# Scheduling vehicles in automated transportation systems : algorithms and case study

OR Spectrum (2002) 24: 31–58

OR Spectrum

# Scheduling vehicles in automated transportation systems

## Algorithms and case study[*]

**Matthieu van der Heijden, Mark Ebben, Noud Gademann, and Aart van Harten**

University of Twente, Centre for Production, Logistics and Operations Management,
Faculty of Technology and Management, P.O. Box 217, 7500 AE Enschede, The Netherlands
(e-mail: m.c.vanderheijden@sms.utwente.nl)

Received: June 21, 2000 / Accepted: January 22, 2001

**Abstract.** One of the major planning issues in large scale automated transportation systems is so-called *empty vehicle management*, the timely supply of vehicles to terminals in order to reduce cargo waiting times. Motivated by a Dutch pilot project on an underground cargo transportation system using Automated Guided Vehicles (AGVs), we developed several rules and algorithms for empty vehicle management, varying from trivial First-Come, First-Served (FCFS) via look-ahead rules to integral planning. For our application, we focus on attaining customer service levels in the presence of varying order priorities, taking into account resource capacities and the relation to other planning decisions, such as terminal management. We show how the various rules are embedded in a framework for logistics control of automated transportation networks. Using simulation, the planning options are evaluated on their performance in terms of customer service levels, AGV requirements and empty travel distances. Based on our experiments, we conclude that look-ahead rules have significant advantages above FCFS. A more advanced so-called *serial scheduling method* outperforms the look-ahead rules if the peak demand quickly moves amongst routes in the system.

**Key words:** Freight transportation – Vehicle scheduling – Simulation

[*] We thank the Dutch Centre for Transportation Technology (CTT) for their funding of the simulation study that has been the basis of our research results. CTT is initiator and coordinator of the project to design and develop the underground logistics system around Amsterdam Airport Schiphol that has been used as a case study in this paper.

*Correspondence to:* M. van der Heijden

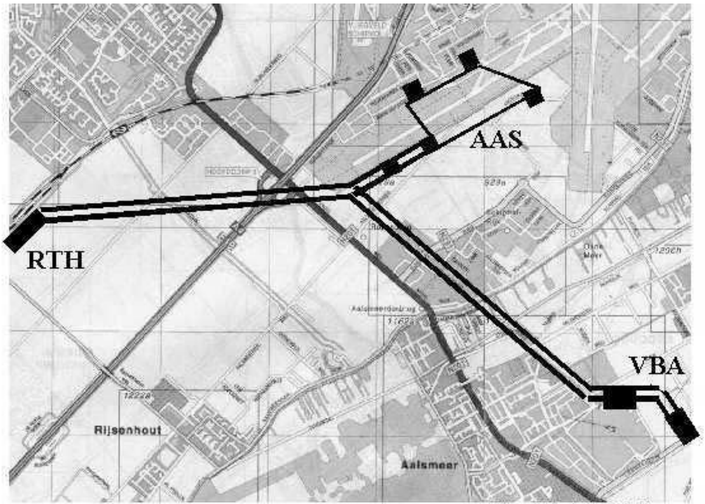
## 1 Introduction

As in many countries, traffic congestion is becoming a severe problem in the Netherlands, causing delay to both private and commercial transportation. Aiming to be a main hub for Western Europe, this forces the Netherlands to search for solutions in order to guarantee rapid processing of transportation orders within and through the country, particularly when the freight supply keeps growing as has occurred in the past decade.

One of the options currently considered is to move part of the freight underground via a fully automated transportation system. In several governmental studies, the technical and economical feasibility of such a system has been established. As a next step, a pilot system is currently developed, focussing on the processing of time-critical products between Amsterdam Airport Schiphol, the flower auction in Aalsmeer and a future rail terminal in Hoofddorp (cf. van der Heijden et al., 2000). All three locations are situated in the western part of the Netherlands. An example of time-critical transportation is the supply of flowers to the auction in Aalsmeer, which have to arrive in time in order to be processed the same day. Also, the transportation of export flowers to Amsterdam Airport Schiphol is critical, as delay may cause that the cargo arrives when the plane has already left. Because the automated transportation system should be rapid and especially reliable, a competitive advantage on other transportation modes, such as traditional door-to-door road transportation, can be obtained.

Such an automated transportation system consists of a number of terminals, connected by an underground tube system (see Fig. 1). The situation in Figure 1 is the primary layout option at the time of our research for the automated transportation network around Amsterdam Airport Schiphol. Automatic Guided Vehicles (AGVs) carry cargo between terminals in standardised load units (air pallets, flower mid boxes). Order patterns are usually time dependent and may vary over days in the week and over hours on a day. Each terminal consists of a number of docks where vehicles can be loaded or unloaded. Transportation is constrained by arrival times and due times. These due times can be met by a combination of sufficient resources (vehicles, terminals, docks) and a set of logistics planning and control rules. Of course, both issues are interrelated, because efficiency gained by clever planning and control rules leads to reduced capacity requirements. On the other hand, simple myopic or look-ahead rules are usually easier to implement, require less information exchange and are more robust to disturbances.

The network as shown in Figure 1 consists of three main locations, namely Amsterdam Airport Schiphol (AAS) in the north consisting of five terminals, Aalsmeer flower auction (VBA = “Verenigde Bloemenveiling Aalsmeer”) in the east consisting of two terminals and Rail Terminal Hoofddorp (RTH) in the southwest consisting of one terminal. AGVs that are not needed for a while or for which there is no room in the local parking are dispatched to a central parking that is located just south of Schiphol Airport, close to the intersection. All terminals and the central parking will be located at the surface, while just the tube system is underground (about 15 meters below the surface). Therefore slopes are planned between each terminal entrance / exit and the tube system. AGVs drive slower on these slopes



Copyright Andes Eindhoven

**Fig. 1.** A layout option for the automated transportation system around Amsterdam Airport Schiphol

than in the tube, which is included in the simulation model. The slopes have a length of approximately 150 meters each.

A large part of the system has two-way traffic, only the five terminals at Schiphol Airport are connected by a loop where only one-way traffic (counter clockwise) is possible. To give an indication for the scale, the distance between RTH and VBA is approximately 10 km. Because of legal speed limitations on the terminals, a small terminal size is more efficient regarding throughput times. Furthermore, the terminal size has significant impact on construction costs, in particular in the case of subterranean construction. Therefore, parking capacity in a terminal is small, both in local parkings and at docks.

In this paper we shall discuss various options for the logistics planning and control of an automated transportation network as described above. Key planning decisions include:

1) empty vehicle management, i.e. the pre-positioning of empty vehicles in the network to anticipate known and predicted demand;
2) order release, i.e. the assignment of transportation orders to load docks in the terminal of origin;
3) order scheduling, i.e. the assignment of empty vehicles to loads at docks when starting a transportation job and the assignment of loaded vehicles to docks after arrival at the terminal of destination;
4) task allocation to docks, i.e. the assignment of load and/or unload activities. Depending on the logistics control concept chosen, these decisions can be taken separately or (partly) simultaneously.

Although we shall discuss a decision structure containing all these questions, this paper focuses on the issue of empty vehicle management. 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 vehicle travel and loaded vehicle travel. Especially, we are 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 orders?
- To which extent can central coordination contribute to a more efficient system, compared to decentralised planning and control?

These issues will be discussed for general transportation networks. The pilot project as discussed above serves as a test case. For ease of reference, we will describe the methodology in relation to our case study. We constructed an object-oriented simulation model based on the object-oriented simulation software eM-Plant and on the Logistics Modeling Framework as defined by Van der Zee (1997). This framework gives guidelines for a systematic construction of logistics simulation models. We used this model to test and evaluate the various options for empty vehicle management.

The remaining part of this paper is organised as follows. In the next section, we present the model for the transportation network, discuss the assumptions and describe the case study to which we applied our decision structure. An overview of related literature is given in Section 3. Section 4 deals with a decision structure of the logistics planning and control. Various options for empty vehicle management, the focus of our paper, are presented in Section 5. The simulation study with numerical results, based on the case study, is the subject of Section 6. Finally, we summarise our main conclusions and give some directions for further research (Sect. 7).

## 2 Model and assumptions

We consider a closed transportation network consisting of a fixed number of origin / destination nodes (terminals), connected by an asymmetric road network, see Figure 1 for an example. Each terminal consists of a fixed number of docks, where vehicles can load or unload (one vehicle at a time), (see Fig. 2 for an example). The vehicles transport loads from the terminal of origin to the terminal of destination. Each vehicle can handle only one load at a time. A load may result from the consolidation of several small volume orders with the same origin and destination and similar due times. However, this is a separate decision that is not discussed in this paper. The route between a given origin-destination pair is fixed and determined in advance using a common shortest path algorithm. In the presence of uni-directional flow paths (as is the case in the pilot project), the distance matrix is asymmetric. When a vehicle is not needed for a while, it may be parked in the terminal (local parking) or on a separate parking lot (a central empty vehicle buffer) that can be considered as a network node.

Transportation jobs arrive in the system (e.g. as a result of train arrivals carrying multiple loads to be transported). The orders may be (partly) known some time before the actual load arrival in the system. Each load has a specific *due time* before

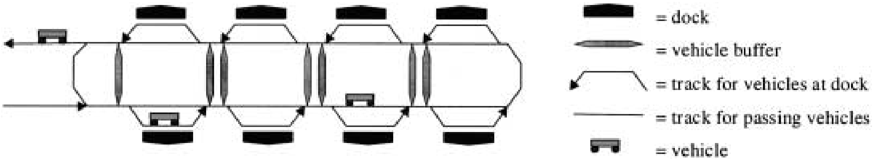


**Fig. 2.** Example of a terminal

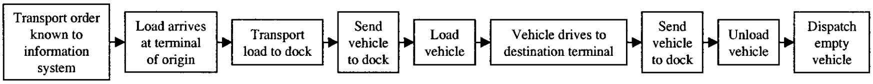

Transport order known to information system
Load arrives at terminal of origin
Transport load to dock
Send vehicle to dock
Load vehicle
Vehicle drives to destination terminal
Send vehicle to dock
Unload vehicle
Dispatch empty vehicle

**Fig. 3.** Order processing

which it should be delivered at the destination terminal, defined as the time at which the unload operation has to be finished. Key performance measures considered are:

- the order throughput time, i.e. the time from order arrival at the origin until the job is unloaded at the destination;
- the fill rate, i.e. the fraction of orders that is delivered before the due time;
- the lateness, i.e. the difference between actual job completion time and due time
- average empty vehicle kilometres per day, being an indicator for energy waste and loss of vehicle capacity.

The processing of a transportation order is depicted in Figure 3. First an order is issued, which may occur some time before the actual arrival of the load. Next the load is moved to a dock (after the *order release* decision), where it waits for a vehicle to be assigned to (*order scheduling*). After arrival at the dock, the empty vehicle picks up the load and drives to the terminal of destination. There it is assigned to an unload dock (possibly after some delay in the case of high utilisation), based on a second order scheduling decision. Then the vehicle drives to the dock and unloads as soon as the dock becomes free. After unloading, the transportation job is considered as completed, so further order handling is not considered. Then the vehicle is available to transport another order. Note that the order release and both order scheduling decisions may be taken simultaneously or sequentially, depending on the logistics control concept chosen (cf. Sect. 4). With respect to the second option, it is relevant to note that there may be a significant time lag between the three actions corresponding to the three decisions (say 5-30 minutes in the Schiphol case). This may be an argument to postpone the second and/or third decision.
As mentioned before, an empty vehicle that cannot be used immediately can be moved to a parking to wait for orders, either locally (possibly short waiting time, limited space) or centrally (possibly longer waiting times, high capacity). So an empty vehicle that has just unloaded may receive one of the following directions:

1) Move to another dock at the same terminal to pick up a load.
2) Move to a local parking to wait until a load order is assigned.
3) Move to another terminal, because it is expected or known that a (high priority) transportation order is available over there.

4) Move to a central parking, because a transportation order for this vehicle is not expected in the short run and/or because there is no room in the local parking.

In our definition, empty vehicle management includes the decisions regarding these four actions. Note that additional actions may be defined, such as vehicle repair after failure, vehicle maintenance and battery changing or recharching. Although we skip these issues for convenience, we remark that the planning framework as discussed in this paper can be extended to include both failure and energy management (cf. Ebben, 2001).

Empty vehicle management is needed because transportation flows will generally be unbalanced over time, i.e. 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. It is especially critical in situations where due times are tight and resource utilisation (vehicles, docks) is high. Therefore, we shall pay special attention to empty vehicle management.

Another part of the logistics control structure is the assignment of tasks to docks (load or unload). Obviously, it is most convenient if a dock can load and unload in random order without capacity loss. In some cases however, separation of incoming and outgoing flows may be preferred for organisational and technical reasons. If the ratio between incoming and outgoing transportation flows is stable in time, docks can be dedicated to either load or unload activities. Otherwise, we may consider using dynamically dedicated docks whose function can periodically be switched (e.g. each hour). Then, the periodic decision how many docks should load and how many should unload is referred to as task allocation to docks.

For the model as described above, we further make the following technical assumptions:

1) All vehicles are identical, so we have a homogeneous fleet.
2) The number of vehicles is fixed.
3) All orders have to be executed, even when they are late.
4) Loading and unloading times at docks vary over jobs; therefore these times are represented by independent and identically distributed random variables.
5) Travel times between nodes are deterministic and known in advance.
6) At the terminal entrance there are two infinite capacity FIFO vehicle buffers, one for loaded and one for empty vehicles.

Regarding the last assumption, we note that 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) or if the destination to which it has already been assigned is occupied. In practice, these queues 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 types. Although infinite queuing capacity implies that terminal parking capacity may be omitted from vehicle planning, queuing may have impact on the logistics performance. The reason is that loaded vehicles cannot overtake in the queue whereas this may be desirable because of different job priorities.

In contrast to travel times, the *order handling times* may fluctuate considerably between orders having the same origin and destination. These handling times consist of

- variations in loading and unloading time,
- waiting times at the destination terminal (which depend on the time-dependent work load)
- dock-dependent distances between dock and terminal entrance/exit.

For a specific job, the handling time can only be observed *after* job completion, whereas this value is needed for planning *in advance*. As the actual handling time of a specific job is unknown at the time of planning, we treat these as random variables for which the mean and standard deviation can be estimated (cf. Sect. 4 for more details).

## 3 Literature

Related literature includes AGV scheduling in internal transportation systems and vehicle scheduling in other external transportation systems (rail, road) and container scheduling.

Vehicle selection and dispatching rules are important for a good performance in AGV systems. Vehicle dispatching involves a decision rule for selecting a vehicle or dock for a specific transportation order. Egbelu and Tanchoco (1984) classify the dispatching rules into two categories, vehicle and work center initiated dispatching, and evaluate some heuristic rules. Klein and Kim (1996) propose various multi-criteria dispatching rules and show better performance than single-criteria rules. These 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. Akturk and Yilmaz (1996) propose an analytical model to incorporate the AGV system into the overall decision-making hierarchy. Ulusoy and Bilge (1993) make the scheduling of AGVs an integral part of the overall scheduling activity; machines and AGVs are scheduled simultaneously.

Directly related to the vehicle dispatching problem is the problem of positioning idle vehicles. The most common objective is to minimize the maximum or mean response time. Kim and Kim (1997) propose a procedure to determine the home location of idle vehicles in a loop layout in order to minimize the mean response time, while 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. These papers deal with so-called dwell-point strategies, i.e. the positioning of vehicles that are currently not needed for transportation. The focus is on AGV systems for internal transport in a warehouse or production facility. Usually such networks are relatively small. On the other hand, we consider *external* transport between various facilities, in which case the networks are much larger. As a 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 the empty vehicle management in this paper differs from the usual idle vehicle positioning in AGV networks.

One of the first papers in the other related area, empty vehicle management, is by White and Bomberault (1969), who model the allocation of empty freight cars in a railroad system with deterministic demand as a transhipment problem. Later on, Jordan and Turnquist (1983) formulated a dynamic model for empty vehicle allocation with stochastic demand and stochastic travel times. A recent contribution is by Holmberg et al. (1998), who develop an optimisation model for a situation with deterministic supply and demand, pre-specified train schedules, heterogeneous fleet and limited train capacity. In the field of empty truck distribution, especially Powell 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. Theoretical improvements and model extensions are presented amongst others in Cheung and Powell (1994), Powell (1996) and Powell and Carvalho (1998). A recent example of empty container allocation is by Cheung and Chen (1998), who construct a single-commodity stochastic network flow model for the distribution of empty containers under random demand and transportation capacities.

For a literature review of earlier work in the area of empty vehicle management, we refer to Dejax and Crainic (1987). They present an overview and classification of empty vehicle 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 vehicle dispatching with single transportation mode and homogeneous fleet in a dynamic setting with stochastic demand.

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

1) We face capacity restrictions at each network node, arising from a limited number of load docks and unload docks.
2) The time horizon is very short compared to applications in railcar distribution, truck distribution or container distribution. While in existing applications the planning horizon covers at least a couple of 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 optimisation models.
3) We focus on attaining customer service levels, measured by fill rates, instead of cost minimisation.
4) We allow an asymmetric road network, which is quite uncommon in other applications.
5) The presence of terminals requiring operational control rules for order release and scheduling asks for a proper planning decomposition. A logical choice

is to use a hierarchical empty vehicle planning procedure, namely a *global* empty vehicle manager to distribute vehicles between network nodes (terminals, central parkings) and a *local* empty vehicle manager covering the assignment of load orders or empty vehicle dispatch orders to the vehicles available within a node. For local empty vehicle management, we can use principles from AGV dispatching and positioning of idle vehicles.

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

## 4 Decision structure and options for planning and control

As stated in the introduction, the amount of information used and the level of planning coordination can affect the logistics performance significantly. It is likely that the best system performance can be attained if all major decisions (order release, order scheduling, task assignment to docks and empty vehicle management) are taken at a central level using all system information available. This implies that one central organisational unit should be responsible for integral system planning and that all relevant information should be available at this central level. The relevant information includes

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 (load in process, 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 status (assigned loading or unloading task; free or occupied)
   b. the expected finishing time of the current activity

Order information and dock information are usually readily available at the local terminal level, but it takes more effort to collect and maintain this information centrally. As a consequence of frequent data alteration, extensive and reliable data exchange is essential, leading to an extensive (and possibly vulnerable) information and communication system. A compromise may be to exchange aggregate information only, such as the number of load jobs for the next period at each terminal (without knowledge of destination or due times). Furthermore, central authority causes additional communication with local authorities such as terminal managers. Also, it may be more difficult for a central authority to react quickly on 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 two hierarchical layers should communicate in a simple, yet efficient way. A local control approach will therefore be the basis of our control framework. Still, we shall analyse the effect of local versus central decision making on the logistics performance.

When decisions are taken at a central level, it is possible to take the order release and order scheduling decisions at once. That is, the central authority may decide simultaneously at which dock an order should be loaded by which AGV and at which dock the AGV will unload at the terminal of destination. Because there can be a significant time lag between the load and unload activity (up to half an hour in our case study), it is not guaranteed that the second order scheduling decision is still optimal when the AGV arrives at the terminal of destination. If congestion and/or AGV failures cause (unpredictable) fluctuations in the travel times, the actual AGV queue length for each dock at the destination terminal can be different from the expectation at planning time. In such a situation, it can be preferable to postpone the second order scheduling decision. The latter will happen automatically in a local control concept.

As total system behaviour is mainly affected by empty vehicle management, we focus on this decision area. We choose some plausible decision rules for the other four decision areas. Further analysis of the impact of local decision rules is a separate research topic. In this section, we describe the rules for order release, order scheduling and task assignment. Further we give the general outline of empty vehicle management. Variants for the latter decision area are discussed in Section 5.

To define the decision rules in this section and the next section, we use the following notation:

i, j = node index (terminal, central parking)
n = order index
k = vehicle index
$t_0$ = current clock time
$tK_n$ = time at which order n is known in the system
$tA_n$ = arrival time of order n (load arrives at terminal of origin)
$tD_n$ = due time of order n, i.e. the latest time at which order n should be unloaded at the terminal of destination
$tL_n$ = latest departure time of order n (see below for the calculation)
$sL_n$ = the latest time that an empty vehicle has to be dispatched in order to pick up load n before the latest departure time $tL_n$
$\tau_{ij}$ = the travel time of a vehicle from the exit of node i to the entrance of node j
$H_{ij}$ = handling time for an order to be transported from node i to node j, consisting of loading time, travel time, waiting time and unloading time; for planning purposes, $H_{ij}$ is modelled as a random variable with mean $E[H_{ij}]$ and standard deviation $\sigma[H_{ij}]$

Note that the order handling time is *not* equal to the throughput time, as the latter also includes the time that the order has to wait for an empty AGV.

### 4.1 Order release

By order release we mean the assignment of transportation orders to load docks in the terminal of origin. The procedure is as follows. At each dock, one or more loads can be placed in a small dock buffer, waiting to be loaded. When a load arrives at the terminal of origin, it is assigned to the load dock having the shortest queue of unprocessed orders. If all load dock buffers are occupied (which will generally be true in peak periods), the load waits at the terminal until a buffer position becomes free. At that time, the order with highest priority is released to the dock where a vacancy arises. This implies that the general load buffer at the terminal should have random access (orders can be released in any arbitrary order, independent of the arrival order). The load buffers at the *docks* can both be random access buffers or channel-like devices. In the latter case, orders cannot overtake. We use the latter assumption in our case study, but modification to random access buffers is straightforward within our logistics control framework.

Priority is assigned to the orders according to the minimum latest departure time $\mathrm{tL_n}$. For each order, this time $\mathrm{tL_n}$ is set such that the due time for order n is met with high probability:

$$tL_{\mathrm{n}} = tD_{\mathrm{n}} - E[H_{\mathrm{ij}}] - k\sigma[H_{\mathrm{ij}}] \qquad (1)$$

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

- variation in load and unload times;
- possible variation in transportation times (congestion effects, variation in route length depending on the dock and/or local parking choice)
- (unpredictable) waiting time at the terminal of destination depending on the work load.

Note that the mean and standard deviation of the handling time can be estimated using a standard exponential smoothing procedure (see e.g. Silver et al., 1998).

Order release according to the priority rule based on minimum latest departure time is applicable to most look-ahead rules for empty vehicle management (Sects. 5.1 and 5.2). However, integral planning of empty vehicles may conflict with this priority rule and therefore the order release heuristic may be overruled (see Sect. 5.3 for more details).

### *4.2 Order scheduling*

Order scheduling defines the start times for load orders and unload orders at a dock. That is, an empty vehicle is assigned to a specific order and dock to pick up the load at the origin terminal and a loaded vehicle is assigned to a dock at the destination terminal. Here we have to take into account that a limited number of positions may be available to park vehicles at each dock, e.g. one or two. If all positions are occupied, a vehicle cannot be sent to a dock, because it could block other traffic in the terminal. Then a vehicle can either wait in front of the terminal entrance or drive to the local parking if there is space left. For both parts of order scheduling (load orders and unload orders), we shall use simple decision rules:

- A vehicle that becomes available for loading at a terminal is assigned to the load dock with the highest priority order that is released but not yet scheduled (i.e.

no vehicle is assigned); it is dispatched to that dock if a free AGV position is available, otherwise it is sent to a local parking to wait until it can be dispatched to the dock.

- A vehicle arriving at the terminal of destination is assigned to the unload dock with the shortest queue if a free AGV position is available, otherwise it is sent to a local parking to wait.

### *4.3 Task allocation to docks*

Assigning load and/or unload activities to docks (time-dependent docks dedication) should obviously be based on the expected number of load orders and unload orders in the system. Tasks are changed periodically only, so that (i) incoming and outgoing product flows can be clearly separated and (ii) the allocation of dock capacity can be fitted to the expected work load. The period during which the task allocation at a terminal is fixed should preferably be not too short and tuned on the frequency in which the balance between load and unload orders changes. For this decision, simple rules of thumb can be used. For example, choose the number of load docks and unload docks such, that the utilisation is balanced for orders that have to be handled in the next period (with latest departure time as selection criterion for load jobs that have to be handled). We refer to Ebben (2001) for more details.

### *4.4 Empty vehicle management*

Empty vehicle management can be considered at two hierarchical levels, namely at network level (how to allocate empty vehicles amongst terminals and central parkings) and at terminal level (how to handle an empty vehicle in a terminal). Although it is possible to consider these decisions simultaneously in a central decision setting, a hierarchical decision structure seems to be more convenient. At a local level, it is easier to react on unexpected events, such as the arrival of rush orders and the failure of docks or vehicles. At a network level, we can focus on balancing vehicle flows without taking into account all detailed events at terminals. If local problems are solved locally and network problems centrally, we can construct a relatively simple and robust control structure without excessive information traffic. To this end, we distinguish between *global empty vehicle management* that covers the network level and *local empty vehicle management* that covers the terminal level. Such an approach is to be preferred from a practical point of view, but requires coordination between the two hierarchical levels. Variants for both levels are discussed in the next section.

## 5 Options for empty vehicle management

In this section, we shall discuss four options for empty vehicle management (EVM). We shall focus on global empty vehicle management and explain the connection to the local empty vehicle manager per option. The variants differ with respect

**Table 1.** Variants for empty vehicle management (EVM)

| Variant for EVM | Central information | Coordination |
|---|---|---|
| EVM1:<br>FCFS, myopic | 1. release time and route for<br>• orders present<br>2. vehicle status, location and time ready | local |
| EVM2:<br>FCFS, look-ahead | 1. release time and route for<br>• orders present<br>• future orders known<br>2. vehicle status, location and time ready | local |
| EVM3:<br>Hierarchical coordination | 1. release time, latest departure time and route for<br>• orders present<br>• future orders known<br>2. vehicle status, location and time ready | hierarchical |
| EVM4:<br>Integral planning | 1. release time, latest departure time and route for<br>• orders present<br>• future orders known<br>2. vehicle status, location and time ready | central |

to the amount of information used and the level of coordination (see Table 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 on each order arrival. The last two variants however are capacity planning procedures that rebalance vehicle flows periodically. The next three subsections will describe the variants in more detail.

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

*5.1.1 First-come-first-served (EVM1).* 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 $tA_n$. At any point in time, the set of available vehicles is defined as:

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

Hence every vehicle (loaded and empty) has as attribute the location where the next order should be picked up. Available vehicles are vehicles without next pickup location. Note that a vehicle becomes available for assignment if it starts loading at a dock in the terminal given by its next pickup location. The time at which this vehicle can pick up the next order can be estimated from the expected order handling time.

Now the procedure for *global empty vehicle management* is as follows. Each time when a load, say index n, arrives at a terminal i (at time $tA_n$), 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.

As mentioned in the previous section, *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. The vehicle may have already been assigned to a dock to pick up an order, but in the case of a local control concept, this decision has not been taken yet (see Sect. 4). In the latter case, the order scheduling has to decide upon the load dock to be assigned. If all load docks are occupied or no unassigned load orders are present, the empty vehicle is sent to a local parking. If the local parking is full, the empty vehicle waits in front of the terminal until room 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 central parking. In periods of heavy traffic, this situation is not likely to occur because the backorder list will not be empty then.
   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 an order with the same destination is available at the terminal and a load dock is available too, the vehicle loads 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 (via order scheduling), if possible. If the vehicle cannot be assigned to a load dock, it is sent to a local parking. If the local parkings are full, the vehicle is dispatched to the nearest central parking with notification to the global empty vehicle manager. As the vehicle was assigned to pick up an order, 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. As another option, one may postpone the replacement until free space becomes available at the terminal. However, then significant time may pass until the free space is occupied, because of the relative long travel times for external transportation systems that we have in mind. Therefore we decided to implement an immediate reaction by the global empty car manager.

Note that a simple FCFS rule may not be as bad as it seems, because orders are released in each terminal according to a latest departure time priority rule (see Sect. 4) and because empty vehicle dispatch is combined with load orders if possible.

*5.1.2 Look-ahead variant (EVM2).* If orders are known some time in advance (for example $^1/_2$-1 hour), we may use this information to improve the planning procedure. When order n becomes known in the system (at $tK_n$), 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 at the terminal and cause congestion. For vehicles from the central parking, the empty vehicle manager can prevent this by delaying its departure (which 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 central parking, 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 central parking.

Note that the earliest dispatch time is introduced here, because early empty vehicle arrival is only possible in the case of prior information about orders. Under myopic planning, a load will always be present for every vehicle that arrives at a terminal, because the vehicle has been requested at $t_{A,n}$. Further, note that local empty vehicle management is equivalent to the procedure from the previous subsection.

### *5.2 Hierarchical coordination (EVM3)*

In the first two options, the global empty vehicle manager does not take into account priorities when dispatching vehicles to terminals. 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, a global empty vehicle manager only controls the *number* of vehicles. This fits into the local logistics control concept, even though complete scheduling is possible (cf. Sect. 5.3). A global empty vehicle manager may request for empty vehicles to be sent to another location (central parking, terminal), but the local authority may decide *which* vehicles *when* to dispatch. The global empty vehicle manager sets priorities to empty vehicle requests using the latest dispatch time $sL_n$, i.e. the latest time that an empty vehicle has to be dispatched in order to pick up load n at terminal j before the latest departure time $tL_n$.

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

- a list of all known load orders 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 load orders is sorted in increasing order of latest departure times $tL_n$. This list is processed sequentially. To each order, we assign the vehicle that can be available at the earliest point in time. The planning procedure continues until the order list is fully assigned or all vehicles are assigned.

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 some 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 orders to dispatch empty vehicles, sorted on latest dispatch time $\underline{sL}_m$, m=1,2,...
B. a list of known load orders, sorted on latest departure time $\underline{tL}_n$, n=1, 2 ...

The underlined variables refer to the respective sorted lists. If $\underline{sL}_1$¡$\underline{tL}_1$ (or if the list of known load orders is empty), the local empty vehicle manager dispatches the vehicle according to the first destination on list A. If $\underline{tL}_1 \leq \underline{sL}_1$ (or if the list of empty vehicles to be dispatched is empty)[1] , the local empty vehicle manager assigns the empty vehicle to the first load order on list B. If both lists A and B are empty, the local empty vehicle manager sends the vehicle to the local parking if it is needed in the near future and if the local parking is not full. Otherwise the vehicle is sent to the nearest central parking (for which a notification is send to the global empty vehicle manager).

The local empty vehicle manager tries to combine orders if possible. That is, if an empty vehicle dispatch order to terminal j has highest priority ($sL_1$¡$tL_1$), a load order for the same terminal j with smallest latest departure time $tL_n$ is searched. If such an order is available and a load dock is available too, this order is scheduled, otherwise the vehicle is dispatched empty to terminal j. Vice versa, if a load order for some terminal j has highest priority ($tL_1 \leq sL_1$), the list of empty vehicle dispatch orders is searched for an order with smallest latest dispatch time having the same destination j. If such an order is found, it is removed from the list and the global empty vehicle manager is informed that the request is fulfilled. In response, the

[1] Note that the indices of $\underline{sL}_m$ and $\underline{tL}_n$ refer to *different* order lists, one for load orders at the terminal under consideration and one for load orders for which this terminal should supply an empty vehicle. Hence $\underline{tL}_1$ and $\underline{sL}_1$ refer to the first orders on each sorted list and thus *not* to the same order.

global empty vehicle manager may provide another empty vehicle dispatch order to the terminal, but this order will have (considerably) less priority.

This procedure has advantages above the simple rules from Section 5.1 in the sense that priorities are used by both the global and the local empty vehicle manager and the two decision layers with responsibilities are properly separated.

### *5.3 Integral 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 through the transportation system later on. For example, the global empty vehicle manager may dispatch empty vehicles to terminal 1, knowing that they will be merely used to transport cargo to terminal 2. Hence these vehicles can be used to transport loads from terminal 2 to another destination later on. If the global empty vehicle manager ignores this information, as in EVM3, the consequence can be that empty vehicles will be dispatched to terminal 2 that appear not to be necessary later on. This may be overcome by making an integral planning in which all orders and empty vehicle trips are considered. This integral planning is the main difference between EVM4 and EVM3. A requirement for integral planning is that all information about orders, vehicles and docks is centrally available.

*5.3.1 Relation between global and local empty vehicle manager.* Based on information about the system status and all orders, the global empty vehicle manager optimises a transportation schedule for all orders 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 integral 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 load 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 load jobs are handled in the prescribed sequence and empty vehicles are dispatched at the prescribed time or as soon as possible afterwards. Note that the dispatch times of empty vehicles are required to avoid that an available empty vehicle, waiting for a load job, is used for another empty vehicle job. Recall that the global empty vehicle manager as described in the previous section (EVM3) just provides a list of empty vehicle dispatch jobs with *latest dispatch time* to the terminals.

The assignment of docks, the scheduling of unload jobs and the exact timing of load jobs, given the order prescribed by the global empty vehicle manager, is left to the local order scheduler. Also, the local empty vehicle manager is 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 provides the local empty vehicle managers with the authority to deal with local circumstances, such as vehicle positioning related to terminal layout and reacting on 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).

*5.3.2 The integral approach for the global empty vehicle manager.* The idea for the integral 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 orders that become known later on are neglected. New orders 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 large, since otherwise too many new orders are missed, but it should also not be too small since otherwise the efficiency of the global plan is not obtained. Of course, it may vary from case to case what too small and too large means. We will deal with this issue in Section 6 when we discuss the numerical experiments.

For the integral planning by the global empty vehicle manager EVM4 we apply a so called serial scheduling method which uses priorities of jobs to sequentially build a schedule (cf. Kolisch, 1996). The objective is to optimise due time performance, i.e. to minimise the maximum lateness of the jobs. Next, we describe this method to optimise a schedule, given a fixed system status. As mentioned, 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 in our application for scheduling with a fixed time interval.

*5.3.3 The serial scheduling method.* Input to the scheduling are the current status of the system and a list of orders. 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 orders contains all orders n known to the system ($tK_n \leq t_0$), including orders in process. Recall that an order in process may either be on a load dock waiting to be handled, it may be in transportation on a vehicle, or it may be at a dock, being unloaded.

We apply 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) *Initialisation*. From the status of the system, compute so called *initial empty vehicle profiles* for every central and local parking. An initial empty vehicle

profile describes the number of available empty vehicles for any $t \geq t_0$, given that no further jobs are started. Furthermore, we compose a sorted list of jobs to be scheduled, containing all jobs n that still have to be started, sorted by increasing latest departure time $tL_n$.

2) Let job n be the first job on the sorted list. Evaluate all possible assignments of empty vehicles to job n, and assign that empty vehicle that serves job n as early as possible. The resulting assignment leads to an *earliest serving moment* $tE_n$ and a slack $tS_n = tL_n - tE_n$ for job n.
3) Try if any of the other jobs on the list can be *combined* with job n. By our definition, a job m can be combined with job n, if and only if the destination of job m equals the origin of job n. If such a combination is possible, it prevents unnecessary empty vehicle movements, but it also may cause a delay compared to the earliest serving moment $tE_n$. Combined transportation is controlled by two parameters, namely the *maximum delay* in serving time and the *minimum slack* that must remain for job n (i.e. the time gap between the scheduled starting time and the latest departure time of job n). To combine transportation, compute per terminal the job with the highest priority that can be combined with job n, such that the minimum slack and maximum delay are not violated. Amongst all these jobs, accept the job m that leads to the smallest delay for job n. Note that job m, and afterwards job n, may use an other empty vehicle than the one assigned to job n in step 2. In that case the empty vehicle assignment of step 2 is cancelled.
4) Remove job n and, in case of a combined transportation, remove job m 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 orders are leading in the scheduling process for EVM4. That is, unlike the planning options as described in Sections 5.1 and 5.2, in the integral planning approach a vehicle may wait empty for an order to be processed instead of starting with the most urgent order 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 minimal slack, that may be worthwhile to be tuned 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 integral 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 capacity of the parkings. Thus it assumes that (1) every job can be handled immediately at the dock 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 our specific application, 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, as excess empty vehicles will be sent to another parking by the empty vehicle manager anyway. Besides, each terminal has queuing space in front of the entrance (see Sect. 2). As we shall see in Section 6, the simulation results suggest that these limiting assumptions are not too harmful in the case under consideration. This could change however if terminals hardly have 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 can quickly be generated, based on different performance criteria, priority rules and parameter settings (maximum delay, minimum slack). We can also use a given maximum computation time as restriction for the number of schedules to generate.

## 6 Numerical investigation

We applied our logistics planning and control structure to the Dutch pilot project around Amsterdam Airport Schiphol as described in the introduction. An object-oriented simulation model based on eM-Plant has been constructed and all planning variants have been embedded. This discrete event simulation model reflects many details of the logistics system, such as individual load movements between buffers and docks and individual AGV movements between locations in the system along some well-defined track system (docks, parkings, junctions, etc.). To this end, we could use predefined objects from the eM-Plant simulation library. Detailed AGV-behaviour (acceleration, distance control) has not been included in our model, as we focused on network performance. Where required, we used a rough model of the AGV behaviour. For example, we used a statistical relation between the number of vehicles on a terminal and the effective AGV speed. This relation has been estimated using a separate, detailed simulation model that has been constructed by Verbraeck et al. (1998).

Although the eM-Plant object library provides a sound basis for our simulation model, the logistics control system almost had to be constructed from scratch. Most heuristics and rules were implemented in eM-Plants' own programming language. Only for empty vehicle manager EVM4, we have coded the serial scheduling method as an external routine in C++ that is periodically called by the eM-Plant simulation model.

In the remainder of this section, we first present the experimental setting and procedure (6.1). Next, we discuss some initial experiments on planning parameters (6.2). The numerical results of our simulation experiments are the subject of Subsection 6.3.

### *6.1 Experimental setting and procedure*

For all our experiments, we use the layout as shown in Figure 1. The transportation system is further specified by the following parameters:

1) The time required for loading and unloading has mean 2 minutes and standard deviation 15 seconds. In the simulation experiments, these values were sampled from a gamma distribution with these characteristics.
2) At each dock only one AGV can load or unload, but there is a waiting position for a second AGV.
3) The number of locations to park empty AGVs temporarily between unloading and loading equals the number of docks.
4) The network contains one central parking that is located between the AAS-terminals and the junction to Hoofddorp and Aalsmeer.
5) The AGV speed is 6 m/s in the underground tubes, 2 m/s in the terminal and 3m/s on slopes between the underground tubes and the terminals at the surface.
6) The order pattern is based on projections for the year 2020 as generated by the Dutch Economical Institute and expressed as the number of standard load units. It is expected that the system will be in full operation in 2020. We used a peak day with heavily imbalanced transportation flows (Tuesday), a peak day with more balanced transportation flows (Monday), and an average day (Friday) to establish the capacities required (number of docks and AGVs) and to test our decision rules. In the simulation model, orders were sampled from a Poisson distribution at the VBA and AAS terminals. At the rail terminal (RTH), 6 batches per hour arrive where the number of orders in each batch were sampled from a Poisson distribution. Batch arrivals at the rail terminal are justified because trains containing a large number of loads arrive according to some fixed schedule.
7) We have various priority classes of orders, specified by their throughput time requirements. The throughput time requirements are as follows:
   - to and from Amsterdam Airport Schiphol (AAS): within 45 (10% of the orders), 60 (20% ) 90 (20% ) and 120 minutes (50% );
   - between Hoofddorp (RTH) and Aalsmeer (VBA): within 60 (10% of the orders), 75 (20% ), 90 (20% ) and 120 minutes (50% ).
8) Energy provisioning of AGVs is arranged via electric wires in the tube system. While driving on the terminals, the AGVs use batteries that are recharged while driving in the tubes. As a consequence, battery recharging or changing is not required (cf. Ebben, 2001 for the analysis of alternative options of energy provisioning using batteries).
9) Equipment failures (AGVs, docks) are not taken into account in the numerical experiments, although the control frameworks and rules still apply (cf. Ebben, 2001 for the impact of equipment failures and failure management).

The logistics performance is measured using a large number of statistics, such as dock and AGV utilisation, throughput times, order waiting times, lateness and fill rates. The statistics are collected time dependent (e.g. for each day or even hour) and per route (terminal of origin and destination). In this section, we focus on the key performance characteristic, namely the fill rate. Recall that the fill rate is defined as the fraction of orders that is delivered before the due time (cf. Section 2). All numerical results presented in this paper 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 hardly any unserviced orders left 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 for the accuracy of the results corresponding to a run length of 30 days. We found that the confidence intervals of the fill rates are quite small in the range that is of interest to us, a fill rate of about 99% . Then the 0.95 confidence interval length for the fill rate is in the range 0.1–0.25% . When the fill rates 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 four options EVM1-EVM4.
2) The impact of the period for which orders are known in advance (value of information).

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

a) Resource capacities
b) The choice of the planning frequency of task allocation to docks, see Section 4
c) Parameter settings for EVM4
d) The choice of the planning frequency when empty vehicle management is coordinated at network level (EVM3 and EVM4)

The results are presented in the following subsection.

### *6.2 Initial experiments*

*6.2.1 Resource capacities.* Before examining the performance of the various logistics 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 orders. The system should contain at least 185 AGVs to handle all loads in peak hours on Tuesday. In this situation, vehicle availability is the bottleneck rather than dock capacity. For Monday, 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 Friday, transportation flows are quite imbalanced, yielding a number of 165 AGVs to handle the peak demand. We used these figures for our simulation experiments.

*6.2.2 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

**Table 2.** Effect of fixed dock task period (Tuesday, 185 AGVs, EVM3 planning each 30 minutes with orders being known 30 minutes in advance)

| | Fill rate | | | | | | |
|---|---|---|---|---|---|---|---|
| Fixed dock task period | AAS-RTH | AAS-VBA | RTH-AAS | RTH-VBA | VBA-AAS | VBA-RTH | Overall |
| 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 |

period of 15-30 minutes has only a slight impact on system performance, see Table 2. Once the period increases to one hour or more, the performance may decrease significantly. As a longer period has technical and organisational advantages (less task changes), we chose 30 minutes for the rest of our numerical experiments.

*6.2.3 Parameter settings for EVM4.* As mentioned in Section 5.3, the empty vehicle manager EVM4 periodically performs an integral planning run. We may call the serial scheduling method once at each run, but we can also call it several times for different parameter settings. Then, the best schedule obtained is 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 than 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, we evaluated only four schedules per integral planning run to keep the run time acceptable. 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 to wait for an empty vehicle or for the release time of a job for 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 larger maximum lateness. We found that a minimum slack between 5 and 20 minutes yields good results. A minimum slack smaller than 5 minutes reduces the on-line fill rate significantly because too many jobs are scheduled close to their latest departure time. On the other hand, a minimum slack larger than 20 generally did not result in significantly different schedules. Among the four schedules generated, we selected the schedule with the smallest maximum lateness. In case of ties, the schedule with the smallest total travel time for empty vehicles was selected.

*6.2.4 Planning frequency for coordinated empty vehicle management.* For coordinated empty vehicle management (EVM3 and EVM4), the planning frequency has to be determined. As the system situation changes quickly, infrequent rescheduling may decrease the system performance. Table 3 shows some corresponding simula-

**Table 3.** Effect of empty vehicle planning frequency (Tuesday, 185 AGVs, orders being known 30 minutes in advance)

| | Fill rate | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| EVM<br>period | AAS-<br>RTH | AAS-<br>VBA | RTH-<br>AAS | RTH-<br>VBA | VBA-<br>AAS | VBA-<br>RTH | EVM3<br>Overall | EVM4<br>Overall |
| 10 min. | 99.3 | 99.3 | 99.5 | 100.0 | 97.8 | 98.8 | **99.1** | **98.9** |
| 20 min. | 99.0 | 99.4 | 99.1 | 100.0 | 96.6 | 97.7 | **98.4** | **98.4** |
| 30 min. | 98.9 | 99.3 | 99.2 | 100.0 | 97.0 | 97.4 | **98.2** | **96.8** |
| 40 min. | 98.2 | 98.8 | 99.0 | 100.0 | 95.7 | 95.6 | **97.0** | **X** |
| 50 min. | 96.8 | 97.6 | 99.5 | 100.0 | 90.9 | 91.9 | **94.7** | **X** |
| 60 min. | 93.3 | 96.0 | 99.0 | 99.9 | 82.1 | 84.6 | **89.8** | **X** |

**Table 4.** Effect of empty vehicle management (orders being known 30 minutes in advance)

| | | Overall fill rate | | | |
|---|---|---|---|---|---|
| Day | AGVs | EVM1 | EVM2 | EVM3 | EVM4 |
| Monday | 120 | 94.7 | 97.1 | 99.8 | 100 |
| Tuesday | 185 | 80.3 | 89.9 | 99.1 | 98.9 |
| Friday | 165 | 85.5 | 93.2 | 99.7 | 99.6 |

tion results when orders are known 30 minutes before cargo arrival at the terminal of departure. Note that a prerequisite for proper functioning of EVM4 is that the planning period should be less than or equal to the time for which orders are known in advance. For example, if the planning period is 40 minutes and orders are known 30 minutes in advance, the orders 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 3 only shows the performance of EVM4 if the planning frequency is at least once per half hour.

Surprisingly, Table 3 shows that the fill rate is not extremely sensitive to the empty vehicle manager planning frequency. As long as the frequency is at least once every half hour, the fill rate remains high. For the time being we choose to reschedule every 10 minutes, yielding high customer service.

### 6.3 Numerical results

*6.3.1 Comparison of empty vehicle management strategies: the value of coordination.* An important research question is the effectiveness of the various options for empty vehicle management. We tested these options for the three weekdays as mentioned in Section 6.1. The overall fill rates are shown in Table 4 for all cases.

We see that planning coordination (included in EVM3 and EVM4) has a significant impact on the customer service. It is remarkable that the effects are considerably smaller for Monday than for the other two days. A possible explanation for this phenomenon is the fact that transportation flows are more balanced on Monday.

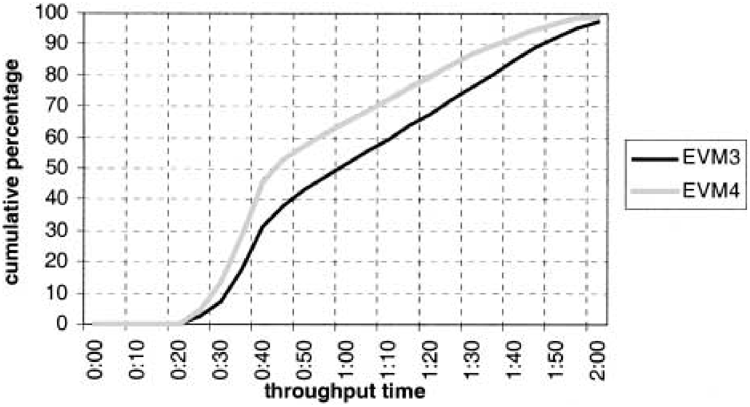


**Fig. 4.** Throughput time distribution for EVM3 and EVM4 in the case of fictitious transportation flows with quickly moving peak levels between routes

To give an indication, on Monday the flows from VBA to RTH are similar to or at most double 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 Monday. Especially when transportation flows are heavily imbalanced (Friday and particularly Tuesday), proper empty vehicle management may increase the system performance considerably.

For the cases mentioned in Table 4, EVM3 and EVM4 have more or less the same overall fill rate. To judge whether this is a general result, we constructed an additional artificial case. This case is 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 anticipation of heavily fluctuating transportation demand is essential to attain high customer service levels. For this case, we found that EVM4 performs significantly better than EVM3 (fill rate 98.5% versus 95% ). Also, the throughput time distribution under EVM4 appears to be significantly better, see Figure 4. Hence we conclude that EVM4 is preferable in the sense that it performs at least equal to EVM3, and under difficult conditions considerably better than EVM3. Moreover, as stated in Section 6.2.3., we could run EVM4 with more parameter settings or priority rules to improve its performance.

For the two best empty vehicle managers, EVM and EVM4, we compared the average empty travel distance per day. In this respect, we found that EVM4 requires 3-10% less empty travel than EVM3, depending on the specific case. Hence EVM4 is also more efficient from an environmental point of view. Besides, empty travel reduction is relevant if AGVs use batteries for energy provisioning, because the recharging frequency is reduced. The choice of energy provisioning (electric or combustion engine) is beyond the scope of this paper.

*6.3.2 Impact of the information horizon: the value of information.* As mentioned before, pre-information about orders may facilitate planning and improve the sys-

**Table 5.** Effect of the period for which orders are known in advance

| Day | AGVs | orders | Overall fill rate | | |
|---|---|---|---|---|---|
| | | known | EVM2 | EVM3 | EVM4 |
| Monday | 120 | 0 min. | 94.7 | 99.4 | 99.0 |
| | | 30 min. | 97.1 | 99.8 | 100 |
| | | 60 min. | 90.3 | 99.9 | 100 |
| Tuesday | 185 | 0 min. | 80.3 | 84.7 | 92.4 |
| | | 30 min. | 89.9 | 99.1 | 98.9 |
| | | 60 min. | 93.0 | 98.5 | 99.2 |
| Friday | 165 | 0 min. | 85.5 | 90.0 | 94.8 |
| | | 30 min. | 93.2 | 99.7 | 99.6 |
| | | 60 min. | 95.6 | 99.9 | 99.6 |

tem performance. However, this requires additional communication and hence more sophisticated organisation and information systems. Therefore we analyse the effect of the period for which orders are known in advance. As pre-information is only used for the empty vehicle manager options 2, 3 and 4, we omit option 1 (FCFS myopic). Obviously, EVM2 (FCFS-look ahead) is identical to EVM1 if pre-information is not available (horizon = 0). The main results are shown in Table 5.

We notice a similar phenomenon as in the previous subsection, namely that the impact of pre-information is highest if the transportation flows are heavily imbalanced (Tuesday and Friday). Pre-information appears to have even a 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 is even considerably less when orders are known 60 minutes in advance compared to only 30 minutes pre-information. This phenomenon can be explained as follows. Vehicles are dispatched to the terminals FCFS once the orders are known. If orders are known 60 minutes in advance, they arrive too early at the terminal and have to wait for a long time before they can be loaded, causing a significant capacity loss on a bottleneck resource (AGVs).

In the pilot project one hour of pre-information will be sufficient for practical purposes because of the maximum travel times between the various locations in the network.

## 7 Conclusions and directions for further research

In this paper, we developed several options for empty vehicle management in automated transportation networks and embedded these planning methods in a logistics planning and control framework. The various methods have been implemented and tested in a simulation environment for the case of an underground transportation system using AGVs around Amsterdam Airport Schiphol.

We found that both information about future orders and planning coordination between terminals gives considerable advantage in terms of customer service (fill rates). 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 in many cases. 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. Then it is worthwhile to plan a sequence of orders in an integral way for a longer time horizon. Also, integral planning has the advantage of empty travel reduction. 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.

Regarding further research, another interesting topic is the consideration of additional resources in the integral planning aproach. For example, the pilot project on underground transportation around Schiphol Airport has evolved to a network containing bi-directional tubes (cf. van der Heijden et al., 2000). That is, the system contains single tubes for traffic in two directions in order to save investment. As a consequence, travel times between locations may fluctuate heavily, because vehicles may have to wait before entering a bi-directional tube. The simple coordinated method EVM3 can easily incorporate these fluctuating travel times, but the performance of this method may decrease. In general, the combination with finite terminal capacities is especially hard, as scheduling on finite capacities requires predictable travel times. So modelling the bi-directional tubes simply as random variables with considerable variance does not seem to be appropriate. A challenging alternative is the extension of EVM4 to incorporate both bi-directional tube scheduling and finite terminal capacities. We expect that the flexible approach should allow this global scheduling method. This issue is the main focus for our further research.

## References

Akturk MS, Yilmaz H (1996) Scheduling of automatically guided vehicles in a decision making hierarchy. International Journal of Production Research 34(2): 577–591

Cheung RK, Chen C-Y (1998) A two-stage stochastic network model and solution methods for the dynamic empty container allocation problem. Transportation Science 32(2): 142–162

Cheung RK, Powell WB (1994) An algorithm for multistage dynamic networks with random arc capacities, with an application to dynamic fleet management. Operations Research 44(6): 951–963

Dejax PJ, Crainic TG (1987) A review of empty flows and fleet management models in freight transportation Transportation Science 21(4): 227–247

Ebben MJR (2001) Logistics control of automated transportation networks. PhD thesis, University of Twente, Faculty of Technology and Management (forthcoming)

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

Gademann AJRM, van de Velde SL (2000) Positioning automated guided vehicles in loop layout. European Journal of Operational Research 127: 565–573

van der Heijden MC, van Harten A, Ebben MJR, Saanen Y, Verbraeck A, 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 TRAIL Research School (submitted for publication)

Holmberg K, Joborn M, Lundgren JT (1998) Improved empty freight car distribution. Transportation Science 32(2): 163–173

Hu CH, Egbelu PJ (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

Jordan WC, Turnquist MA (1983) A stochastic, dynamic network model for railroad car distribution. Transportation Science 17(2): 117–145

Kim KH, Kim JY (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 CM, Kim J (1996) AGV dispatching. International Journal of Production Research 34: 95–110

Kolisch R (1996) Serial and parallel resource-constrained project scheduling methods revisited: Theory and computation. European Journal of Operational Research 90(2): 320–333

Powell WB (1996) A stochastic formulation of the dynamic assignment problem, with an application to truckload motor carriers. Transportation Science 30(3): 195–219

Powell WB, Carvalho TA (1998) Dynamic control of logistics queuing networks for large–scale fleet management. Transportation Science 32(2): 90–109

Powell WB, Sheffi Y, Nickerson KS, Butterbaugh K, Atherton S (1988) Maximizing profits for North American Van Lines' truckload division: A new framework for pricing and operations. Interfaces 18: 21–41

Silver EA, Pyke DF, Peterson R (1998) Inventory management and production planning and scheduling. Wiley, New York

Ulusoy G, Bilge U (1993) Simultaneous scheduling of machines and automated guided vehicles, International Journal of Production Research 31: 2857–2873

Verbraeck A, Saanen YA, Valentin E (1998) Logistic modeling and simulation of automated guided vehicles. In: Bargiela A, Kerckhoffs E (eds) Simulation technology: science and Art, pp. 514–519. SCS, San Diego CA

White W, Bomberault A (1969) A network algorithm for empty freight car allocation. IBM Systems Technical Journal 8: 147–171

van der Zee DJ (1997) Simulation as a tool for logistics management. PhD Thesis, University of Twente