *Resource capacities*

Before examining the performance of the various logistic control procedures, we studied the key resource capacities. Our analysis revealed that 2 docks at each AAS terminal, 5 docks at each VBA terminal and 10 docks at the RTH terminal are sufficient to handle all transportation jobs. The system should contain at least 185 AGVs to handle all transportation jobs in peak hours on Tuesdays. In this situation, vehicle availability is the bottleneck rather than dock capacity. For Mondays, 120 AGVs appeared to be sufficient. Because the transportation flows are better balanced, less empty vehicle movements are needed and hence vehicles are better utilized. For Fridays, transportation flows are quite imbalanced, yielding a number

of 165 AGVs to handle the peak demand. We used these figures for our simulation experiments.

*Planning frequency of task allocation to docks*

For the period during which the task allocation is fixed, we found that the effect of fixing the tasks for some period of 15-30 minutes has only a slight impact on system performance, see Table 4.2. Once the period increases to one hour or more, the performance may decrease significantly. As a longer period has technical and organizational advantages, we chose 30 minutes for the rest of our numerical experiments.

**Table 4.2.** *Effect of fixed dock task period (applied to the network layout in Figure 3.1 and transportation flows from Table 3.1 for a Tuesday with 185 AGVs, the EVM3 control rule planning every 30 minutes with orders being known 30 minutes in advance)*

Fixed dock task period	Service level						Overall
	AAS-RTH	AAS-VBA	RTH-AAS	RTH-VBA	VBA-AAS	VBA-RTH	
15 min.	99.7	99.7	99.6	100.0	98.9	99.0	99.3
30 min.	99.3	99.3	99.5	100.0	97.8	98.8	99.1
60 min.	98.1	98.9	98.4	100.0	96.5	96.9	97.7

*Parameter settings for EVM4*

As mentioned in Section 4.5, the empty vehicle manager EVM4 periodically performs an integrated planning run. We may call the serial scheduling method once for each run, but we can also call it several times for different parameter settings. The best schedule that has been obtained is then selected. Possible parameters to be varied are the maximum delay and the minimum slack, the control parameters for combined transportation. We may also use other priority rules instead of the latest departure time. However, using the latest departure time as priority rule allows EVM4 to be evaluated against other empty vehicle managers. We found that EVM4 generally requires about half a second of CPU time per schedule for our application. Because we needed lengthy simulation runs to compare the various empty vehicle managers, to keep the run time acceptable we evaluated only four schedules per integrated planning run. Preliminary analysis shows that the schedules are more sensitive to the minimum slack than to the maximum delay. Therefore, we decided to use only one value for the maximum delay, namely 8 minutes. Since the sum of the expected loading and unloading time is 4 minutes, a maximum delay of 8 minutes allows a wait for an empty vehicle or for the release time of a job of at most 4 minutes. A small minimum slack generally increases the number of combined transportation jobs and reduces empty vehicle travel. However, this may lead to a greater maximum lateness. We found that a minimum slack of between 5 and 20 minutes yields good results. A minimum slack less than 5 minutes reduces the on-line service level significantly because too many jobs are scheduled close to their latest departure time. On the other hand, a minimum slack longer than 20 minutes



usually did not result in significantly different schedules. From the four schedules generated, we selected the schedule with the lowest maximum lateness. In the case of ties, the schedule with the shortest total travel time for empty vehicles was selected.

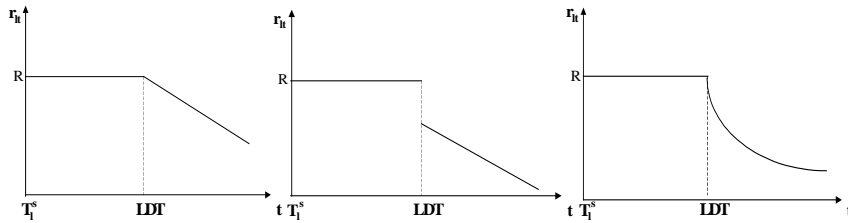
#### *Parameter settings for EVM5*

We investigated the sensitivity of several input parameters of EVM5. These parameters are  $\gamma$ ,  $T$ , the length of one time period in the discrete time model and the maximum number of iterations. The results were rather insensitive to these parameters. Therefore, we choose the following values:  $\gamma=0.2$ , the length of one time period is 4 minutes,  $T=160$  minutes (40 time periods), and the maximum number of iterations is 1000. The revenue of a transportation job is 1000 if it is transported on time ( $t \leq \text{LDT}$ ) and  $1000-t+\text{LDT}$  if it is transported late ( $t > \text{LDT}$ ), see the left function in Figure 4.5. Notice that we stopped the algorithm when no improvements in the objective function were observed in the last 50 iterations (Section 4.6.4). In the basic model of EVM5, we stored the best solution and afterwards we rearranged this solution with respect to the latest departure times (cf. Section 4.6.7). The expected handling times from the simulation were used, together with the standard deviation to compute the latest departure time (see Section 4.6.7). After determining the input parameters, we investigated the effect of extensions of the basic model, as described in Sections 4.6.6 and 4.6.7. In Table 4.3 we show the results for the basic model and the several extensions. We considered the use of the results of the previous planning operation,  $u_{ijt}^{n-1}$  or  $v_{it}^{n-1}$ , the use of the task gradient and the LAMA method.

**Table 4.3.** *Service level for the basic model and several extensions*

<b>Day</b>	<b>Basic</b>	$u_{ijt}^{n-1}$	$v_{it}^{n-1}$	<b>task gradient</b>	<b>LAMA</b>	<b><math>r_{it}</math>-variant</b>
Monday	99.0	98.8	98.8	99.3	98.6	99.5
Tuesday	96.8	97.4	96.8	97.6	96.9	97.7
Friday	97.6	97.8	97.3	98.4	97.6	98.8
Fictitious	94.8	94.0	94.6	96.0	93.2	97.8

Besides the model extensions, we tested several revenue functions, see Figure 4.5. The left revenue function, with  $R=1000$ , was used in the basic model. The results were not very sensitive to the value of  $R$ , we tried  $R=500$  and  $R=2000$ , but the results were not significantly better or worse than the results for  $R=1000$ . Furthermore, we tried two alternative revenue functions, both with the idea that when the job is late, it should be heavily penalized, but the lateness is less important.



**Figure 4.5.** *Tested revenue functions*

The other revenue functions are:

$$r_{it} = 1000 - \max\{500(t - LDT), 0\} \text{ and } r_{it} = 1000 - 100 * \max\{\sqrt{t - LDT}, 0\}$$

The last function did not show significant improvements in the results, but the function in the middle did. The results for the function in the middle of Figure 4.5 are shown in the last column of Table 4.3 (task gradient included). This revenue function gives slightly better results for Mondays and Tuesdays, but significantly better results for the fictitious day.

Based on the results for these possible extensions, we chose to use the basic model with the task gradient and the revenue in function in the middle of Figure 4.5 for further experiments with EVM5. The claim made by Powell and Carvalho (1998a) holds: the solution is within a few percent of the optimum solution. An upper bound on the optimum solution is when all jobs are delivered on time. The computation times of the LQN algorithm are in the range of 1-15 seconds. Using the results of the previous planning procedure can significantly decrease the computation times to 1-2 seconds. A dynamic planning horizon can decrease the computation times, but did not improve the results.

#### *Planning frequency for coordinated empty vehicle management*

For coordinated empty vehicle management (EVM3, EVM4 and EVM5), the planning frequency has to be determined. As the system situation changes quickly, infrequent rescheduling may decrease the system performance. Table 4.4 shows some corresponding simulation results when transportation jobs are known 30 minutes before cargo arrival at the terminal of departure. Note that a prerequisite for proper functioning of EVM4 and EVM5 is that the planning period should be less than or equal to the time for which transportation jobs are known in advance. For example, if the planning period is 40 minutes and transportation jobs are known 30 minutes in advance, the jobs arriving after 30 minutes will not be scheduled until the next EVM planning. Hence, these loads will wait until the next schedule is generated, even if sufficient vehicle and dock capacity is available. Obviously, this is not a practical option. Therefore, Table 4.4 only shows the performance of EVM4 and EVM5 if the planning frequency is at least once per half-hour.

**Table 4.4.** *Effect of empty vehicle planning frequency on service level (applied to the network layout in Figure 3.1 and transportation flows from Table 3.1, runs for a Tuesday with 185 AGVs and orders being known 30 minutes in advance)*

EVM period	EVM3	EVM4	EVM5
10 min.	99.1	98.9	97.7
20 min.	98.4	98.4	97.2
30 min.	98.2	96.8	95.4
40 min.	97.0	X	X
50 min.	94.7	X	X
60 min.	89.8	X	X

Surprisingly, Table 4.4 shows that the service level is not extremely sensitive to the empty vehicle manager planning frequency. As long as the frequency is at least once every half-hour, the service level remains high. We chose to reschedule every 10 minutes, yielding high customer service.

### 4.7.3. Numerical results

*Comparison of empty vehicle management strategies: the value of coordination*

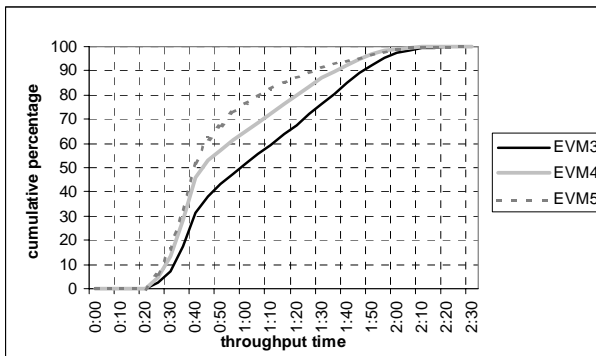
An important research question relates to the effectiveness of the various options for empty vehicle management. We tested these options for the three busiest weekdays as mentioned in Chapter 3. The overall service levels are shown in Table 4.5 for all cases.

**Table 4.5.** *Effect of empty vehicle management (applied to the network layout in Figure 3.1 and transportation flows from Table 3.1, orders being known 30 minutes in advance)*

Day	AGVs	Overall service level				
		EVM1	EVM2	EVM3	EVM4	EVM5
Monday	120	94.7	97.1	99.8	100	99.5
Tuesday	185	80.3	89.9	99.1	98.9	97.7
Friday	165	85.5	93.2	99.7	99.6	98.8

We see that planning coordination (included in EVM3, EVM4 and EVM5) has a significant impact on the customer service. It is remarkable that the effects are considerably smaller for Mondays than for the other two days. A possible explanation for this phenomenon is the fact that transportation flows are more balanced on Mondays. To give an indication, on Mondays the flows from VBA to RTH are similar to, or at most twice the return flows, whereas this ratio is 5-10 on the other days. Hence, vehicles tend to be positioned at the right location without much coordination on Mondays. Proper empty vehicle management increases the system performance considerably, especially when transportation flows are heavily imbalanced (Friday and especially on Tuesday),

For the cases mentioned in Table 4.5, EVM3 and EVM4 have more-or-less the same overall service level, while EVM5 shows slightly worse results. To ascertain whether this is a general result, we constructed an additional artificial case. This case was based on the same network, but with a different transportation pattern. To test the empty vehicle managers under difficult conditions, we defined transportation patterns with peak levels moving quickly from one route to the other. This means that to attain high customer service levels it is essential to anticipate heavily fluctuating transportation demands. For this case, we found that EVM4 performs significantly better than EVM3 (service level 98.5% versus 95%), while EVM5 also performs rather well with a service level of 97.8%. In Figure 4.6, EVM5 has the best throughput time distribution, but there are several jobs that have a throughput time longer than 2 hours. The throughput time distribution under EVM4 is also significantly better than that of EVM3. We conclude that EVM4 is preferable in the sense that its performance is at least equal to that of EVM3, and under difficult conditions considerably better than that of EVM3. EVM4 also outperforms EVM5 with regard to service level. Moreover, as stated in Section 4.7.2, we could run EVM4 with more parameter settings or priority rules to improve its performance.



**Figure 4.6.** Throughput time distribution for EVM3, EVM4 and EVM5 in the case of fictitious transportation flows with quickly moving peak levels between routes

For the three best empty vehicle managers, EVM3, EVM4 and EVM5, we compared the average empty travel distance per day. We found that EVM4 requires 3-10% less empty travel than EVM3, and 15-30% less empty travel than EVM5, depending on the day. In case of sufficient AGV capacity, EVM5 tends to send more empty AGVs to a terminal than is strictly necessary. Hence, EVM4 is also more efficient from an environmental point of view. Furthermore, empty travel reduction is relevant when AGVs use batteries for energy provisioning, because the recharging frequency is reduced.

*Impact of the information horizon: the value of information*

As mentioned before, pre-information about transportation jobs may facilitate planning and improve the system performance. However, this requires additional communication and hence more sophisticated organization and information systems. We analyze the effect of the period for which transportation jobs are known in advance. As pre-information is only used for the empty vehicle manager options 2, 3, 4 and 5, we omit option 1 (FCFS myopic). EVM2 (FCFS-look ahead) is identical to EVM1 if pre-information is not available (horizon = 0). For EVM5 with an information horizon of 60 minutes we use a planning horizon ( $T$ ) of 220 minutes, because more transportation jobs have to be planned. The main results are shown in Table 4.6.

**Table 4.6.** *Effect of the period for which transportation jobs are known in advance.*

Day	AGVs	Orders known	Overall service level			
			EVM2	EVM3	EVM4	EVM5
Monday	120	0 min.	94.7	99.4	99.0	98.1
		30 min.	97.1	99.8	100	99.5
		60 min.	90.3	99.9	100	99.3
Tuesday	185	0 min.	80.3	84.7	92.4	91.5
		30 min.	89.9	99.1	98.9	97.7
		60 min.	93.0	98.5	99.2	98.0
Friday	165	0 min.	85.5	90.0	94.8	93.0
		30 min.	93.2	99.7	99.6	98.8
		60 min.	95.6	99.9	99.6	99.1

We notice a similar phenomenon to that in the previous section, namely that the impact of pre-information is highest if the transportation flows are heavily imbalanced (Tuesday and Friday). Pre-information appears to have an even stronger effect on customer service than coordination. Of course, proper coordination is facilitated by the availability of sufficient information. The marginal value of additional pre-information clearly decreases.

It is remarkable that for EVM2 the overall fill rate on Monday, with transportation jobs known 60 minutes in advance, is even considerably less than the rate when only 30 minutes pre-information is available. This phenomenon can be explained as follows. Vehicles are dispatched to the terminals FCFS once the transportation jobs are known. If transportation jobs are known 60 minutes in advance, AGVs arrive too early at the terminal and have to wait for a long time before they can be loaded, causing a significant capacity loss in a bottleneck resource (AGVs). In the OLS 30 minutes to one hour of pre-information will be sufficient for practical purposes, because of the maximum travel times between the various locations in the network.

## 4.8 Conclusions

In this chapter, we developed several options for vehicle management in automated transportation networks. The various methods have been implemented and tested in a simulation environment for the OLS-case. We found that both information about future orders and planning coordination between terminals gives considerable advantage in terms of customer service levels. In many cases a relatively simple method to balance empty vehicle flows in the system (EVM3), taking into account some future order information and communication with local (terminal) levels, already appears to provide considerable benefits. The advantage of EVM3 is the relatively low level of complexity and information exchange between local (terminal) and central (network) level.

A more advanced serial scheduling method (EVM4) can improve the performance further, especially in difficult cases when peak demand quickly moves from one route to another. In this case it is worthwhile to plan a sequence of orders in an integrated way for a longer time horizon. Integrated planning also has the advantage of reducing empty travel. Moreover, the serial scheduling method offers a broad range of possibilities for further refinements. In this respect, inclusion of finite terminal capacities, both docking capacities and parking space, is an interesting subject for further research (not in this thesis).

Another approach based on logistics queuing networks shows slightly worse results than EVM3 and EVM4, but an advantage of the LQN approach of EVM5 is the flexibility. Due to the time-space decomposition, complicating constraints can be added without a lot of extra computation time. Order acceptance could easily be implemented, meaning that not all transportation jobs have to be transported. Nevertheless, we also encountered several disadvantages of the LQN algorithm. The choice of discrete time intervals may have some negative effects in the case of long time intervals. The use of a linear approximation leads to an unstable objective function, which does not converge to an optimal solution, as also noticed by Carvalho and Powell (1998). The LQN algorithm shows results that are within a few percent of the optimal solution, as was also claimed by Powell and Carvalho (1998a). However, we note that the same applies to the much simpler coordinated look-ahead rule (EVM3).

An advantage of the coordinated look-ahead rule (EVM3) is that because of the local control concept it can easily be extended to include complications such as two-way tracks. Two-way tracks are more difficult to incorporate in an integrated planning approach, such as EVM4 or EVM5. For this reason, and because in the practical cases with a positive information horizon the performance of EVM3 is comparable to the performance of EVM4, we will use EVM3 in the remainder of this thesis.



# Chapter 5

## Control of an infrastructure bottleneck: two-way track control<sup>3</sup>

### 5.1 Introduction

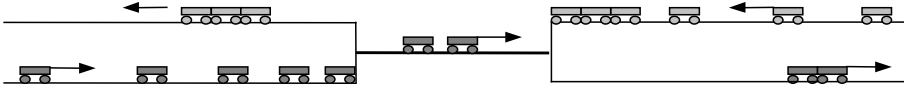
The realization of an underground transportation system, such as the OLS, requires high investments in both infrastructure and control systems. Dominant cost factors are the total length of the tubes and their diameter. To ensure that the project is feasible from an economic perspective, configurations are studied in which terminals are connected by a single tube, allowing for traffic flows from one end at a time (cf. Figure 3.2). However, such options for the system layout may have a severe impact on the logistic performance, because AGVs are often forced to wait at the tube entrances. Consequently, in order to guarantee acceptable throughput times there is a need for intelligent control of the driving direction.

In this chapter, we address a track/tube allowing for alternating traffic as a *two-way track/tube*. We describe the two-way track as a part of an underground automated transportation network, although the same setting applies to systems like bridges, tunnels or roadblocks caused by traffic accidents or road maintenance. As a starting point for our discussion we use Figure 5.1.

---

<sup>3</sup> This chapter is partly based on Van der Heijden et al. (2001b).





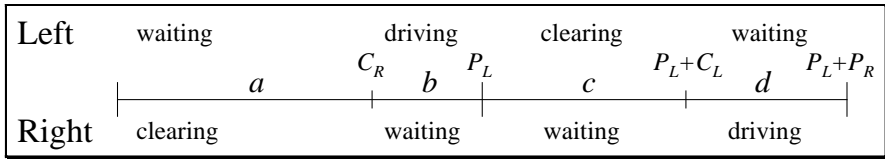
**Figure 5.1.** Two-way track

The central part of Figure 5.1 is the two-way track, used for traffic in both directions. The AGVs arrive and depart on tracks that only allow one-way traffic. According to a control rule, AGVs arriving at the tube may either enter the tube immediately or queue until a signal is given that access to the single tube is allowed. Each control rule generates the system states, shown in Table 5.1, in cyclic order.

**Table 5.1.** Description of the system states

State	Description
a	AGVs from the right have to wait, but AGVs from the left are not yet able to enter the track, because it still contains AGVs driving from right to left; we use the term <i>clearing the tube</i> for this system state
b	AGVs from the left can enter and AGVs from the right have to wait
c	AGVs from the left are stopped and AGVs from the right wait until the tube is cleared
d	AGVs from the right can enter and AGVs from the left have to wait

The time spent in the states *b* and *d* is called the *green time* (since there is a green “traffic light” at the right and the left, respectively) and the time spent in states *a* and *c* is called the *clearance time*. The clearance time refers to the period with a red traffic light at both ends, while vehicles still have to leave the tube. The cumulative time spent in the four system states makes up the *cycle time*. The time spent in each system state is variable and depends on the driving time on the two-way track, the arrival distributions at both sides and the control rule.



**Figure 5.2.** System states under a periodic control rule

A main assumption in our modeling is that at each moment *t* it is known how many vehicles are present on a track. Technologically, this can easily be accomplished, for example by using induction wiring at both ends of the two-way track. Let us denote the number of vehicles on the track driving from right to left and from left to right, as *R(t)* and *L(t)* respectively. The operation of the two-way track will always be such that the collision avoidance condition  $R(t)L(t)=0$  holds true. Updating the counter *R(t)* is simple: in the case of an entrance at the right end of the two-way track we

have  $R(t+h)=R(t)+1$ , in the case of a departure at the left end we have  $R(t+h)=R(t)-1$ . For  $L(t)$  analogous updating rules with right and left interchanged hold true. In these rules  $h$  denotes a small time interval associated with electronic data transmission. In addition, we assume that an end-of-clearance signal is available. The starts of the red light intervals can be considered as decision variables in the operational control of the two-way track. As we will see shortly, there are several ways to set a decision rule for these decision variables:  $P_L$  and  $P_R$ .

Our study of the OLS raised the question of which rule should be used to control access to the two-way track in order to minimize delay. Although central scheduling of all activities, including the two-way track, is theoretically possible for the OLS system, we chose to focus on local control rules fitting into the general framework as discussed in Chapter 2. This control rule for the *isolated* subsystem may take into account information on the environment, such as expected AGV arrival times.

A straightforward well-known control rule in production management, which uses information on arrival times of vehicles, is First-Come First-Served (FCFS). However, this rule will only lead to a reasonable performance in the case of very low traffic intensity and relatively short driving times along a two-way section (for example crossroads). Therefore it is not applicable to the OLS, where a high traffic intensity and long tube driving times are envisaged (5-10 minutes). Another obvious option is switching the driving direction periodically:  $P_L$  and  $P_R$  are constants. That is, the system spends a fixed time in both the system states  $a&b$  ( $P_L$ ) and in the states  $c&d$  ( $P_R$ ), with  $P > T$ , where  $T$  is a measure of the driving time on the two-way track. So the *green time* depends on the preceding clearance time. An alternative simple control rule could be to fix the *green time* instead of the cycle time, i.e. spend a fixed time in state  $b$  and  $d$ . In this chapter we focus on the first possibility. An obvious drawback of these simple rules is that they do not use information on AGV arrivals and queues at both two-way track entrances. Hence, it is possible that AGVs have to wait at one entrance, while the two-way track is free and no AGVs are present or approaching the other entrance. In particular when the traffic intensities at both sides of the two-way track are different and/or fluctuate in time, a simple periodic control rule may lead to excessive waiting times. Therefore, we also developed adaptive control rules with increasing complexity and information usage. Information on future arrivals at the two-way track can be obtained, for example from induction wiring at a certain distance in front of the entrances of the two-way track. One can install counters for the numbers in the queue at both ends of the two-way track. We focus on control rules that are suitable for on-line control: they have to be efficient in terms of computation time. The potential of the new control rules is demonstrated by an extensive simulation study based on independent Poisson arrivals of AGVs, as is plausible in an open system. We will also discuss the implications of embedding the two-way track in the OLS. More specifically, we will consider the effects of the convoys created at a two-way track on both terminal operations and on other two-way tracks in the system.

This chapter is structured as follows. In the next section we give an overview of the literature, in which we relate the system studied to similar traffic, queuing and production systems. In Section 5.3, our model is defined in more detail. In Section 5.4 we describe the periodic control rule, followed by the adaptive control rules in Section 5.5. The design of the simulation study is described in Section 5.6. In Section 5.7 we present the results of a simulation study of an isolated two-way track, while we focus on the implications of a two-way track in a closed system in Section 5.8. Finally, in Section 5.9, we summarize the main conclusions.

## **5.2 Literature review**

The problem of controlling a two-way track appears to be at the intersection of a number of research fields. While the link with the traffic literature, as described in Section 5.2.1, is intrinsically clear, other research is also relevant. In addition to those mentioned in the traffic literature, we can find similar models in the queuing literature, as described in Section 5.2.2. Furthermore, one can formulate the problem as a machine-scheduling problem, in which a planner has to decide when the next batch of products (product 1 is an AGV from the right, product 2 is an AGV from the left) should be processed (Section 5.2.3). Below we relate the two-way track control problem to these research fields.

### **5.2.1. Traffic literature**

Linking two-way track control to what is mentioned in the dynamic traffic control literature seems natural as the system is intended to be an integral part of traffic systems. Haight (1963) generalizes the problem of controlling two-way tracks and similar systems as the control of a road section that allows one-way traffic only. Given the arrival distributions and the allowed driving directions, either left-right or right-left, formulas are derived which specify queuing behavior. Two-way track control is also related to junction control supported by traffic signs. In the early days traffic signals were usually scheduled according to a pre-determined scheme (cf. Bell, 1992). Such a scheme allowed traffic to be handled following a fixed sequence. A periodic control rule is a good example of such a scheme. Mung, Poon and Lam (1996) elaborated on Haight's model and derive distributions of queue lengths at fixed time traffic signals. Heidemann (1994) derives analytical results on statistical distributions of queue lengths and delays at traffic signals, given Poisson arrivals and a fixed-time control. The results are compared with several approximations (Webster, 1958; Miller, 1968). Hu et al. (1997) extend Heidemann's model to the multi-lane case, where multiple vehicles may enter the traffic intersection simultaneously. Nowadays, traffic signals are usually controlled by adaptive rules, which take the actual traffic situation into account. Usually traffic control at intersections is *vehicle* dependent: the arrival of one vehicle can switch the traffic light to green if there is no other demand for the resource. Increasingly, these

control rules become *traffic* dependent. A common objective is to minimize the average loss times, i.e. minimizing average waiting times. For example, Robertson and Bretherton (1974) describe an optimal control policy for an intersection for any known sequence of vehicle arrivals. They compare the optimum policy with a fixed time control policy and responsive control policies, a “saturation flow” and “no-flow” policy. The responsive control policies cannot be used for a two-way track due to the long clearance time of the two-way track.

In the traffic literature, it is usually assumed that the effective green period and the effective red period are *constant* and known. If this assumption is valid, we could simply use the exact method proposed by Heidemann (1994). As appears from our problem statement in the introduction, this is not the case in our situation. The clearance time is a random variable and depends on the last moment at which a vehicle enters the two-way track before the traffic light is switched to red. The use of fixed effective green times would be reasonable if the passing time of the lane was negligible. It is also a reasonable assumption if the system runs close to maximum capacity, so that the vehicle queue has usually not yet disappeared or has just disappeared when the traffic light is switched to red. In the latter case, the clearance time equals the deterministic driving time along the shared lane, so the effective green time is fixed too. Some preliminary numerical experiments applying Heidemann’s (1994) method to our modified model, revealed that in this way the waiting time can be overestimated by 5-10% over the entire parameter range. This is a result of the fact that the clearance time is less than or equal to the driving time along the lane, causing a longer effective green time. Therefore, we conclude that it is useful to construct a dedicated method to calculate mean waiting times.

An essential difference between an automated transportation system and traffic systems, as discussed here, is that the behavior of AGVs can be directly influenced. Human drivers feel a need for a “fair” handling of traffic and therefore restrictions are set to maximum cycle times, which is roughly the maximum waiting time for a car. Moreover, quite precise information is available on expected arrival times of AGVs at the two-way track entrance, because the driving behavior of AGVs is far more predictable than that of passenger cars.

### **5.2.2. Queuing models**

As well as in the traffic literature, we can also find similar models in the queuing literature. To be specific, our model has similarities with polling systems (cf. Takagi, 1990), in which a single server handles two queues and switches between them according to some control rule. For our model, the service time should be deterministic and equal to the minimum distance between successive vehicles, expressed in time. Furthermore, the server attends both queues for a fixed period of time. This aspect is not common in polling models, but can be found in STDM (Service Time Division Multiplexing, cf. Kleinrock, 1976). However, STDM models do not include switchover times. In our setting, the clearance time could be

modeled as a random set-up time, depending on the timing of the last service at the other queue. This aspect is not common in queuing systems.

Related to polling systems are queuing models with vacations (Doshi, 1990; Takagi, 1991). The server leaves the queue regularly to perform other tasks. In our case, the vacation period can be modeled as some random variable that depends on the switching interval  $P$ , the driving time along the lane,  $T$ , and the vehicle arrival process. We would still need a decomposition between the queues at both sides of the shared lane, whereas these queuing processes are actually interrelated. In Doshi's (1990) classification, our model is closest to the category with asynchronous vacations and general vacation rules. We did not find a model similar to ours in the literature on vacation queues.

### **5.2.3. Machine scheduling**

In the machine scheduling literature, problems rather similar to the two-way track problem are studied. Several batching strategies are developed within the context of deterministic scheduling. For overviews see e.g. Uzsoy et al. (1992, 1994), Webster and Baker (1995), and Potts (2000). An alternative approach is taken by another group of authors (Glasse and Weng, 1991; Fowler et al., 1992; Weng and Leachman, 1993; Van der Zee et al., 1997), who study the batching problem in a dynamic context, where jobs have to be scheduled realtime given the availability of information on some or all near future arrivals. Many of these studies focus on the control of oven systems found in semiconductor manufacturing and aircraft industry. Similarities between the two-way track system and such systems can be found in the fact that goods (AGVs) have to be "batched", with the processing time of a batch equal to the driving time on the two-way track, and the dynamic nature of the problem. An essential difference can be found in the fact that for the two-way track system there is no fixed processing time. While for oven systems processing times are related to static product and process characteristics, the processing time of a convoy on a two-way track depends on the time between the first and the last AGV in a convoy. Furthermore, oven systems set restrictions on the number of products in a batch, but there is no a priori limit on the number of AGVs in a convoy.

Summarizing, the most important differences compared to the literature are:

- (a) Significant throughput time on the two-way track (several minutes).
- (b) The clearance time is variable, depending on the time at which the last AGV entered the two-way track.
- (c) No maximum batch size.

## 5.3 Control rule concepts

### 5.3.1. Assumptions and notation

For reasons of clarity of understanding, and to keep the formulas simple, we make the following simplifying assumptions for the system characteristics:

- 1) All AGVs are identical.
- 2) When driving, the AGVs travel at a constant speed.
- 3) Queuing AGVs accelerate instantaneously to their normal speed, when activated, and AGVs can stop instantaneously when arriving at a queue.
- 4) AGVs waiting in queue enter the shared lane with a fixed minimum succession time ( $\delta$ ), that remains constant while driving on the two-way track.
- 5) The length of an AGV is negligible (can be included in  $\delta$ ).

In practice, the succession time between vehicles will be strictly positive. An obvious reason is the length of a vehicle; two vehicles cannot enter a single lane simultaneously. Also, safety margins can be included to take into account a finite deceleration. These safety margins should be set such, that the collision probability is negligible. We use the *brick wall principle* (cf. Chapter 3, Section 3.3.3). This principle states that when travelling with the minimum succession interval a vehicle should be able to stop in time if the preceding vehicle comes to a dead stop. Safety margins for finite acceleration can be incorporated in the minimum distance. The effect of stochasticity in the AGV speed (for example load dependent) can be put into a safety margin by working with a slightly lower constant speed than the maximum speed. In the OLS-case, for example, the maximum speed of an AGV is 7 m/s (cf. Chapter 3), but we use an operational speed of 6 m/s. Note that we do not make an explicit assumption about the arrival processes as far as lot-size is concerned. Both single and batch arrivals are considered. In the remainder of this section we characterize the rules for real-time two-way track control in terms of information usage and decision structure. For a detailed description of the control rules we refer to Sections 5.4 and 5.5.

### 5.3.2. Characterization of control rules

If information is available on all AGV arrivals it is possible to construct an optimal schedule (cf. the concept of deterministic machine scheduling). In a dynamic situation however, information changes as a result of additional AGV arrivals or better forecasts of arrival times. We consider the case in which arrival information is known for arrivals up to  $t_0+H$ , with  $t_0$  a decision moment and  $H$  the information horizon. Arrivals after time  $t_0+H$  are assumed to be unknown. Hence, as time proceeds, new information may become available. This leads to a rolling horizon approach. In consequence, rescheduling is necessary. This implies that the rescheduling frequency appears as an additional parameter.

We consider three situations as far as information on AGV arrivals is concerned:

- (a) No information on AGV queues and arrivals.
- (b) Only local information on queue lengths at the tube entrances is available ( $H=0$ ).
- (c) Prior information, i.e. both on current queue lengths and on future arrivals within a certain information horizon  $H > 0$  is available.

Now we can classify the control rules according to the available information and decision options, see Table 5.2.

**Table 5.2.** *Classification of the control rules*

	<b>no information</b>	<b>local information</b>	<b>prior information</b>
<b>one change</b>	periodic control	adaptive local control	adaptive look-ahead control
<b>multiple changes</b>		dynamic programming	

If no information on the AGVs waiting or arriving is available, a *periodic control* rule is the only realistic option. In that case, the only issue is to determine the fixed time interval  $P$ , as introduced in the previous section. The interval chosen is based on off-line optimization, considering the historical and/or expected traffic flows, as in off-line junction control (Bell, 1992). The periodic control rule will be discussed in Section 5.4. The advantage of a periodic control rule lies in its simplicity; it requires no information on system status.

In the other cases a decision policy has to be constructed. A policy has to define the set of decision moments  $t_0$ , and for each decision moment the allowed decisions. Policies open to the controller are either to take a single decision on the timing of the next direction change, or to make a schedule of multiple direction changes within the information horizon  $H$ , i.e. all known AGV arrivals (jobs) are included in the schedule. We restrict the decision moments to the following list of discrete events:

- (I) the end of state  $a$ , where the two-way track is cleared for traffic from the left;
- (II) the arrival of a new AGV at the left or the right end of the two-way track in state  $b$ , with a green traffic light at the left entrance;
- (III) the end of state  $c$ , where the two-way track is cleared for traffic from the right;
- (IV) the arrival of a new AGV at the left or the right end of the two-way track in state  $d$ , with a green traffic light at the right entrance.

At each decision moment the allowed decisions are: “let the green traffic light remain green until the next decision moment” or “put the green light on red so that a change of direction is enforced immediately after this decision moment”. Sometimes

the choice of a next decision moment is modified in the sense that it is the first event of type (I-IV) after a lower bound in time after the previous decision moment. In this way, one can enforce that a certain batch of vehicles enters the tube. Usually, it only makes sense to change the two-way track direction directly after an AGV has entered the two-way track. The additional decision moments are sketched above. Too frequent rescheduling may result in excessive computation times.

Information on the AGVs in queues and/or information on arrivals can be used to improve system performance by developing control rules that use this information. These are *adaptive control* rules (see Section 5.5). The construction of such an adaptive control rule implies finding an optimization criterion to determine the direction changes. When only local information on queue lengths is available, and the direction is changed to right-left, then obviously the whole queue at the right first passes the tube. It can easily be shown that any other option, for example processing the queue at the right in two separate batches, is inferior in terms of total waiting time for the vehicles in queue because of the additional clearance time in case of an additional direction change. For this reason, a *dynamic programming* approach that considers multiple changeovers naturally leads to batch processing; the entire queue present at the decision moment enters the two-way track.

We consider two decision options based on information on future arrivals (see Table 5.2). First, we develop a look-ahead criterion for determining the next direction change (Section 5.5.2). Second, a dynamic programming rule is introduced (Section 5.5.3), which computes an optimal sequence of direction changes. In a rolling horizon approach only the first change in this sequence is used, but the computations take future direction changes into account.

The decision about changing the direction should be based on some optimization criterion. As our focus in the OLS is to minimize average order throughput times, we chose minimum average waiting time at the two-way track as criterion. The waiting time is defined as the time from the arrival of an AGV at the entrance of the two-way track until it enters the two-way track. The basic ideas presented in this chapter can also be applied to other criteria, such as minimum lateness.

## **5.4 Periodic control**

### ***5.4.1. Introduction and assumptions***

In Section 5.2 we have seen that similar models to the one under consideration have been analyzed, but also that as far as we know the combination of a fixed (periodic) service schedule, random clearance times and correlated queuing processes is new.



For the periodic control rule we proceed from Figure 5.2 in Section 5.1. The four system states constitute a cycle. The time spent in system state  $b$  ( $d$ ) is called the *effective green time* from the left (right). We refer to the time spent in system state  $a$  ( $c$ ) as the *clearance time*, denoted by  $C_L$  ( $C_R$ ). Obviously, the clearance time is variable and depends on the moment the last vehicle enters the lane before the traffic light is switched to red. In all cases, the clearance time varies between 0 and the driving time on the two-way track, denoted by  $T$ .

We define a *periodic* control rule such that the time spent in system states  $a$  plus  $b$  is constant and equal to  $P_L$ , see Figure 5.2. Analogously, the time spent in system states  $c$  and  $d$  is constant and equal to  $P_R$ . Hence, the total cycle length equals  $P_R+P_L$  and the effective green time from the right (left) equals  $P_R-C_L$  ( $P_L-C_R$ ). The key problem is to determine  $P_L$  and  $P_R$  such, that vehicle waiting times are minimized. To this end, we need to calculate the mean waiting times for vehicles arriving at both sides of the two-way track as a function of the control parameters  $P_L$  and  $P_R$ . If the vehicle arrival processes at both sides of the two-way track are the same, it is reasonable to take  $P_L=P_R=P$ . We refer to this situation as the *symmetric* case. In this section, we address the symmetric and the asymmetric case ( $P_L \neq P_R$ ).

#### *Assumptions*

We focus on the computation of the mean waiting time at a periodically switched two-way track as described in the introduction. If we have an expression for the mean waiting time, we can use a standard numerical search procedure to find the optimal value of the switching period  $P$  in the symmetric case; we have to find both  $P_R$  and  $P_L$  in the asymmetric case.

We make an additional assumption to the ones already mentioned in Section 5.3.1. AGVs arrive one-by-one according to a Poisson process, not necessarily identical at both sides of the two-way track. With respect to this assumption, we note that Poisson arrivals are theoretically conflicting with the assumption on minimum succession times, as Poisson arrivals allow succession times that are almost zero. Many papers use the assumption of Poisson arrivals for convenience. This does not seem to be very harmful from a practical point of view if the probability that a vehicle arrives within the minimum succession time is negligible, which is satisfied in many practical cases. Although we base our derivations on Poisson arrivals, we will discuss the generalization to compound Poisson arrivals in Section 5.4.7.

#### *Model input:*

- $T$  = the driving time of a single vehicle to pass the two-way track
- $A(t)$  = the number of vehicles arriving at one side of the two-way track during an arbitrary time period with length  $t$ , a random variable
- $\lambda$  = the vehicle arrival rate at one side of the two-way track
- $\delta$  = the minimum time between two successive vehicles entering the two-way track

*Model output:*

$C$  = clearance time of the two-way track, a random variable  
 $W$  = vehicle waiting time, a random variable

*Control parameter:*

$P$  = Fixed period for traffic from one side to enter the two-way track; the cycle length equals  $2P$  (symmetric case)

As stated before, we may give the parameters  $P$  and  $\lambda$  and the random variables  $W$ ,  $C$  and  $A()$  a subscript  $R$  or  $L$ , denoting the respective characteristics of the right and left side of the two-way track. In an asymmetric case with different arrival rates  $\lambda_L$  and  $\lambda_R$ , the length of the green period at the left (right) end of the two-way track equals  $P_L - C_R$  ( $P_R - C_L$ ). Analogously, the length of the red period at the left (right) end of the two-way track equals  $P_R + C_R$  ( $P_L + C_L$ ). As we consider identical vehicles, the driving time on the two-way track,  $T$ , as well as the minimum mutual distance between vehicles,  $\delta$ , does not depend on the driving direction. Therefore, the subscripts  $R$  and  $L$  are omitted for these parameters. From a theoretical point of view, it is straightforward to include non-identical values of  $T$  and  $\delta$ .

As we explained in Section 5.2, this problem is related to other models encountered in traffic management literature and has also similarities with some queuing models (polling systems,  $M/D/1$  vacation queue). However, we did not encounter this particular model in the literature. Therefore, we developed approximate methods to calculate the mean waiting time at a two-way track for several variants. We started with the analysis of a simple case, namely a symmetric model where the *succession time*, i.e. the minimum time between successive vehicles entering the two-way track, is zero. For this special case, we can develop an exact method to calculate the mean waiting time as well as a simple approximation. It is more realistic that minimum succession times are strictly positive, taking into account the vehicle length and a minimum safety distance to avoid collisions. For this more general case, we are only able to derive approximate expressions. Therefore, we validated our approximation method by comparing the outcomes to simulation results. We subsequently addressed the symmetric and asymmetric cases. In all our models, we assume Poisson arrivals at both ends of the two-way track. However, extension to more general arrival processes is possible, and we will briefly discuss how to accomplish that.

The remainder of this section is structured as follows. First, we give the notation to be used throughout this section. The simple case with zero minimum vehicle succession time is discussed in Section 5.4.3, and some numerical results are shown in Section 5.4.4. The analysis for strictly positive minimum succession times is given in Section 5.4.5, for both the symmetric and the asymmetric cases. The corresponding numerical results are shown in Section 5.4.6. In Section 5.4.7 we describe how our method can be extended to compound Poisson arrivals.

### 5.4.2. Stability conditions

In a periodic control rule, the setting of the time interval  $P$  is restricted by the need to clear the two-way track every time the driving direction is changed. In consequence, the time interval  $P$  should exceed the maximum clearance time: the driving time  $T$ . It is straightforward to define necessary and sufficient conditions for system stability. We present these conditions for the symmetric as well as the asymmetric case. For the symmetric case, we can derive the stability condition by intuition. First, we note that the effective green period equals  $P-T$  in the worst case, because the clearance time  $C$  equals the driving time  $T$  if there is heavy traffic. Second, we observe that the maximum number of vehicles that can be processed during the effective green period should exceed the total number of arrivals in a cycle. Hence we find that  $(P-T)/\delta > 2\lambda P$  or, equivalently,

$$P > \frac{T}{1-2\lambda\delta} \quad (1)$$

This condition was formally derived by Meissl (1963) and can easily be extended to the asymmetric case. Then it is necessary that, at both sides of the two-way track, the expected number of arrivals in a cycle with length  $P_L+P_R$  can be served in the expected green time. This expected green time equals  $P_L-T$  at the left and  $P_R-T$  at the right side of the two-way track. Therefore, the following two stability conditions apply:

$$\lambda_L(P_L + P_R) < \frac{1}{\delta}(P_L - T) \Leftrightarrow P_L(1 - \lambda_L\delta) > T + \lambda_L\delta P_R \quad (2)$$

$$\lambda_R(P_L + P_R) < \frac{1}{\delta}(P_R - T) \Leftrightarrow P_R(1 - \lambda_R\delta) > T + \lambda_R\delta P_L \quad (3)$$

These two equations define the feasible area in  $\mathbf{R}^2$  for which the system is stable. As can be expected, conditions (2) and (3) reduce to the stability condition for the symmetric case (1) if  $\lambda_L = \lambda_R = \lambda$ .

### 5.4.3. Waiting times if the succession time is negligible ( $\delta=0$ )

In this section, we derive expressions for the mean waiting time  $E[W]$  for the symmetric case with minimum succession time  $\delta=0$ . For ease of notation, we omit the subscripts  $R$  and  $L$ . First, we note that, conditional on the clearance time  $C$ , the waiting probability equals  $(P+C)/2P$ . Second, we exploit the fact that the exponential inter-arrival distribution corresponding to Poisson arrivals has the well-known memory-less property. So if a vehicle has to wait, the mean waiting time equals  $(P+C)/2$ . This gives:

$$E[W | C] = \frac{(P+C)^2}{4P} = \frac{P}{4} + \frac{C}{2} + \frac{C^2}{4P} \quad \Rightarrow \quad E[W] = \frac{P}{4} + \frac{E[C]}{2} + \frac{E[C^2]}{4P} \quad (4)$$

Therefore in order to compute  $E[W]$  it is sufficient to derive an expression for the first two moments of the clearing time  $C$ . We show how we can compute the probability distribution of the clearance time  $C$  exactly, so we can compute the first two moments of  $C$  exactly as well.

When  $P > 2T$ , the effective green time  $P-C$  certainly exceeds  $T$ , because  $C \leq T$ . In that case, the clearance time is fully determined by the last arrival before the driving direction is changed. Then we find that

$$\begin{aligned} \Pr\{C = 0\} &= \Pr\{\text{no arrivals in } [0, T]\} = e^{-\lambda T} \\ \Pr\{0 < t < C < T\} &= \Pr\{\geq 1 \text{ arrivals in } [0, T-t]\} = 1 - e^{-\lambda(T-t)} \end{aligned}$$

Note that in this case the distribution is independent of  $P$ .

When  $T < P \leq 2T$ , the situation is more difficult. Nevertheless, we can exploit the fact that  $C$  has to satisfy a fundamental relationship. Let  $I_R$  and  $I_L$  denote indicators whether at the start of a green traffic light interval at the right and left end of the two-way track, respectively, vehicles are present (indicator=1) or not (indicator=0). Let  $Y$  be the length of the interval beginning when the traffic light becomes green and ending when the last vehicle during that green traffic period enters the two-way track. It is clear that

$$\begin{aligned} C_L &= C_R + Y + T - P \quad \text{if } I_L = 1 \quad \text{and} \quad C_R + Y + T - P > 0 \\ C_L &= 0 \quad \text{else} \end{aligned} \quad (5)$$

Noting that  $\Pr\{I_L = 0 | C_R = \tau\} = e^{-\lambda(P+\tau)}$ , we find that

$$\begin{aligned} \Pr\{0 < t < C_L < T | C_R = \tau\} &= 1 - e^{-\lambda(T-t)} \quad \text{if } \max(0, \tau - \Delta) < t < T \\ \Pr\{0 < t < C_L < T | C_R = \tau\} &= (1 - e^{-\lambda(P+\tau)}) + (1 - e^{-\lambda(P-\tau)})e^{-\lambda(P+\tau)} = 1 - e^{-2\lambda P} \quad \text{if } T > \tau > t + \Delta \end{aligned}$$

Due to the symmetry in the arrival processes,  $C_R$  and  $C_L$  are identically distributed. Denote their probability density as  $f(t)$  and let us introduce the shorthand notation

$$G(t) \stackrel{\text{def}}{=} \Pr\{0 < t < C < T\} = \int_0^T \Pr\{0 < t < C_L < T | C_R = \tau\} f(\tau) d\tau$$

Note that  $f(t)$  has a point mass at  $t = 0$ ; the integral should be interpreted as including this effect. It is crucial that  $G(t)$  satisfies a fundamental difference equation. Using

the relation with the conditional probabilities given above, we find after integration with respect to  $\tau$

$$G(t) = (1 - e^{-\lambda(T-t)}) \left(1 - \int_{\min(t+\Delta, T)}^T f(\tau) d\tau\right) + (1 - e^{-2\lambda P}) \int_{\min(t+\Delta, T)}^T f(\tau) d\tau = 1 - e^{-\lambda(T-t)} + (e^{-\lambda(T-t)} - e^{-2\lambda P}) G(t + \Delta) \quad (6)$$

where  $\Delta = P - T$ . Of course,  $G(t + \Delta)$  has to be interpreted as 0 if  $t + \Delta > T$ . The structure of this equation is such that it can easily be solved recursively: first  $t \in [T - \Delta, T]$ , next  $t \in [T - 2\Delta, T - \Delta]$ , etc. This leads us to

$$G(t) = \begin{cases} 1 - e^{-\lambda(T-t)} & t \in [T - \Delta, T] \\ 1 - e^{-\lambda(T-t)} + (1 - e^{-\lambda(T-t-\Delta)})(e^{-\lambda(T-t)} - e^{-2\lambda P}) & t \in [T - 2\Delta, T - \Delta] \\ (1 - e^{-\lambda(T-t)}) + \sum_{k=1}^n (1 - e^{-\lambda(T-t-k\Delta)}) \prod_{l=0}^{k-1} (e^{-\lambda(T-t-l\Delta)} - e^{-2\lambda P}) & t \in [T - (n+1)\Delta, T - n\Delta] \end{cases}$$

Equation (6) can easily be proved using induction. The analysis leads us to sums of exponential functions for  $G(t)$  and  $f(t)$  for  $t > 0$ . This computation is completed by noticing that the clearance distribution has a point mass at  $t = 0$ , such that its total mass equals 1. In this way, all moments of  $C$  can be computed. Note that in general the probability of no arrivals in a cycle will be negligible, so  $e^{-2\lambda P} \approx 0$ . If we substitute this in the equations above, we can derive a simple and accurate approximation for the clearance time distribution:

$$G(t) \approx 1 - e^{-(n+1)\lambda(T-t-\frac{1}{2}n\Delta)} \quad \text{for } t \in [T - (n+1)\Delta, T - n\Delta] \quad (7)$$

It is easy to derive the first two moments of the clearance time from this expression. Defining the integer  $N$  as  $\lfloor T/\Delta \rfloor$ , the largest number smaller than or equal to  $T/\Delta$ , we find:

$$E[C] \approx T - \int_0^{T-N\Delta} e^{-(N+1)\lambda(T-t-\frac{1}{2}N\Delta)} dt - \sum_{n=0}^{N-1} \int_{T-(n+1)\Delta}^{T-n\Delta} e^{-(n+1)\lambda(T-t-\frac{1}{2}n\Delta)} dt$$

After some manipulations, this equation can be reduced to

$$E[C] \approx T - \frac{e^{-(N+1)(T-\frac{1}{2}N\Delta)\lambda} - e^{-\frac{1}{2}(N+1)(N+2)\lambda\Delta}}{(N+1)\lambda} - \frac{1}{\lambda} + \frac{e^{-\frac{1}{2}N(N+1)\lambda\Delta}}{N\lambda} + \sum_{n=1}^{N-1} \frac{e^{-\frac{1}{2}(n+1)n\lambda\Delta}}{n(n+1)\lambda} \quad (8)$$

Analogously, we can derive for the second moment

$$E[C^2] \approx T^2 + 2 \frac{\left( [1 - (N+1)(T - N\Delta)\lambda] e^{-\frac{1}{2}N(N+1)\lambda\Delta} - e^{-(N+1)(T - \frac{1}{2}N\Delta)\lambda} \right)}{(N+1)^2 \lambda^2} \quad (9)$$

$$+ 2 \sum_{n=0}^{N-1} \frac{\left( [1 - (n+1)(T - n\Delta)\lambda] e^{-\frac{1}{2}n(n+1)\lambda\Delta} - [1 - (n+1)(T - (n+1)\Delta)\lambda] e^{-\frac{1}{2}(n+1)(n+2)\lambda\Delta} \right)}{(n+1)^2 \lambda^2}$$

The calculation of the approximate values of  $E[C]$  and  $E[C^2]$  from the expressions above is straightforward. The most convenient way to compute the mean waiting time is to combine the equations (4), (8) and (9).

#### 5.4.4. Numerical results for $\delta = 0$

We use the equations obtained in the previous section for a numerical analysis which gives some insight into the optimal value of  $P$ . We choose  $T = 7$  minutes for all our numerical experiments, this being a representative value for the OLS-case. Figure 5.3 shows the average waiting time as a function of the switching period  $P$  for various vehicle arrival rates  $\lambda$  (all time units in minutes).

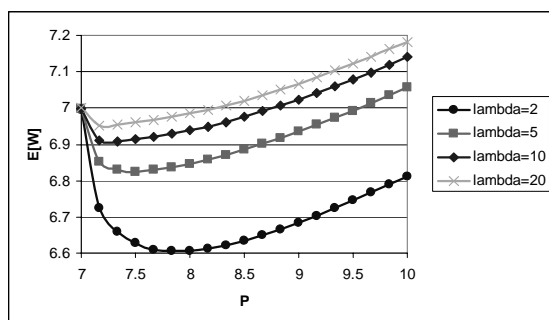


Figure 5.3. Expected waiting times for periodic control,  $T=7$  minutes

We can see that around the optimum the average waiting time only varies by a few percent. As can be expected, the average waiting time increases with the traffic intensity. Another observation is that the optimum switching period  $P$  decreases (and so does the effective green time) as traffic intensity increases. This is due to the assumption that a batch of vehicles can enter the two-way track in negligible time, irrespective of the batch size. If the driving distance between vehicles is positive ( $\delta > 0$ ), we expect an opposite effect, which will be confirmed in Section 5.4.6. Therefore, the equations as derived in the previous section are *only* useful if the minimum driving distance between vehicles is negligible indeed, for example, if a

train of vehicles can be constructed by magnetic coupling while vehicles are waiting in front of the entrance.

#### **5.4.5. Waiting times in the case of strictly positive succession time ( $\delta > 0$ )**

When mutual distances between vehicles are significant, we construct an approximation that is closely related to the case with  $\delta = 0$ . We start from the following assumption:

*The probability that a vehicle encounters more than two traffic light switches during its waiting time is negligible.*

Based on this assumption, we can use the following decomposition of the mean waiting time:

$$E[W] * E[A(cycle)] = \sum_{s=0}^2 E[K_s * W | S = s] \quad (10)$$

where

$E[W]$  = waiting time per vehicle.

$S$  = the number of traffic light switches during the AGV waiting time.

$K_s$  = the number of vehicles in a cycle for which  $S=s$  and  $W>0$ , note that  $K_s$  and  $W|S=s$  are not independent.

$E[K_s * W|S=s]$  = the total expected waiting time of all vehicles which encounter  $S=s$

and where  $E[A(cycle)] = \lambda(P_R + P_L)$  in case of Poisson arrivals. Note that the assumption above is formalized as  $\Pr\{S \geq 3\} = 0$ .

In this section we use the following, additional notation:

$M$  = the number of AGVs remaining in queue when the traffic light is switched to red

$N$  = the number of AGVs in queue when the traffic light is switched to green

$Z(n)$  = the time until a queue with length  $n$  vanishes, if the traffic light is green and not switched to red; AGVs enter the tube at rate  $1/\delta$  and additional AGVs arrive according to the process  $A(t)$ , being a Poisson process with rate  $\lambda$

$\lceil x \rceil$  = the smallest integer larger than or equal to  $x$

$x^+$  =  $\max\{x, 0\}$

For the asymmetric case, the parameter  $S$  and the random variables  $K_s$ ,  $M$ ,  $N$  and  $Z(n)$  have a subscript  $R$  or  $L$ , referring to the side of the two-way track (right or left). We derive all equations for the asymmetric case and state which simplifications apply to the symmetric case. The equations will be stated in terms of the waiting

time at the left side. Obviously, similar expressions apply for the right side of the two-way track, where the subscripts  $L$  and  $R$  are interchanged in all expressions.

We proceed as follows. First we derive expressions for the three components of equation (10),  $E[K_s * W | S = s]$  for  $s=0, 1$  and  $2$ . We will see that these expressions contain the first and/or second moment of the random variables  $M, N, Z(n)$  and  $C$ . Therefore, we derive expressions for these components. At the end of this section we summarize our algorithm.

*Basic expression for the waiting time components*

We first classify all vehicles arriving at the left side of the two-way track in a cycle with length  $P_L+P_R$  as a member of one of the sets having  $S_L=0, S_L=1$  or  $S_L=2$ . In this way, we obtain approximate expressions for the numbers  $K_{L,s}$  ( $s=0,1,2$ ).

For vehicles arriving at the left side of the two-way track, a cycle with length  $P_L+P_R$ , consists of an effective red period of length  $P_R+C_R$  and a subsequent effective green period of length  $P_L-C_R$ . The mean number of vehicles arriving in this cycle equals  $\lambda_L(P_L+P_R)$ . By definition the number of vehicles in queue at the start of the cycle (start of the red period) equals  $M_L$ . All vehicles arriving in a green period have an even value of  $S_L$  (0 or 2, ignoring higher order terms) and all vehicles arriving in a red period have an odd value of  $S_L$  (1, ignoring higher order terms). The latter implies that

$$K_{L,1} \approx A_L(P_R + C_R) \quad (11)$$

The vehicles arriving during the green period that cannot be handled in the same green period have  $S_L=2$ . This number equals the number of vehicles in queue at the end of the first green period ( $M_L$ ), excluding the vehicles that were already in queue at the end of the preceding red period and that could not be served in the first green period. Based on our assumption, we can ignore the latter number, as  $S_L=3$  for those vehicles. Therefore, we find

$$K_{L,2} \approx M_L \quad (12)$$

The number of vehicles that arrives in a green period and can be handled in the same green period equals  $K_{L,0} = \delta^{-1} * \text{Min}\{Z_L(N_L), P - C_R\} - N_L$ . This stochastic variable is difficult to handle because it requires the complete distribution of  $Z_L(N_L)$ . However, if the system is not close to its stability boundary  $\text{Pr}\{Z_L(N_L) > P - C_R\} \approx 0$ . Therefore, we use the following approximation for  $K_{L,0}$ :

$$K_{L,0} \approx \frac{Z_L(N_L)}{\delta} - N_L \quad (13)$$



We use (11)-(13) to derive expressions for the waiting time components  $E[K_{L,s} * W_L | S_L = s]$  ( $s=0, 1, 2$ ).

$S_L=1$ .

In this case the average waiting time consists of the following two components:

- the waiting time during the red period; on average  $\frac{1}{2}(P_R + C_R)$ .
- the time until the vehicle can enter the two-way track after the light has switched to red. The number of vehicles waiting at the start of a green period equals  $(M_L + K_{L,1})$ ; as they enter the two-way track at rate  $\delta$ , the average additional delay is  $M_L\delta + \frac{1}{2}(K_{L,1} - 1)\delta$ . This formula covers the fact that all  $K_{L,1}$  vehicles have to wait until the first  $M_L$  vehicles enter the two-way track, and the fact that the  $n^{\text{th}}$  vehicle enters the two-way track at time  $(n-1)\delta$  after the traffic light turns to green.

$$\text{So } E[K_{L,1} * (W_L | S_L = 1)] = E\left[\frac{1}{2}K_{L,1}(P_R + C_R + K_{L,1} - 1)\delta + K_{L,1}M_L\delta\right].$$

Noting that  $M_L$  and  $K_{L,1}$  are independent, but  $K_{L,1}$  and  $C_R$  are not, we can write this equation as

$$E[K_{L,1} * (W_L | S_L = 1)] = \frac{1}{2}E[K_{L,1}C_R] + E[K_{L,1}]\left\{\frac{1}{2}P_R + E[M_L]\delta - \frac{1}{2}\delta\right\} + \frac{1}{2}\delta E[K_{L,1}^2] \quad (14)$$

Note that  $C_R$  refers to the clearance time as part of the red period after the queue with length  $M_L$  has been created, so  $K_{L,1}$  and  $M_L$  are mutually independent. To evaluate (14), we need:

$$E[K_{L,1}] \approx E[A_L(P_R + E[C_R])] = \lambda_L(P_R + E[C_R]) \quad (15)$$

$$\text{Var}[K_{L,1}] \approx \text{Var}[A_L(P_R + C_R)] = \lambda_L(P_R + E[C_R]) + \lambda_L^2 \text{Var}[C_R] \quad (16)$$

$$E[K_{L,1}C_R] \approx E[C_R A_L(P_R + C_R)] = E[\lambda_L C_R(P_R + C_R)] = \lambda_L P_R E[C_R] + \lambda_L E[C_R^2] \quad (17)$$

Here we used the fact that for any correlated pair of random variables  $X$  and  $Y$  the well-known conditioning formulas apply:

$$E[X] = E\{E[X | Y]\}$$

$$\text{Var}[X] = E\{\text{Var}[X | Y]\} + \text{Var}\{E[X | Y]\}.$$

$S_L=2$ .

The waiting time of the  $K_{L,2} \approx M_L$  vehicles remaining at the end of a green period consists of the following components:

- the length of the next red period  $P_R + C_R$ .

- the average time until the vehicles enter the lane when the light is switched to green,  $\frac{1}{2}(M_L - 1)\delta$ .
- the average time that the vehicles have been waiting when the light is switched to red,  $\frac{1}{2}M_L / \lambda_L$ .

Analogously to the derivation of (14), and using the fact that  $M_L$  and  $C_R$  are independent, we can state that:

$$E\left[K_{L,2} * (W_L | S_L = 2)\right] \approx E[M_L](P_R + E[C_R] - \frac{1}{2}\delta) + \frac{1}{2}E[M_L^2](\lambda_L^{-1} + \delta) \quad (18)$$

$S_L=0$ .

$K_{L,0}$  vehicles arrive during a green period when the queue of vehicles that is present at the start of the green period (with length  $N_L$ ) has not yet entered the tube. As noted before, this period has length  $Z_L(N_L)$ . The corresponding waiting time consists of the following components:

- the average time until the vehicles can enter the lane, calculated from the moment the light is switched to green,  $N_L\delta + \frac{1}{2}(K_{L,0} - 1)\delta$ .
- *minus* the average time that the vehicles arrive after the traffic light is switched to green; using that the vehicles arrive during the period  $Z_L(N_L)$ , and using equation (13), we find that this correction term equals  $\frac{1}{2}Z_L(N_L) = \frac{1}{2}(K_{L,0} + N_L)\delta$ .

So after some algebraic procedures we find:

$$E[K_{L,0} * (W_L | S_L = 0)] \approx \frac{1}{2}\{E[K_{L,0}N_L] - E[K_{L,0}]\}\delta \quad (19)$$

where

$$E[K_0] \approx \frac{E[Z_L(N_L)]}{\delta} - E[N_L] \quad (20)$$

and

$$E[K_0N_L] \approx \frac{1}{\delta}E\{N_L Z_L(N_L)\} - E[N_L^2] \quad (21)$$

*The auxiliary random variables  $M$ ,  $N$ ,  $Z(N)$  and  $C$*

Equations (14)-(21) together with (10) give an approximation of the mean waiting time, but before we can solve these equations we need the following characteristics of the random variables  $M$ ,  $N$ ,  $Z(N)$  and  $C$ :

- the first two moments of the number of vehicles remaining in queue at the end of a green period,  $M_L$ , for equations (14) and (18);
- the first two moments of the number of vehicles in queue at the start of a green period,  $N_L$ , for the equations (20) and (21);
- the first two moments of the clearance time,  $C_R$ , for the equations (15)-(17);
- the expectation of the time taken by the queue to enter the tube from the start of a green period,  $Z_L(N_L)$ , for equation (20);
- the expectation of the product of  $Z_L(N_L)$  and  $N_L$  for equation (21).

Below we derive formulas for these characteristics.

*The queue lengths  $M_L$  and  $N_L$ .*

First, we observe that the following relations exist between  $M$ ,  $N$  and  $C$ :

$$N_L = M_L + A_L(P_R + C_R) \quad (22)$$

$$M_L = [N_L + A_L(P_L - C_R) - [(P_L - C_R)/\delta]]^+ = [M_L + A_L(P_L + P_R) - [(P_L - C_R)/\delta]]^+ \quad (23)$$

where the random variables  $M_L$  at the right and left hand side refer to two subsequent cycles.

Equation (23) can be solved analytically to obtain the distribution of  $M_L$ , conditional on  $C_R$ . Trying to obtain the unconditional distribution in this way leads to an unattractive analysis. Therefore, we chose a simpler approach: the moment iteration method for the mean waiting time in the  $G/G/1$  queue, as developed by De Kok (1989). This method works as follows:

1. Initialization: assume that  $M_L=0$ .
2. Calculate the first two moments of the random part in the right hand side of (23),  $X = M_L + A_L(P_L + P_R) + C_R / \delta$ ; note that this random variable has some unknown discrete probability distribution on  $\{0, 1, 2, \dots\}$ .
3. Choose a simple discrete probability distribution function that has the same first two moments as the random variable  $X$ . For this step, we applied the procedure developed by Adan et al. (1996).
4. Use these simple probability distributions to approximate the first two moments of  $M_L = (X - [P_R / \delta])^+$ .
5. Repeat the steps 2-4 until convergence is reached.

Note that we modified the approach of De Kok (1989). We also used a procedure to fit *discrete* distributions onto the first two moments of a discrete random variable rather than the continuous equivalent, as in De Kok (1989). When we have computed the first two moments of  $M_L$ , the calculation of the first two moments of  $N_L$  is straightforward. From (22) and using (15) and (16), it is immediately clear that

$$E[N_L] = E[M_L] + E[A_L(P_R + C_R)] = E[M_L] + \lambda_L(P_R + C_R) \quad (24)$$

$$Var[N_L] = Var[M_L] + Var[A_L(P_R + C_R)] = Var[M_L] + \lambda_L(P_R + E[C_R]) + \lambda_L^2 Var[C_R] \quad (25)$$

The iterative procedure above can be followed, but only if one knows the first two moments of the clearance time  $C_R$ .

*The clearance time  $C_R$ .*

Equations for the distribution of  $C_R$  can be derived from equation (5). For the case  $\delta > 0$ , this leads to an intricate analysis. In any case we have to make several approximations to obtain an explicit solution. Observe that the distribution of  $C_R$  has

a point mass in  $T$  with strength  $Pr\{C_R = T\} = Pr\{M_R > 0\}$ , but the distribution for  $0 < t < T$  is considerably more complicated. Therefore, we select a much simpler approach.

Note that the probability distribution of  $C_R$  is simple for the limiting cases  $P_R \rightarrow \infty$  and  $P_R \downarrow P_{R,\min}$ , where  $P_{R,\min}$  represents the lower bound of  $P_R$  according to (1). If  $P_R \rightarrow \infty$ , the probability that the queue has vanished before the traffic light is switched to red goes to zero. Hence, the limiting behavior of the clearance time is specified by  $C_R = T - B_R$ , where  $B_R$  refers to the timing of the last arrival before the traffic light is switched to red. So  $B_R$  is exponentially distributed with mean  $1/\lambda_R$ . Therefore, for the mean and standard deviation ( $\sigma$ ) of the clearance time we find:

$$\lim_{P_R \rightarrow \infty} E[C_R] = T - \lambda_R^{-1} \quad \text{and} \quad \lim_{P_R \rightarrow \infty} \sigma[C_R] = \lambda_R^{-1} \quad (26)$$

On the other hand, under high utilization the clearance time approached the driving time along the two-way track,  $T$ , so

$$\lim_{P_R \downarrow P_{R,\min}} E[C_R] = T \quad \text{and} \quad \lim_{P_R \downarrow P_{R,\min}} \sigma[C_R] = 0 \quad (27)$$

The idea is that we can use a simple weighted average of these limiting values for all other values of  $P_R$ . As weighting factor, we chose  $Pr\{M_R = 0\}$ , because it is clear that (27) is correct if  $M_R > 0$  and (26) will be a rather good approximation if the queue has vanished at the end of a green period ( $M_R = 0$ ). So we use the approximations

$$E[C_R] = T - \lambda_R^{-1} * Pr\{M_R = 0\} \quad \text{and} \quad \sigma[C_R] = \lambda_R^{-1} Pr\{M_R = 0\} \quad (28)$$

As an alternative to this approximation, we may consider using a weighted average for the variance instead of the standard deviation and also we may weigh the complete probability distributions instead of just the first and second moments. We have tested both alternatives numerically and found that the performance of the approach used in equation (28) is slightly better. Therefore, we chose to use the approximations in (28).

We can approximate  $Pr\{M_R = 0\}$  during the iterative procedure for the calculation of the first two moments of  $M_R$  as described above. Because this moment-iteration method for  $M_L$  ( $M_R$ ) depends on the first two moments of the clearance time  $C_R$  ( $C_L$ ), we should include the approximation of  $C_R$  ( $C_L$ ) according to (28) in the iterative procedure. That is, after each iteration, we update  $Pr\{M_L = 0\}$ , and also  $E[C_R]$  and  $\sigma[C_R]$ , according to (28). In the asymmetric case, we have to perform the iterative procedure in parallel: during each iteration, we subsequently approximate  $M_L$ , update  $C_L$ , approximate  $M_R$  and update  $C_R$ .

The time  $Z_L(N_L)$  until the queue vanishes.

For  $Z_L(N_L)$ , it is clear that the time until an initial queue with length  $N_L$  vanishes equals the time to process all vehicles in queue at the start of a green period ( $N_L\delta$ ) plus the time required to serve all vehicles arriving in the meantime ( $A_L[Z_L(N_L)]\delta$ ). So we find the following equation:

$$Z_L(N_L) = N_L\delta + A_L[Z_L(N_L)]\delta \quad (29)$$

It is not straightforward to derive the distribution of  $Z_L(N_L)$  from equation (29), but we can easily find the mean (and variance, if we wish to), conditionally on  $N_L$ . In case of Poisson arrivals, we find

$$E[Z(N_L) | N_L] = N_L\delta + \delta\lambda_L E[Z_L(N_L) | N_L] \Rightarrow E[Z(N_L) | N_L] = \frac{N_L\delta}{1 - \lambda_L\delta} \quad (30)$$

From this equation, we can derive the unconditional mean of  $Z_L(N_L)$  as

$$E[Z_L(N_L)] = \frac{E[N_L]\delta}{1 - \lambda_L\delta} \quad (31)$$

Finally, we find for the cross-product of  $Z_L(N_L)$  and  $N_L$ :

$$E[N_L Z_L(N_L)] = E[E[N_L Z_L(N_L) | N_L]] = \frac{E[N_L^2]\delta}{1 - \lambda_L\delta} \quad (32)$$

Now that we have formulas for all variables required to approximate the mean waiting time, we can summarize our algorithm.

#### *Computation of the mean waiting time: algorithm*

Because we have made a distinction between queuing behavior at the right side and at the left side, handling both the symmetric and asymmetric case is easy. In fact, to a large extent we can analyze both queues separately. The dependencies between the queues are completely specified by the clearance times  $C_L$  and  $C_R$ . So in the symmetric case, we have a single moment iteration procedure for  $C$  and  $M$ , whereas we have a joint iteration procedure for  $M_L$ ,  $C_L$ ,  $M_R$  and  $C_R$  in the asymmetric case.

We have to take the following steps to arrive at our approximation for the mean waiting time:

1. Jointly approximate the first two moments of  $M_L$ ,  $C_L$ ,  $M_R$  and  $C_R$  (only  $M$  and  $C$  in the symmetric case), using the moment-iteration method as described before.

2. Calculate the first component of the waiting time at the left side,  $E[K_{L,0} * (W_L | S_L = 0)]$ , from equation (19) using (20) and (21), and based on the auxiliary expressions for:
  - the first two moments of  $N_L$ , as specified by (24) and (25),
  - the mean of  $Z_L(N_L)$ , as specified by (31)
  - the expectation of  $N_L Z_L(N_L)$ , as specified by (32)
3. Calculate the second component of the waiting time at the left side,  $E[K_{L,1} * (W_L | S_L = 1)]$  from equation (14) using (15), (16) and (17), where the first two moments of  $C_R$  and the mean of  $M_L$  as obtained in Step 1 are substituted.
4. Calculate the third component of the waiting time at the left side,  $E[K_{L,2} * (W_L | S_L = 2)]$ , from equation (18), where the first two moments of  $M_L$  (see Step 1) and the mean of  $C_R$  are substituted.
5. Use the three components to approximate the mean waiting time at the left side of the two-way track according to (10), i.e. add the three terms and divide the sum by  $\lambda_L(P_R + P_L)$ .
6. Repeat Step 2-5 for the right side, where all indices  $L$  and  $R$  are interchanged.

In the next section, we discuss the results from the numerical tests for this algorithm.

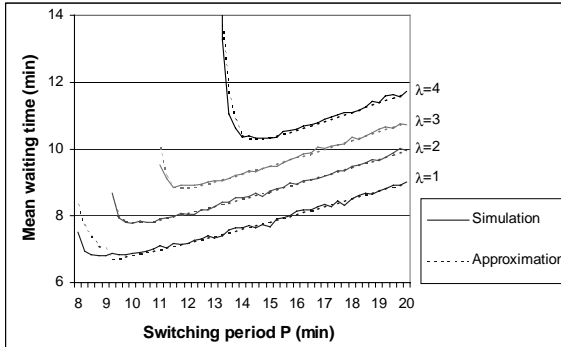
#### **5.4.6. Numerical experiments for $\delta > 0$**

In order to verify the accuracy of our approximations, we constructed a discrete-event simulation model to run a range of numerical experiments. We compare the simulation results to the approximation values. First we present the results for the symmetric case and afterwards for the asymmetric case. We designed our simulation experiments in such a way that the relative width of the 95% confidence interval for the average waiting time is less than 1% of the estimated value for  $E[W]$ .

##### *Symmetric case*

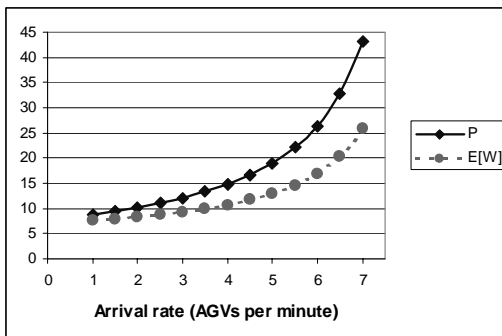
We use the algorithm as summarized in Section 5.4.5 to calculate the mean waiting time as a function of the switching period  $P$  for various vehicle arrival rates  $\lambda$  (all time units in minutes). In all cases, the driving time on the two-way track equals  $T=7$  minutes and the minimum driving time between vehicles equals  $\delta=3.5$  seconds, which are values taken from the OLS-case (cf. Chapter 3). Figure 5.4 shows a comparison between approximation and simulation results. The figure reveals that the approximation is accurate, except when the system operates close to its stability bounds. This is due to the approximation of the number of items remaining in the queue at the end of a green period,  $M$ . There can be a significant approximation error when the system is running close to its stability bound. From a practical point of view, this is not a limitation of our method, as waiting times for these cases are high anyway.

We also observe that the quality of the approximation is less for  $\lambda=1$ . However, we can see that this is a rather exceptional case. Close to the stability bound ( $P=8$ ), the effective red time equals  $P+T=15$  and the effective green time equals  $P-T=1$ , whereas only one vehicle arrives on average during the effective green time ( $\lambda=1$ ). Thus the ratio between effective red time and effective green time is very high, whereas traffic intensity is low. Therefore, this does not seem to be a very realistic case. Nevertheless, we conclude that the location of the minimum is estimated very well in all cases, also for  $\lambda=1$ .



**Figure 5.4.** Mean waiting time as function of the switching period  $P$ , analytical versus simulation results

As can be expected, the average waiting time increases with the traffic intensity. We also see that the optimal switching period  $P$  increases as traffic intensity increases. The same holds for the green time. This is due to the fact that more green time is required to allow the waiting vehicles to enter the two-way track. The latter observation is further clarified in Figure 5.5, where the optimal switching period  $P$  and the corresponding mean waiting time  $E[W]$  are shown as a function of the traffic intensity  $\lambda$ .

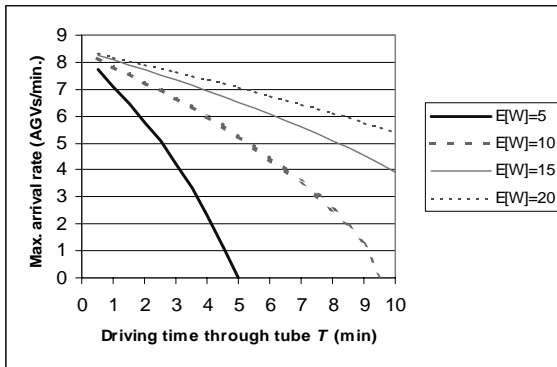


**Figure 5.5.** Optimal switching interval  $P$  and mean waiting time  $E[W]$  in minutes as function of the arrival rate  $\lambda$  ( $T=7$ )

Another interesting issue is the capacity of the two-way track. From equation (1), we obtain an upper bound for  $\lambda$  given a fixed value of  $P$ :

$$\lambda < \frac{1 - T/P}{2\delta} \quad (33)$$

The capacity equals  $\lambda = 1/(2\delta)$  vehicles per time unit for each side, but unfortunately this level is reached if  $P \rightarrow \infty$ ; in that case  $E[W] \rightarrow \infty$ . Therefore, a more realistic figure is the capacity of the two-way track *given an upper bound on the mean waiting time*. Using the algorithm from Section 5.4.5, we conduct a numerical grid search over  $P$  and  $\lambda$  to find the maximum value of  $\lambda$  satisfying the mean waiting time requirement. The results are shown in Figure 5.6, where the capacity of the two-way track is expressed in  $\lambda$ , the number of vehicles to be processed per time unit at each side of the two-way track. We see that the capacity of the two-way track, given a maximum mean waiting time, seriously decreases with increasing length, especially if only limited waiting time is accepted.



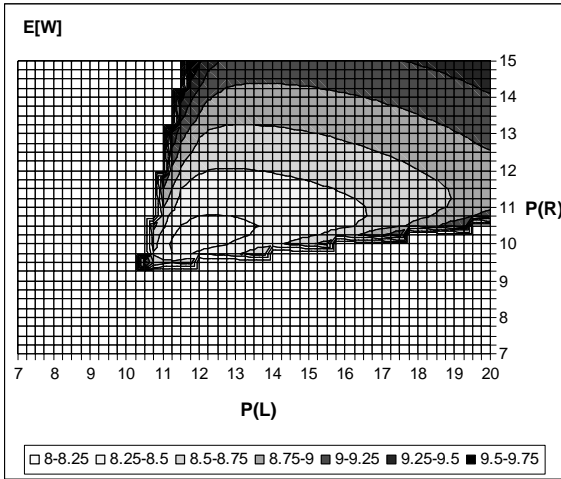
**Figure 5.6.** Two-way track capacity (number of vehicles per time limit per side of the two-way track), given a maximum mean waiting time

#### Asymmetric case

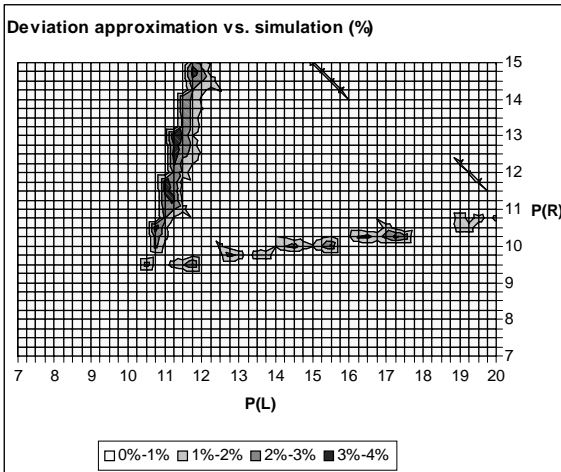
We can also use the algorithm from Section 5.4.5 to calculate the mean waiting time at each side of the two-way track as a function of the two switching periods,  $P_L$  and  $P_R$ . As an example, in Figure 5.7 we show the weighted mean waiting time  $E[W] = (\lambda_R E[W_R] + \lambda_L E[W_L]) / (\lambda_R + \lambda_L)$ , based on the input data:  $T=7$  minutes,  $\delta=3.5$  seconds,  $\lambda_R = 2$  per minute and  $\lambda_L = 3$  per minute. This figure is a contour plot; it shows areas with similar values of the mean waiting time as indicated by the legend. It is clear that the weighted mean waiting time increases faster with  $P_R$  than with  $P_L$ , given the higher arrival rate at the left side of the two-way track. The optimum is found around  $(P_L, P_R) = (11.75, 10.0)$ , where the weighted mean waiting time equals  $E[W] \approx 8.2$ .



To give an indication of the accuracy of our approximation, we make a comparison with simulation results. The relative deviation between simulated and approximated values, as function of  $P_L$  and  $P_R$ , is shown in Figure 5.8. The relative error is less than 1% for most of the cases. The error is somewhat larger close to the stability bounds, but as indicated before, this region is less interesting from a practical point of view. The location of the minimum is slightly different for the simulated values:  $(P_L, P_R) = (11.25, 10.0)$ . In our opinion, taking into account statistical fluctuations arising from the discrete event simulation, this deviation is within reasonable limits.



**Figure 5.7.** Approximation for the mean waiting time (weighted average of right and left side) as function of the switching periods  $P_R$  and  $P_L$ .



**Figure 5.8.** Relative deviation between approximation and simulation results

### 5.4.7. Generalization to compound Poisson arrivals

The method can be generalized to compound Poisson arrivals. In fact, the arrival process has an impact only on a few characteristics to be discussed in this section. If we modify these characteristics to account for other arrival processes and plug these formulae into the expressions at the appropriate places, the algorithm still applies. We show that this can easily be accomplished for *compound* Poisson arrivals. That is, batches of vehicles arrive at the left (right) side of the two-way track according to a Poisson process with rate  $\mu_L$  ( $\mu_R$ ), where the batch size has some discrete probability distribution function  $D_L$  ( $D_R$ ) with known mean and variance. Then it is straightforward to derive that

$$E[A_L(P_L + P_R)] = \mu_L E[D_L](P_L + P_R) \quad (34)$$

$$E[A_L(P_R + C_R)] = \mu_L E[D_L](P_L + E[C_R]) \quad (35)$$

$$\text{Var}[A_L(P_R + C_R)] = \mu_L E[D_L^2](P_L + E[C_R]) + \mu_L^2 E^2[D_L] \text{Var}[C_R] \quad (36)$$

$$E[C_R A_L(P_R + C_R)] = \mu_L E[D_L](P_L E[C_R] + E[C_R^2]) \quad (37)$$

Further, we find from equation (29) that

$$E[Z_L(N_L)] = \frac{E[N_L]\delta}{1 - \mu_L E[D_L]\delta} \quad (38)$$

$$E[N_L Z_L(N_L)] = \frac{E[N_L^2]\delta}{1 - \mu_L E[D_L]\delta} \quad (39)$$

When using equations (34)-(39), we can still use our algorithm. When considering further generalizations, for example to compound renewal arrivals, the expressions become more complicated, because the memory-less property of the exponential inter-arrival distribution is lost. We believe that it is still possible to generate approximations for the expressions as mentioned above, but the quality of the approximation has to be tested. This is a subject for further research.

## 5.5 Adaptive control rules

In this section, we discuss the adaptive control rules in more detail. Subsequently, we address adaptive local control, adaptive look-ahead control and dynamic programming. For these control rules we present the case with  $\delta=0$  and  $\delta>0$ . First, we introduce some notation:

- $t_0$  = decision moment
- $H$  = information horizon, i.e. at a decision moment  $t_0$ , AGV arrivals up to  $t_0+H$  are known

- $C(t_0)$  = the clearance time of the two-way track at  $t_0$ , i.e. the time needed for the last AGV to leave the two-way track, assuming that no additional AGVs enter the two-way track (note that this definition is different from the one used in Section 5.4)
- $t_i^r; t_i^l$  = the  $i$ -th arrival at the right (left) side of the two-way track, where  $t_i^r (t_i^l) \leq t_0$  if an AGV has already arrived
- $q^r(t); q^l(t)$  = the number of AGVs in queue at the right (left) at time  $t$

### 5.5.1. Adaptive local control

In this local control rule we consider the situation in which only the queue size at either side of the two-way track is known. Without loss of generality, we assume that the current driving direction is from left to right. In accordance with Section 5.3, we distinguish between two decision options: either the queue at the left passes the two-way track first or the queue at the right passes the two-way track first.

*The case  $\delta = 0$*

A direction change, meaning that the AGVs at the right side will pass the two-way track first, will induce waiting time for the AGVs in queue at the left. These AGVs are forced to wait for at least a period equal to the clearance time ( $C(t_0)$ ) plus the driving time ( $T$ ). The waiting time caused by a direction change is:

$$W_{change} = q^l(t_0)(C(t_0) + T) \quad (40)$$

When the direction is not changed immediately, but after the AGVs from the left have entered the two-way track, all AGVs in queue at the right have to wait longer. The waiting time per AGV equals the difference between the clearance time before making a decision ( $C(t_0)$ ) and the new clearance time caused by AGVs entering the two-way track from the left. The new clearance time is equal to the driving time ( $T$ ). Therefore, the cumulative additional waiting time for all vehicles in the queue at the right ( $q^r(t_0)$ ) is:

$$W_{nochange} = q^r(t_0)(T - C(t_0)) \quad (41)$$

The direction is changed immediately when  $W_{change} < W_{nochange}$ . Otherwise, in a static situation the direction change is planned after all AGVs in the queue at the left entrance have entered the two-way track. In a dynamic situation, it will be natural to make a new decision when the next AGV arrives. Note that the waiting times for the AGVs at the right side of the two-way track caused by the clearance time are regarded as *sunk costs*; the waiting time already incurred at the decision moment is not relevant.

The case  $\delta > 0$

If  $\delta > 0$ , an additional term has to be added to equations (40) and (41) to account for the time it takes a convoy of AGVs to enter the two-way track. The first AGV in queue can enter the two-way track directly, while all other AGVs have to remain at a distance  $\delta$  from their predecessor.

$$W_{change} = q^l(t_0)(C(t_0) + T + \delta \cdot \max\{q^r(t_0) - 1, 0\}) \quad (42)$$

$$W_{nochange} = q^r(t_0)(T - C(t_0) + \delta \cdot \max\{q^l(t_0) - 1, 0\}) \quad (43)$$

When the convoy of AGVs enters the two-way track, new AGVs might arrive. We do not take these additional terms into account. Note that these formulas reduce to (40) and (41) when  $\delta=0$ .

### 5.5.2. Adaptive look-ahead control

Adaptive look-ahead control can be considered as an extension of adaptive local control, taking into account future arrivals within the information horizon  $H$ . Let us consider the cost functions associated with the decision options: changing the direction at the decision moment or after the  $i$ -th arrival from the left.

The case  $\delta = 0$

The cost function for the case where the two-way track direction is changed directly can easily be found by a straightforward extension of formula (40):

$$W_{change} = \sum_{t_k^l \leq t_0 + T + C(t_0)} \min\{t_0 + T + C(t_0) - t_k^l, T + C(t_0)\} \quad (44)$$

Note that this equation reduces to (40) when  $t_k^l = t_0$ . The formula includes the waiting time of arrivals ( $t_k^l$ ) at the left entrance of the two-way track up to  $t_0 + C(t_0) + T$  unless, given the restrictions set on the information horizon  $H$ , this information is not available. Note how (44) also covers those situations in which decision making does not coincide with the arrival of an AGV ( $t_k^l < t_0$ ).

The alternative is to postpone the change in direction. The direction change can be scheduled after the  $i$ -th AGV entering the two-way track. At that time  $q^r(t_i^l + T)$  AGVs are queuing at the right entrance. For each option costs are computed as:

$$W_{nochange,i} = \sum_{t_i^l < t_k^l < t_i^l + 2T} (t_i^l + 2T - t_k^l) + \sum_{t_k^r < t_i^l + T} (t_i^l + T - \max\{t_k^r, t_0 + C(t_0)\}) \quad (45)$$

The first term considers the waiting times of AGVs arriving at the left side of the two-way track within the interval  $(t_i^l, t_i^l + 2T)$ , i.e., after  $i$  AGVs have entered the two-way track and before the scheduled AGVs from the right have passed the two-way track. The second term of (45) represents waiting times for AGVs in the queue

and arriving at the right side of the two-way track up to a time horizon  $t_i^l + T$ . Again, only arrivals within the information horizon  $H$  are taken into account. The choice of the number of options to be considered can be independent of the information horizon.

A straightforward comparison between the options is not possible, because we compute waiting times for different schedules, each of which has its own planning horizon. The moment that the last scheduled AGV leaves the two-way track is different for all decision options. Thus the computed waiting time for each option corresponds with a different planning horizon. To facilitate comparisons we weigh the waiting times with the planning period. Weighted costs associated with each option are formulated as:

$$WC_{change} = \frac{W_{change}}{T + C(t_0)} \quad (46)$$

$$WC_{nochange} = \min_i \left\{ \frac{W_{nochange,i}}{t_i^l + 2T - t_0} \right\}$$

The direction is changed immediately when  $WC_{change} < WC_{nochange}$ . Otherwise the direction change is planned after the  $i$ -th arrival which causes the minimum  $WC_{nochange}$  (static situation) or may be postponed (dynamic situation). Another possibility is to weigh by using the number of vehicles affected by the decision, but then the total waiting time should be considered instead of only the additional waiting time.

*The case  $\delta > 0$*

If  $\delta > 0$  additional terms have to be added which take into account the additional waiting time caused by the required distance between AGVs.

$$W_{change} = \sum_{t_i^l \leq t_0 + T + C(t_0) + \delta \cdot (q^r(t_0 + C(t_0)) - 1)} \min \{ t_0 + T + C(t_0) + \delta \cdot (q^r(t_0 + C(t_0)) - 1) - t_k^l, T + C(t_0) + \delta \cdot (q^r(t_0 + C(t_0)) - 1) \} \quad (47)$$

$$W_{nochange,i} = \sum_{t_i^l < \max \{ t_i^l, t_0 + \delta \cdot (i-1) \} + 2T + \delta \cdot (q^r(\max \{ t_i^l, t_0 + \delta \cdot (i-1) \} + T) - 1)} \left( \max \{ t_i^l, t_0 + \delta \cdot (i-1) \} + 2T - t_k^l + \delta \cdot (q^r(\max \{ t_i^l, t_0 + \delta \cdot (i-1) \} + T) - 1) \right) + \sum_{t_k^l < \max \{ t_i^l, t_0 + \delta \cdot (i-1) \} + T} \left( \max \{ t_i^l, t_0 + \delta \cdot (i-1) \} + T - \max \{ t_k^l, t_0 + C(t_0) \} \right) \quad (48)$$

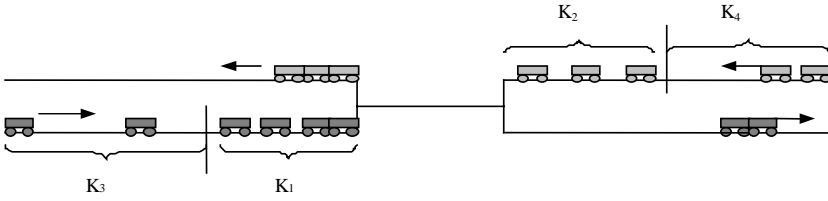
We assume that the number in the queue is greater than zero. Otherwise we have to take the maximum of  $q^r(\cdot) - 1$  and 0, as is shown in (42) and (43). The number of options may increase as a result of the entrance time. The weighted costs associated with each option are formulated as:

$$WC_{change} = \frac{W_{change}}{T + C(t_0) + \delta \cdot (q^r(t_0 + C(t_0)) - 1)} \quad (49)$$

$$WC_{nochange} = \min_i \left\{ \frac{W_{nochange,i}}{\max\{t'_i, t_0 + \delta \cdot (i-1)\} + 2T - t_0 + \delta \cdot (q^r(\max\{t'_i, t_0 + \delta \cdot (i-1)\} + T) - 1)} \right\}$$

### 5.5.3. Dynamic programming

For those situations where multiple direction changes have to be scheduled, we propose a Dynamic Programming approach (DP).



**Figure 5.9.** Sequencing of convoys

The scheduling decision involves finding the sequence of convoys which minimizes total waiting time (see Figure 5.9). Denoting by  $N_L$  ( $N_R$ ) the number of AGVs from the left (right) within the information horizon at  $t_0$ , we can formulate the objective function as:

$$\text{MIN}_{n=L,R} \left\{ f_n(N_L, N_R, t^*) \right\} \quad (50)$$

The case  $\delta = 0$

The cost function  $f_n(i, j, t)$  is defined as the minimum total waiting time for all AGVs present at the two-way track, or arriving within the horizon  $H$ , if at time  $t$  already  $i$  AGVs from the left and  $j$  AGVs from the right have passed the two-way track ( $i = 0, 1, \dots, N_L$ ,  $j = 0, 1, \dots, N_R$ ), and the last convoy passing the two-way track came from direction  $n$  ( $n = L, R$ ). Hence  $t^*$  is a point in time at which all AGVs, known at  $t_0$ , have been processed. Note that  $t^*$  is just an auxiliary variable for the recursion. Given the initial conditions at the decision moment ( $t_0$ ):

$$f_n(0, 0, t_0) = 0 \quad (51)$$

The cost function  $f_n(\cdot)$  can be formulated, starting with the left side, as:

$$\begin{aligned}
f_L(i, j - k_2, t) &= \min_{k_1=1, \dots, i} \{c[(i - k_1, j - k_2, t), (i, j - k_2, t')] + f_R(i - k_1, j - k_2, t)\} \\
f_R(i, j, t') &= \min_{k_2=1, \dots, j} \{c[(i, j - k_2, t'), (i, j, t'')] + f_L(i, j - k_2, t')\} \\
\text{with:} & \\
c[(i - k_1, j - k_2, t), (i, j - k_2, t')] &= \sum_{y=i-k_1+1}^{y=i} \max\{t - t'_y, 0\} \\
c[(i, j - k_2, t'), (i, j, t'')] &= \sum_{y=j-k_2+1}^{y=j} \max\{t' - t''_y, 0\} \\
t' &= T + \max\{t, t_i^1\} \\
t'' &= T + \max\{t', t_j^1\}
\end{aligned} \tag{52}$$

We assume that a convoy contains at least 1 AGV, without affecting optimality, because otherwise there is no reason to change the direction. The function  $c[(i - k_1, j - k_2, t), (i, j - k_2, t')]$  computes waiting times for a convoy of  $k_1$  AGVs that has entered the two-way track at time  $t$  and left it at time  $t'$ . This time is determined by the question of whether the last AGV arrived during the red ( $t_i^1 \leq t$ ) or green period ( $t_i^1 > t$ ).

The computation times do not necessarily increase enormously with more arrivals, because not all discrete points in time have to be computed; the algorithm takes long time steps of length  $T$ . AGVs that wait in queue when the traffic light becomes green will all enter the two-way track. It is not optimal to cut a queue into two separate convoys.

#### *The case $\delta > 0$*

In this case only the cost functions and the decision times change, the recursion remains unchanged.

$$\begin{aligned}
c[(i - k_1, j - k_2, t), (i, j - k_2, t')] &= \sum_{y=i-k_1+1}^{y=i} \max\{t - t'_y + \delta \cdot (y - i + k_1 - 2), 0\} \\
c[(i, j - k_2, t'), (i, j, t'')] &= \sum_{y=j-k_2+1}^{y=j} \max\{t' - t''_y + \delta \cdot (y - j + k_2 - 2), 0\} \\
t' &= T + \max\{t + \delta \cdot (q^1(t) - 1), t_i^1\} \\
t'' &= T + \max\{t' + \delta \cdot (q^1(t') - 1), t_j^1\}
\end{aligned} \tag{53}$$

Here we also neglect the fact that AGVs might arrive while the queue is entering the two-way track. Furthermore, we assume that arrivals at the two-way track satisfy the same restrictions with respect to intermediate distances  $\delta$ . In the implementation of the algorithm we assume that all AGVs that are waiting in the queue when the traffic light switches to green enter the two-way track in one convoy. A queue is not separated into two convoys, which might happen for a  $\delta > 0$ . This assumption significantly reduces computation times.

## 5.6 Design of the simulation study

To demonstrate the potential of the strategies for two-way track control, we set up an extensive simulation study. The design of the simulation study was based on figures from the OLS-case (cf. Chapter 3). In this way, practical settings are guaranteed which may also serve as examples for future systems. In Table 5.3, the experimental factors and their ranges are shown.

**Table 5.3.** *Experimental factors*

<b>Factor</b>	<b>Range</b>
1. Control rule	<b>PCR, ARloc, Arla, DP</b>
2. Average inter-arrival time (minutes)	<b>0.5, 1, 1.5, 2, 4</b>
3. Arrival distribution	<b>Poisson</b> , Uniform
4. Lot size	<b>1</b> , Uniform(3,9)
5. Tube length (meters)	1000, <b>2000</b> , 3000

For convenience, the control rules introduced in Sections 5.4 and 5.5 are abbreviated as PCR (Periodic Control Rule), ARloc (Adaptive Rule with local control), Arla (Adaptive Rule with look-ahead), and DP (Dynamic Programming). The experimental factors concern the arrival patterns of AGVs (interval, distribution and lot size) and two-way track length. The driving speed of the AGVs is constant and equals 6 m/s. For each control rule average waiting time per AGV has been measured for the default settings (marked boldly). Alternative system configurations were chosen by changing the value of only one of the factors 2-5. For all control rules we consider a dynamic setting where re-planning coincides with the moment one or multiple AGVs arrive at the two-way track and information changes or has changed. The fixed time interval of the periodic control rule is determined by using the formulas derived in Section 5.4. We use the symmetric case ( $P_L = P_R$ ) because we do not have information about arrival patterns at both ends. This results in change-over times of 6.3, 6.8, 7.1, 7.3 and 8 minutes corresponding to average inter-arrival times of 0.5, 1, 1.5, 2 and 4 minutes.

For the adaptive look-ahead rule (AR<sub>LA</sub>), we chose to consider all options for changing the driving direction within the interval  $[t_0, t_0+T]$ . Many choices are possible here; our choice of the interval was motivated by two arguments. First, considering system characteristics, the driving time on the two-way track forms a natural horizon. Secondly, as a result of simulation experiments, we found that the choice of the interval is an acceptable trade-off between the information requirement and the performance improvement. We assumed that the information on AGV arrivals would be available for a period equal to  $H = 3T$ , i.e., three times the driving time for the two-way track. This period logically follows from the above choice to consider all possible moments for changing the two-way track direction up to once the driving time on the two-way track ( $T$ ). Given a maximum clearance time of  $T$ , waiting times for AGVs may be influenced up to  $3T$ . After a change of direction



they will not be able to enter the two-way track until the last AGV from the left and the AGVs queuing at the right have passed the two-way track.

In addition to the experiments mentioned in Table 5.3, we studied the practical use of a DP-approach in somewhat more detail by means of a sensitivity analysis. Computation times were recorded and the effects of the planning frequency on system performance were studied. To get an idea of the effect of the planning frequency, we tested the DP-rule for a smaller planning frequency. Furthermore, we considered the effect of the minimum distance between AGVs on system performance ( $\delta > 0$ ).

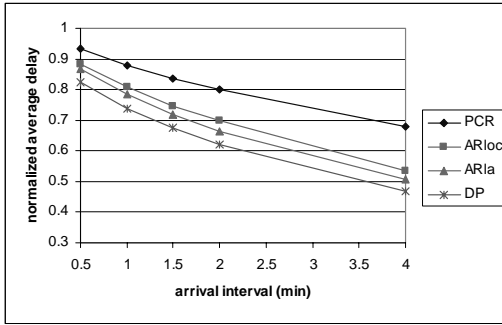
In our experiments we adopted the batch means method (cf. Law and Kelton, 1991; Hoover and Perry, 1989). Each batch equals one day. The first batch is discarded to account for any start-up bias. We related the number of batches to the relative width ( $\gamma = 0.01$ ) of the confidence interval for the average waiting time, where the significance level  $\alpha$  is set to 0.05. Lack of correlation of the batches was tested by using the runs test (cf. Hoover and Perry, 1989). Given a significance level  $\alpha = 0.05$ , this showed no significant correlation.

## **5.7 Analysis of simulation results**

In this section, we analyze the outcomes of the simulation study. First, we present results for the default settings (Section 5.7.1). Next, we discuss how the choice of alternative system configurations in terms of arrival distributions and two-way track length may influence system performance (Section 5.7.2). Sections 5.7.3-5.7.5 consider the results of the sensitivity analysis concerning the length of the information horizon, computation times and the intermediate distance required for AGVs ( $\delta > 0$ ).

### **5.7.1. Default settings**

In this section, outcomes of the simulation study are presented for the default settings: single AGVs arriving according to a Poisson process at a 2000 meter long two-way track. In order to facilitate the comparison of the results, the performance for each setting is indicated in terms of normalized average delay, i.e., the average waiting time at the two-way track divided by the driving time on the two-way track ( $T$ ).

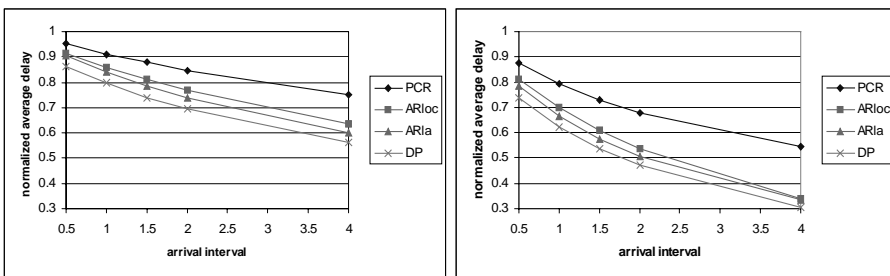


**Figure 5.10.** Normalized average delay for the default settings.

Figure 5.10 shows that the adaptive rules clearly beat the PCR-rule by percentages up to 25% at low arrival rates. The differences between the adaptive rules are smaller. The look-ahead rule performs 2-3% better than the local control rule, while the dynamic programming rule performs 5-8% better than the look-ahead rule. The outcomes clearly confirm the general proposition that the more information on future arrivals is included in decision-making, the better the results. Note that the performance differences presented in Figure 5.10 may all be considered significant, given a paired t-test with significance level 95%.

### 5.7.2. Alternative system configurations

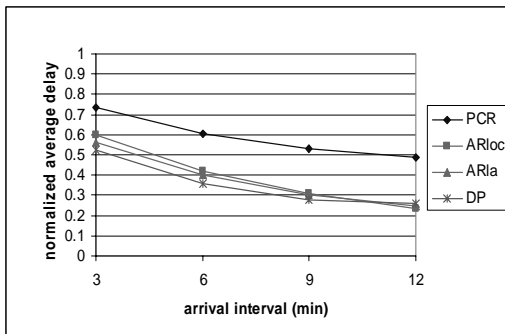
We found that the *relative* performance of the rules is hardly influenced by two-way track length. Figure 5.11 shows the results for two-way track lengths of 1000 and 3000 meters. We see the rigidity of periodic control in the sense that PCR performs very poorly for long arrival intervals in combination with a short two-way track (1000 meters).



**Figure 5.11.** Normalized average delay for a two-way track of length 3000 meter (left figure) and 1000 meter

The influence of the arrival distribution on system performance is studied in two ways. In the first series of experiments the Poisson process with a negative exponential distribution is replaced by a more “regular” Uniform distribution.

Although most results are comparable to those found for Poisson arrivals, relative performance differences between the control rules tend to be much smaller, usually less than 6% (see Figure 5.13). Secondly, we considered compound Poisson arrivals, where the lot-size of arriving AGVs is uniformly drawn from the interval [3,9]. Note that we increased the average inter-arrival times (3, 6, 9 and 12 minutes) correspondingly. If results (Figure 5.12) are compared with those for the default setting, two effects are apparent. First, a significant decrease in normalized average delay. Secondly, relative performance of PCR is worse. Both effects can be explained by the fact that irregularity of arrivals is increased. In consequence of the lengthy arrival intervals on many occasions the two-way track will be idle. However, the irregularity makes on-line information more important; the two-way track direction should be changed when a convoy arrives. Clearly, a periodic control rule will operate less efficiently under these circumstances.

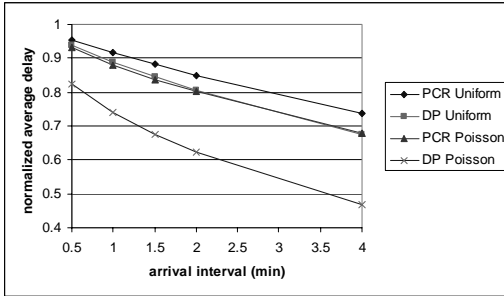


**Figure 5.12.** Normalized averaged delay for compound arrivals

Two conclusions can be drawn from this series of experiments:

1. The greater the irregularity of the arrival pattern, the better the adaptive control rules perform in comparison to the periodic control rule. This is due to their responsiveness to system state.
2. More irregular arrival patterns lead to lower normalized average delay. More frequently, large time gaps occur between AGVs, which are efficiently used by the adaptive rules for changing the two-way track direction.

These conclusions are illustrated in Figure 5.13. The difference between the performances of the periodic control rule and the dynamic programming rule is much smaller for Uniform arrivals, than it is for Poisson arrivals. We also see that normalized average delay is much smaller for Poisson arrivals, than it is for Uniform arrivals (cf. DP Poisson and DP Uniform). Similar conclusions were found by Glassey et al. (1991) in the context of dynamic machine scheduling.

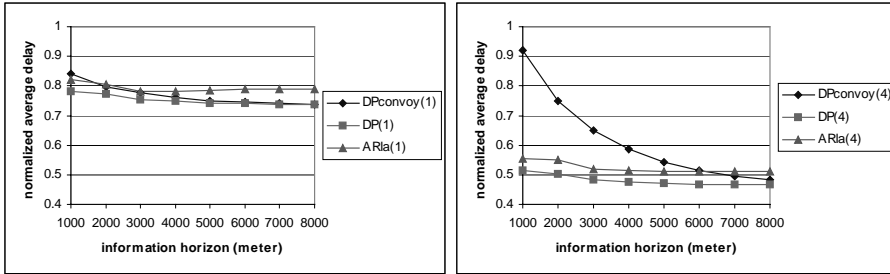


**Figure 5.13.** *Periodic control and dynamic programming for Uniform and Poisson arrivals*

In the previous experiments we considered the symmetrical case. However, if the arrival rates from the left and right side differ (asymmetric case), we expect that the performance of the periodic control rule will be worse. This hypothesis was confirmed in an additional experiment. Given an average inter-arrival time of 0.5 minutes from one side and 1 minute for the other side, the relative difference between the periodic control rule and the dynamic programming rule rises to 22% (compare Figure 5.10).

### **5.7.3. Sensitivity for the information horizon and planning frequency**

An interesting question from a practical perspective is how the performance of the new control rules relates to the availability of information on future arrivals. To answer this question, we study the relationship between the length of the information horizon and system performance. In this section, we express the information horizon in meters rather than in time, because this naturally relates to facilities for vehicle detection like sensors or inductive loops. The experiment includes the DP-rule and the  $AR_{LA}$  rule. As far as the DP rule is concerned, two scheduling frequencies are considered: per arriving AGV (DP as considered in Section 5.7.1) and per convoy (DPconvoy). The scheduling frequency for the DPconvoy rule is related to a planned direction change, i.e. no new decision is made before the planned convoy has left the tube. The two-way track length is set to 2000 meters, while AGVs arrive one by one according to a Poisson process with average arrival intervals of 1 and 4 minutes. Figure 5.14 shows the numerical results, where the average inter-arrival time in minutes is given between parentheses. We see that the dynamic programming rule (DP) and the look-ahead rule ( $AR_{LA}$ ) are not very sensitive to the length of the information horizon. An information horizon of 3000 meters already gives good results, while more distant information horizons do not lead to considerable improvement. The fact that performance is most influenced at low arrival rates can easily be explained by the relative weight of information on arriving AGVs under these circumstances.



**Figure 5.14.** Sensitivity for the length of the information horizon and planning frequency for  $\lambda=1$  (left figure) and  $\lambda=4$  minutes respectively

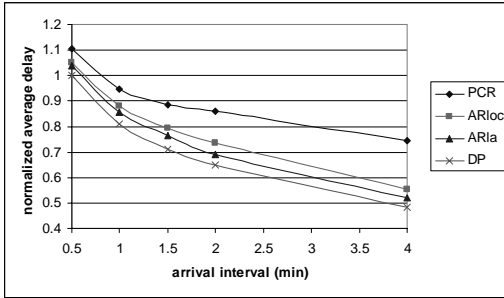
Another effect, which is clearly shown by Figure 5.14, is the effect of the re-planning frequency in a dynamic context; compare the results for both DP rules. Clearly, a lower planning frequency can significantly reduce system performance. For longer information horizons, the effect is smaller.

#### 5.7.4. Practical use of the DP-approach

The outcomes of the simulation study clearly point out dynamic programming as the best performing heuristic for two-way track control. However, whether the dynamic programming approach is useful in practice also depends on the computational effort. A series of experiments indicated that even in the worst case evaluated in the simulation study (see Section 5.6; average inter-arrival time of 0.5 minutes), the dynamic programming rule only requires a few seconds per changeover using a Pentium III 500 MHz. Although computation times for the dynamic programming rules tend to increase exponentially for higher arrival rates, this outcome still leaves a lot of room for practical application. This is especially true when one considers the fact that for high arrival rates the performance of the DP control rule is not very sensitive to the planning frequency (see Section 5.7.3).

#### 5.7.5. AGV length and safety precautions

So far in the experiments we have assumed that the minimum distance between two successive AGVs and the AGV length are negligible ( $\delta=0$ ). In this section we study the effect of both factors on system performance. The parameter setting for the periodic control rule is supported by the results from Section 5.4. This results in switching periods of  $P = 8\frac{1}{4}, 7\frac{1}{2}, 7, 6\frac{1}{2},$  and  $6\frac{1}{4}$  minutes corresponding to  $\lambda$  equal to 0.5, 1, 1.5, 2 and 4 minutes. Figure 5.15 presents results for settings in which the required distance between AGVs is set to 15 meters and the AGV length is 6 meters ( $\delta = 3.5$  seconds). The remaining settings correspond to the default settings. The results in Figure 5.15 are similar to those in Figure 5.10, although we clearly see the effect of safety precautions in terms of longer waiting times in the case of high traffic intensities.



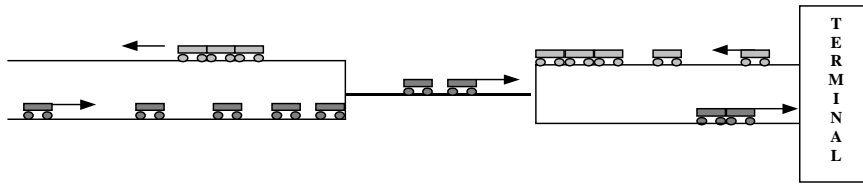
**Figure 5.15.** Intermediate distances and AGV length considered ( $\delta = 3.5$  seconds)

## 5.8 A two-way track in a closed system

### 5.8.1. The effects of convoys on the system

Until now we have studied the two-way track in isolation, as part of an open system. As a consequence, arrivals were considered independently of each other and an infinite population was assumed. It is reasonable to assume an open system in the case of a road network or a large transportation network. In these cases, the convoys that are formed by the two-way track are dispersed to different destinations and the population is also large compared to the traffic in the section under consideration. In a closed system these assumptions can no longer be justified, making it necessary to study the effects of interaction with other elements in the network (compare the attention paid in the traffic literature to coordinating traffic signs, see e.g. Bell, 1992). For example, let us consider the situation where the network at one side of the two-way track is a single terminal (see Figure 5.16). The convoys that leave the two-way track drive to the terminal, load/unload, and probably soon return to the same two-way track. So the two-way track will face a batch arrival of AGVs, where the batch size is heavily influenced by a previous direction change. Decisions on two-way track control may therefore have serious impact on future decisions, as well as having a significant impact on total system performance.

We consider a system in which there is a single terminal at one side of the two-way track (see Figure 5.16). The other side of the two-way track connects to a large transportation network, such that arrivals at the left entrance are still independent.

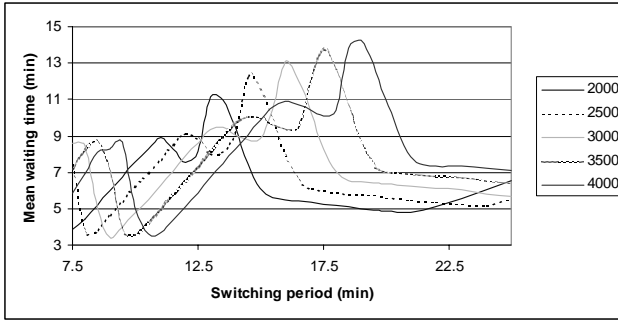


**Figure 5.16.** *A system with one terminal and one two-way track*

AGVs arrive at the left entrance of the two-way track according to a Poisson process. These AGVs pass the two-way track and drive to the terminal. The time an AGV spends at the terminal depends on the required actions (loading, unloading or both) and on the waiting and driving times on the terminal. After leaving the terminal an AGV returns to the two-way track. The terminal process is represented by a Uniform distribution with service times between 3 and 6 minutes. These times are related to the OLS-case, where loading or unloading takes 2 minutes and driving times on the terminal vary strongly.

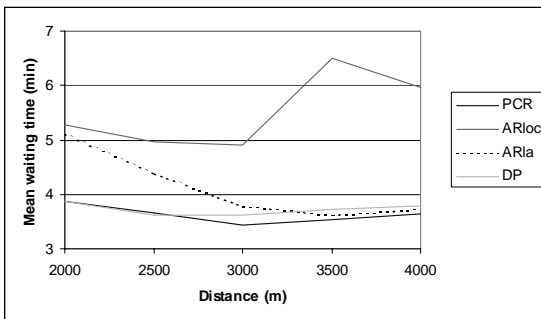
We expect that the switching period  $P$  of the periodic control rule is very sensitive to the distance between the two-way track and the terminal. The switching period can no longer be determined by the formulas in Section 5.4, because the arrivals at the right entrance do not follow a Poisson process. The question is whether the control rules are sensitive to the distance between the terminal and the two-way track. To have sufficient information on AGV arrivals, we varied the distance between the two-way track and the terminal from 2000-4000 meters in steps of 500 meters. The average inter-arrival interval at the left entrance is 0.5 minutes.

In Figure 5.17 we clearly see the sensitivity of the periodic control rule to the distance between the terminal and the two-way track. A wrong choice of the switching period can more than double the mean waiting time (left and right) at the two-way track. On the other hand, we see that the minimum mean waiting time is not very sensitive to the distance between the terminal and the two-way track. The minimum value for all lines is almost the same.



**Figure 5.17.** Mean waiting time as a function of the switching period for different distances between two-way track and terminal for  $\lambda=2$

These experiments show that it is difficult to determine a switching period for the periodic control rule in a closed system. Therefore, periodic control is not a suitable option in the design phase of an automated transportation system as the layout frequently changes and parameter tuning is required for each variant. Arrival distributions are time-dependent, because order patterns vary during the day. The switching period corresponding with the minimum mean waiting time is different for other values of  $\lambda$ . Figure 5.18 shows that all control rules are rather sensitive to the distance between the terminal and the two-way track, except for the dynamic programming rule. The periodic control rule shows very good results, but Figure 5.17 showed that it is almost impossible to determine the optimal switching period. In Figure 5.18 the best solution for the different switching periods is shown for the periodic control rule. This was only found after a series of simulation experiments.



**Figure 5.18.** Sensitivity for the distance between one terminal and a two-way track for  $\lambda=2$

Similar problems occur in the case of a second two-way track with no crossings or intersections in between both two-way track sections. The convoys created can cause congestion problems at terminals and other two-way tracks. Therefore, an interesting question is how the local control rules effect total system performance. Below we try to answer this question by studying the OLS.



### 5.8.2. Numerical results for the OLS-case

To check whether our conclusions for two-way track control in an open system are equally valid for a closed system, we embedded the rules in a simulation model for the OLS (see Chapter 2). The layout that was studied contains one two-way track (cf. Figure 3.3). At one side of the two-way track there is only one terminal (VBA). At the other side the two-way track is connected to several terminals at AAS and RTZ. The two-way track has a length of 1500 meters and for both sides accurate information on arrivals is available by recording AGV movements 1500 meters before the entrance. Note that we shortened the two-way track to ensure that sufficient arrival information is available. The required distance between two AGVs is 3.5 seconds. The control rules still aim to minimize the waiting time locally at the start of the two-way track, but we measure the total system performance in terms of the service level, i.e. the percentage of transportation jobs that is handled before the due time. Note that we take the same information horizon for both sides. This is always possible if the minimum is taken from the information horizons. Using different horizons at each side is not a problem for PCR and  $AR_{LOC}$ , but it might influence the performance of the other two control rules. We did not investigate this effect.

The question is whether the local control rules that perform best in minimizing the average waiting time at an isolated two-way track also result in best system performance. For PCR after some initial experiments we chose a switching period of 7 minutes. The simulation results confirm the ranking of the control rules with respect to our performance measure, which can be seen in Table 5.4.

**Table 5.4.** Performance of the control rules in a closed system as in Figure 3.3

Control rule	180 AGVs	190 AGVs
PCR (7 minutes)	90.2	99.7
$AR_{LOC}$	92.4	99.5
$AR_{LA}$	93.5	99.6
DP	99.1	100

Note that for a closed system such as our application, we may adjust the control rule, e.g. by including information on the situation at the surrounding terminals and two-way tracks. The dock capacity of a terminal is known, together with the number of jobs. A convoy that passes a two-way track and arrives at a terminal will incur waiting time because of the restricted dock capacity. The length of the convoy is known and also the number of unloading operations. With this information the expected waiting time at the terminal can be estimated and these waiting times could be included in the control rules. We did not consider such a method because we focus on local control. Furthermore, such an approach might be applicable in a situation with only one terminal at one side of the two-way track (see Figure 5.16) but is very difficult to generalize. The effects of a large convoy in a network with a lot of terminals are difficult to estimate.

It is also possible to use another objective function. Instead of minimizing the average waiting time at the two-way track, a local objective, we could minimize a systems related objective, for example the expected number of late AGVs. An AGV is late when it arrives at its destination after its due time. This means that all AGVs need to have a due time, including AGVs that are driving empty toward a terminal or central parking area. As their due time loaded AGVs have the due time of the transportation job. Empty AGVs driving toward a terminal can be given the release time of the transportation job they are supposed to pick-up as a due time. Empty AGVs driving toward a central parking area can receive a due time far in the future; they are not needed in the near future. Furthermore, the expected travel time from the two-way track to all possible destinations should be available. With the due times and the expected travel times it is possible to determine whether an AGV is expected to be late at its destination. The control rule at the two-way track should try to minimize this expected number of late AGVs. We did not investigate a due time based rule because such a rule is very situation specific and cannot be used in general traffic applications where no information about due times is available.

## 5.9 Conclusions

In this chapter we addressed the issue of real time control of a two-way track. Several new control rules were defined as an alternative to the classic periodic control rule, which manages traffic flows on the two-way track by giving priority to either direction according to a fixed time interval. For the periodic control rule we derived an algorithm to approximate the mean waiting time. We showed that the algorithm works well by comparing it to simulation for both the symmetric ( $\lambda_R = \lambda_L$  and  $P_R = P_L$ ) and the asymmetric cases. For the most relevant cases, the approximation error is less than 1%. Moreover, the location of the optimum (minimum mean waiting time as function of  $P_L$  and  $P_R$ ) is estimated well. Although we developed our method for Poisson arrivals, we showed that extension to compound Poisson processes is straightforward. Further generalization to compound renewal arrivals requires additional approximations and is a subject for further research.

The adaptive control rules try to make good use of local information on queue lengths and look-ahead information on future arrivals in order to improve on system performance. Simulation results indicate that, depending on the amount of information available, adaptive control rules improve system performance up to 10% for high arrival rates and up to 25% for low arrival rates. Best performance is realized by a rule based on dynamic programming, even in the case of a short information horizon. While it is known that a dynamic programming approach may involve high computational costs, simulation results for a realistic large-scale transportation network indicate that computation times for the dynamic programming rules are acceptable for real-time applications.

In general, the performance of the rules is not much influenced by the length of the information horizon. Only in the case of a low planning frequency (e.g. related to the planned change-over times), is the dynamic programming rule sensitive to the length of the information horizon. This effect is most noticeable at low traffic intensities. The introduction of safety precautions, such as the requirement that a minimum distance should be kept between AGVs to prevent collisions, does not significantly change the relative performance of the rules. Nevertheless, it does significantly change the optimum changeover time for the periodic control rule. An experiment that included the two-way track in a closed network, the OLS, showed that our conclusions still hold for a closed system. We examine the robustness of these control rules to disturbances caused by equipment failures in Chapter 6 (Section 6.3.5)

# Chapter 6

## Secondary processes battery and failure management, and the control of restricted storage capacities

### 6.1 Introduction

In this chapter, we describe explorative research on some of the secondary processes in automated transportation networks that were introduced in Chapter 2 including battery management and failure management, and the management of the primary process with restricted storage capacities. We describe only one possible way to take these aspects into account in the control of the network and do not compare several control strategies. We want to show that the framework described in Chapter 2 is flexible enough to take these processes into account. Furthermore, we want to investigate the effects of constraints imposed by the secondary processes on the behavior of the model, and the implications of these in the OLS-case. The inclusion of battery constraints is discussed in Section 6.2. We discuss how the results from this section can be used to choose a way of energy provisioning for the OLS-case. In Section 6.3, we describe the implications of failures of AGVs and docks. The effects of restricted storage capacities, i.e. an additional bottleneck resource in the primary process, are discussed in Section 6.4. Optimization of such control activities within the OLS-project strongly depends on technological choices still to be made. Therefore, the goal is to indicate the general effects and to highlight important aspects that should be taken into account in answering some of the technological

questions. In this chapter we answer research questions 5 and 7 as stated in Chapter 1 (Section 1.2.2). For all experiments in this chapter we use EVM3 (look-ahead rule with hierarchical coordination, cf. Chapter 4, Section 4.4) for vehicle management. This control rule did not require adjustments when used in a system with two-way tubes. Two-way tubes themselves are controlled according to an adaptive rule without look-ahead, as discussed in Chapter 5, Section 5.4. This choice was made because the two-way tracks are close to the terminals; thus, there is little arrival information available.

## **6.2 Battery management**

### **6.2.1. Introduction**

AGV battery replenishment or recharge is neglected in most studies on AGV systems. Omitting battery constraints does not always have serious consequences (McHaney, 1995), particularly in internal AGV systems (production, warehousing) with short distances and with sufficient idle time to replenish or swap batteries. Nevertheless, in the case of high AGV utilization and long distances, these battery constraints cannot be neglected. McHaney (1995) also shows some examples in which the required number of AGVs significantly increased after the inclusion of battery constraints, and in some cases even doubled. In this section, we focus on the implications of battery constraints for the logistic control. Furthermore, we present several options for battery replenishment and show results for different strategies and battery types.

We assume that the vehicles are driven by an electric motor (cf. Chapter 3). A decision has to be made concerning the battery type: lead-acid or nickel-cadmium. Furthermore, there are several technical options with respect to battery replenishment:

- a battery swap
- recharging the battery inside the AGV
- charge-rails

A battery swap strategy has limited influence on the operational time of the AGVs. The batteries are recharged outside the AGV. A disadvantage is the requirement of special battery stations to execute the swap operation. Recharging the battery inside the AGV means that an AGV is not available for several hours. In the OLS-case, the recharging time of a battery is longer than the possible driving time on a full battery, for a lead-acid battery recharging takes 8 hours, while the maximum driving time on a fully recharged battery is 6.4 hours (Van Harten et al., 1999). We do not consider this option in this chapter, because it would more than double the required number of AGVs. The same holds for recharging a nickel-cadmium battery inside the AGV.

The last option is charge-rails, where the AGVs are coupled to a charge-rail in part of the track network. The battery is recharged to such a level that the AGVs can drive freely in the other parts of the network (e.g. on the terminals). The option of charge-rails does not influence system performance because batteries are replenished while AGVs are moving. An additional advantage of charge-rails, compared to swapping and recharging batteries, is that only as many batteries as AGVs are required, whereas additional batteries are required for the other two options. A drawback is the additional investment in charge-rails. In order to make a proper trade-off the results of our simulation studies can be coupled to a cost analysis. In Section 6.2.4, we give a hypothetical example of the way in which such a trade-off can be made.

Logistic choices have to be made, depending on the chosen technical solution. The questions we want to answer with respect to battery management are:

1. What are the implications of the technical solutions on the logistic control?
2. What are the effects of the different technical solutions and battery types on the logistic performance?
3. Where should the battery stations be located?
4. What are the required resources for the different options?
5. What are rough indications for the costs of the different options?
6. What is the most attractive solution with respect to costs and logistic performance?

These questions will be answered in the following sections.

### **6.2.2. Implications of battery constraints for the logistic control**

The introduction of battery constraints has implications for the logistic control. We discuss these implications for two options: a battery swap and charge-rails. The battery type and the option of charge-rails do not influence the logistic control. In the case of charge-rails, the only prerequisite is that the replenishment by charge-rails is sufficient to permit the AGV to drive freely in those parts of the network without charge-rails. For the OLS-case, this condition is satisfied if charge-rails are constructed in the tubes and AGVs only have to drive freely on the terminals. No battery stations are needed, but investments are required for the infrastructure on the tracks. In case of a battery swap, the AGV has to be sent to a battery station where the battery will be changed. Two questions have to be answered by the logistic control. It is necessary to determine *when* and *where* the battery of an AGV should be changed.

*When?*

Three principles regarding the replacement of batteries can be considered. The first option is to replace a battery when its charge is not sufficient to perform the next job. An AGV receives a job from vehicle scheduling and checks whether its battery charge is sufficient to complete the job. If the battery charge is not sufficient,

vehicle scheduling is notified. Vehicle scheduling has to determine whether another AGV should be used for the job, if not the selected AGV has to go to a battery station while executing the job.

A second option is to change the batteries of all AGVs before a peak period. Such a preventive replacement of batteries is attractive from the point of availability when a fully charged battery is sufficient for this peak. Unfortunately, this is not the case in the OLS-system; batteries have to be changed at least once during the peak due to the length of the peak period, therefore, we do not take this strategy into further consideration.

Another option is to change the battery when an *opportunity arises*, i.e. a possibility to change the battery when an AGV has to wait for some reason. Using such opportunities could reduce the operational AGV time lost by battery changes but it does increase the number of battery swaps and so the battery swap stations should have higher capacity. Opportunities arise when:

- An empty AGV is temporarily parked at a terminal waiting for a new load job. When an AGV is sent to a parking place, the battery manager is notified and, based on the charge information supplied by the AGV, can determine whether it is possible and useful, to change the battery of the AGV. It supposes that there is a battery station at or near the terminal in question.
- An AGV has to wait at the entrance of a two-way track or terminal, due to the limited capacity of such resources. When an AGV passes a sensor close to a two-way track or terminal, it supplies information about its battery charge to the battery manager. The battery manager can determine whether the battery can be changed within the expected waiting time at the entrance of the terminal or two-way track, or by losing only a fraction of the time normally required for a battery swap. This opportunity can only be used if there is a battery station close to the two-way track or terminal.
- An AGV that is on its way to or is at a central parking area. When an AGV is sent to a parking area it is apparently not needed directly, so this is an opportunity to change the battery without seriously affecting the logistic performance. In fact, this is a rather safe opportunity, as there are apparently sufficient resources if an AGV is sent to a central parking area. However, these opportunities do not occur during peak periods. Logically, a central parking area is an attractive location for a battery station.
- An AGV that is loading or unloading at a dock. If batteries can be replaced at a dock while an AGV is being loaded or unloaded, the logistic performance is not influenced by this activity. Such a solution sounds attractive, but from a technical point of view it is probably more difficult. Given the uncertainty about whether it is technically possible, we do not consider this option in this research.

### *Where?*

This decision covers the choice of locations for battery stations and, given that an AGV has to change its battery, the selection of one of the available battery stations for a battery change. In choosing locations for battery stations, the opportunities should be taken into account (terminal, parking area, dock, and two-way track). Locations with high traffic intensity might be attractive as well. Besides logistical reasons, costs and the available space are also important factors in determining the possible locations for battery stations.

Given that an AGV has to change its battery, one of the battery stations has to be selected. Several heuristic rules are possible:

- Nearest battery station
- Farthest reachable battery station on the current route
- First battery station encountered on the current route
- Battery station that leads to minimum delay

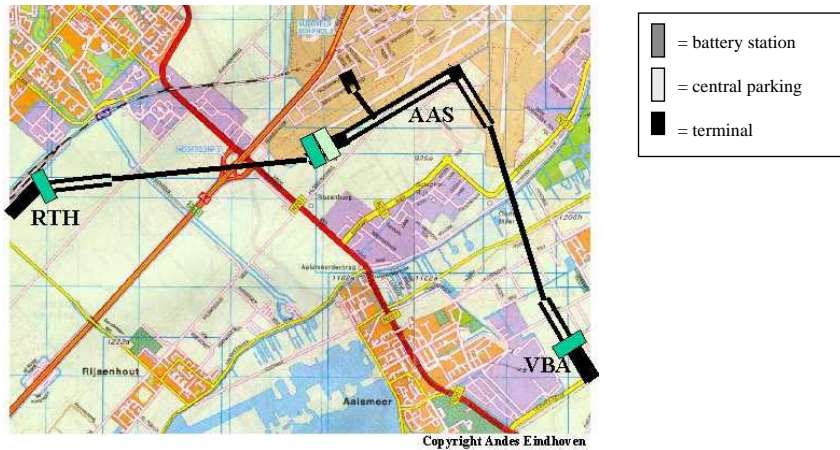
The choice of where to change the battery depends on which battery stations can still be reached. It must always be possible to reach the nearest battery station; otherwise, the battery should have been swapped during or before the previous job. In addition to the nearest battery station, other battery stations might also be reachable. One battery station should be selected from the set of reachable battery stations. It seems attractive to select a battery station on the route of the AGV. When there are several battery stations on the current route one might want to minimize the remaining charge of the batteries. In this case, battery management should choose the farthest accessible battery station on the current route. When no battery stations are present on the current route, battery management can choose the battery station that results in minimum delay for the job in question. Besides driving times, also the expected waiting time at a battery station could be taken into account. Somewhat different aspects play a role in case of an opportunity swap. In case of an opportunity at a terminal, where a battery has to be swapped within limited waiting time, it is logical to use the nearest battery station to perform this battery swap. If an AGV is on its way to the central parking area, the choice of a battery station is less important. However, when a battery station is located at the central parking area it is logical to use that one, if it can still be reached.

### *The OLS-case*

For the OLS-case, we first have to choose the locations of the battery stations. The layout with battery stations is shown in Figure 6.1, which was the preferred system layout at the time of research on battery constraints (Van Harten et al., 1999). In determining the possible locations of the battery stations, several criteria were important. Based on opportunities, available space, traffic intensities, dispersion over the network and costs, three locations were selected. Because of the high traffic intensity between VBA and RTH and the location of these terminals at the network boundaries, these locations seem logical. A central location near AAS seems a good option as well. As most waiting times (and hence opportunities) arise at the two-way



tube, the location between the western terminal at AAS and the two-way tube between AAS and RTH seems preferable to a different location at Schiphol. Moreover, the central parking area is also planned at that same location. The three locations indicated in Figure 6.1 might not be cost and space optimal for the battery stations, but no detailed information about costs and available space was available. The number of docks at a battery station is limited, but the recharging capacity, i.e. the capacity to recharge the batteries outside the AGVs, is unlimited.



**Figure 6.1.** *Layout with two-way tracks and battery stations*

The battery stations are located as follows: one battery station near the rail terminal, one battery station near the flower auction market, and one battery station close to the central parking area at Schiphol Airport. With these locations, opportunities for a battery swap at the terminals RTH and VBA can be used, and the AGVs driving toward the parking area can also change their batteries. A terminal cannot handle an entire convoy (created by the two-way track) directly, and therefore waiting times occur at the entrances of the terminals. While waiting at the entrance, the battery could be changed if a battery station was nearby. The effects of the number of battery stations are determined by a simulation study. We compare the case with 3 battery stations with that where there is only 1 battery station at the central parking area.

Battery management is implemented as follows. When an AGV receives a new job, the AGV does not have to swap its battery if the remaining charge is sufficient to drive to its new destination and from this destination to the nearest battery station. When the remaining charge is not sufficient, the AGV is sent to the nearest battery station to swap its battery. From this battery station, the AGV can carry on to its destination. We assume that both loaded and empty AGVs can swap the battery. Usually, the future destination of an AGV is not known. This means that an AGV that is driving to a terminal will receive a new job (empty or loaded) at the terminal,

so whether the remaining battery charge is sufficient for this future job is also not known. We choose to use a swap opportunity when the charge of a battery is not sufficient to drive from the current location to the farthest possible destination in the network and from there to the nearest battery station, since this is rather robust in terms of transport planning. AGVs will swap batteries more often than strictly necessary, but the battery swaps can take place without real loss of time. An AGV that drives to the central parking area changes its battery when the charge would not be sufficient to drive from the parking area to the farthest possible destination and from there to the nearest battery station. The criterion of farthest possible destination might not always be appropriate. If one location is situated far from all other terminals, this criterion might lead to unnecessary battery changes. A possible solution is to swap the battery if the charge is not sufficient for X% (e.g. 70%) of the jobs at that specific terminal.

Notice that these control rules for swapping batteries need estimates of the energy consumption on all routes. These estimates have to be accurate, because an empty battery, and as a result a stationary vehicle, has serious effects on the logistic performance of the network in terms of throughput times and service levels (cf. Section 6.3 on failure management). The actual energy consumption on the routes can be recorded and average energy consumption is computed using an exponential smoothing technique (cf. Section 2.6). A safety factor is added to this average to minimize the chance that the AGV will fail to reach its destination. If in spite of this the AGV has to stop because of insufficient battery charge, it can be considered as a failed AGV, and therefore it has to be treated in the way as described in Section 6.3.

### *Experiments*

With the simulation model, we investigated the effects of different options on the logistic performance and resource requirements. We investigated different options for battery replenishment, charge-rails or battery swap, and different variants with respect to battery type (lead-acid, nickel-cadmium), number of battery stations and the battery swap time (see Table 6.1).

**Table 6.1.** *Experimental factors*

<b>Parameter</b>	<b>Options</b>
Battery option	Swap, charge-rails
Swap time (min.)	1, 5
Battery type	Lead-acid, Nickel-cadmium
Number of battery stations	1, 3

The swap time of 1 or 5 minutes was chosen after discussions with experts within the OLS-project in the field of automated docking and charging technology. If only one battery station is included, it is logical to choose the most central location: the battery station near the parking area. Only one battery station reduces the possibilities to use waiting times or AGV idle time for a battery swap. For the experiments, we used the transportation flows from Table 3.2 in Chapter 3, together

with the general assumptions made in Chapter 3 with respect to speed, length of an AGV, etc.

Two options with respect to battery type were investigated, lead-acid batteries and nickel-cadmium batteries. For the nickel-cadmium battery we also consider a quick-charge option, which means that the battery is recharged up to 80% of its capacity, which can be done in 20% of the time for a complete recharge (see Table 6.2). An advantage of the nickel-cadmium quick-charge option is that fewer batteries are needed in the system because of the short charge time. A big disadvantage is that vehicles have to swap these batteries more often. In consequence, more battery swapping capacity is required at the battery stations, order throughput times may increase and probably more vehicles are required to make up for this lost time. The battery charge capacity and required charging times of the battery types are shown in Table 6.2.

**Table 6.2.** *Battery characteristics*

<b>Battery</b>	<b>Capacity (kWh)</b>	<b>Charge time (hour)</b>
Lead-acid	26.9	8
Nickel-cadmium	17.9	5
Nickel-cadmium quick-charge	13.1	1

We assumed that the recharging capacity of the battery charge stations, is not a bottleneck, i.e. there are always fully charged batteries available when a vehicle wants to swap its battery. Nevertheless, the number of batteries recharging is measured in the simulation model to facilitate the cost computations. The recharging time of a single battery is proportional to the amount of charge that has to be replenished.

The energy consumption of the batteries is not the same in all parts of the system since it depends on speed, status (loaded or empty), and also on possible slopes. Different figures are available for these cases, see Table 6.3 (Van der Heide, 1999). The energy consumption is independent of battery type, but driving in a two-way tube requires more energy than in a one-way tube, due to the convoys created in the two-way tube. Clearly driving up a slope requires a lot of energy.

**Table 6.3.** *Battery consumption while driving*

<b>Status vehicle</b>	<b>Consumption per hour (kWh)</b>
Empty in one-way tube (6m/s)	4.2
Loaded in one-way tube (6m/s)	6.5
Empty in two-way tube (6m/s)	6.8
Loaded in two-way tube (6m/s)	9.0
Empty on terminal (2m/s)	1.0
Loaded on terminal (2m/s)	1.8
Empty up-ramp (3m/s)	17.0
Loaded up-ramp (3m/s)	31.0
Empty down-ramp (6m/s)	0.9
Loaded down-ramp (6m/s)	0.9

Moreover, we took the energy consumption for certain specific events into account. Acceleration requires additional energy, and at locations where AGVs always accelerate, for example at the exit of a terminal, or after a slope, this energy consumption was taken into account (see Table 6.4). We did not take into account the energy used as a result of acceleration on a terminal.

**Table 6.4.** *Energy consumption for acceleration*

<b>Event</b>	<b>Consumption per event (kWh)</b>
empty accel. 2 -> 6 m/s	0.0177
full accel. 2 -> 6 m/s	0.0322
empty accel. 3 -> 6 m/s	0.0165
full accel. 3 -> 6 m/s	0.0283
empty accel. 0 -> 6 m/s	0.0194
full accel. 0 -> 6 m/s	0.0356

### **6.2.3. Logistic performance in case of battery constraints**

In this section, we show the results of the experiments defined in the previous section. First, we determine the system dimensions with regard to the required number of vehicles, number of battery stations and the number of docks in these battery stations. The results of this system without battery replenishment are equal to the option of charge-rails. Using charge-rails means that batteries do not have to be changed. The resource requirements are 8 docks at RTH and VBA, 2 docks per terminal at AAS and 360 AGVs (cf. Table 3.14); the loading and unloading time is 1 minute. The number of docks in the terminals is kept constant because the introduction of battery constraints has no real effect on these required capacities. Some preliminary experiments confirmed this statement. In Table 6.5, we show the most important results for the different options of battery replenishment. The number of battery swaps influences the required docking capacity at the battery stations.

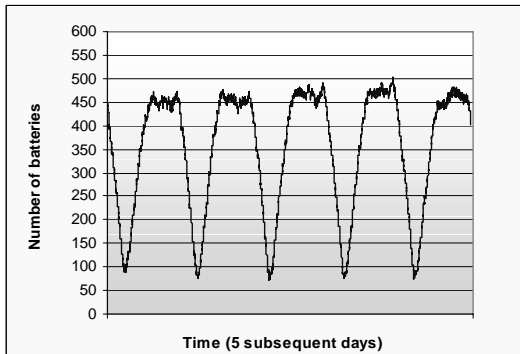
**Table 6.5.** *Overview of the most important results for battery replenishment*

Battery	Swap time (min.)	Battery stations (docks)	Number of AGVs	Number of batteries	Average number of battery swaps per day	Average remaining battery charge (%)
Charge-rails	n.a.	0	360	360	n.a.	n.a.
Lead-acid	5	1 (30)	380	894	1112	17.6
	5	3 (3x15)	370	975	1043	12.5
Nickel-cadmium	5	1 (30)	390	1015	1870	25.7
	5	3 (3x15)	380	1087	1715	18.6
Nickel-cadmium quick-charge	5	1 (40)	410	620	3047	35.7
	5	3 (3x20)	380	664	2679	28.1
Lead-acid	1	3 (3x6)	360	975	1043	12.5
Nickel-cadmium	1	3 (3x6)	360	1087	1715	18.6
Nickel-cadmium quick-charge	1	3 (3x6)	370	664	2679	28.1

The required number of AGVs increases as a result of the battery constraints, since the AGVs lose time during the battery swap and this lost time can only be compensated by adding more AGVs. The extra number of AGVs depends on the number of battery swaps per day and the use of idle time. The number of AGVs chosen is such that the throughput time distributions and service levels are about the same for the different options (97%). It is clear that when looking at the number of AGVs required, preference is given to 3 battery stations, but this requires additional investments in battery stations. In particular, frequent battery swaps (nickel-cadmium quick-charge) and only 1 battery station lead to inefficient AGV-usage. The difference between the options of 1 and 3 battery stations is also due to the fact that in case of 1 battery station fewer changes can take place during opportunities.

In determining the number of docks per battery station (between brackets in Table 6.5) we assumed that the battery swap should not become a bottleneck, causing extra waiting time. Nevertheless, in the case of a swap time of 5 minutes this can easily happen when docking capacity is too small. Waiting times increase quickly with decreasing capacity. This entails that large battery stations are required; in the case of only 1 battery station, 30-40 docks are needed, while in the case of 3 battery stations 15-20 docks per battery station are required. A considerable decrease in swap time leads to much smaller battery stations: 6 docks in case of 3 battery stations. These figures are based on a quick-charge strategy with a nickel-cadmium battery which, owing to the limited capacity, needs significantly more changes than the other two options (in the peak about once per hour, see Table 6.2). Therefore, if lead-acid batteries are used the number of docks can be considerably smaller. Given this large number of docks and corresponding driving times, the design of these battery stations is an important research aspect. However, before spending time on the design, it is necessary to determine whether the use of battery stations is an attractive option, compared to charge-rails. The number of batteries charging in

battery stations varies during the day, and is related to the transportation flows. From the number of batteries recharging at a battery station over 5 days, see Figure 6.2, the necessary charging capacity of the battery stations can be computed. This gives an indication of the required number of locations at which batteries can be recharged outside the AGV. According to Figure 6.2, the required number of charge stations is about 500.



**Figure 6.2.** Number of batteries recharging (lead-acid, 1 battery station)

The required number of batteries strongly depends on the chosen battery type. Quick charging a nickel-cadmium battery can significantly reduce the required number of batteries (by 300-400). The effect of the number of battery stations on the required number of batteries is a reduction of less than 10%. The positive effect of one battery station is reduced by an increase in the required number of AGVs, also because in this case less idle time can be used. The average remaining charge of a battery in absolute figures is not really different for the three battery types. This is not the case when we look at percentages, because there is a big difference between the capacities of the batteries. Part of the lost charge is caused by the chosen logistic control rules. Battery changes are performed as much as possible in idle time, for example when AGVs are waiting in front of a terminal. This leads to more frequent changes, with on average more remaining charge, but less influence on system performance. A large part of the lost charge is caused by the long distances between the terminals.

The number of battery changes largely depends on the net capacity of the battery and less on the number of battery stations. If there are three battery stations, differences in the number of battery changes per station can be explained by the unbalanced transportation flows. Since most AGVs travel between RTH and VBA, many changes take place at these two locations. Most changing occurs at VBA, and this can be explained by the fact that most AGVs travel loaded from VBA to RTH, and empty from RTH to VBA. Loaded trips require more battery charge, and therefore more often a battery has to be changed before a loaded trip.

Decreasing the battery swap time has big positive effects. A battery swap time of only 1 minute diminishes the performance loss, but the battery swap still has serious impact on system performance. Shorter swap times make the option of quick charging nickel-cadmium batteries competitive, because only a few extra AGVs are required to reach the demanded system performance. The result is that the required number of batteries is substantially less than for the other options (see Table 6.5). Notice that with a swap time of 1 minute, combining the battery swap with the loading/unloading operations at a dock could be considered. This might be technically complicated, but it would have positive effects on the logistic performance. The performance is comparable with the option of charge-rails, and no battery stations are required.

We conclude that all options have strengths and weaknesses from the logistic and financial points of view. In order to make a choice between the various options for energy provisioning, in the next section we present an approach for the preparation of a cost analysis. Because of the lack of sufficiently accurate cost data, the calculations are hypothetical.

#### **6.2.4. Structure for a cost trade-off**

The choice of one of the technical options, (a battery swap or charge-rails), for one of the battery types, lead-acid or nickel-cadmium, and the number of battery stations, cannot be based solely on logistical arguments. We present a way to assess the costs of the different alternatives, based on artificial cost figures. More research is required to estimate real costs, for example those of construction, vehicles and batteries.

**Table 6.6.** *Indicative costs for an automated transportation network*

<b>Costs (indicative)</b>	<b>Description</b>	<b>Unit</b>	<b>Price/unit (x1000)</b>
Battery	Lead-acid	piece	21.5
	Nickel-cadmium	piece	41.8
AGV		piece	150
Terminal	Loading/unloading dock	piece	75
Charging	Charging location battery	piece	12
	Charge-rail in tube	m	0.75
Battery station	Battery station (space)	m <sup>3</sup>	0.3
	Battery change dock	piece	75
Maintenance	Charge-rails, battery stations, charge locations, etc.		P.M.

The costs of batteries, additional AGVs, battery stations, and charge-rails are included. The costs of battery stations include those of docks, space for storing and charging batteries, and charge stations.

**Table 6.7.** *Input data for cost computations*

Data	Value
Life span lead-acid battery	40320 kWh
Life span nickel-cadmium battery	57360 kWh
Planning period	20 year
Interest rate on initial investments	4.0 %
Size of the battery stations	2 m <sup>3</sup> /charge location
Total length of charge-rails	17460 m

The life span of a nickel-cadmium battery is longer than the life span of a lead-acid battery, but the costs of using a nickel-cadmium battery are also higher. In Table 6.8 the average energy consumption per day is shown. For the option of charge-rails, the energy consumption is not exactly known, but it is estimated that it will be comparable to the situation with battery stations and a lead-acid battery.

**Table 6.8.** *Input data for costs computations*

Battery	Battery stations	Swap time (min)	m <sup>3</sup> battery stations	Gross capacity batteries (kWh)	Net capacity batteries (kWh)	Avg. energy consumption per day
Charge-rails, with lead-acid battery	0	n.a.	0	33.6	26.88	15820
Charge-rails, nickel-cadmium battery	0	n.a.	0	23.9	17.93	15820
Lead-acid	1	5	1028	33.6	26.88	15820
	3	5	1210	33.6	26.88	15841
Nickel-cadmium	1	5	1250	23.9	17.93	15935
	3	5	1414	23.9	17.93	16050
Nickel-cadmium, quick-charge	1	5	420	23.9	13.15	16452
	3	5	568	23.9	13.15	16412
Lead-acid	3	1	1230	33.6	26.88	15841
Nickel-cadmium	3	1	1454	23.9	17.93	16050
Nickel-cadmium, quick-charge	3	1	588	23.9	13.15	16412

Other input data, such as the required number of AGVs and batteries, is shown in Table 6.5. It might seem that battery costs are much higher for the option of a battery swap. However, we have to take into account that these batteries are not used continuously, while in case of charge-rails the batteries are used continuously. Therefore, the battery costs will not be very different, because the driven kilometers are more or less the same.



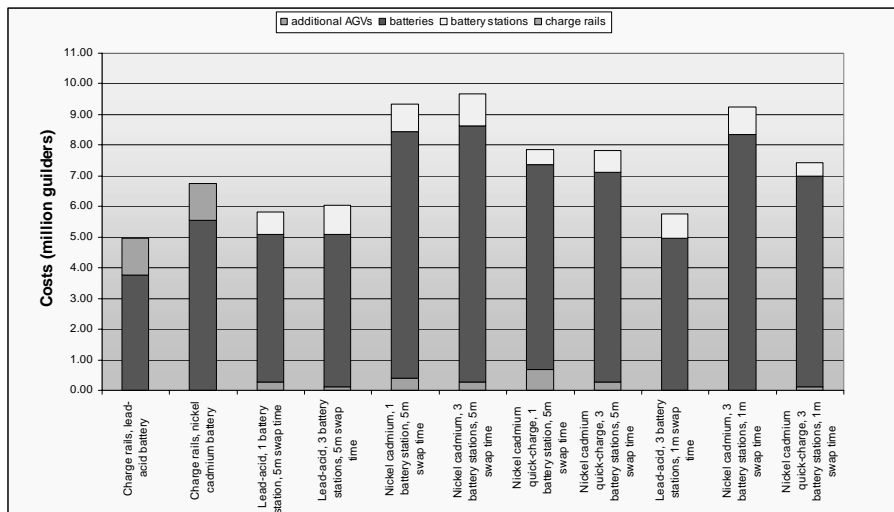


Figure 6.3. Cost comparison of the different alternatives

In Figure 6.3 we see the estimated costs for the different alternatives. The batteries are responsible for the largest part of the costs, followed by charge-rails, battery stations and additional AGVs. The costs for the nickel-cadmium options are higher because this battery is assumed to be more expensive than a lead-acid battery. The difference in cost between charge-rails or a battery swap is not very great for the lead-acid battery. In this indicative example, the difference between 1 and 3 battery stations is small. The extra costs of 2 battery stations are comparable with the costs of additional AGVs in the case of only 1 battery station. From a cost point of view, quick-charging a nickel-cadmium battery seems to be more attractive than a normal charging procedure. Overall, the charge-rails option is best, from both the logistical and cost points of view, but a lead-acid battery swap follows closely.

### 6.2.5. Conclusions

The main conclusions with respect to battery management are that:

- Battery constraints in automated transportation networks should be considered, because they can have a serious impact on the logistic performance and system costs, depending on the chosen option.
- The logistic control can be adapted to take battery constraints into account. The logistic control might be further optimized when some technical choices are made.
- In the case of a battery swap strategy, these battery swaps should take place as much as possible during idle time or waiting time.

In making a choice between the replenishment options, the logistical implications should be taken into account although of course, financial aspects should also be considered when choosing one of the replenishment options or battery types. Given

the results in the previous section and the fact that a lot of technical choices have yet to be made, in the rest of this chapter we assume that charge-rails are incorporated in the system.

For the OLS-case, we conclude that:

1. Given the relation between charging time and the driving time of a battery, replenishing the batteries inside the AGV at battery stations is not a real option. In the best case, the required number of AGVs increases with about 50%.
2. The best logistic performance can be obtained by using charge-rails, because no extra constraints are introduced.
3. With a battery swap time of 5 minutes, very large battery stations are required to reduce the effects on system performance. In this case the layout of the battery stations becomes an important design question. Small battery stations lead to long waiting times and consequently to an increase in the required number of AGVs or throughput times.
4. Decreasing the swap time to 1 minute leads to big improvements in performance. Battery replenishment without serious performance loss is then possible.
5. In the case of a battery swap option a network with several battery stations is to be preferred. Idle time can be exploited better and this leads to a smaller impact on the logistic performance. Because there are more battery stations fewer additional AGVs are required.
6. The additional number of batteries required for a battery swap strategy can increase to twice the number of AGVs.
7. Based on the logistic performance, we would prefer a lead-acid battery. The large capacity of this battery leads to less battery swaps. The use of a nickel-cadmium battery may be interesting when the battery swap time can be significantly reduced.

### **6.3 Failure management**

Failures can have serious consequences for the logistic performance of AGV systems, but no literature was found on this topic. Besides speed, flexibility and environmental issues, the reliability of an automated transportation network is very important. The resources most susceptible to failures are AGVs and docks. Such failures can have a serious effect on the logistic performance and therefore they cannot be neglected. A reason that we did not find something related in the literature might be the fact that in existing relatively small scale systems AGV failures can be quickly resolved. In an internal transportation system, a failed AGV can usually be removed from the system in a few minutes, because it can be reached rather easily. Furthermore, if there are only a few AGVs and preventive maintenance is carried out, AGV failures will be rare. In underground transportation systems, AGVs in underground tubes are difficult to reach and it takes a considerable amount of time

to get them out of the tube; meanwhile the tube is blocked. Moreover, with a large number of AGVs, failures are no longer rare, but are an almost daily occurrence.

The objective of this section is to describe the design of control methods that can handle the equipment failures of AGVs and docks. The model can be used to determine acceptable failure rates, i.e. failure rates for docks and AGVs that only slightly influence the overall system performance, compared to the case without failures. In the OLS-case, such acceptable failure rates derived from the simulations can be used as targets in the design of AGVs and docks. Another option is to determine the influence on system performance at a given failure rate. Electricity failures, personnel failures and software failures are not taken into account; we restrict ourselves to AGV and dock failures. Questions we want to answer with respect to failure management are:

- What are the implications of failures for the logistic control?
- What are the consequences for the logistic performance, depending on the location of the failure and time of failure?
- What are acceptable failure rates in the OLS-case?
- Can additional AGVs compensate the effects of failures?

In Section 6.3.1, we describe the consequences of AGV and dock failures and possible ways to handle these failures. In Section 6.3.2, we give the assumptions made in the OLS-case. To determine the effect of a specific failure, dock or AGV, given its failure location and the time of failure, we look at isolated disturbances in Section 6.3.3. Here the failure is dealt with as an isolated event and we determine the system recovery time given a specific failure. A model with integrated failures is used in Section 6.3.4 to assess the logistic performance of the OLS system with failures, and to determine acceptable failure rates. Furthermore, we investigate the degree to which introducing a higher speed for a recovery vehicle or additional AGVs can reduce the negative effects of failures. In Section 6.3.5 we investigate the robustness of two-way track control with respect to failures. In Section 6.3.6 we present the conclusions.

### **6.3.1. Model for failure management**

An AGV or dock failure influences the system for a certain amount of time. A failed dock cannot be used, which reduces dock capacity. A failed AGV possibly leads to late delivery of the cargo on the AGV, but it also affects the operations in a part of the system. AGVs blocked behind a failed AGV have to wait until the failed AGV is towed or pushed away. Such a failed AGV can block a tube, a part of the terminal or even an entire two-way track. Adequate and fast failure handling is therefore important. In this section, we describe the modeling and control of AGV and dock failures, starting with dock failures.

### *Dock failures*

In the case of a dock failure, the docking operation is stopped immediately. The dock control object notices the dock failure and notifies the terminal manager. The AGV that was involved in the dock operation is held up until the dock has been repaired. Other AGVs that are waiting at the dock might be able to drive around the AGV that is held up at the failed dock. The terminal manager has to take the dock failure into account in local vehicle management, order release, and the task allocation to docks. *Local vehicle control* takes action by no longer sending AGVs to that particular dock and rerouting AGVs already driving to the dock. *Order release* cannot release new transportation jobs to this failed dock and *task allocation to docks* cannot assign a new functionality to the dock. Transportation jobs already at the dock, or on their way to the failed dock, have to be recalled and sent to another dock.

Repairing a dock takes a certain amount of time. After repair, the dock will restart the loading/unloading operation it was working on at the time of failure. This operation requires the normal loading or unloading time. When the dock is unblocked, the terminal manager is notified, and new jobs can be planned for the repaired dock.

### *AGV failures*

Handling the AGV failures requires close attention. Failed AGVs should be repaired or taken out of the system as quickly as possible, because they block other traffic. Possibilities are to repair the AGV at the failure location or to bring the failed AGV to a repair station. Repairing an AGV in a tube is virtually impossible because there is almost no space around the AGV. Moreover, the repair times will be very unpredictable and other traffic is blocked for a long time. The other option is to bring the failed AGV to a repair station, towed by a recovery vehicle or pushed by other AGVs. In this case, choices have to be made with regard to the locations and number of repair stations. A terminal and a parking area are good locations for such repair stations.

Letting another AGV push a failed AGV might seem attractive, because the first AGV that encounters the failed AGV can push the failed AGV to a repair station. It seems that other traffic is not really influenced, but a failed AGV can still block a two-way track for a long time, while waiting for another AGV to push it away. Furthermore, there should always be an AGV behind every other AGV; otherwise, it is possible that there will be no AGV to push the failed AGV. Other difficulties arise from a technical viewpoint. The AGVs have to be equipped with more powerful engines to be able to push other AGVs, and it might not be possible to push a failed AGV at all, for example in case of a broken axle.

Recovery vehicles tow a failed AGV to a repair station. This means that special vehicles are required and that these vehicles drive part of the route against the usual driving direction to approach the failed AGV. Therefore, there should always be a

recovery vehicle in front of every AGV. In order to be able to reach the failed AGV, the tracks in front of the failed AGV should be freed of other traffic. All failed AGVs can be handled, because a failed AGV can also be lifted somewhat by a recovery vehicle, which might be necessary in case of a broken axle.

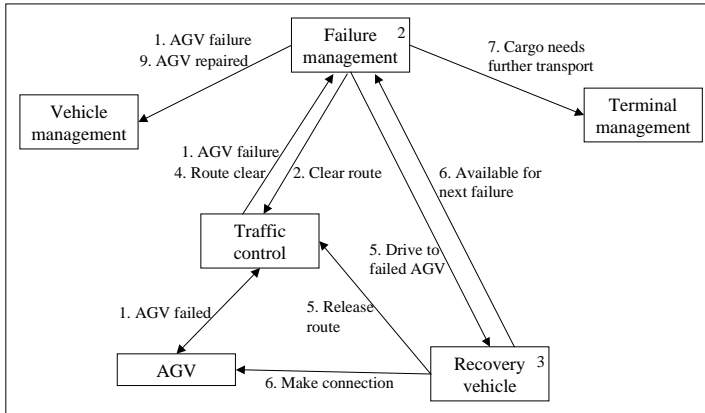
Both methods have their advantages and disadvantages, but the latter option is preferred. It seems that this solution is most likely to be used within the OLS project, because of the large number of AGVs in the network that have to be upgraded in case of the push approach. Therefore, we will investigate the effects of failures if a failed AGV is towed by a recovery vehicle. Once the failed AGV has been taken out of the system, it can be repaired. In the remainder of this section, we use the following general assumptions:

- A failed AGV will be recovered by a recovery vehicle, which tows the failed AGV to one of the repair stations.
- Recovery vehicles do not fail.
- There are sufficient recovery vehicles in the system, for example one at every terminal.
- A recovery vehicle can always reach a failed AGV, i.e. in case of two-way tracks there should always be a recovery vehicle at both ends of the two-way track.
- We do not explicitly take the repair capacity into account; the repair process is modeled by a repair throughput time that may consist of repair time and waiting time.

In general, the recovery of a vehicle is modeled by using the following steps:

1. Detecting an AGV failure
2. Selecting the appropriate recovery vehicle and determining the route of this recovery vehicle
3. Preparing the recovery vehicle
4. Clearing the route of the recovery vehicle
5. Driving of the recovery vehicle to the failed AGV
6. Towing the failed AGV to a repair station
7. Handling the load when the failed AGV was loaded
8. Repairing the failed AGV
9. Returning the repaired AGV to the system

Figure 6.4 shows the communication between the objects involved, that results from the activities described above. Some of the activities give rise to a new control object; for example the selection of a recovery vehicle is done by *failure management*.



**Figure 6.4.** Communication between the objects involved in failure handling

### 1. Detecting an AGV failure

When an AGV breaks down, it can take some time before the failure is actually detected and the failure handling can start. The detection time depends on the failure location. If the AGV can still communicate, it notifies traffic control that it has broken down. Otherwise, traffic control should detect the failure based upon the expected arrival time of AGVs at specific locations in the network. When an AGV is late at a specific point, it is necessary to check whether something has happened to it. In a terminal, a failure will probably be detected earlier than in a tube between terminals. Once *traffic control* knows about an AGV failure, *failure management* is informed about the location and identity of the failed AGV, and it can start handling the failure. *Failure management* notifies *vehicle management*, because the failed AGV cannot be used for a long time.

### 2. Selecting a recovery vehicle and determining the route

When a failure is detected, *failure management* has to determine which recovery vehicle will handle the failure. At the same time, the route of the recovery vehicle has to be determined. The problem is that other AGVs may be driving in front of and behind the failed AGV. We want to develop a strategy that can handle all AGV failures at all locations in the network. It may be possible that the recovery can be speeded up in very specific situations, for example when no AGVs are behind a failed AGV that is just outside a terminal. However, these situations will not occur frequently during peak hours and they require very specific and exact information about the system state. Usually this information will not be available or will not be accurate enough. In the chosen strategy, the failed AGV is always approached from the front. This means that a recovery vehicle drives part of the route against the normal driving direction. In order not to disturb other traffic, an attempt is made to travel as far as possible in the normal driving direction. At the exit of a two-way track a recovery vehicle can change its driving direction once the route is cleared.

The choice of a specific recovery vehicle depends on the exact location of the failed AGV. We choose the recovery vehicle that is closest to the failed AGV according to our recovery strategy. Once we have chosen a recovery vehicle, the route of the recovery vehicle is given, namely the shortest route, irrespective of the number of AGVs occupying the tracks. To reach the failed AGV, the recovery vehicle will drive as far as possible in the normal driving direction, and only in the last part against the driving direction. It is possible that an AGV breaks down when the recovery vehicle is working on another AGV failure. In this case, the AGV failure is placed in a backorder list for that particular recovery vehicle. Once the recovery vehicle is available again, it will start the next recovery from the backorder list. AGV failures are therefore handled on a first-come first-served basis. A possible improvement could result from the selection of the recovery vehicle that can reach the location of the failed AGV in the shortest time. This would take into account that a particular vehicle might already be working on a failure. We did not implement this last option. Once the route has been determined, *failure management* notifies *traffic control* that a particular route has to be cleared.

### *3. Preparing the recovery vehicle*

In addition to the detection time, we include a reaction time. This reaction time is required to prepare the recovery vehicle for the recovery operation. The reaction time is only relevant when no other activities, such as clearing the route, take more time than the reaction time. Therefore, the reaction time is mainly important in case of disturbances at a terminal.

### *4. Clearing the route of the recovery vehicle*

Before a recovery vehicle can start the recovery, the part of the route against the driving direction should be free of AGVs. This means that *traffic control* will not allow AGVs to enter this particular part of the route. AGVs that are still driving on the route drive to their destination and the route will automatically be cleared. Once the route is free, *traffic control* notifies *failure management* that the recovery vehicle can start driving toward the failed AGV. A possible way to reduce the time for clearing the route of the recovery vehicle is to redirect AGVs that are driving on the route. These AGVs could leave the route of the recovery vehicle at the first branch they encounter, which means that the route to their final destination has to be changed. This might speed up the failure handling, but the redirected AGVs are delayed because they no longer follow the original shortest route to their destination. This option was not implemented in our control.

### *5. Driving of the recovery vehicle to the failed AGV*

When the route of the recovery vehicle has been cleared, *traffic control* notifies *failure management*, and the recovery vehicle can start driving to the failed AGV. The speed of a recovery vehicle could differ from the speed of AGVs. This speed is one of the aspects that determine the time needed to recover the failed AGV. While the recovery vehicle is driving to the failed AGV, *traffic control* releases that part of the route the recovery vehicle has passed.

#### *6. Towing the failed AGV to a repair station*

When the recovery vehicle reaches the failed AGV, they are coupled. The recovery vehicle drives to the nearest repair station, leaves the failed AGV and receives a new destination from failure management. Another option is to select a repair station based on how much effect the route to the repair station has on other traffic. If the recovery vehicle drives more slowly than the other AGVs when towing a failed AGV, the recovery vehicle should leave a busy route as quickly as possible in order to minimize its effect on the other AGVs. We choose to send the recovery vehicle to the nearest repair station and from there to its home base.

#### *7. Handling the load when the failed AGV was loaded*

It is possible that the failed AGV was loaded so the cargo should be delivered to its final destination. We assume that it takes a specific time to unload the failed AGV and to prepare the cargo for further transport. It is possible that coincidentally the cargo is already at its destination. However, when the cargo is not at its destination, the cargo should be transported to the correct destination. *Failure management* will put this transportation job into the job list of the terminal. The transportation job will probably receive high priority, because it was already being transported to its destination and time was lost because of the failure.

#### *8. Repairing the failed AGV*

The failed AGV is repaired in the repair station but it will not be available for transport for several hours.

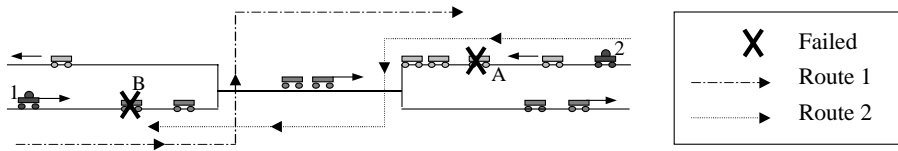
#### *9. Returning the repaired AGV to the system*

A repaired AGV should return smoothly into the system, i.e. the return of the AGV should be announced to the information system so that it can again be used for transportation jobs. The AGV returns in the system at the terminal at which it was repaired. The *local empty vehicle manager* is notified and can assign a new job to this AGV. Because the repaired AGV is always empty, it can be used directly for a new transportation job, can be assigned to a parking place, or sent to another terminal or parking area.

#### *Handling multiple AGV failures*

The recovery procedures described above work in the case of isolated disturbances, i.e. only one AGV breaks down, and in most cases of multiple disturbances. Nevertheless, there are situations that lead to a deadlock. It is possible that the route of the recovery vehicle is blocked by another failed AGV, see Figure 6.5. One recovery vehicle wants to follow route 1 to recover failed AGV A at the right. Route 1 is blocked by failed AGV B at the left side of the two-way track. This failed AGV also cannot be reached because failed AGV A blocks the route of the recovery vehicle (route 2). We have a deadlock.





**Figure 6.5.** Example of a deadlock situation in case of multiple failures

The two failures interfere with each other. Such situations can normally be solved rather easily by human intervention. In the example above, a possible solution is that recovery vehicle 1 (at the left side) takes the other tube, that has to be cleared first. Then failed AGV A can be removed, and the problem is solved. However, not all possible combinations of AGV failures can be easily generalized to one method for recovery. Theoretically, the possibility of 3 or more simultaneous AGV failures in the same part of the network cannot be excluded. With another recovery strategy, the deadlock depicted in Figure 6.5 might not have occurred, but in that case, a larger part of the system would have to have been cleared of traffic. In any case, these deadlock situations still have to be solved in our simulation model. Therefore, we introduce a special emergency method for these situations, which is put into effect when the failed AGV is not reached within a particular period of time. We assume that within this particular time a solution has been found by human intervention. After this maximum period, we assume that in one way or another, the failed AGV has been reached and that it will move to the nearest repair station, at a speed equal to that of the recovery vehicle. Notice that this emergency procedure will not be called upon very often when the failure rate is low. Only in the case of very high hypothetical failure rates, would this emergency procedure be frequently called upon. After an emergency has been handled, the recovery vehicle involved can start to cope with another failure.

### **6.3.2. Parameters for the OLS-case**

To assess the implications of AGV and dock failures, we used the layout with two-way tracks (Figure 6.1), with the general input data with respect to transportation flows (Table 3.2), AGV speeds, terminals and throughput time requirements, described in Chapter 3. We made the following assumptions for our simulation study:

- Failures are proportional with the operational time of AGVs and docks, i.e. AGVs and docks only fail when they are active.
- The time between two failures of the same object is exponentially distributed, where the mean value is varied in our experiments.
- There is a recovery vehicle on each terminal, combined with a repair station.
- The repair time of a dock is exponentially distributed with an average of 1 hour.
- The repair of an AGV takes 12 hours (deterministic).
- The reaction time is 1 minute, recovery vehicles are always ready to handle a failure.

- The emergency time, the maximum time until a failed AGV is reached by the recovery vehicle, is 7 minutes on a terminal, 20 minutes for a failure on a two-way track and 25 minutes for a failure at other locations. These times are based on a worst-case of the time it might take to free the route of the recovery vehicle plus the time for the recovery vehicle to drive toward the failed AGV.
- Suppose that a loaded, failed AGV is towed to a terminal that is not its original destination. The time required to unload the cargo and prepare it for further transport is 15 minutes, starting from the arrival of the failed AGV in the repair station.
- Batteries are charged by charge-rails; the batteries do not have to be changed and there are no battery stations in the model.

An AGV failure on a terminal is detected almost instantaneously because on the terminal communication is possible and personnel is present. When communication is also possible in the tubes, the failure can also be detected almost instantaneously there, but otherwise an AGV failure outside a terminal or parking area is not detected directly. Because there was no clarity on this issue at the time of our research, we chose to assume that there would be no communication inside the tubes. *Traffic control* knows the expected arrival time of the AGV at specific locations in the system (terminal, two-way track, etc.) and this information can be used to detect an AGV failure. When an AGV does not arrive at the specified time, or somewhat later (a safety margin), probably something has happened to the AGV and failure handling should be started. We therefore assume that the detection time is equal to the remaining driving time that is needed to cover a part of the tube (track). Several sensors could be located in the tubes to reduce these detection times.

### **6.3.3. Experiments with a single controlled failure**

To investigate the effect of a disturbance on system performance, we first analyzed the effect of an isolated disturbance. In fact, we controlled the conditions under which an AGV or dock fails (timing, location in the network), and investigated the effects of such a specific failure. The logistic performance was measured by using the following performance indicators:

1. the loss in service level on the day of failure (percentage of late orders).
2. the recovery time, i.e. the time the system needs to reach a state that is comparable with the state of the system without failure.

For the analysis of isolated disturbances, we designed a series of experiments for one of the OLS-layouts (Figure 6.1). We took the following aspects into account:

- a) Traffic intensity
  - Outside the peak (Tuesday morning between 9.30 and 10.00)
  - Start of the afternoon peak (Tuesday afternoon between 12.30 and 13.00)
  - In the middle of the evening peak (Tuesday evening 19.00 and 20.00)
- b) Speed of the recovery vehicle: 3, 6 and 12 m/s
- c) Location of the dock:

- At a busy terminal (RTH)
  - At a quiet terminal (AAS2)
- d) Location of the failed AGV in the transportation system:
- On a terminal (RTH): blocking the entry and on the middle of the terminal
  - On a one-way track (OWT), on a central location and between AAS1 and AAS3
  - On a busy two-way track (TWT), the one between VBA and AAS3
  - On a quiet two-way track, between AAS1/AAS3 and AAS2
- e) Position of the failed AGV in the convoy, i.e. first, second or one of the last.

We looked at the first and second AGV of a convoy, because these situations can be very different. When the first AGV breaks down, the recovery vehicle can start driving immediately, because the track in front of the failed AGV is already empty. When the second AGV breaks down, the first AGV has to clear the route, and this can take a while. In the meantime, this failed AGV is blocking the rest of the convoy, which might have serious consequences. For other AGVs in the convoy this situation occurs in a similar way, therefore we restricted the analyses to the 2<sup>nd</sup> AGV.

**Table 6.9.** *Twelve failure situations*

Case	Description
0	Base case: no failure
1	Dock on RTH, failed during one hour
2	Dock on AAS2, failed during one hour
3	AGV blocking entrance RTH
4	AGV in the middle of RTH
5	AGV on OWT AAS3-AAS1, first one in convoy
6	AGV on OWT AAS3-AAS1, second in convoy
7	AGV on OWT AAS3-AAS1, last in convoy
8	AGV on TWT AAS1-AAS2, first in convoy
9	AGV on TWT AAS1-AAS2, last in convoy
10	AGV on TWT VBA-AAS3, first in convoy, directly after entering the tube
11	AGV on TWT VBA-AAS3, second in convoy, directly after entering the tube
12	AGV on TWT VBA-AAS3, one of the last in the convoy

Summarizing, we had twelve different failure locations for AGVs and docks (c, d and e), see Table 6.9. Every failure situation was combined with three different types of traffic intensity (point a) and three different recovery speeds (point b). Therefore, we had 108 different controlled disturbances. Notice that the experiments were performed for a peak day (Tuesday) in an average week, so that the worst case was included.

### *Results of the controlled failures*

In most cases, the impact of a single failure on the overall system performance in terms of service level was rather marginal. The decrease in service level is usually less than 1 percent. However, in some cases we saw a serious negative effect. This occurred especially if an AGV failed on the two-way track between VBA and AAS (cases 10-12). When the traffic intensity is high and the recovery speed low the loss can increase to 10 percent for a single bad failure. There also seemed to be an effect caused by the recovery speed, especially when the recovery vehicle moves slowly (3 m/s) leading to much delay and a poor performance. The additional improvement in performance brought about by the use of a recovery vehicle moving at a higher speed (12 m/s instead of 6 m/s) seems to be small.

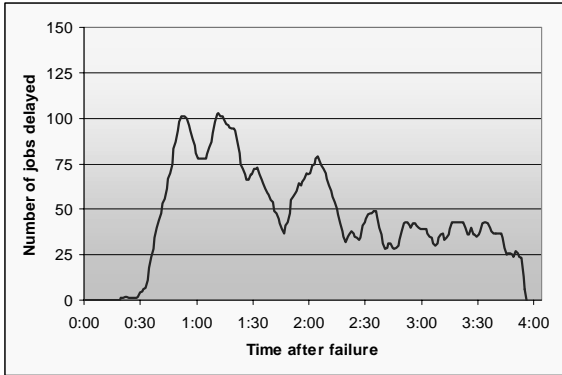
The question is how robust these results are; they only represent one disturbance on a given day with a given stochastic realization. To investigate the robustness, we simulated similar disturbances 4 times. Thus, we obtained an indication of the reliability and reproducibility of the results.

**Table 6.10.** *Variation in performance loss*

Case	Description	(i)	(ii)	(iii)	(iv)
10	AGV on TWT VBA-AAS3, 1 <sup>st</sup> in convoy	1.7%	1.3%	0.6%	4.5%
11	AGV on TWT VBA-AAS3, 2 <sup>nd</sup> in convoy	10.1%	2.7%	2.0%	8.1%

Table 6.10 shows the variation in performance loss for the two most serious cases, case 10 and 11. The same failure was simulated with four different sets of seed numbers for the random generators (variants (i)-(iv)). We saw that the effect of a similar disturbance could vary greatly, especially for case 11. Therefore, these isolated disturbances only gave an indication of the possible performance loss in particular situations.

Despite the stochasticity in the effects of a controlled disturbance, it is interesting to see how long it takes until the transportation jobs are no longer delayed, compared to a run without disturbance. We looked at the number of delayed orders some time after the disturbance. The number of delayed orders was derived from the results of the same day without a disturbance. Most cases only had a small effect on the number of delayed orders. The system did not really run behind schedule and thus there was no real recovery time. With the disturbance, the total process developed somewhat differently, but was not really worse. In the most serious case (case 11(i)), the number of delayed orders was really large. In the evening peak, there was almost no time to recover from the failure. Only after 4 hours, when it was quiet again, did the system seem to have recovered (see Figure 6.6).



**Figure 6.6.** Recovery time case 11, in the middle of the evening peak, recovery speed 6 m/s

We also looked at the distribution of the delays of all orders after the disturbance. About 20% of the orders were still on time, while 95% of the orders had a delay of less than 30 minutes. There were also orders that were delayed by 50 minutes or more. Again, we see that this particular failure (case 11) had serious consequences on system performance. The question is how often these serious disturbances occur in an automated transportation network and to answer this question, an integral analysis is needed.

#### *Conclusions for controlled disturbances*

Experiments with isolated disturbances showed that the effect of a particular disturbance is unpredictable and depends on many factors. Nevertheless, we can already give some preliminary conclusions with respect to failures in automated transportation networks:

1. Most disturbances only have a marginal effect on system performance, but serious performance loss can occur in the case of a failure in a busy period at a bad location. The recovery time of the system can accumulate to several hours.
2. The disturbances on a terminal (AGVs and docks) usually have only a limited effect. A failed AGV on a terminal can be taken out of the system relatively quickly (within several minutes).

#### **6.3.4. Experiments with an integrated failure model**

For integrated failures, we investigated the following variants:

- The Mean Time Between Failures (MTBF) of AGVs is varied between 100 and 1500 operational hours.
- The Mean Time Between Failures of docks is 10, 25 or 100 operational hours.
- The speed of the recovery vehicle is 3, 6 or 12 m/s.

Failure behavior of docks and AGVs was first investigated separately. Afterwards, AGV and dock failures were integrated, and interaction effects could be

investigated. The simulation ran for a peak day with a run-length of 31 days, and the first day was deleted from the results to account for start-up effects. Furthermore, experiments were performed for an average week (Saturday to Friday), to see whether the main conclusions from a peak day in an average month still held. For an average week the run-length is 71 days, i.e. 10 weeks.

#### *Analysis of dock failures*

From the transportation flows, we can easily derive a relation between the average number of failures and the failure rate of the docks, without simulation. A MTBF of docks of 25 hours is equal to, on average, 3.7 dock failures per day, of which 0.7 at AAS, 1.7 at RTH, and 1.3 at VBA. Dock failures have only a marginal influence on the throughput times, even in case of frequent disturbances (MTBF=10, on average 9.1 failures per day). Besides throughput times we also looked at the effect of failure behavior on the service levels, i.e. the number of orders that is on time at its destination. From Table 6.11 it can be concluded that dock failures only have marginal effects on system performance.

**Table 6.11.** *Effect of dock failures on the service levels (peak day in average week)*

MTBF	TOTAL
--	97.2
100 hour	97.0
25 hour	97.0
10 hour	96.4

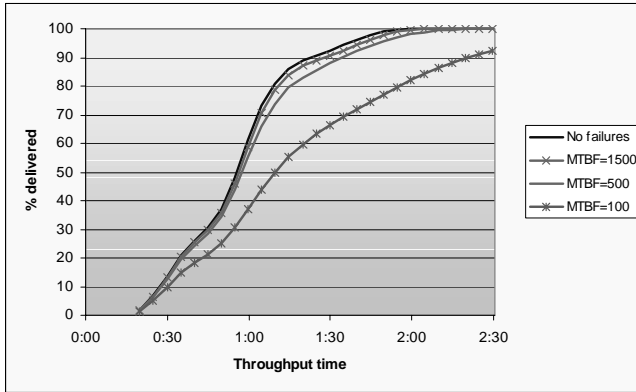
#### *Analysis of AGV failures*

From the simulation results, we can compute some statistics with respect to AGV failures, such as the average number of AGV failures per day. With an MTBF of 1000 hours, there are on average 1.7 AGV failures per day. On a peak day, this number can be three times larger. Furthermore, we can make a distribution of the failures over the different parts of the system, i.e. parking area, terminals, on a one-way track and on a two-way track. We can also register the average time required to solve an AGV failure. Table 6.12 shows that more than 80% percent of the failures occur outside a terminal or parking area. These failures are precisely the situations that, on average, require a long recovery time.

**Table 6.12.** *Indication of the handling time of AGV failures, dependent on location*

Location	% failures	Average handling time (min.)
One-way track	44%	18
Two-way track	38%	20
Terminal	12%	3
Parking area	7%	7

We investigated the effect on throughput times of an MTBF from 100 to 1500 operational hours. The speed of the recovery vehicle was equal to 6 m/s. We compared the results with a simulation experiment without failures, see Figure 6.7. Even with an MTBF of 300-500 hours, the increase in the throughput times stays within limits. For smaller MTBF, i.e. 100 or 200 hours, the results are poor; the service level is below 90%.



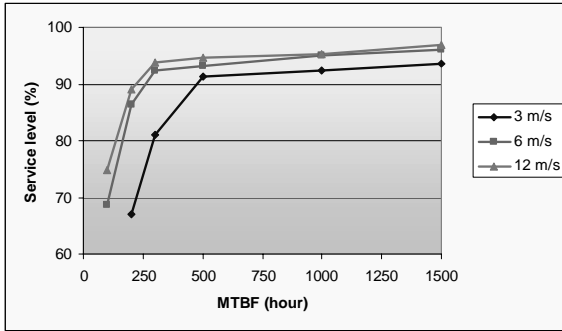
**Figure 6.7.** Effect of AGV failures on throughput times (peak day in average week)

In Table 6.13, we see the effect of AGV failures on the service levels for a peak day in an average week and a recovery speed of 6 m/s. An MTBF of 1000 hours or more gives acceptable results since the service level is greater than 95%.

**Table 6.13.** Effect of the MTBF of AGVs on the service levels (peak day in average week)

MTBF	TOTAL
--	97.2
1500 hour	96.2
1000 hour	95.2
500 hour	93.2
300 hour	92.3
200 hour	86.4
100 hour	68.7

The effects of AGV failures may be influenced by the speed of the recovery vehicle. In Figure 6.8, we show the service levels for three different speeds of the recovery vehicle (3, 6 and 12 m/s), and for different values of the MTBF.



**Figure 6.8.** Service levels dependent on MTBF of AGVs and speed of the recovery vehicle

We can explain the different effects. The speed of the recovery vehicle influences the driving time from the terminal to the failed AGV and the driving time from the failure location to the terminal (repair shop). At a speed of 3 m/s it takes a long time for the recovery vehicle to reach the failed AGV. While the recovery vehicle is driving back to the terminal, all AGVs driving behind it can only travel at 3 m/s instead of 6 m/s. The advantage of a speed of 6 m/s is that the failed AGV is reached more quickly, and while driving back to the terminal all other AGVs can drive at their maximum speed. At a speed of 12 m/s, the recovery vehicle reaches the failure location more quickly, but while it is driving back to the terminal all AGVs drive at 6 m/s. In consequence, the difference between the performances at 3 m/s and at 6 m/s is larger than that for speeds of 6 m/s and 12 m/s. An additional advantage of a driving speed of 12 m/s is that the recovery vehicle is ready to attend another failure sooner, which happens more often with small MTBF.

#### *Interaction of AGV and dock failures*

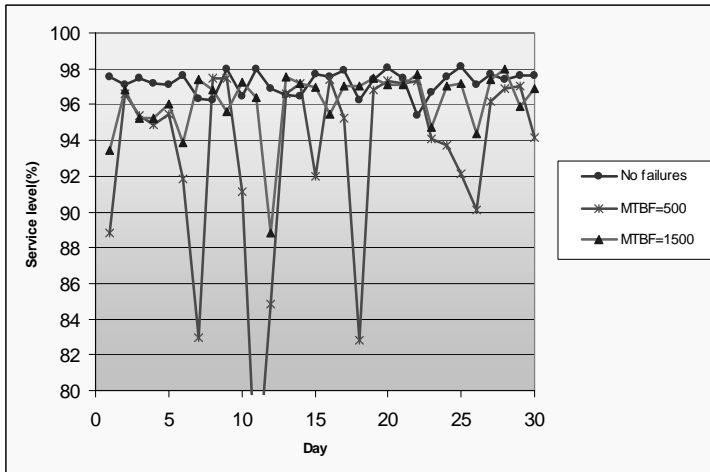
In this section, the AGV and dock failures are incorporated into one model. Because the effects of dock failures seem rather small, we chose a rather high failure rate for docks (MTBF=25 hour). We wanted to investigate whether the conclusions of the previous sections still held. We chose an MTBF for AGVs varying from 500 to 1500 hours and determined the effect on the service levels. The speed of the recovery vehicle was 6 m/s. The results will be compared with a simulation run without disturbances.

**Table 6.14.** Effect of AGV failures on service levels (peak day in average week)

AGV MTBF	Dock MTBF	
	--	25 hour
--	97.2	97.0
1500 hour	96.2	95.4
1000 hour	95.2	95.3
500 hour	93.2	92.7



Table 6.14 suggests that there is a small interaction effect between AGV and dock failures. The performance loss in the case of simultaneous failures is somewhat larger than for both failures separately. It is also interesting to see how the service levels fluctuate from day to day, instead of only looking at an average number. The results are shown in Figure 6.9 and indicate the daily service levels for 30 peak days in an average week, depending on the MTBF of AGVs (500 or 1500 hours) and an MTBF for docks of 25 hours.



**Figure 6.9.** *Fluctuations in service levels (peak days in average weeks)*

In the case of low failure rates (large MTBF), the service levels are not only much larger, but also much more stable due to less failures. Nevertheless, also with a high MTBF bad days can happen. Even with an MTBF of 1500 hours, the service level is below 90% for one day. An AGV failure at a less accessible location and a busy time can lead to a performance decrease of 10% (see isolated failures). Such a serious disturbance can occur with all positive failure rates.

It may be possible that the negative effects of failures are partially or entirely compensated by additional resources (AGVs or docks). Extra AGVs might lead to a faster recovery of the system. We looked at the effect of extra AGVs on the service levels. Because of the marginal effect of dock failures on system performance, we did not look at the possibility of extra docks. Table 6.15 shows the results for different numbers of AGVs.

**Table 6.15.** *Service levels dependent on the number of extra AGVs*

MTBF AGV	360 AGVs	+10 AGVs	+20 AGVs
--	97.2		
1500 hour	95.4	96.0	95.8
1000 hour	95.3	95.6	94.6
500 hour	92.7	94.4	94.8

The reduction in system performance resulting from failures can only be compensated slightly by extra AGVs. One reason is that more AGVs automatically lead to more failures so in this case increasing the number of AGVs would not really be useful.

*Effects for an average week (instead of a peak day in an average week)*

In this section we analyze the performance in an average week rather than just a peak day in such a week. We use an MTBF for docks of 25 hours and a speed of the recovery vehicle of 6 m/s in all cases. Table 6.16 shows the relation between the MTBF of AGVs and the service level in an average week. We see that the effects in an average week are much smaller. Even an MTBF of 500 hours leads to acceptable results. The service levels fluctuate heavily between the different days (cf. Figure 6.9). This is mainly due to the differences in transportation flows. Of course the busiest days show the lowest performance.

**Table 6.16.** *Effect of AGV failures on throughput times (all days in an average week)*

AGV MTBF	Service levels						TOTAL
	AAS- RTH	AAS- VBA	RTH- AAS	RTH- VBA	VBA- AAS	VBA- RTH	
--	99.9	99.8	100.0	99.4	99.7	97.0	98.6
1500 hour	99.5	99.3	99.5	99.1	99.2	96.6	98.2
1000 hour	99.5	99.5	99.6	98.8	98.4	95.4	97.7
500 hour	99.5	99.6	99.5	98.6	99.2	95.1	97.5

### **6.3.5. Robustness of two-way track control with respect to failures**

In the previous sections we used the adaptive control rule without look-ahead for two-way track control (cf. Chapter 5). This rule only takes the information on queue length into account, and is therefore probably less susceptible to failures. Failures lead to unreliable data on future arrivals, and given unreliable information, the adaptive control rule with look-ahead and the DP-rule might perform worse and take a bad decision for the actual situation. We investigated this effect in the system with one two-way track (cf. Chapter 3, Figure 3.3), with a length of 2800 meters. The available information horizon was about 900 meters, equal to 2.5 minutes, given a

speed of 6 m/s. We used an MTBF of 500 hours for AGVs and an MTBF of 100 hours for docks. For the periodic control rule we used a switchover time of 12 minutes. In Table 6.17 we see the robustness of the two-way track control rules for these failures.

**Table 6.17.** *Robustness of the two-way track control rules, service level in case of failures*

Control	No failures		Failures	
	230 AGVs	250 AGVs	230 AGVs	250 AGVs
PCR	90.4	97.6	89.0	96.6
AR <sub>LOC</sub>	95.8	98.0	94.3	97.9
AR <sub>LA</sub>	94.3	97.8	79.8	91.2
DP	97.2	98.4	94.4	97.9

The inclusion of failures has effects on the logistic performance, although these effects are smaller than in Section 6.3.4. This is due to a different system layout with only one two-way track and a shorter length, which results in less AGVs. The adaptive control rule with look-ahead is very sensitive to inaccurate information. When this rule is applied the performance of this rule drops seriously. The poor performance is partly due to the short information horizon; this rule was designed for a long information horizon. For the other control rules, PCR, AR<sub>LOC</sub> and DP, the performance loss is limited to about 1%. The periodic control rule does not use any information, and therefore is not influenced by inaccurate information; the loss of performance is entirely due to the failure process itself. More or less the same holds for the local adaptive control rule. For the case under consideration the information about queue lengths will be rather accurate, and therefore the periodic control rule is not really influenced. The dynamic programming rule is less susceptible to inaccurate information than the look-ahead rule, but still we see a serious effect in case of 230 AGVs and failures. The performance of the DP-rule drops to the level of the local adaptive control rule. Therefore, the periodic control rule and the local adaptive control rule are robust with respect to AGV failures; these rules are not affected by inaccurate arrival information. We can conclude that given its robustness, the simplicity and low computational costs a local adaptive control rule seems well-suited for a system with failures.

It is important to mention that the performance of the information-intensive rules might be improved by notifying two-way track control about a failure, in which case two-way track control might switch to a local adaptive control rule. In such a case, it would temporarily not use the probably inaccurate information.

We can also compare the results without failures with the results from Chapter 5. The ranking of the control rules is almost similar to the one found in Chapter 5. The periodic control rule performs worst, and the dynamic programming rule shows the best results; significantly better than the rest. The adaptive control rule with look-ahead shows worse results than the local adaptive control rule, which can be a result

of the short information horizon. The dynamic programming rule appeared to be less sensitive with respect to the information horizon (cf. Chapter 5).

### **6.3.6. Conclusions and further research**

Here we highlight the most important conclusions from the previous sections:

1. Failures cannot be neglected in modeling an automated transportation network; in the OLS-case failure handling is a daily activity.
2. Logistic control can be adapted to handle AGV and dock failures.
3. In the OLS case, AGV failures have more impact than dock failures.
4. Many disturbances have little effect on system performance, but one unfortunate failure can result in a performance loss on one day of 10%.
5. Failures on terminals have marginal effects on system performance because they can be solved in little time.
6. Additional AGVs can only slightly compensate for the loss in system performance; the introduction of failures always leads to a lower logistic performance.
7. The speed of the recovery vehicles should at least be equal to the speed of the AGVs. A lower speed is too slow and the effect of a higher speed seems limited.
8. For the OLS case, the AGV failure rate should not exceed once per 500 active hours in order to preserve acceptable logistic performance.
9. A local adaptive control rule for two-way track control seems most robust with respect to disturbances. The dynamic programming rule might still be the best performing rule after some slight modifications.

## **6.4 Restricted storage capacities of terminals**

So far, we have neglected capacity restrictions on storage capacity at the terminals. Besides storage capacity restrictions, other capacity restrictions also exist (docking and parking places). The latter two capacities are included in the simulation model, although the logistic control does not take these restrictions into account because they do not seem to cause an actual problem. AGVs can wait at the entrance of the RTZ and VBA terminal when there is no place inside. Only at the small Schiphol terminals is there a lack of parking space. When this appears to be a bottleneck, these capacities should also be taken into account in the control. The consequence of the relocation of the rail terminal to Schiphol Airport (cf. Chapter 3, Figure 3.3) was that the storage space had to be limited because of very high costs and limited space. Therefore, these storage capacity restrictions should be taken into account. In this section we want to investigate the effects of limited storage capacities and adapt the logistic control to take these constraints into account. The OLS-layout with the rail terminal near Schiphol is used in an example.

In the remainder of this section we use the following assumptions:

- The terminals have separate in-buffers and out-buffers, each with their own capacities.
- Cargo remains in the out-buffer until the due time; at the due time the cargo leaves the terminal by truck, train or plane. So space is occupied in the out buffer from delivery at the destination terminal until the due time.

#### **6.4.1. Adjustments in logistic control**

Before going into more detail, we can state that the consequences of storage capacity restrictions probably are:

- More just-in-time control, which leads to less opportunities for peak shaving, so probably more AGVs will be required.
- More coordination between the terminals is required.
- Cargo should be delivered in a specific order (order of departure from the terminal of destination = departure of trains or planes).
- Changes in order release: *order release* has to ask permission of the destination terminal to release a specific transportation job.

A terminal with restricted storage capacity has to control the amount of cargo in storage. Cargo should arrive more or less in order of due time. A specific transportation job with low priority (e.g. due time in 1.5 hours) should not occupy the storage space of a transportation job with high priority. In order to control the storage on the terminal, the terminal manager needs to communicate with other terminal managers. All terminal managers that want to start a transportation job, with another terminal as a destination, have to ask permission to start the transportation job. If there is insufficient capacity, the destination terminal does not give permission. When the destination terminal is able to receive new orders, it gives a signal to all terminals that new dispatch requests are allowed. To this end, each terminal manager maintains a storage occupation profile, describing the expected storage utilization in the next period (say 1-2 hours) based on expected load arrivals and departures.

The acceptance procedure by the destination terminal works as follows. Assume that the total storage capacity on terminal  $i$  is limited to  $C$ . The terminal manager can obtain information about all known transportation jobs that have terminal  $i$  as a destination, together with the due times of these jobs. For the first  $C$  transportation jobs on this list, sorted on due time, a place should be reserved in the cargo storage. When one of these  $C$  jobs leaves the storage (exits the system), another job can be unloaded. Therefore, when a request arrives from order release at terminal  $j$  to start a transportation job  $k$  with destination terminal  $i$ , the terminal manager has to determine:

- the number in storage ( $S$ );
- the handling time for job  $k$ ,  $\tau_{ji}^k$ , assumed to be deterministic;
- the number of departures from storage before time  $t + \tau_{ji}^k$ :  $N_d(t + \tau_{ji}^k)$ ;

- the number of transportation jobs that have higher priority than job  $k$  (higher on the sorted list) and have not yet arrived:  $N_a(k)$ .

Transportation job  $k$  is allowed to start when:  $C - S + N_d(t + \tau_{ij}^l) - N_a(k) > 0$ .

The remaining capacity at time  $t$  plus the capacity that comes available before the arrival of job  $k$ , minus the capacity that has to be reserved for higher priority transportation jobs, should be larger than 0. Otherwise, all capacity is reserved for transportation jobs with shorter due times. This means that orders do not have to be delivered exactly in order of due time, as long as a place in storage is available for all (known) jobs with higher priority (shorter due time). It can still happen that a transportation job is delivered too early, because the travel times are not completely deterministic. In that case, the AGV has to wait at a parking place or at a dock, until a storage location is available and it can be unloaded. A two-way track makes it difficult to estimate the travel times. In this case, a safety margin should be included to ensure that a transportation job is delivered at its destination before its due time.

The terminal manager of terminal  $i$  will notify the terminal manager of terminal  $j$  whether transportation job  $k$  can be released. If the request is rejected, transportation job  $k$  should be released at some other time in the future. The only times at which this is possible, is at moments when capacity comes available on terminal  $i$ . Therefore, when capacity becomes available on the terminal, terminal manager  $i$  will notify all other terminal managers. At that moment these terminal managers can request permission to release transportation jobs for destination terminal  $i$ . The communication can be reduced by informing only those terminal managers that are allowed to release transportation jobs. This will not affect the outcome of the planning procedure.

The approach presented here works when only one terminal has a storage capacity restriction. If several terminals have storage capacity restrictions, this method is no longer guaranteed to work. It is possible that one terminal needs to send cargo to another terminal, but that the receiving terminal has no cargo space available at that moment. Additional control rules are required to solve these conflicts. Possibly other terminals with sufficient storage capacity could be used for temporary storage or even a special storage facility could be introduced for this function.

#### **6.4.2. The OLS-case**

The methods explained in the previous sections could be applied to all three system designs. We chose the last system design, with internal transportation on Schiphol Airport, but with the two-way track replaced by two one-way tracks, in order to have (more or less) deterministic travel times. Failures are included in the model, with an MTBF of 500 hours for AGVs and 100 hours for docks, both exponentially distributed. The repair time for an AGV is 12 hours (deterministic), and the repair time for a dock is exponentially distributed with an average of 1 hour. In all experiments so far have we assumed that the storage space for cargo at the terminals

is unlimited. At Schiphol Airport the investment costs are very high and therefore the terminals should be made as small as possible. To get an idea of the size of the storage facilities on the terminals, we first measure the storage distribution per location, including the in-buffer and out-buffer, without taking into account finite storage capacities.

Figure 6.10 shows the distribution of the storage capacity used on terminals RTZ and VBA. We show the percentage of time that the storage capacity used is lower than or equal to the amount shown on the horizontal axis. It turns out that a storage utilization of more than 200 TU is rare at RTZ (about 10% of the time). At VBA for 95% of the time there is less than 150 TU in the cargo storage. For most of the time the storage capacity used at VBA is much lower than the storage utilization at RTZ. This is due to the logistic control, in which cargo is transported as quickly as possible to handle most transportation jobs before the peak starts (peak shaving). Peak shaving reduces the required number of AGVs and it reduces the negative effects of two-way tracks. Now suppose that we restrict the storage capacity at RTZ. In consequence of such a storage capacity restriction at RTZ, the storage usage at VBA and AAS will rise. The total amount of cargo in the system remains the same, so it is just a matter of balancing loads over buffers.

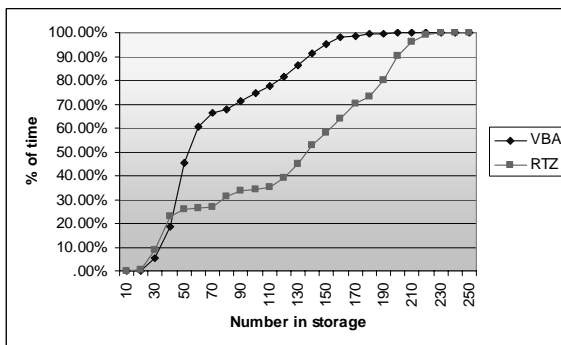


Figure 6.10. Storage utilization, cumulative percentages in 2020, Tuesday with 250 AGVs

### 6.4.3. Effects of limited storage capacities in the OLS-case

We used the OLS-case to investigate the effects of limited storage capacities. We assumed that the incoming and outgoing storage capacity at RTZ is equal to 80 TU, the choice of which was based on the number of docks in the terminal and the available space at each dock. The terminal at RTZ contains 16 docks, and at each dock there is space for 5 outgoing TUs and 5 incoming TUs. For a technical solution with respect to docking technology, refer to Rijssenbrij et al. (2000). The other terminals do not have storage capacity restrictions. We distinguished an incoming and an outgoing flow; the outgoing flow departs with the train and the incoming flow arrives with the train and is transported to other terminals in the OLS-system.

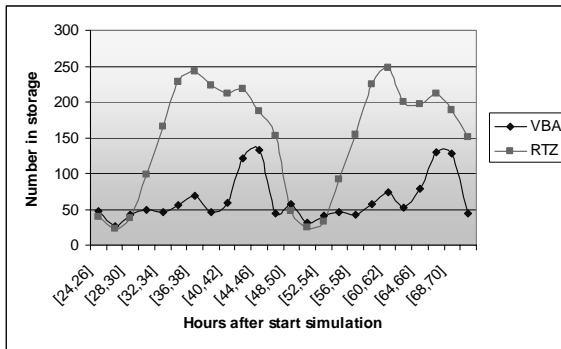
Therefore, in total there could be 160 TU at RTZ. In Table 6.18 we show the service levels with and without capacity restrictions for different numbers of AGVs.

**Table 6.18.** Service levels, layout with internal transport on Schiphol Airport, Tuesday 2020

Number of AGVs	With capacity restriction	Without capacity restriction
150	53.2	84.8
160	75.2	99.1
170	84.4	99.8
180	98.8	99.9

The storage capacity restriction on RTZ results in a requirement of 20 extra AGVs to reach a similar service level. This can be explained by the fact that the possibilities for peak shaving are limited. All orders to RTZ are planned closer to their due time. This makes the system also more vulnerable for disturbances. In the case of a failure, many orders will be late because they were already close to their due time. Therefore, service levels will fall.

Comparing the throughput times makes clear that the throughput times on the route to RTZ rise strongly after the introduction of limited storage capacity. Orders have to wait longer at AAS or VBA until there is space at RTZ. Besides throughput times, it is interesting to see what happens to the storage utilization at the different terminals. In Figure 6.11 we see the total number of TU that is stored at VBA or RTZ at a particular time *without* limited storage capacity.



**Figure 6.11.** Storage without capacity restriction

Without capacity restriction, the maximum number of TU at RTZ is 250 (in- and outgoing). At VBA, the number of TU in storage stays below 150. Figure 6.12 shows that the introduction of capacity restrictions led to a shift of cargo in storage from RTZ to VBA. At RTZ, the total number of TU does not exceed 100 (80 for the outgoing flow). The storage levels at VBA increase to more than 200 TU at certain times. In addition to the rise at VBA, the storage utilization at AAS will also rise.



The required storage capacity at some of the AAS terminals might double as a result of the limited storage capacity at RTZ.

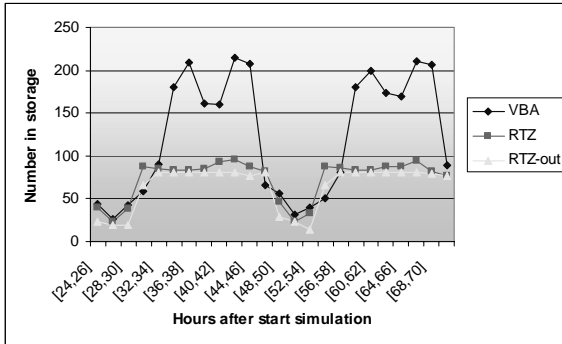


Figure 6.12. Storage with capacity restriction

#### 6.4.4. Conclusions

In this section we showed a first approach to the incorporation of storage capacity restrictions in the model. The logistic control was adapted to take these restrictions into account. These adaptations work in case of rather deterministic travel times, together with storage capacity restrictions at only one terminal. Nevertheless, the principles explained here can also be used with storage capacity restrictions at several terminals. Owing to capacity restrictions at several terminals, more coordination is required between the terminals. If dock and parking capacities were also incorporated in the logistic control, even more coordination would be required. In that case there would be so many restrictions that a scheduling approach would seem to be more appropriate. In Chapter 4 we showed a serial scheduling approach for the vehicle management problem. It might be also possible to incorporate these capacity restrictions in such a method.

A disadvantage of the just-in-time principle is that the storage capacity utilization at the other terminals increases. A possible solution is that the customers also deliver their goods to the OLS just-in-time, and not 2 hours before train departure. During these 2 hours, the cargo has to be stored somewhere in the OLS system. An option is to include a specific buffer facility in the network. Cargo could then be stored temporarily at this facility when there was no space left at the terminals. Additional activities could take place at such a facility, such as the consolidation of cargo according to flight number or train compartment. A disadvantage of such a facility is the additional number of transportation movements and the extra handling. Another disadvantage of just-in-time control is that it becomes difficult when travel times are stochastic. Long two-way tracks cause great variations in travel times. The question is whether these travel times can be predicted. Safety margins should probably be

incorporated to ensure that AGVs arrive at their destinations before the due time of the transportation jobs. The chance that an AGV arrives at its destination too early will also increase. More research is required to investigate whether two-way tracks can be incorporated in the previously described scheduling method. Ultimately, this would lead to a global optimization model in which we depart from the local control concept.



# Chapter 7

## Conclusions and further research

### 7.1 Introduction

Recent developments in the area of automated transportation networks provided the motivation for this research. In Chapter 1 we presented the research goal and the research questions to be answered in this thesis (Section 1.2.2). Let us now reflect on how the research questions have been answered. To start with we note that all methods considered in this thesis conform to the real-time constraint stated in the research goal. Furthermore, we introduced several new elements in our research. Chapter 2 described the object model designed for automated transportation networks, based on a local control concept (*cf. research question 3*). This object model was the basis for the development of an object-oriented simulation model in eM-Plant, which was used for experiments with control methods and system layouts, described in later chapters. The OLS-case, described in Chapter 3, is one that was used to test the different control methods and options. In Chapters 4, 5, and 6, the research focused on the design and evaluation of control methods for several important aspects of automated transportation networks. Vehicle management and two-way track control relate to the primary process, and for these primary objects we developed several planning and control variants. We also developed control objects for the secondary processes, such as battery management and failure management, in Chapter 6. In that chapter, we also examined the impact of storage capacity restrictions on the system performance.

Summarizing, we developed a flexible logistic control structure that can be used for automated transportation networks. Some of the developed control methods may also be useful in a broader context, such as fleet management problems (Chapter 4) and traffic applications (Chapter 5). A simulation library was designed to quickly construct different system layouts and evaluate alternative control methods. These products proved very valuable in the OLS-case. We used the simulation library to determine the logistical consequences of the ideas from several groups within the OLS project. For the OLS-case we showed that two-way tracks are feasible from a logistic point of view, resulting in significant savings, but with serious consequences for the logistic performance and resource requirements. Simulation results were also important input for the energy study and financial computations. The simulation library is still used in the OLS project.

Turning to the new elements and answers to the research questions, in Section 7.2 we discuss the conclusions relating to the topics discussed above. Topics for further research are discussed in Section 7.3.

## 7.2 Conclusions

*Object model and simulation model (research questions 1, 2, 4)*

In Chapter 2 we introduced an object model for automated transportation networks. In this object model, we made a clear distinction between physical, information and control objects according to a logistic modeling framework. Furthermore, we made a classification according to primary and secondary processes, transport and logistic activities, and flow, handling and storage objects. By defining the control activities we answered research questions 1 and 4. We also described the performance information that can be used to evaluate control methods. Performance measures to evaluate several methods for several control activities were further explored in subsequent chapters (cf. Chapters 4, 5, and 6).

The chosen approach proved useful, especially in evaluating methods for the different planning activities, but also in the evaluation of system layouts. The result is a simulation library in eM-Plant, which can be used to model automated transportation networks. New terminal designs, control methods (e.g. order acceptance) and objects such as a load bearer, can be added to the library to increase the options provided by the simulation library. The simulation library was used to determine resource requirements for specific OLS system layouts, to determine the effects of a two-way tube versus two one-way tubes, to evaluate a growth scenario for the years 2005-2020, and to evaluate several methods for the control activities.

Though our object-oriented approach provides modular objects (building blocks), tuning of the control is always necessary. A specific combination of objects does not automatically guarantee a good logistic performance. For example, a new vehicle

manager might require specific behavior from the terminal manager. Owing to these dependencies, the construction of a model with good logistic performance is not always straightforward. Performance is determined by the content and interaction of objects, not by an object-oriented structure.

*Vehicle management (cf. research questions 4, 6, 8)*

For vehicle management, we designed several control rules, a First-Come First-Served heuristic, a rule with hierarchical coordination, a serial scheduling method and an algorithm based on Powell and Carvalho's logistic queuing network approach (see Table 4.1). These options were evaluated on customer service levels, resource requirements and empty travel distance. We can conclude that a simple FCFS heuristic gives the worst results; all other rules yield significantly better results (see Table 4.5). In most cases, the local control rule with hierarchical coordination shows similar results compared to the rules with central coordination. Only in cases of demand that is quickly changing between routes and locations do the rules with central coordination perform significantly better. The serial scheduling approach gave the best results, also with respect to empty travel distance, which means a reduction in energy consumption. The logistic queuing network approach shows slightly worse results than the rule with hierarchical coordination and the serial scheduling approach. From these results, we can conclude that the performance of a decentralized control rule is not necessarily worse than that of a centralized control rule. Only when the transportation flows are changing very dynamically, do the centralized control rules perform significantly better. Given the fact that we used a rather small network, it might be possible that in larger transportation networks the gap between centralized and decentralized control increases in favor of centralized control rules.

Pre-arrival information on transportation jobs improves the planning results, but information is only useful up to a certain time period (see Table 4.6). Given frequent re-planning, e.g. every 10 minutes, an information horizon equal to the maximum travel time between two locations in the system (30 minutes in the OLS-case) seems sufficient. A longer information horizon only leads to marginal improvements (research question 6).

*Two-way track control (research questions 4, 6, 8)*

A two-way track can considerably reduce infrastructure investment. Access control rules were designed to manage vehicle flows, in such a way that collisions are avoided and waiting times are minimized. We developed both periodic control rules and adaptive control rules that react to known and/or predicted vehicle arrivals (cf. research question 4). For the periodic control rule, theoretical approximations for the mean waiting time as a function of the switching frequency were derived, for both the symmetric and asymmetric cases (Section 5.4). Comparison with simulation results indicated that the accuracy is usually in the range of 1-2% in the case of Poisson arrivals. Based on these approximations, the optimum switching frequency

can be accurately approximated for a class of arrival processes (Poisson, compound Poisson).

Intelligent adaptive rules vary from local adaptive control via look-ahead heuristics to dynamic programming solutions. Numerical experiments showed that intelligent adaptive control rules reduce waiting times by 10-25% (see Figure 5.10), compared to a straightforward periodic rule. In theory, the best adaptive control rule is a dynamic programming rule, but this rule also requires the most intensive exchange of information. It turned out that the dynamic programming method was not very sensitive to the information horizon, but in practice, it will probably be difficult to generate the necessary information on future arrivals reliably. The less information-intensive, intelligent adaptive rules, even without look-ahead, can already realize the larger part of the performance improvement (cf. research question 6).

#### *Battery management (research questions 7, 8)*

From Chapter 6 (Section 6.2), we conclude that, unlike most internal transportation systems, an automated transportation network cannot neglect battery constraints. Only when charge-rails are used, are the logistic effects of the use of batteries negligible, but this has the disadvantage that serious investment costs are required. The option of a 'battery swap' requires battery stations where batteries can be exchanged. Furthermore, the required number of AGVs slightly increases (up to 10%) and the required number of batteries might even triple, depending on the battery type and the possibility of using a quick-charge strategy. The logistic control had to be adapted to take battery constraints into account. AGVs have to be sent to battery stations before the battery is empty, and whenever possible changing the battery should take place during idle or waiting times. Additional AGVs can compensate the negative effects of a battery swap strategy. The choice between one of the alternatives, charge-rails or battery swap, and of a specific battery type (lead-acid or nickel-cadmium), cannot be based solely on logistic aspects. Detailed cost computations are also required to support such a choice. We showed an approach to assessing the costs of the different alternatives (Section 6.2.4), which can be used when detailed cost figures are available.

#### *Failure management (research questions 5, 8)*

Handling AGV failures is a daily activity in a system such as the OLS, with serious consequences for the logistic performance, as shown in Section 6.3. We designed control methods to handle AGV and dock failures, and to try to minimize the negative effects of these failures. The model can be used to assess the effects of given failure rates, or to determine acceptable failure rates, i.e. failure rates that do not really affect the logistic performance and could be used as a target in the design process. For the OLS case, we found that AGV failures affect the system performance much more than dock failures. When the AGV failure rate is smaller than once per 500 operational hours, the decrease in system performance remains within reasonable limits. We only performed explorative research on failures. From experiments with isolated controlled disturbances, we found that one AGV failure in

a two-way tube in a busy time period can reduce service levels for that particular day by 10%. Dock failures and AGV failures on terminals have only limited effects on system performance. In general, logistic performance decreases after the introduction of failures. Additional AGVs can only slightly compensate these negative effects.

#### *Storage capacity restrictions (research questions 4, 8)*

Several capacity restrictions can be present in automated transportation networks. The number of docks, parking places and storage locations is limited. Dock and parking capacities are included in the model, but not taken into account in the logistic control. These capacity restrictions appeared to have limited impact on system performance, which can also be derived from the marginal effects of dock failures. The storage capacity restriction was the most important capacity restriction in the OLS-project at the time of the research. Owing to the very high investment costs at Schiphol Airport, storage space at the Zwanenburg rail terminal had to be limited. We extended the control methods to take this restriction into account, assuming that travel times vary only slightly (Section 6.4). The coordination between the terminals has to increase when storage capacities are included. Given the fact that between arrival and departure cargo has to be stored somewhere in the system, a decrease in storage utilization at one terminal leads to an increase on other terminals. Furthermore, a restriction on storage capacity reduces the possibilities for peak shaving. As a consequence, the resource requirements increase. Further research has to be performed to investigate whether the same procedures can be used with stochastic travel times (see Section 7.3).

#### *Impact of information (research question 6)*

The availability of information was an important aspect in this research. It turned out that limited pre-arrival information, of about 15-30 minutes, improves the logistic performance significantly (cf. Table 4.6). In controlling two-way tracks, information on future arrivals can also be used to reduce the waiting times at these two-way tracks.

Another aspect is the accuracy of the information. The transportation flows used in the OLS-case were estimated for the years 2005-2020. It is difficult to estimate these transportation flows, especially because there is not even a rail terminal at the moment and also, the throughput time requirements, which are still very unclear, were not known at the time of the research project. Therefore, the outcomes of the simulation model with respect to the number of AGVs and docks are not exact answers. Just as the inputs are estimates for 2020, the outcomes are also approximations of the performance and resource requirements in the year 2020. The strong feature of the simulation model is that several scenarios or growth paths can easily be computed. The model does not have to be changed; all methods work if other transportation flows are introduced. Recall, for example, that the three different system designs were substantially different, but that the methods developed work for all these system designs.



### 7.3 Further research

In Chapter 1, we made some assumptions and simplifications for the research in this thesis. Several interesting topics for further research, partly based on the adaptation of these assumptions, are discussed in this section.

#### *Capacity restrictions*

A topic already mentioned in the previous section is the problem of capacity restrictions. A method to incorporate storage capacity restrictions was discussed in Chapter 6, but dock and parking capacities are not integrated in the logistic planning. The small Schiphol terminals, without parking places and with only one or two docks, might profit from an approach that takes these dock and parking capacities into account. Sometimes AGVs have to be sent away, because there is no space on the terminal and AGVs cannot wait in front of the terminal because they would block all other traffic. Including these restrictions in the control rules might prevent several AGVs from arriving at the same time. Otherwise some of these AGVs may have to make another circuit at Schiphol. An interesting question is whether these capacity restrictions can be implemented according to a local control principle, or whether a scheduling method, such as the one presented in Chapter 4, is better suited to incorporate these capacity restrictions (integrated planning). Another question is how a two-way track can be incorporated in the integrated planning, resulting in a model in which the travel times can be predicted because the direction changes of the two-way track are scheduled by the same method that schedules the vehicles.

#### *Just-in-time delivery*

Related to the aspect of capacity restrictions is a just-in-time control concept. In an ideal situation, cargo is delivered at its destination just-in-time before train or plane departure. This is a different control concept (pull approach) from the one used in this thesis, which is more a push approach. Cargo is delivered at its destination as quickly as possible. In the push approach, the best possible use is made of opportunities for peak shaving. This reduces the peak flows and thus the resource requirements. In the case of just-in-time delivery (pull approach) peak-shaving opportunities can no longer be used, which leads to an increase in the resource requirements. It is probably more important to incorporate capacity restrictions in a just-in-time concept, because there is less slack time to wait for a dock. Interesting questions concern which control methods are best suited for just-in-time deliveries.

#### *Consolidation*

Consolidation of cargo was not an issue in this thesis; we assumed that cargo arrives in complete transportation units. This is not the case in practice; cargo may be delivered in boxes, on pallets, or in other ways. To incorporate this requirement into the model, consolidation methods have to be developed. Given the limited capacity of an AGV, the time needed to combine several cargo types to one transportation unit and the customs regulations at Schiphol Airport, it is likely that for the OLS case consolidation can only take place at a terminal, not between terminals. In view

of the fact that all transportation jobs have a specific due-time, and possibly different weights and dimensions, this consolidation problem is not trivial. How long should one wait for additional cargo when an AGV is still not completely loaded? According to which rules can different cargo types be combined? An additional constraint might be that cargo has to be consolidated according to flight number or train compartment.

#### *Terminal scheduling*

The research in this thesis was focused on the higher levels of the logistic planning. The implementation of terminal control, vehicle control and traffic control was rather simplified. Research on vehicle and traffic control was performed by Verbraeck et al. (1998a, 1998b). Scheduling within a terminal was not a major aspect of our research. Probably some principles from internal transportation systems can be used for terminal control (cf. Van der Meer, 2000). Interesting topics could include the pre-positioning of AGVs within the terminal, the scheduling of docks, the selection of a dock or AGV for a specific job, based on the position of the AGVs on the terminal or on the battery charge of an AGV. Consolidation of cargo is another activity at a terminal, as described above, but also the process after arrival at the destination terminal, for example the link with connecting flights or train departures, could be modeled in more detail.

#### *Multi-agent approach*

Several options for vehicle management were discussed in this research, but other approaches are also possible, a multi-agent approach being one of them. At the moment, agent systems receive a lot of attention in the literature; see for example Jennings (2000). It is possible to design a marketplace, where AGVs and transportation jobs are brought together and bidding principles are used to link AGVs and jobs. It is desirable that AGVs should maximize their revenues (or loaded kilometers), while transportation jobs need to be at their destination on time, and the closer the due time, the more a job might be willing to bid for an AGV. It is interesting to find out whether market forces can achieve a result that is close to the global optimum. One of the problems in such an approach might be to estimate the 'revenue' of being at a specific terminal in a future time period, as was encountered in the LQN approach presented in Chapter 4. After having transported one job, the AGV will be at the destination of this job, and it is important to know whether high revenues can be attained from this terminal at that time period. Further research is required to determine the attractiveness of a multi-agent approach for this particular application.

#### *Exploitation model*

In Chapter 3 we presented an analytical model to estimate the resource requirements in a layout with deterministic travel times. This method could not be used for layouts with two-way tracks, because these two-way tracks result in long waiting times for the AGVs at the ends of the two-way track and at the terminals. Extending the analytical model to take the stochastic travel times into account might be

worthwhile. Such an analytical model could be a valuable tool during the design process, because the resource requirements of several designs can be quickly compared, without the need to run many simulation experiments. Such an analytical model could be part of a larger exploitation model, in which a total cost figure is determined for the system layout, resource requirements, and energy consumption.

*More complex network structures*

In this thesis we considered network structures that have a direct link with a specific application, the OLS. The size of these networks is comparable to a small regional transportation network or a city distribution network. The question is whether the same performance as that obtained in our research can be attained for larger and/or more complex network structures, such as a national network with several regional and/or city distribution sub-networks.

# References

- Adan, I.J.B.F., Van Eenige, M.J.A. and Resing, J.A.C. (1996) Fitting discrete distributions on the first two moments. *Probability in the Engineering and Informational Sciences*, 9, 623-632.
- Akturk, M.S. and Yilmaz, H. (1996) Scheduling of automated guided vehicles in a decision making hierarchy. *International Journal of Production Research*, 34, 577-592.
- Arifin, R. and Egbelu, P.J. (2000) Determination of vehicle requirements in automated guided vehicle systems: a statistical approach. *Production Planning and Control*, 11, 258-270.
- Ball, M.O., Magnanti, T.L., Monma, C.L. and Nemhauser, G.L. (1995) *Network Routing: Handbooks in operations research and management science*. Volume 8, Elsevier, Amsterdam.
- Bell, M.G.H. (1992) Future directions in traffic signal control. *Transportation Research A*, 26, 303-313.
- Bliss, D. (2000) Mail Rail: 70 years of automated underground freight transport. *Proceedings ISUFT 2000*, TRAIL, Delft.
- Booch, G. (1994) *Object-oriented analysis and design, with applications*. Addison-Wesley, Reading.
- Buck, R., Wever, E., Van Binsbergen, A.J., Visser, J.G.S.N. , Katgerman, J. and Piebenga, D. (1999) *Gaat Leiden Ondergronds?*. Hoofdrapport, Buck Consultants International (in Dutch).
- Carvalho, T.A. and Powell, W.B. (1998) *A multiplier adjustment method for dynamic resource allocation problems*. Technical Report SOR-96-03, Princeton University.
- Center for Transportation Technology (1997) *Definitiestudie ondergronds logistiek systeem*. Deelrapportages 1 t/m 5, Rotterdam, the Netherlands (in Dutch)
- Centrum Ondergronds Bouwen (2000) *Projectdocument COB*. Centrum Ondergronds Bouwen, the Netherlands (in Dutch)
- Cheng, T.C.E. (1987) Simulation study of automated guided vehicle dispatching. *Robotics and Computer-Integrated Manufacturing*, 3, 335-338.
- Cheung, R.K., and Chen, C-Y. (1998) A two-stage stochastic network model and solution methods for the dynamic empty container allocation problem. *Transportation Science*, 32, 142-162.

- Cheung, R.K. and Powell, W.B. (1994) An algorithm for multistage dynamic networks with random arc capacities, with an application to dynamic fleet management. *Operations Research*, 44, 951-963.
- Co, C.G. and Tanchoco, J.M.A. (1991) A review of research on AGVS vehicle management. *Engineering Costs and Production Economics*, 21, 35-42.
- Dejax, P.J. and Crainic, T.G. (1987) A review of empty flows and fleet management models in freight transportation. *Transportation Science*, 21, 227-247.
- De Kok, A.G. (1989) A moment-iteration method for approximating the waiting time characteristics of the G/G/1 queue. *Probability in the engineering and informational sciences*, 3, 273-287.
- De Koster, R. and Van der Meer, J.R. (1998) Centralized versus decentralized control of internal transport, a case study. In: B. Fleischmann, J.A.E.E. van Nunen, M. Grazia Speranza and P. Stähly (eds.), *Advances in distribution logistics*, Springer, Berlin, 403-420.
- Desrosiers, J., Dumas, Y., Solomon, M.M. and Soumis, F. (1995) Time Constrained Routing and Scheduling, in Ball, M.O., Magnanti, T.L., Monma, C.L. and Nemhauser, G.L. (eds.) *Network Routing: Handbooks in operations research and management science*. Elsevier, Vol. 8, Amsterdam.
- Dijkstra, E.W. (1959) A note on two problems in connection with graphs. *Numerische Mathematik*, 1, 269-271.
- Doshi, B. (1990) Single server queues with vacations. In: *Stochastic analysis of computer and communication systems*. H. Takagi (ed.), Elsevier Science, North-Holland, Amsterdam.
- DynaVision (1999) OLS-Utrecht: *Een uitweg onder de grond*. DynaVision Management Consultancy, Wassenaar, the Netherlands (in Dutch).
- ECMT (1995) *Transport: New problems, new solutions*. OECD Publications Service, Paris, France.
- Egbelu, P.J. and Tanchoco, J.M.A. (1984) Characterization of automated guided vehicle dispatching rules. *International Journal of Production Research*, 22, 359-374.
- Egbelu, P.J. and Tanchoco, J.M.A. (1986) Potentials for bidirectional guidepath for automatic guided vehicle based systems. *International Journal of Production Research*, 24, 1075-1099.
- Egbelu, P.J. (1987) The use of non-simulation approaches in estimating vehicle requirements in an automated guided vehicle based transport system. *Material Flow*, 4, 17-32.
- Egbelu, P.J. (1993) Positioning of automated guided vehicles in a loop layout to improve response time. *European Journal of Operational Research*, 71, 32-44.
- Espinasse, B., Cloutier, L. and Lefrancois, P. (1998) Globalization of manufacturing in the digital communications era of the 21<sup>st</sup> century: Innovation, agility and the virtual enterprise. *Proceedings of the tenth international IFIP international conference PROLAMAT 98*, Trento, Italy.
- Evers, J. and Lindeijer, D.G. (1999) *The agile traffic-control and engineering system: TRACES*. Working paper, TRAIL Research School, the Netherlands.

- Evers, J.J.M., Loeve, L. and Lindeijer, D.G. (2000) The service-oriented agile logistic control and engineering system: SERVICES. *Logistics Information Management*, 13, 77-90.
- Fowler, J.W., Hogg, G.L. and Phillips, D.T. (1992) Control of multiproduct bulk service diffusion/oxidation processes. *IIE Transactions*, 24, 84-96.
- Gademann, A.J.R.M. and Van de Velde, S.M. (2000) Positioning automated guided vehicles in a loop layout. *European Journal of Operational Research*, 127, 565-573.
- Glasse, C.R. and Weng, W.W. (1991) Dynamic batching heuristic for simultaneous processing. *IEEE transactions on semiconductor manufacturing*, 4, 77-82.
- Golden, B.L. and Assad, A. (1988) *Vehicle Routing: Methods and Studies*, North Holland, Amsterdam.
- Haight, F.A. (1963) *Mathematical theories of traffic flow*. Academic press, New York.
- Heidemann, D. (1994) Queue length and delay distributions at traffic signals. *Transportation Research B*, 28, 377-389.
- Hollier, R.H. (1987) *Automated guided vehicle systems*. Springer, Berlin.
- Holmberg, K., Joborn, M. and Lundgren, J.T. (1998) Improved empty freight car distribution. *Transportation Science*, 32, 163-173.
- Hoover, S.V. and Perry, R.F. (1989) *Simulation, a problem solving approach*. Addison-Wesley, Reading.
- Hu, C. and Egbelu, P.J. (2000) A framework for the selection of idle vehicle home locations in an automated guided vehicle system. *International Journal of Production Research*, 38, 543-562.
- Hu, X.N., Tang, L.C. and Ong, H.L. (1997) A  $M/D^x/1$  vacation queue model for a signalized intersection. *Computers and Industrial Engineering*, 33, 801-804.
- Hwang, H. and Kim, S.H. (1998) Development of dispatching rules for automated guided vehicle systems. *Journal of Manufacturing Systems*, 17, 137-143.
- Iding, M. and Van der Heijden, M.C. (2000) Towards a public freight network with underground pipeline transportation. *Proceedings ISUFT 2000*, TRAIL, Delft.
- Ilic, O.R. (1994) Analysis of the number of automated guided vehicles required in flexible manufacturing systems. *International Journal of Advanced Manufacturing Technology*, 9, 382-389.
- Jennings, N.R. and Wooldridge, M.J. (1998) *Agent technology: foundations, applications, and markets*. Springer, Berlin
- Jennings, N.R. (2000) On agent-based software engineering. *Artificial Intelligence*, 117, 277-296.
- Jordan, W.C. and Turnquist, M.A. (1983) A stochastic, dynamic network model for railroad car distribution. *Transportation Science*, 17, 117-145.
- Kim, K.H. (1995) Positioning of automated guided vehicles in a loop layout to minimize the mean vehicle response time. *International Journal of Production Economics*, 39, 201-214.
- Kim, K.H. and Kim, J.Y. (1997) Estimating mean response time and positioning idle vehicles of automated guided vehicle systems in loop layout. *Computers and Industrial Engineering*, 33, 669-672.

- Klein, C.M. and Kim, J. (1996) AGV dispatching. *International Journal of Production Research*, 34, 95-110.
- Kleinrock, L. (1976) *Queueing systems, volume 2: Computer applications*. Wiley, New York
- Kolisch, R. (1996) Serial and parallel resource-constrained project scheduling methods revisited: Theory and computation. *European Journal of Operational Research*, 90, 320-333.
- Law, A.M. and Kelton, W.D. (1991) *Simulation modeling and analysis*. McGraw-Hill Inc, New York.
- Liu, H. (2000) Capsule pipeline: An overview. *Proceedings ISUFT 2000*, TRAIL, Delft.
- Mahadevan, B. and Narendran, T.T. (1990) Design of an automated guided vehicle-based material handling system for a FMS. *International Journal of Production Research*, 28, 1611-1622.
- Mahadevan, B. and Narendran, T.T. (1993) Estimation of number of AGVs for an FMS: an analytical model. *International Journal of Production Research*, 31, 81-106.
- Mahadevan, B. and Narendran, T.T. (1994) A hybrid modeling approach to the design of an AGV-based material system for an FMS. *International Journal of Production Research*, 32, 2015-2030.
- Majety, S.V. and Wang, M.H. (1995) Terminal location and guide path design in terminal based AGV systems. *International Journal of Production Research*, 33, 1925-1938.
- Maxwell, W.L. and Muckstadt, J.A. (1982) Design of automated guided vehicle systems. *IIE Transactions*, 14, 114-124.
- McHaney, R. (1995) Modelling battery constraints in discrete event automated guided vehicles simulations. *International Journal of Production Research*, 33, 3023-3040.
- Meissl, P. (1963) Zufallsmodell einer Lichtsignalanlage mit mehrspurigem Stauraum. *Mathematik-Teknik-Wirtschaft* Heft 1/62, 11-18 and Heft 2/62, 65-69, Wien.
- Miller, A.J. (1968) Settings for fixed-cycle traffic signals. *Operations Research Quarterly*, 14, 373-386.
- Ministry of Transport, Public Works and Water Management (2000a) <http://www.bereikbaarheidsoffensief.nl/Details.asp?Action=FAQDetails&CatID=-1&rID=57>
- Ministry of Transport, Public Works and Water Management (2000b) <http://www.minvenw.nl/rws/wnt/avg/>
- Ministry of Transport, Public Works and Water Management (2000c) <http://www.bereikbaarheidsoffensief.nl/index.asp?Action=Visie&mID=1460&CatID=3>
- Mung, G.K.S., Poon, C.K.A. and Lam, W.H.K. (1996) Distributions of queue lengths at fixed time traffic signals. *Transportation Research B*, 30, 421-439.
- Neuts, M.F. (1967) A general class of bulk queues with Poisson input. *Annals of mathematical statistics*, 38, 759-770.

- Potts, C.N. (2000) Scheduling with batching: a review. *European Journal of Operational Research*, 120, 228-249.
- Powell, W.B., Sheffi, Y., Nickerson, K.S., Butterbaugh, K. and Atherton, S. (1988) Maximizing profits for North American Van Lines' truckload division: A new framework for pricing and operations. *Interfaces*, 18, 21-41.
- Powell, W.B., Jaillet, P. and Odoni, A. (1995a) Stochastic and dynamic networks and routing, in Ball, M.O., Magnanti, T.L., Monma, C.L. and Nemhauser, G.L. (eds.) *Network Routing: Handbooks in operations research and management science*. Elsevier, Vol. 8, Amsterdam, 141-295.
- Powell, W.B., Carvalho, T.A., Godfrey, G. and Simao, H. (1995b) Dynamic fleet management as a logistics queuing network. *Annals of Operations Research*, 61, 165-188.
- Powell, W.B. (1996) A stochastic formulation of the dynamic assignment problem, with an application to truckload motor carriers. *Transportation Science*, 30, 195-219.
- Powell, W.B. and Carvalho, T.A. (1998a) Dynamic control of logistics queuing networks for large-scale fleet management. *Transportation Science*, 32, 90-109.
- Powell, W.B. and Carvalho, T.A. (1998b) Real-time optimization of containers and flatcars for intermodal operations. *Transportation Science*, 32, 110-126.
- Rajotia, S., Shanker, K. and Batra, J.L. (1998) Determination of optimal fleet size for an FMS. *International Journal of Production Research*, 36, 1177-1198.
- Rijsenbrij, J.C., Dunselman, J.R. and Pielage, B.A. (2000) *Ondergronds Logistiek Systeem ASH: Projectdocument van het deelproject transporttechniek en besturingssysteem*. TRAIL Research School, the Netherlands (in Dutch)
- Robertson, D.I. and Bretherton, R.D. (1974) Optimum control of an intersection for any known sequence of vehicle arrivals. *Traffic control and transportation systems*, AFCET, pp. 3-17. North-Holland, Amsterdam.
- Russel, R.S. and Tanchoco, J.M.A. (1984) An evaluation of vehicle dispatching rules and their effects on shop performance. *Material Flow*, 1, 271-280.
- Saanen, Y.A., Valentin, E. and Verbraeck, A. (1999) *Tussenrapportage deel II: Logistieke prestaties terminals in het OLS*. TRAIL Research School, the Netherlands (in Dutch)
- Sabuncuoglu, I. and Hommertzhaim, D.L. (1992) Dynamic dispatching algorithm for scheduling machines and automated guided vehicles in a flexible manufacturing system. *International Journal of Production Research*, 30, 1059-1079.
- Schiphol Project Consult (2000) *Deelprojecten OLS cluster: samenvatting*. Document code: L85.03-00.8014.R, Schiphol Project Consult, the Netherlands (in Dutch)
- Seifert, R.W., Kay, M.G. and Wilson, J.R. (1998) Evaluation of AGV routing strategies using hierarchical simulation. *International Journal of Production Research*, 36, 1961-1976.
- Silver, E.A., Pyke, D.F. and Peterson, R. (1998) *Inventory management and production planning and scheduling*. Wiley, New York.



- Smith, B.C., Leimkuhler, J.F. and Darrow, R.M. (1992) Yield management at American Airlines. *Interfaces* 22, 8-31.
- Soh, J.T.L., Hsu, W.J., Huang, S.Y. and Ong, A.C.Y. (1996) Decentralized routing algorithms for automated guided vehicles. *ACM SIGICE Bulletin*, 22, 12-18.
- Solomon, M.M., and Desrosiers, J. (1988) Time window constrained routing and scheduling problems. *Transportation Science* 22, 1-13.
- Stein, D. and Schoesser, B. (2000) Current progress of the research project "Underground transportation and utility systems" at the Ruhr-University Bochum. *Proceedings ISUFT 2000*, TRAIL, Delft.
- Taghaboni, F. and Tanchoco, J.M.A. (1995) Comparison of dynamic routing techniques for automated guided vehicle systems. *International Journal of Production Research*, 33, 2653-2669.
- Takagi, H. (1990) Queueing analysis of polling models: An update. In: *Stochastic analysis of computer and communication systems*, H. Takagi (ed.), Elsevier Science, North-Holland, Amsterdam, 267-318
- Takagi, H. (1991) *Queueing analysis, a foundation of performance evaluation, Volume 1: Vacation and priority systems*. Elsevier Science, North-Holland, Amsterdam.
- Tanchoco, J.M.A., Egbelu, P.J. and Taghaboni, F. (1987) Determination of the total number of vehicles in an AGV-based material transport system. *Material Flow*, 4, 33-51.
- Taniguchi, E., Ooishi, R. and Kono, T. (2000) Development and future perspectives for underground freight transport systems in Japan. *Proceedings ISUFT 2000*, TRAIL, Delft.
- Tecnomatix (2000) *eM-Plant reference manual*. Version 4.5, Tecnomatix Technologies.
- Ulusoy, G. and Bilge, U. (1993) Simultaneous scheduling of machines and automated guided vehicles. *International Journal of Production Research*, 31, 2857-2873.
- Uzsoy, R., Lee, C.Y. and Martin Vega, L.A. (1992) A review of production planning and scheduling models in the semiconductor industry, Part II: Shop floor control. *IIE Transactions*, 26, 44-55.
- Uzsoy, R., Lee, C.Y. and Martin Vega, L.A. (1994) A review of production planning and scheduling models in the semiconductor industry, Part I: System characteristics, performance evaluation and production planning. *IIE Transactions*, 24, 47-60.
- Van de Geijn, W.E. and Katgerman, J. (1996) *Haalbaarheidsonderzoek ondergronds logistiek systeem tussen luchthaven Schiphol en Bloemenveiling Aalsmeer*. Van de Geijn Partners B.V., Houten, the Netherlands (in Dutch)
- Van der Heide, J. (1999) *Energiestudie OLS*. TNO report 99.OR.NT/005.1/JvdH, TNO Wegtransportmiddelen, Delft, the Netherlands (in Dutch).
- Van der Heijden, M.C., Van Harten, A., Ebben, M.J.R., Saanen, Y.A., Verbraeck, A. and Valentin, E. (2000) *Safeguarding Schiphol Airport accessibility for freight transport: The design of a fully automated underground transportation*

- system and the role of simulation*. Working paper, University of Twente and Delft University of Technology, the Netherlands, submitted for publication.
- Van der Heijden, M.C., Ebben, M.J.R., Gademann, A.J.R.M. and Van Harten, A. (2001a) Scheduling vehicles in automated transportation systems: Algorithms and case study. To appear in *OR Spektrum*, 23.
- Van der Heijden, M.C., Van Harten, A. and Ebben, M.J.R. (2001b) Waiting times at periodically switched one-way traffic lanes: A periodic, two-queue polling system with random setup times. To appear in *Probability in the engineering and informational sciences*.
- Van der Meer, J.R. (2000) *Operational control of internal transport*. Ph.D. Thesis, Erasmus University Rotterdam, the Netherlands.
- Van der Zee, D.J., Van Harten, A. and Schuur, P.C. (1997) Dynamic Job Assignment Heuristics for multi-server batch operations – A cost based approach. *International Journal of Production Research*, 35, 3063-3093.
- Van der Zee, D.J. (1997) *Simulation as a tool for logistics management*. Ph.D. Thesis, University of Twente, the Netherlands.
- Van der Zwaan, A.H. (1995) *Organisatie onderzoek; leerboek voor de praktijk: het ontwerpen van onderzoek in organisaties*. Van Gorcum, Assen, (in Dutch).
- Van Harten, A., Van der Heijden, M.C. and Ebben, M.J.R. (1999) *Deelproject simulatie OLS: Rapportage accumanagement en energiestudie*. Report OLS simulation group, University of Twente, the Netherlands (in Dutch).
- Van Harten, A., Ebben, M.J.R. and Van der Heijden, M.C. (2000) *Deelproject simulatie OLS: Effecten trace met rail terminal Zwanenburg en analyse groeipad*. Report OLS simulation group, University of Twente, the Netherlands (in Dutch)
- Verbraeck A., Saanen, Y.A. and E. Valentin, E. (1998) Logistic Modeling and Simulation of Automated Guided Vehicles. In: A. Bargiela, E. Kerckhoffs (Eds.), *Simulation Technology: Science and Art*. SCS, San Diego CA, 514-519.
- Verbraeck, A., Valentin, E. and Saanen, Y.A. (1999) Simulatie voor het Ondergronds Logistiek Systeem Schiphol. In *SimLog: Simulatie en logistiek rond de haven*, Van Nunen, J. and Verspui, L. (eds.), Eburon, Delft, the Netherlands (in Dutch).
- Verbraeck, A., Valentin, E. and Saanen, Y.A. (1998a) *OLS Simulatie Voertuigbesturing: Voortgangrapportage TRACES implementatie in SiMPLE++*. Report OLS simulation group, TRAIL Research School, the Netherlands (in Dutch).
- Verbraeck, A., Valentin, E. and Saanen, Y.A. (1998b) *OLS Simulatie Voertuigbesturing: Tussenrapportage logistieke prestaties terminals in het OLS*. Report OLS simulation group, TRAIL Research School, the Netherlands (in Dutch).
- Verbraeck, A., Saanen, Y. and Valentin, E. (2000) Designing effective terminals and their control systems for the underground logistic system Schiphol. *Proceedings ISUFT 2000*, TRAIL, Delft.

- Verduijn, T.M. and Broens, D.F. (2000) Goederen Clearing House – Schakeloverschrijdende transportplanning in logistieke ketens. *TNO Inro report Log2000-11*, Delft, The Netherlands (in Dutch).
- Webster, F.W. (1958) Traffic Signal Settings, Road Research Technical paper 39. *Her Majesty's Stationary Office*, London.
- Webster, S. and Baker, K.R. (1995) Scheduling groups of jobs on a single machine. *Operations Research*, 43, 692-703.
- Weng, W.W. and Leachman, R.C. (1993) An improved methodology for real-time production decisions at batch-process work stations. *IEEE transactions on semiconductor manufacturing*, 6, 219-225.
- Winston, W.L. (1991) *Operations Research: Applications and algorithms*. Duxbury Press, Belmont, California.
- White, W. and Bomberault, A. (1969) A network algorithm for empty freight car allocation. *IBM Systems Technical Journal*, 8, 147-171.
- Wysk, R.A., Egbelu, P.J. and Tanchoco, J.M.A. (1987) Use of spread sheet analysis for evaluating AGV systems. *Material Flow*, 4, 53-64.
- Yim, D.S. and Linn, R.J. (1993) Push and pull rules for dispatching automated guided vehicles in a flexible manufacturing system. *International Journal of Production Research*, 31, 43-57.

## Samenvatting (Summary in Dutch)

De toenemende drukte op het wegennet bedreigt de bereikbaarheid van met name de stedelijke gebieden en verkeersknooppunten als vliegvelden en havens. Daarnaast heeft de sterke groei in het wegverkeer invloed op milieu, ruimtegebruik en verkeersveiligheid. Voorzien wordt dat de groei in het goederenvervoer zich in de komende decennia zal voortzetten, waardoor de genoemde problemen steeds groter zullen worden. Vandaar dat gezocht wordt naar alternatieve transportmodaliteiten. Voor een deel van de goederenstromen zou vervoer per Ondergronds Logistiek Systeem (OLS) een aantrekkelijk alternatief kunnen zijn. Dat betreft met name de zogenaamde “tijdkritische goederen”, dat wil zeggen die goederen waarbij een vertraging in de aflevering serieuze gevolgen heeft. Als bloemen, verse groente of fruit het vliegtuig of de trein missen, kan dit de waarde van deze goederen op de plaats van bestemming sterk beïnvloeden. Dat geldt ook voor bijvoorbeeld kranten en dure reserve-onderdelen die voor de reparatie van high-tech apparatuur dringend nodig zijn.

Doel van dit proefschrift is het ontwerp en de beoordeling van een logistieke besturingsstructuur voor geautomatiseerde transportnetwerken, die garant staat voor een hoge logistieke performance, gebruikmakend van een acceptabele hoeveelheid resources als Automatisch Geleide Voertuigen (AGV's) en docks.

In dit promotie-onderzoek is daartoe een objectmodel ontwikkeld (*Hoofdstuk 2*) dat gebruikt kan worden voor de modellering van geautomatiseerde transportnetwerken. In dit objectmodel is een onderscheid gemaakt tussen fysieke, informatie en besturingsobjecten. De processen in het netwerk zijn verder onderverdeeld in een primair proces (transport en logistieke afhandeling) en secundaire processen zoals accu-management en het afhandelen van storingsen. Bij de ontwikkeling van de besturingsstructuur is uitgegaan van een lokaal besturingsconcept. Een dergelijke benadering levert flexibiliteit op in de modellering, is robuust gebleken en is op een aantal punten vergeleken met een meer centraal gecoördineerde besturing. Het objectmodel is geïmplementeerd in het simulatiepakket eM-Plant.

Het OLS Schiphol-Aalsmeer is een goed voorbeeld van een geautomatiseerd transportnetwerk, en is gebruikt als case in dit proefschrift. De input van het simulatiemodel met betrekking tot systeem layout, terminal layouts, goederenstromen, etc., is gebaseerd op de OLS-case, beschreven in *Hoofdstuk 3*. De

luchthaven Schiphol en de bloemenveiling te Aalsmeer hebben al langere tijd problemen met de bereikbaarheid over de weg. Files zorgen voor veel onzekerheid in de aankomsttijden, waardoor de leverbetrouwbaarheid in het gedrang komt. Het behouden en verbeteren van hun bereikbaarheid is van levensbelang voor de concurrentiepositie van Schiphol en de bloemenveiling Aalsmeer. Een mogelijk structurele oplossing is het OLS, dat tevens een link verzorgd met railvervoer. Karakteristiek voor dit systeem zijn onder andere het aantal AGV's, 200-300, en de afstanden tussen de terminals, oplopend tot meer dan 10 kilometer.

Het simulatiemodel is gebruikt om verschillende alternatieven voor de besturingsactiviteiten te testen en te beoordelen, maar ook voor de vergelijking van systeem layouts, scenario's voor goederenstromen en alternatieve AGV snelheden. Interessante vragen waren welke methoden qua performance het beste werken en wat de invloed is van de beschikbare informatie. Een aantal besturingsaspecten is verder uitgediept: voertuig management, de besturing van een buis met wisselende rijrichting, accu-management en de afhandeling van AGV en dock steringen. In dit proefschrift ligt de focus op de primaire processen van voertuig management (*Hoofdstuk 4*) en de enkele buis besturing (*Hoofdstuk 5*).

Door de onbalans in de goederenstromen in een transport netwerk is het noodzakelijk dat de voertuigen worden herverdeeld over de terminals. Voertuigen moeten verplaatst worden van een terminal met een voertuig overschot naar terminals met een tekort aan voertuigen. Voor dit probleem zijn verschillende methoden ontwikkeld (*Hoofdstuk 4*), variërend van simpele heuristieken tot integrale plannings- en schedulings methoden. Een lokaal besturingsconcept leidt niet noodzakelijkerwijs tot een slechtere prestatie van het systeem. Een slimme lokale besturing met coördinatie tussen de terminals levert, voor de netwerken zoals onderzocht in dit proefschrift, vergelijkbare prestaties op als een integrale planning. Informatie over het aanlevertijdstip van goederen kan de planning aanzienlijk verbeteren. Wanneer aanlevertijdstippen een bepaalde tijd van tevoren bekend zijn kan de besturing daarop anticiperen. Voor de OLS-case lijkt 30 minuten voldoende, wat gelijk is aan de maximale rijtijd tussen twee terminals. Een langere informatie horizon levert in deze case weinig extra voordelen op.

De investeringskosten voor een OLS zijn hoog, met name vanwege de ondergrondse infrastructuur. Deze kosten kunnen aanzienlijk omlaag als er in plaats van twee buizen (één voor elke rijrichting) maar één buis wordt aangelegd tussen twee terminals, waarbij de rijrichting van deze buis wordt gevarieerd om de wachttijden voor de voertuigen te minimaliseren. Hiervoor is een besturingsobject nodig welke de rijrichting bepaald, bijvoorbeeld afhankelijk van de tijd of de drukte in beide richtingen. Voor deze besturingsactiviteit zijn een aantal methoden ontwikkeld (*Hoofdstuk 5*), variërend van een periodieke omstelregel (vaste omsteltijd) tot een dynamische programmerings aanpak. We hebben theoretische resultaten afgeleid voor de gemiddelde wachttijd bij een periodieke omsteltijd regel met Poisson verdeelde aankomsten en positieve tussenafstanden tussen twee voertuigen.

Simulatie resultaten wijken voor dit geval nauwelijks af van de theoretische resultaten. De adaptieve besturingsregels, die rekening houden met informatie over wachtende voertuigen bij de ingang van de buis en beschikbare informatie over voertuigaankomsten, laten aanzienlijk betere resultaten zien in vergelijking met de starre periodieke regel. In dit geval is look-ahead informatie dus ook zeer waardevol. Een dynamisch programmerings regel blijkt bruikbaar in een praktijksituatie als het OLS.

Verder vond er een exploratief onderzoek plaats naar de secundaire processen (*Hoofdstuk 6*). Vragen die onderzocht werden, waren:

Hoe moet een AGV van stroom worden voorzien en wat zijn de gevolgen van de verschillende alternatieven met betrekking tot accu-laden/wisselen en accu-type?

Wat zijn de gevolgen van AGV en dock storingen?

Uit het onderzoek blijkt dat de gevolgen van deze secundaire processen voor de prestatie van het systeem en voor het benodigde aantal AGV's aanzienlijk kunnen zijn, afhankelijk van de gemaakte keuzes. Voor AGV en dock storingen konden wij de gevolgen laten zien bij een bepaalde storingsfrequentie, maar ook bepalen welke storingsfrequentie nog acceptabel is met het oog op de logistieke prestaties van het systeem. Vooral AGV storingen hebben een aanzienlijke invloed op de prestaties. Extra AGV's kunnen deze negatieve gevolgen maar ten dele opheffen.

Een flexibele, robuuste logistieke besturingsstructuur is noodzakelijk voor een succesvolle implementatie van grootschalige (externe) geautomatiseerde transport-netwerken. Dit proefschrift beschrijft het ontwerp en de evaluatie van een dergelijke besturingsstructuur en levert daarmee een bijdrage aan de huidige ontwikkelingen op dit gebied. De uitgevoerde simulatiestudies hebben een belangrijke rol gespeeld in het OLS project. Het aantonen van de logistieke haalbaarheid van 1 buis in plaats van 2 buizen op bepaalde trajecten leverde een daling op in investeringskosten van enige honderden miljoenen guldens. Een dergelijke beslissing heeft echter wel een serieuze invloed op de logistieke prestaties en het aantal benodigde AGV's. De simulatiebibliotheek bleek goed in staat verschillende layouts van het systeem, of verschillende input stromen met elkaar te vergelijken. De opgedane ervaring en kennis binnen het OLS project is dan ook zeer relevant voor de ontwikkeling van soortgelijke systemen.



# Curriculum Vitae

Mark Ebben was born on July 5<sup>th</sup> 1973 in Wierden, the Netherlands. In 1991 he finished secondary school at the Pius X College in Almelo. In the same year he started his study econometrics at the University of Groningen, where he specialized in operations research and logistics. In 1996 he graduated after completing his Master's thesis entitled 'A data-correcting algorithm applied to the Traveling Salesman Problem', under supervision of prof. dr. G. Sierksma and prof. dr. B. Goldengorin. In the same year he started as a PhD-student at the University of Twente, faculty Technology and Management under the supervision of prof. dr. A. van Harten and dr. M.C. van der Heijden. He started on a research project called 'Congestion oriented decision support systems for vehicle routing', but during this research the OLS project came along. This project produced many interesting research questions, resulting in this thesis on 'Logistic Control in Automated Transportation Networks'. From July 2001 he will work as a researcher at the University of Twente.