Dispatching and layout rules at an automated container terminal

Iris F.A. Vis[1]
Maurice Bakker
Vrije Universiteit Amsterdam, Faculty of Economics and Business Administration,
Room 3A-31, De Boelelaan 1105, 1081 HV Amsterdam, The Netherlands

Abstract

We study an automated container terminal in which Automated Guided Vehicles (AGVs) are used to internally transport containers and Automated Stacking Cranes (ASCs) to store and retrieve containers. In designing container terminals, the terminal management has to consider the choice for interrelated AGV and ASC dispatching rules and the layout of the stack. In this paper, we, therefore, examine the joint decision problem of dispatching containers to AGVs, selecting ASCs and the layout of the stack. In choosing a certain combination of dispatching rules and layout, we have considered criteria, such as, unloading times of a ship, number of vehicles required and utilisation of equipment. From this specific study, we can conclude that hardly any differences in unloading times can be obtained if we choose another combination of AGV and ASC dispatching rules. However, savings of approximately 20% in the number of AGVs can be obtained, if we apply the nearest-AGV-first rule in combination with the cyclic ASC rule. The results of the layout experiments only show small variations in unloading times, but they justify further research to the options for ASCs to span a wider stack. From the sensitivity analysis, it can be concluded that Quay Cranes should be distributed as evenly as possible over the ship and that twin-load AGVs can be used to obtain significant savings in unloading times and up to 50% reductions in the number of AGVs required.

Keywords: dispatching rules, layout, simulation, automated container terminal

1 Introduction

Since its introduction in the early 1960s, the containerised trade market has been growing rapidly. As a result, the capacity of ships has been extended to 6000 TEU or more (twenty-foot-equivalent-unit; length of a container is twenty feet). Currently, the sector benefits especially from the strong growth of the Chinese economy. To keep up with this trend and to provide the necessary capacity and customer service, ports and terminals should handle their operations efficiently. Especially in Europe, where the competition among the ports in the Le Havre-Hamburg range is fierce (see Kroon and Vis, 2005).

At ports and terminals containers are transshipped from one mode of transportation to another. To ensure an effective coordination of all activities at a terminal, information technology and automated control technology become more and more important. Johansen (1999) provides a detailed description of these technologies in container terminals.

The process of unloading and loading a ship at an automated container terminal is illustrated in Figure 1 and can be described as follows: when a ship arrives at the terminal, the containers need to be unloaded from the ship. Manned Quay Cranes (QCs) take the containers from the ship's hold and deck according to an unloading plan. Usually four QCs perform a common

[1]corresponding author, +31-20-5986067, fax. +31-20-5986005, ivis@feweb.vu.nl

unloading operation. The container is transferred from an QC to an automated transport vehicle (e.g. an automated guided vehicle or an automated lifting vehicle). These automated guided vehicles (AGVs) transport the containers along a predefined path, usually a loop layout, to the stack. This stack consists of a number of blocks where containers can be stored for a certain period of time. Automated Stacking Cranes (ASCs) store and retrieve containers in and from the stack. The retrieved containers are internally transported to other transportation modes, such as barges, trucks or trains. To load containers on a ship, these processes are executed in reverse order.

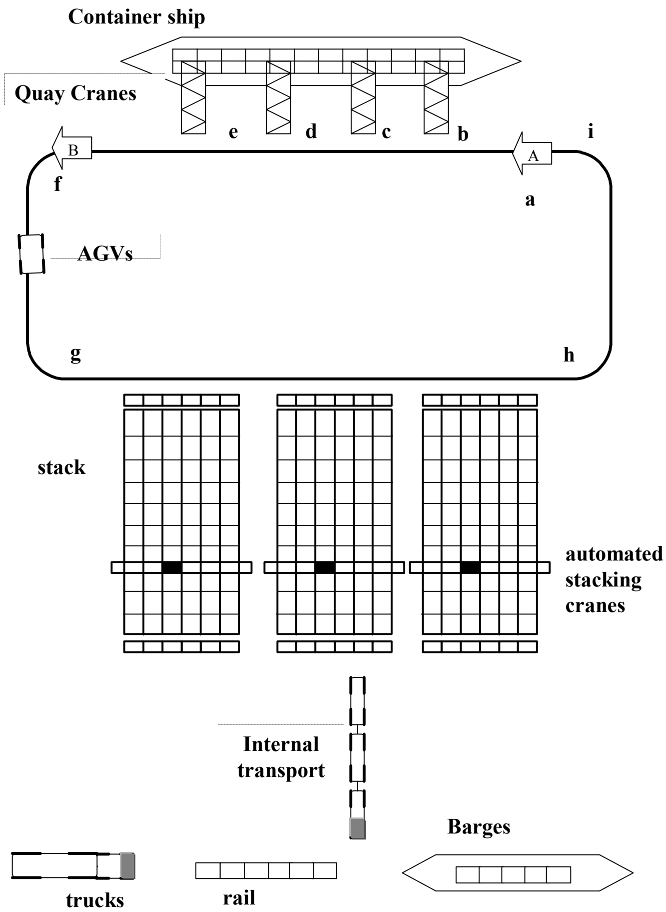

Figure 1: Example of processes and layout of an automated container terminal.

As can be seen in Figure 1, the stack separates deep-sea loading and unloading operations from landside (including short-sea and barges) loading and unloading operations. To ensure short berth times of the ships, the most critical planning and control problems arise at the deep-sea side. In this paper, we will, therefore, focus on this part of the terminal. We can distinguish between four planning and control levels in making decisions, namely the strategic, tactical, operational and the real-time level. At the strategic level it is, for example, decided where to locate a container terminal. The time horizon of decisions at this level covers one to several years. These decisions lead to the definition of a set of constraints under which the decisions at the tactical and operational level have to be made. At the tactical level it is, for example, decided which layout is chosen. The time-horizon of these decisions covers months to years. At the operational level, decisions such as the number of vehicles required, are made of which the time-horizon covers a day to a month. Finally, at the real-time level decisions are made each minute on, for example, the dispatching of containers to vehicles or to stacking cranes. The decisions at the various levels can be made sequentially, but for interrelated issues they can be treated as joint decisions. Vis and De Koster (2003) and Steenken *et al.* (2004) classify the main decision problems at a container terminal and present an overview of relevant literature.

From these overviews, it can be concluded that both analytical approaches and simulation are used to solve decision problems. By using simulation it is possible to solve problems that arise simultaneously at several levels and, furthermore, to investigate results that are obtained by integrating different material handling systems. For example, Van der Meer (2000) uses simulation to study the performance of various dispatching rules for AGVs at container terminals. Vis and Harika (2004) use a simulation model to compare the performance of automated guided vehicles and automated lifting vehicles. Some of the input data from that study are also used in this paper. However, the simulation model of this paper contains much more detail and will be used to compare various dispatching and layout rules.

Analytical models usually address one or two decision problems at the same time. For example, Welgama and Gibson (1996) propose an integrated methodology for simultaneously determining a layout and type of equipment used. Grunow *et al.* (2004) study the dispatching of multi-load AGVs at a container terminal. Kim and Bae (1999) study dispatching of containers to AGVs in a container terminal. The authors propose a mixed integer linear programming model and heuristics to dispatch containers to AGVs such that the delay of the ship is minimised. Bish *et al.* (2001) study an extension of this problem by also considering the storage location problem of containers. The authors propose an heuristic to solve the NP-hard problem of assigning containers to vehicles and to storage locations in the stack such that the total unloading time of the ship is minimised. In this paper, we extend this problem by studying the interrelated planning and control problems of dispatching unloaded containers to automated guided vehicles, the dispatching of these containers to storage locations in the stack and the layout of the stack. To decide for a certain dispatching policy or layout, we use several performance criteria such as the unloading times of the ship, the number of automated guided vehicles required and the utilisation of these vehicles.

Section 2 discusses the specifications of the simulation model. We describe in Section 3 how we have implemented the model in the simulation software. To study the effects of technical aspects (twin-load capacity of AGVs) and characteristics of the ship (type of containers and the distribution of QCs over the ship) on the dispatching rules, we have performed a sensitivity analysis in Section 5. By examining the results of Sections 4 and 5, we provide an advice in Section 6 on the choice for certain dispatching rules in relation to the layout of the stack at an automated container terminal.

2 Conceptual model

We consider an automated container terminal in which Automated Guided Vehicles (AGVs) are used for the transportation of containers and Automated Stacking Cranes (ASCs) for the handling of containers in the stack. In this paper, we only address the unloading process. Usually, Quay Cranes (QCs) first unload a ship and thereafter load containers on the ship. Unloading and loading operations are only mixed during a very short period of time. The way of modelling unloading and loading operations are quite similar. By using AGVs the sequence in which containers are handled is fixed (see also Vis and Harika, 2004). For all experiments with planning and control rules, we consider a similar situation and, therefore, we can compare the results and draw conclusions. In the next sub sections, we will describe the layout of the terminal and the various processes in more detail. The input data are obtained from the report of Celen *et al.* (1997) and from interviews with logistics managers of Europe Combined Terminals at the port of Rotterdam. Some of these data were also used in the study of Vis and Harika (2004).

2.1 Layout of the terminal

We consider an automated container terminal in which the QCs and ASCs are connected by a multiple lane loop layout. This loop is a fixed sequence of pick-up and delivery points at QCs and ASCs. AGVs travel over these fixed guidepaths to handle all transportation requests. During the unloading process AGVs travel full from an QC to an ASC and empty from an ASC to an QC. To avoid the common problems of a loop layout, in which vehicles can not pass each other, container terminals can use the concept of multiple lanes. We refer to Sinriech (1995) for an overview of approaches for guidepath design for AGVs.

Figure 1 and Table 1 indicate the layout of the terminal and the related distances (see also Vis and Harika, 2004). As can be seen, vehicles drive counterclockwise from the parking area A, where they receive a transportation request, from a certain QC. At arrow B, the vehicle is dispatched to an ASC. The applicable dispatching rules will be explained in the following sub sections.

	a	b	c	d	e	f	g	h	i
a	x	10	60	110	160	315	515	920	1120
b	x	x	50	100	150	305	505	910	1110
c	x	x	x	50	100	255	455	860	1060
d	x	x	x	x	50	205	405	810	1010
e	x	x	x	x	x	155	355	760	906
f	x	x	x	x	x	x	200	605	805
g	x	x	x	x	x	x	x	405	605
h	x	x	x	x	x	x	x	x	200
i	90	x	x	x	x	x	x	x	x

Table 1: Distances (in meters) in the terminal. An x indicates an infeasible path. The location of $a - i$ can be found in Figure 1.

2.2 Processes at the QC

We assume that 2500 containers need to be unloaded off the ship by four QCs. Due to space restrictions at both sides of the ship, containers are not equally distributed over the ship. Usually, 15% of the containers is designated for the leftmost QC, 35% for each of the two QCs in the middle and 15% for the rightmost QC. Furthermore, we need to generate the properties of the containers (see Hartmann, 2004). We only focus on the size of containers. Initially, we assume that 50% of the containers has a length of 20 feet (1 TEU) and the other 50% has a length of 40 feet (2 TEU).

Manned QCs unload the containers from the ship. The time (i.e. cycle time) required to unload a container off the ship and to position it on an AGV follows an empirical distribution (see Celen *et al,* 1997). This distribution is represented in Table 2. The average cycle time equals 65.9 seconds. Thus, on average an QC can unload 55 containers per hour. The total unloading time is determined by the two cranes at the middle of the ship, which each need to unload 875 containers. Therefore, a ship can be unloaded on average in 16 hours.

Fraction	Cycle time in seconds
0.05	30-40
0.15	40-50
0.25	50-60
0.20	60-70
0.17	70-80
0.11	80-90
0.04	90-120
0.02	120-150
0.01	150-180

Table 2: Empirical distribution of the cycle times of an QC

The moment an QC starts unloading a container, an AGV is selected from the parking area (arrow A, Figure 1) according to a certain dispatching rule. An overview of dispatching rules for AGVs is given in Vis (2005). In this study, we will compare the performance of the following QC-initiated dispatching rules (i.e. an AGV is selected by an QC from a set of idle AGVs to transport a container):

- *nearest-vehicle-first* (Egbelu and Tanchoco, 1984): a free AGV at the smallest distance is dispatched to the QC needing an AGV,
- *farthest-vehicle-first* (Egbelu and Tanchoco, 1984): a free AGV at the farthest distance is dispatched to the new QC.
- *random vehicle rule* (Egbelu and Tanchoco, 1984): the new pick-up container is randomly dispatched to any available AGV regardless the location of the AGV and the QC.
- *cyclic rule*: select the first available AGV beginning with the successor of the last AGV selected (to balance the workload among all AGVs in the system)
- *preferred order rule*: select the available AGV with the lowest unit number to transport the container

The assigned AGV travels to the QC and waits for the container. The QC positions the container on the AGV. It might also occur that the crane needs to wait for an AGV to arrive. In that case the QC needs to wait before it can unload a new container. After receiving the container, the AGV starts transporting the container to the stack. This part of the unloading process is given in a schematic way in the upper part of Figure 2.

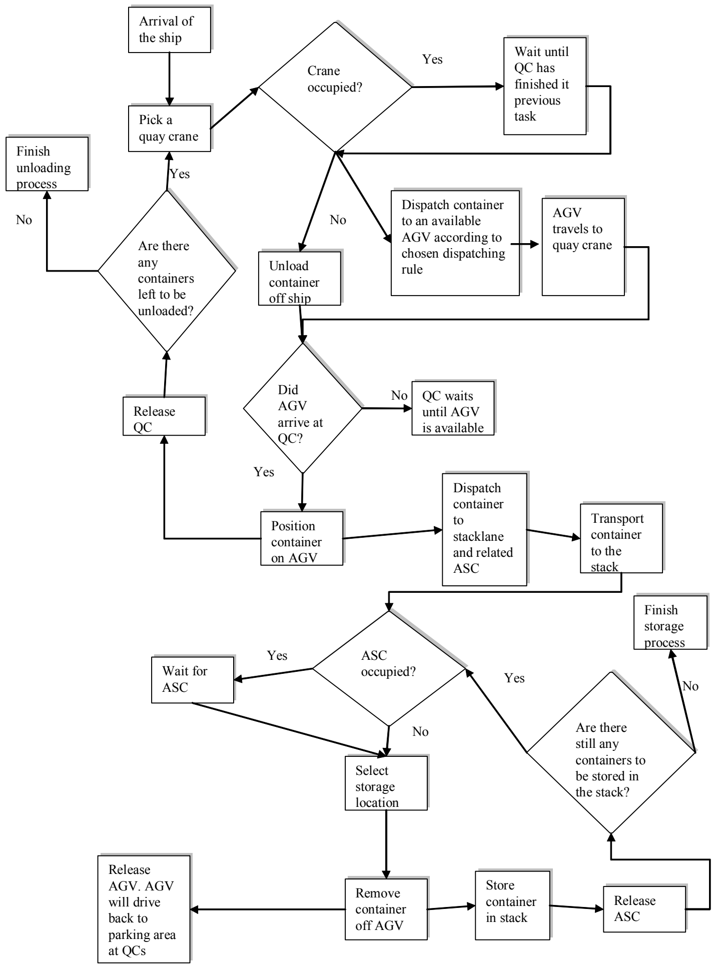

Figure 2: Conceptual model of unloading process at an automated container terminal

2.3 Transportation process

A variable number of AGVs is available to transport containers from the ship to the stack. The number of vehicles required to minimise the unloading times of the ship will be used as a performance measure in the experiments in Sections 4 and 5. Initially, we assume that the capacity of an AGV equals one twenty foot or one forty feet container. The speed of a full AGV equals 4 m/s and of an empty AGV 5.5 m/s.

The AGV with its container will already be dispatched at arrow B (see Figure 1) to an ASC with empty storage locations. In this way, we avoid that the AGV travels to an ASC that does not have free space for a container. In this paper, we will study the performance of the following ASC selection rules:

- *cyclic rule*: select the first available ASC with empty space beginning with the successor of the last ASC selected (to balance the workload among all ASCs in the system)
- *random rule*: randomly select an ASC with empty space
- *farthest/nearest ASC first:* select the ASC (with empty space) at the farthest/smallest distance and with no more than x AGVs on their way to this ASC. The optimal value of x will be determined in the various experiments.

After arrival at the stack, the container needs to be stored in the stack. If the ASC is still handling another container, the AGV needs to wait. The ASC lifts the container off the AGV and the empty AGV travels back to the parking area at the QCs. Also this part of the process is illustrated in Figure 2.

2.4 Processes at the ASC

The stack consists of a number of blocks. In this paper, we assume that 7 blocks have sufficient capacity to store all 2500 containers of a ship. Each ASC serves one block. Currently, the maximum number of parallel rows of containers in a block equals six (25 meters) due to the technical restrictions of the ASCs. The maximum stacking height is currently restricted to three containers. The exact layout of a block of containers needs to be determined. Initially, the layout of a block is set as follows: height of stacking equals 3 containers, length of a row equals 40 containers (1 TEU) and number of rows equals 6 (which is based on the layout at the port of Rotterdam). As a result, we assume a constant capacity of the block of 720 storage locations, which equals 720 1 TEU containers. Clearly, an 2 TEU container needs two storage locations. To determine an optimal layout we need to determine the value of three variables, namely height, width and length of the block. Clearly, if we decrease the number of rows and keep the height at three, we need to extend the length of each row to maintain the desired capacity. In Section 4.2 we will study the layout of the stack in more detail.

After arrival of the AGV at the pick-up and delivery point of the assigned block, the ASC lifts the container and stores it in the stack. First, we select one of the rows in the block according to an uniform distribution, in which each row has an equal probability to be chosen. Thereafter, a storage location is randomly selected from the empty locations in the selected row. The travel speed of an ASC equals 3.5 meters per second. This part of the process is also illustrated in Figure 2.

3 Implementation of the models

We have used Arena 7.01 as simulation software. In this section, we describe how we have implemented the various processes as discussed in the previous section. We will describe the following parts of the model in more detail:

- processes at the QCs (see section 3.1)
- processes at the ASCs (see section 3.2)

3.1 Processes at QCs

As indicated in Figure 2, the implementation starts at the QCs (see upper part of Figure 3). Each *CREATE* block generates containers which are *ASSIGNED* to one of the four QCs according to the distribution as given in Section 2. For each container it is *DECIDED* which size it has. Initially, we use a 50-50 distribution for 20 or 40 feet containers. The QC performs two tasks simultaneously, namely claiming an AGV to transport the container and the unloading of the container. These two issues are interrelated due to the fact that the QC can not finish the unloading of the container if no AGV is present. Therefore, we *DUPLICATE* the container. The original entity claims an AGV, whereas the duplicate is unloaded.

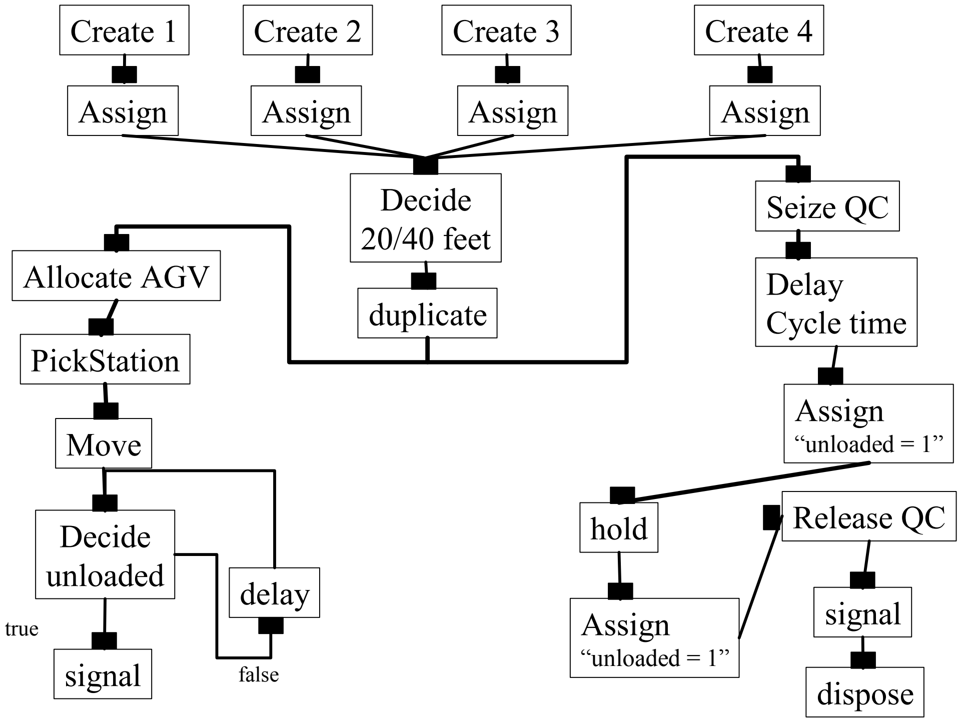

Figure 3: Implementation of processes at QCs

The right side of Figure 3 illustrates the unloading of the duplicate container. A container first needs to *SEIZE* an QC to be unloaded. The container is *DELAYED* during its cycle time which is drawn from the empirical distribution in Table 2. The value *one* is *ASSIGNED* to the variable "unloaded" to signal the AGV that it can receive the container. The QC with container is forced to *HOLD* until the AGV is present. After positioning the container on the AGV the value *zero* is *ASSIGNED* to the variable "unloaded". The crane is *RELEASED* and a *SIGNAL* is given that the next container can be unloaded. With *DISPOSE* we destroy the duplicate of the container such that only one container remains in the system to be transported.

The original container claims an AGV, which is illustrated in the left side of Figure 3. With *ALLOCATE* we dispatch an AGV to the container according to one of the dispatching rules as described in Section 2.2. With *PICKSTATION* the AGV is informed to which QC it should travel to pick-up the container. The AGV *MOVES* to the right QC and needs to *DECIDE* if the container can be received directly or that the AGV needs to wait for the QC. Each second it is *DECIDED* if the variable "unloaded" indicates the value zero (*DELAY* AGV until QC is ready) or one (*SIGNAL* of AGV to crane that unloaded container can be positioned on AGV). After receiving the container the AGV travels to the stack.

3.2 Processes at ASCs

Figure 4 represents the modelling of the selection of storage locations at the ASCs. The container needs to be assigned to a free storage location in the stack and related ASC according to one of the dispatching rules as described in Section 2.3. First, we *SEIZE* (with the chosen dispatching rule) one of the ASCs in the stack. Thereafter, we need to *ASSIGN* one of the rows to the container. Randomly, one of the storage locations (with a size of 1 TEU) in that row is *ASSIGNED* to the container. We need to be sure that the assigned storage location is empty. Therefore, we maintain a database in which we indicate if a storage location is empty or full. We need to consider two cases, namely storing an 20 feet or an 40 feet container. For an 20 feet it is just *DECIDED* if the chosen location is empty. For an 40 feet container we need to *DECIDE* if the neighbouring location with a corresponding height is also available. If the storage location(s) is (are) not empty, we need to choose another location. The value of the variable "number of times" is *ASSIGNED* to its *previous value* +1. Thereafter, a new row and location can be chosen. If we have performed this operation 5 times without success, we will select a new storage block. If the location is empty, the location (and if applicable, the neighbouring location) will be chosen and *ASSIGNED* the value full.

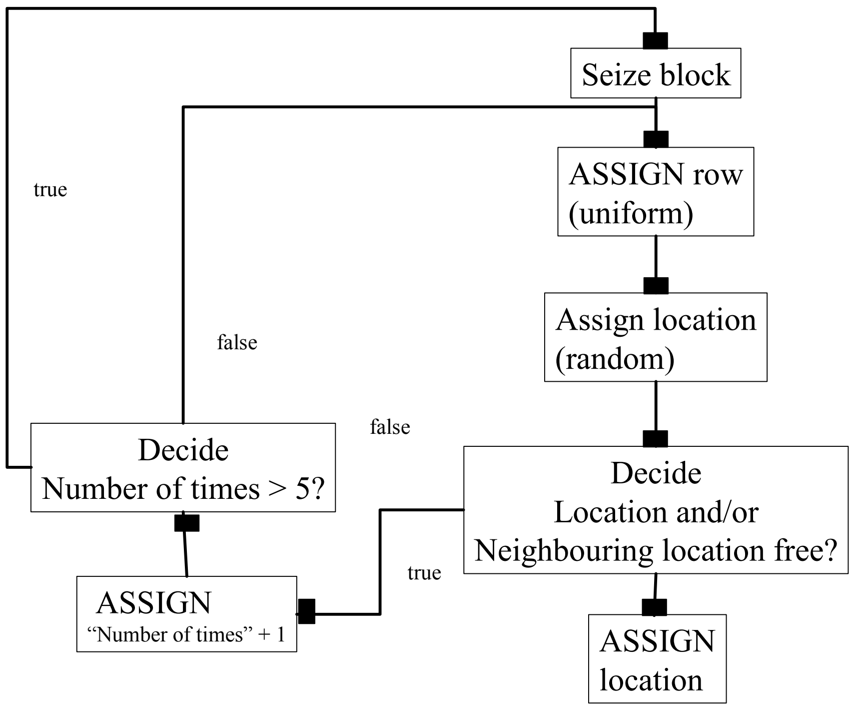

Figure 4: Implementation of selection of storage location at ASC

After arrival at the stack, an ASC needs to lift the container off the AGV. Similar to the processes at the QC, the container needs to perform two tasks, namely requesting storage by an ASC and releasing the AGV. The original container requests for transport and is stored at the assigned storage location. The duplicate frees the AGV, such that it can be dispatched again to an QC. Therefore, a logical loop checks with a low priority each second if there is a free AGV at the ASC and requests the free AGV to travel back to the parking area at the QCs. The duplicate container is disposed.

4 Results

The simulation of the unloading of a ship can be considered as a terminating system. The model has a clear start (arrival of the ship) and end (all containers have been unloaded). To obtain a high level of accuracy (relative error smaller than 2% with a probability of 95%) in the results, we need to perform a sufficient number of replications of each experiment. For all experiments in this paper, we have calculated that a replication size of 100 is sufficient (see formula of Law and Kelton, 1991). Thus, for each simulation study we generate 100 experiments and determine the average value among these experiments. On average, the computation times for 100 replications on a Pentium 4 PC with 1.50 GHz are 7 minutes.

4.1 Control of equipment

In this section, we study the performance of the various AGV dispatching rules in combination with the ASC selection rules, which are described in Sections 2.2 and 2.3. Summarising, we study a ship with 2500 containers, which will be unloaded by four QCs. The left and right QC handle each 15% of the containers. The rest is unloaded by the two other QCs. Seven ASCs serve each a block of containers, which has a height of three, length of 40 and width of six containers. The objective is to minimise the time a ship needs to stay at the terminal. In other words, we want to minimise the unloading times. All other processes in the terminal are inferior to the QCs, which are the bottleneck in the terminal. Therefore, we will compare these rules with the following performance measures:

- *total cycle time* required to unload all containers off the ship and store them in the stack
- *minimum number of AGVs* required to achieve a minimal total cycle time
- *average utilisation* of the AGVs

First, we examine the various AGV dispatching rules in combination with the *cyclic selection* of ASCs. We have varied the number of AGVs from 24 to 36, which corresponds to 6 to 9 AGVs per QC. These numbers are quite realistic in practice. Figure 5 represents for each of the AGV dispatching rules the total cycle times in seconds for a varying number of AGVs.

From Figure 5 we can conclude that the lowest total cycle time can be obtained if we apply the *nearest-vehicle-first rule*. The related cycle time equals 58665 seconds, which corresponds to the deterministic estimate of 16 hours in Section 2.2. However, the differences in cycle times with the *random and cyclic vehicle rule* are small (less than 1%). Contrary, the differences in the minimum number of vehicles required to reach these minimal cycle times are larger (approximately 20%). Namely, while applying the *nearest-vehicle-first* rule, we need 29 AGVs. 34 and 35 AGVs are respectively required for the *random and cyclic rule*. If we incorporate costs of AGVs, which are 318,000 Euro per AGV (estimate of ECT at the end of the year 2000), the difference between the rules becomes more significant. However, the mutual differences in cycle times with a varying number of AGVs for a certain dispatching rule are small. For example, if we allow an extra unloading time of 12 minutes, we can also apply the *random rule* also with 29 AGVs.

The average utilisation (see Table 3) of the AGVs becomes lower if we use more AGVs. If we apply the *nearest-vehicle-first rule*, AGVs are occupied for almost 50% of their time. Therefore, it might be expected that total cycle times do not rise sharply due to unexpected events in the AGV system. Both the *preferred-order rule* and the *farthest-vehicle-first rule* have a lower performance than the other rules. On average, the total cycle times are 3% higher. Furthermore, it can be noticed that adding extra vehicles does not impact the total cycle times.

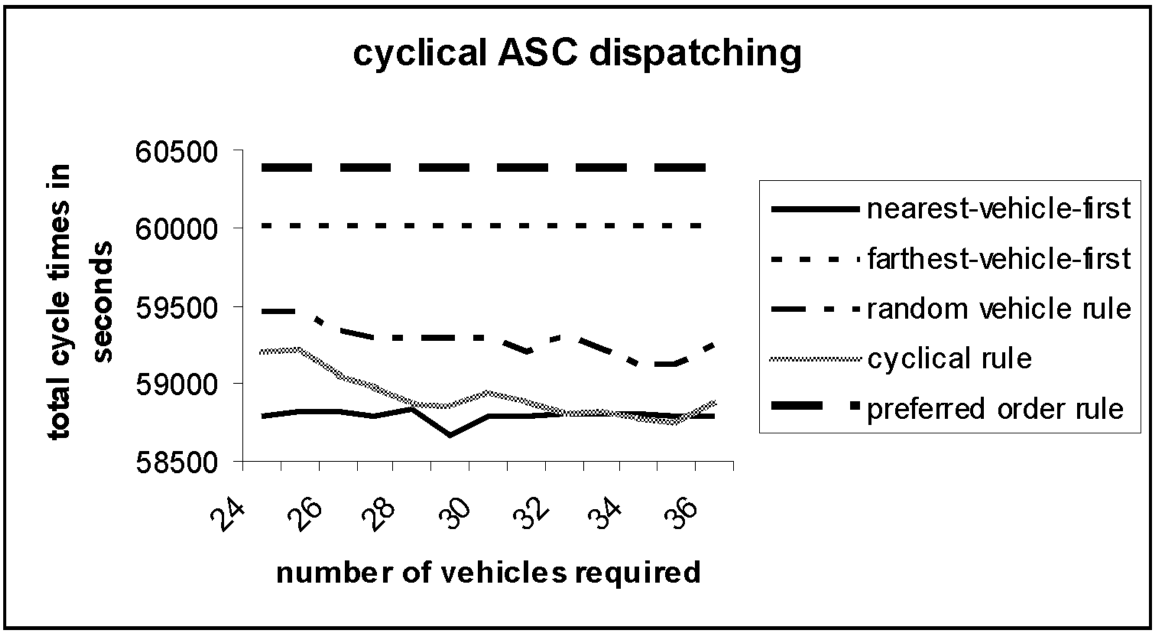

Figure 5: Total cycle times in seconds for a varying number of AGVs for each of the vehicle dispatching rules if we assign vehicles to ASCs according to the cyclic ASC rule.

Figure 6 represents the results of combing the *random ASC selection rule* with the various AGV dispatching rules. We notice similar results as with the *cyclic ASC* rule. If we compare the *cycle and random rule* it can be concluded that in both cases a minimal cycle time is obtained by applying the *nearest-vehicle-first rule.* The smallest cycle time is obtained with the random ASC rule. However, the difference in the total cycle time with the *cyclic rule* is smaller than 1.30 minutes. Furthermore, we need five more AGVs to reach this cycle time.

The final ASC selection rule is the *farthest/nearest ASC first.* We will first study the performance of the *farthest ASC rule* and thereafter decide if it is still interesting to study the nearest ASC rule. As described in Section 2.3, we need to decide how many vehicles ($= x$) may be on their way to the same ASC. Clearly, we want to avoid waiting lines of AGVs at the ASCs. Therefore, we calculate for a deterministic setting the value of x. The average time for an ASC to store a container and return to the pick-up and delivery point equals 70 seconds (2* distance middle of stack/speed). The average travel time of an AGV from the decision point f to the end of the stack h (see Table 1) equals 150 seconds. As a result, no more than two AGVs should be on their way to the same ASC. Therefore, we test the farthest ASC rule with $x = 0, 1$ and 2. From the results in Table 3, we conclude that in the stochastic environment, waiting lines of AGVs at the stack arise and that as a result the average utilisation of AGVs and total cycle times increase. Only with $x = 0$, some acceptable results can be obtained. However, both the *cyclic and random ASC rule* outperform the *farthest ASC rule* in total cycle times and number of vehicles required. We expect similar results for the *nearest ASC rule* and, therefore, we did not perform these experiments.

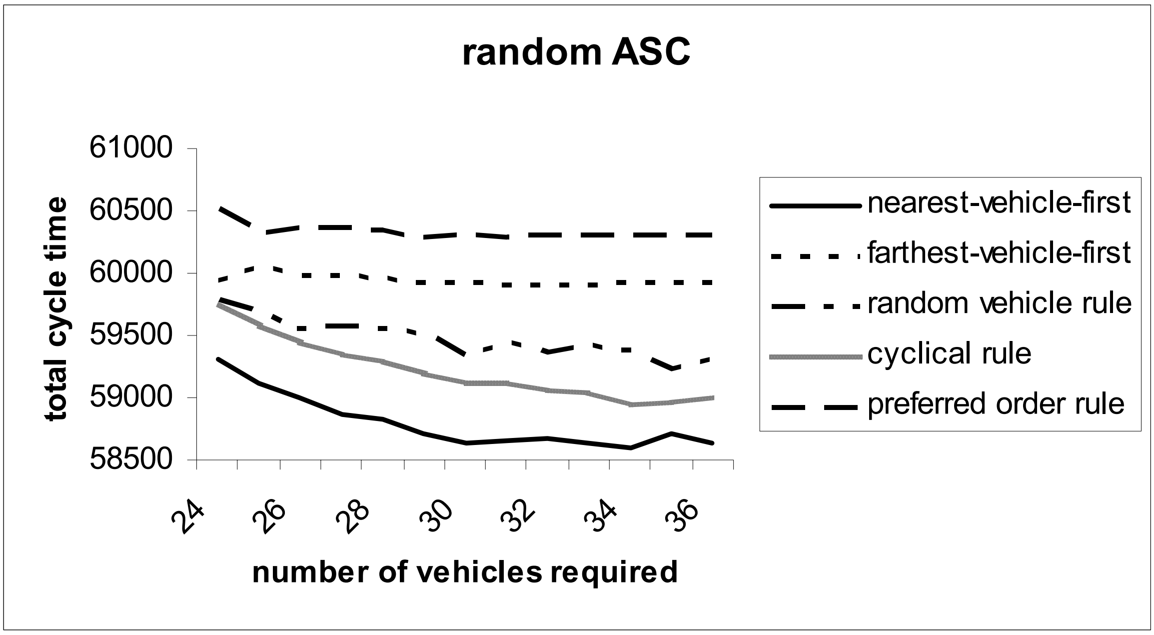

Figure 6: Total cycle times in seconds for a varying number of AGVs for each of the vehicle dispatching rules if we assign vehicles to ASCs according to the random ASC rule.

ASC/AGV	**nearest**	**farthest**	**random**	**cyclic**	**preferred**
cyclic	58758/29/0.49	60371/25/0.54	59264/34/0.41	58865/35/0.40	60004/24/0.57
random	58593/34/0.45	60284/29/0.51	59183/35/0.43	58937/34/0.45	59899/31/0.48
farthest $x = 0$	59232/36/0.48	60761/34/0.49	60011/35/0.49	59539/35/0.49	60476/35/0.48
farthest $x = 1$	60152/34/0.53	61366/34/0.50	59831/42/0.45	60072/36/0.49	60810/38/0.46
farthest $x = 2$	62165/35/0.57	62669/38/0.53	62214/34/0.57	62016/36/0.55	62462/37/0.54

Table 3: Results $x/y/z$ for all combinations of ASC and AGV dispatching rules.
Whereas, $x =$ minimal total cycle time in seconds,
$y =$ minimum number of AGVs and $z =$ average utilisation of AGVs

Table 3 summarises the minimal cycle times with the related number of vehicles and utilisation for all combinations of AGV dispatching and ASC selection rules. Based on these results, we have decided to concentrate on the *nearest-vehicle-first, random and cyclic rule* for dispatching AGVs and on the *cyclic and random rule* for selecting ASCs. However, more experiments are required to decide in favour for a certain combination of these rules. In the next sub section, we also incorporate the layout of the stack.

4.2 Layout of the stack

In the previous experiments, we assumed a fixed layout of the block, which was mainly determined by the technical restrictions of the ASCs. In the following experiments, we assume that the ASC can stack one container higher and span a wider block of containers. To compare the results of the various rules in a stack with varying sizes, we need to keep the storage capacity constant at 720 containers. We vary the width (e.g. number of rows r) of the block from three to ten containers ($3 \leq r \leq 10$). The height of the stack h varies between 2 and 4 ($2 \leq h \leq 4$). The number of locations l in a row easily follows from the values of r and h. Namely, $l = \frac{720}{r \cdot h}$.

We have executed the experiments for all combinations of dispatching rules with their optimal number of vehicles (see Section 4.1). As main performance measure, we have used the total cycle time of the ship. We study the performance of the nearest-vehicle-first rule in combination with the cyclic ASC rule in more detail in Table 4. From the results in Table 4, we can conclude that there is a difference in cycle times of less than 1% between the best and the worst layout. The worst layout ($r = 4, l = 45$ and $h = 4$) has a small number of rows with each a relatively high number of containers, whereas the best layout ($r = 8, l = 30$ and $h = 3$) consists of a wide span with a smaller number of containers in each row. The difference in cycle times with the usual layout, as described in Section 2.4 is less than 0.5%.

# rows/height	**2**	**3**	**4**
3	58745	58680	58876
4	58755	58757	58890
5	58789	58733	58708
6	58685	58758	58760
7	58676	58738	58743
8	58753	**58650**	58746
9	58808	58690	58876
10	58790	58775	58763

Table 4: Total cycle times for nearest-vehicle-first in combination with cyclic ASC for a varying layout with 29 AGVs

The results for the other combinations of AGV dispatching and ASC selection rules are quite similar. As a result, we need to conclude that total cycle times are quite insensitive for the layout of the stack. Table 5 represents an overview of the best layouts for all combinations of dispatching rules. In our opinion, the results in Table 5 justify further study for the technical aspects of an ASC. The first focus in these researches should be more on the span width of the ASC than on the height of stacking.

AGV/ASC	**nearest**	**random**	**cyclic**
cyclic	8/30/3, 58650	5/72/2, 59136	8/45/2, 58753
random	6/40/3, 58593	9/27/3, 59131	9/40/2, 58907

Table 5: Results $r/l/h$, x, for combinations of ASC and AGV dispatching rules where as, r = number of rows, l = length of each row, h = height of each row and x = minimal total cycle time in seconds

5 Sensitivity analysis

To give a better advice on the choice for a certain combination of an AGV dispatching rule and an ASC selection rule at a container terminal, we have performed a sensitivity analysis. We have varied one of the variables in the model while the value of the other variables remains similar to the value indicated in Section 2 . The following aspects have been studied:

- distribution of QCs over the ship
- twin-load capacity of AGVs

- type of containers

We present the results of the various studies in the next subsections.

5.1 Distribution of QCs over the ship

As indicated in Section 2.2 containers, and as a result QCs, are not equally distributed over the ship. In the previous experiments, we assumed a 15/35/35/15 distribution. We will study the effect on the total cycle times for the various dispatching rules if we vary this distribution. The following experiments have been performed:

- 17/33/33/17%
- 20/30/30/20%
- 22/28/28/22%

For each of the scenarios we have determined the minimum total cycle time with the corresponding number of vehicles required to reach this cycle time. From the results in Table 6, we can conclude that the total cycle times sharply decrease if the leftmost and the rightmost QC handle more containers. The total cycle time is still determined by the two cranes in the middle (see Section 2.2). However, their workload has dropped from 875 (35%) to 700 (28%) containers. As a result, it can be noticed that total cycle times decrease with 25%.

QC	**ASC**	**nearest**	**random**	**cyclical**
15/35/35/15	**cyclical**	58758/29	59264/34	58865/35
15/35/35/15	**random**	58593/34	59183/35	58937/34
17/33/33/17	**cyclical**	55335/31	55763/37	55527/29
17/33/33/17	**random**	55303/35	56041/35	55570/36
20/30/30/20	**cyclical**	50323/27	51037/34	50475/35
20/30/30/20	**random**	50279/36	51179/35	50675/35
22/28/28/22	**cyclical**	46964/31	47743/35	47258/35
22/28/28/22	**random**	47024/32	47807/38	47373/36

Table 6: Results x/y for all combinations of ASC and AGV dispatching rules.
Whereas, $x =$ minimal total cycle time in seconds and
$y =$ minimum number of AGVs

Furthermore, it can be concluded that the *nearest-vehicle-first rule* outperforms the other two AGV dispatching rules. The *random ASC rule* results in almost all cases in lower cycle times, while being combined with the *nearest-vehicle-first rule*. However, it still needs to be noticed that we need far more vehicles than with the *cyclic ASC rule (*on average *16.5%).* If we approach an equal distribution of containers over the ship, the *random and cyclic ASC rule* reach an equal performance with a comparable number of AGVs. This can be explained as follows: with an equal distribution of containers over the ship, there is a constant flow of containers from the ship to the stack instead of a busy period of 7 hours in which 4 QCs are sending containers to the stack and a quiet period of 9 hours with only 2 QCs in operation. In the first situation, the workload is balanced over all ASCs for both the cyclic and random selection rule.

5.2 Twin-load capacity of AGVs

From a technical and economical perspective, it is also interesting to study the impact on total cycle times and the number of vehicles required if one AGV can transport two 1 TEU containers simultaneously. 2 TEU containers are still transported one at a time. Just as in the other experiments, we assume a 50-50% distribution of 1 and 2 TEU containers. We study the impact on dispatching rules and cycle times by decreasing the number of twin-load AGVs.

# AGVs	nearest	random	cyclical
35	-	-	58336
34	-	58385	-
29	58225	58306	58337
25	58231	58425	58378
20	58726	58816	58696
15	62245	62373	62252
10	77606	77330	77644

Table 7: minimum cycle times for a varying number of twin-load AGVS while combining AGV dispatching rules and the cyclic ASC rule

Table 7 shows for each of the vehicle dispatching rules and the *cyclic ASC rule*, total cycle times for a varying number of AGVs. The results for the *random ASC rule* are quite comparable and, therefore, not included in this paper. The number of AGVs in the experiments have been started at the optimal number of vehicles as determined in Section 4.1 and have been decreased to 10. The minimal total cycle time decreases on average with 0.1%. Furthermore, it can be noticed that a decrease of, on average, 19% in the number of AGVs can be obtained. For *random and cyclic AGV* dispatching we need to use 29 AGVs. For the *nearest-vehicle-first rule*, 25 AGVs are required to minimise the total cycle time. We might have expected a larger decrease in the number of vehicles. However, on average we have a repeating pattern of unloading an 1 TEU and an 2 TEU container. As a result, on average each 1 TEU container on an AGV needs to wait twice the average cycle time (=132 seconds) before the transport can start. Furthermore, each 2 TEU container requests its own AGV. The *nearest-vehicle-first rule* performs better than the two other rules. The difference in cycle times between the *nearest-vehicle-first rule* and the *cyclic vehicle rule* increases.

5.3 Types of containers

In the previous experiments, we assumed a 50-50% distribution of 1 and 2 TEU containers. In this section, we will perform experiments in which we vary the amount of 1 TEU containers between 40 and 100%. We will study the effect on the total cycle times and the number of vehicles required. For all vehicle rules, we run the model for 29 and 15 AGVs (decision based on the results in Section 5.2). Table 8 represents the results for the *cyclic ASC rule*. The results for the *random ASC rule* are comparable to the results of the *cyclic ASC rule* and, therefore, we did not include them in the paper.

% 1 TEU containers	number of AGVs	nearest	random	cyclic
40%	29	58232	58547	58297
40%	15	63554	63537	63554
50%	29	58225	58306	58337
50%	15	62245	62373	62252
60%	29	58231	58381	58313
60%	15	61183	61140	61243
70%	29	58266	58233	58235
70%	15	60187	60131	60223
80%	29	58237	58349	58237
80%	15	59321	59402	59386
90%	29	58310	58285	58310
90%	15	58621	58705	58649
100%	29	58277	58290	58277
100%	15	58243	58293	58240

Table 8: Total cycle times for the combination of the cyclic AS rule with various AGV dispatching rules for a varying amount of 1 and 2 TEU containers while using respectively 29 and 15 AGVs

From the results in Table 8, we can conclude that we can decrease the number of AGVs if more 1 TEU than 2 TEU containers need to be transported. If we need to transport 90% or more 1 TEU containers it is even possible to use half the number of vehicles to obtain total cycle times which are similar to the total cycle times with single load vehicles. If we need to transport more 2 TEU than 1 TEU containers, the total cycle times and number of vehicles required are similar to the case of single load vehicles. Also in these experiments, the results for the various dispatching rules are close. Furthermore, if we need to transport only 1 TEU containers on twin-load AGVs, there is no difference in performance between the various dispatching rules.

Summarising, we can conclude that twin-load AGVs, regardless the amount of 1 and 2 TEU containers to be transported, do not impact the performance of the various dispatching rules. However, significant savings in the number of AGVs required can be obtained if more 1 TEU than 2 TEU containers need to be transported. Based on these conclusions, we strongly recommend terminal managers to make use of the technical option of AGVs to transport two 1 TEU containers simultaneously during the unloading process.

6 Conclusions

At automated container terminals, containers are transshipped from one mode of transportation to another. Manned Quay Cranes unload containers off the ship. The containers are transported to the stack by Automated Guided Vehicles and stored in the stack by Automated Stacking Cranes. In this paper, we have studied the joint decision problem of dispatching AGVs to containers, selecting ASCs and related storage locations and the layout of the stack. The performance of all combinations of the various AGV-dispatching rules and ASC selection rules and various layout configurations have been studied based on total cycle times, number of vehicles required and utilisation of the AGVs. Furthermore, we have varied the setting of the terminal by studying different distributions of QCs over the ship, usage of twin-load AGVs and different distributions of 1 and 2 TEU containers.

Summarising, we can conclude for all different experiments that the choice for a certain AGV dispatching rule hardly impacts the total cycle times. However, the amount of vehicles required differs per dispatching rule. While considering both the unloading times and the number of vehicles required, we recommend for all different settings the *nearest-vehicle-first rule*. For the selection of the ASC, we suggest that the workload should be evenly distributed over the various ASCs (*cyclic ASC rule*) instead of dispatching AGVs randomly or to the nearest or farthest ASC. The layout of the stack slightly impacts the cycle times. It should be studied in more detail, if it is interesting to invest in ASCs that can stack wider. If sufficient space is available in the terminal, we can conclude that it seems not to be efficient to stack higher than three containers. From the results in Section 5, we conclude that it should be tried to distribute QCs as evenly as possible over the ship to obtain significant decreases in cycle times. Finally, it is wise to use the technical property of twin-load AGVs to transport two 1 TEU containers at the same time, such that transportation costs and annual depreciations of the vehicles can be reduced significantly.

References

Bish EK, Leong TY, Li CL, Cheong Ng JW, Simchi-Levi D (2001) Analysis of a new vehicle scheduling and location problem. Naval Research Logistics 48: 363-385

Celen, HP., Slegtenhorst, RJW, Ham, RTh, van der, Nagel, A, Vos Burchart, R de, Berg, J, van den Evers, JJM, Lindeijer, DG, Dekker, R, Meersmans, PJM, Koster, MBM, de, Meer, R, van der, Carlebur, AFC, Nooijen, FJAM (1997) FAMAS NewCon. Concept TRAIL Report, Technical University of Delft and Erasmus University Rotterdam, The Netherlands

Egbelu, PJ.,Tanchoco, JMA. (1984) Characterization of automatic guided vehicle dispatching rules. International Journal of Production Research 22(3): 359-374

Grunow M, Günther HO, Lehmann M (2004) Dispatching multi-load AGVs in highly automated seaport container terminals. OR Spectrum 26: 211-235.

Hartmann, S (2004) Generating scenarios for simulation and optimization of container terminal logistics. OR Spectrum 26: 171-192.

Johansen, RS (1999) Gate solutions, paper presented at the Containerport and Terminal Performance Conference, February, Amsterdam.

Kim, KH, Bae, JW (1999) A dispatching method for automated guided vehicles to minimize delays of containership operations, International Journal of Management Science 5 (1): 1-25.

Kroon R, Vis IFA (2005) Seriously Ceres. Research Memorandum 2005-3, Faculty of Economics and Business Administration, Vrije Universiteit Amsterdam, Amsterdam, The Netherlands.

Law, AM., Kelton, WD (1991) Simulation Modeling & Analysis, second edition. McGraw-Hill International Editions, New York

Sinriech, D (1995) Network design models for discrete material flow systems: a literature review, International Journal of Advanced Manufacturing Technology 10: 277-291.

Steenken D., Voss S., Stahlbock R. (2004) Container terminal operation and operations research - a classification and literature review. OR Spectrum 26: 3-49.

Van der Meer, JR (2000) Operational control of internal transport. ERIM Ph.D. Series Research in Management 1, Erasmus University Rotterdam, The Netherlands.

Vis IFA (2005) Survey of research in the design and control of automated guided vehicle systems. European Journal of Operational Research, in press.

Vis IFA, Harika I (2004) Comparison of vehicle types at an automated container terminal. OR Spectrum 26: 117-143.

Vis IFA, De Koster R (2003) Transshipment of containers at a container terminal: An overview. European Journal of Operational Research 147: 1-16.

Welgama, PS, Gibson, PR (1996) An integrated methodology for automating the determination of layout and materials handling system, International Journal of Production Research 34(8): 2247-2264.