

Dirk Briskorn · Andreas Drexl · Sönke Hartmann

Inventory-based dispatching of automated guided vehicles on container terminals

Published online: 12 April 2006
© Springer-Verlag 2006

Abstract This paper deals with automated guided vehicles (AGVs) which transport containers between the quay and the stack on automated container terminals. The focus is on the assignment of transportation jobs to AGVs within a terminal control system operating in real time. First, we describe a rather common problem formulation based on due times for the jobs and solve this problem both with a greedy priority rule based heuristic and with an exact algorithm. Subsequently, we present an alternative formulation of the assignment problem, which does not include due times. This formulation is based on a rough analogy to inventory management and is solved using an exact algorithm. The idea behind this alternative formulation is to avoid estimates of driving times, completion times, due times, and tardiness because such estimates are often highly unreliable in practice and do not allow for accurate planning. By means of simulation, we then analyze the different approaches. We show that the inventory-based model leads to better productivity on the terminal than the due-time-based formulation.

Keywords Container logistics · Container terminal · Automated guided vehicles · Dispatching · Assignment problem · Simulation

D. Briskorn (✉) · A. Drexl
Institut für Betriebswirtschaftslehre, Lehrstuhl für Produktion und Logistik,
Christian-Albrechts-Universität zu Kiel, 24098 Kiel, Germany
E-mail: briskorn@bwl.uni-kiel.de, andreas.drexl@bwl.uni-kiel.de

S. Hartmann
HPC Hamburg Port Consulting GmbH, HHLA Container Terminal Burchardkai,
21129 Hamburg, Germany
E-mail: s.hartmann@hpc-hamburg.de

S. Hartmann
HHLA Container Terminal Altenwerder GmbH, Bei St. Annen 1, 20457 Hamburg, Germany

1 Introduction

In various regions of the world, double-digit growth rates in container handling have been common during the last years and, hence, a substantial number of container vessels is built each year. In addition, new vessels are often larger than older ones—currently, modern vessels can carry more than 9,000 standard containers (20-foot equivalent unit, TEU), and even larger ships are already planned. Thus, the capacity of the worldwide container vessel fleet increases year by year. This development puts pressure on container terminal operators to enlarge terminal capacities to avoid congestion in ports. As a consequence, more container terminals are built, and existing ones are expanded. For reasons of efficiency and stacking density, new and extended terminal facilities increasingly make use of automated equipment. This leads to the necessity of complex terminal control systems which allow for an optimized utilization of the automated resources.

Due to its practical relevance, container terminal logistics has been a prominent field of research. A comprehensive-literature survey has recently been given by Steenken et al. (2004). Further overviews have been provided by Meersmans and Dekker (2001), Vis and de Koster (2003), as well as Vis (2006). Important optimization problems include berth planning (see Guan and Cheung 2004; Imai et al. 1997, 2001; Lim 1998; Park and Kim 2003), quay crane planning (see Daganzo 1989; Peterkofsky and Daganzo 1990), and straddle carrier scheduling (see Böse et al. 2000; Kim and Kim 1999b; Steenken et al. 1993). Moreover, approaches for locating containers in the yard have been developed (see de Castilho and Daganzo 1993; Kim and Kim 1999a; Kim et al. 2000; Taleb-Ibrahimi et al. 1993; Zhang et al. 2001).

Several papers have studied specific optimization problems arising in container terminals with automated equipment. Automated guided vehicles (AGVs) have been studied by Bae and Kim (2000). Bish et al. (2005) propose a greedy dispatching method for AGVs. Grunow et al. (2004) consider double load AGVs, that is, AGVs that can carry two 20-ft containers at a time. A general model for scheduling equipment such as AGVs or automated stacking cranes (or non-automated resources such as straddle carriers and reefer mechanics) has been proposed by Hartmann (2004). Meersmans and Wagelmans (2001) discuss an integrated scheduling approach for automated stacking cranes and AGVs. A simulation study to compare AGVs and automated shuttle carriers has been given by Vis and Harika (2004). Kim et al. (2001) employ simulation to provide a test bed for the control system of an automated container terminal. There are numerous papers in which resource allocation/dispatching rules have been applied in different manufacturing settings. For the sake of brevity, we do not go into details in this paper and refer the reader to, e.g., Hwang and Kim (1998), de Koster et al. (2004) and Vis (2006).

In this paper, we focus on highly automated terminals which employ AGVs. This study has been carried out in cooperation with the HHLA Container Terminal Altenwerder in Hamburg, Germany [for details on this terminal, see Baker (1999)]. We consider a container terminal configuration similar to the Altenwerder terminal that employs quay cranes, AGVs, and automated stacking cranes. Quay cranes are used to discharge containers from and load containers onto vessels. AGVs are means for horizontal transport of containers between the stacking area and the quay, and they are unable to load or unload themselves. The yard is organized in a

number of stacks, and each stack (or yard block) is served by one or more stacking cranes. The terminal layout considered throughout this paper is displayed in Fig. 1. In this paper, we only deal with the waterside, that is, containers arriving by a vessel which have to be brought to the stacking area and containers being picked up by a vessel which have to be brought from the stack to the quay (the landside with its outside truck and rail operations is not considered, hence, it is not shown in Fig. 1).

The goal of the paper is to present a method for assigning AGVs to transportation jobs that is applicable to real-world container terminals. Therefore, the main requirements for the method are high waterside productivity, very short

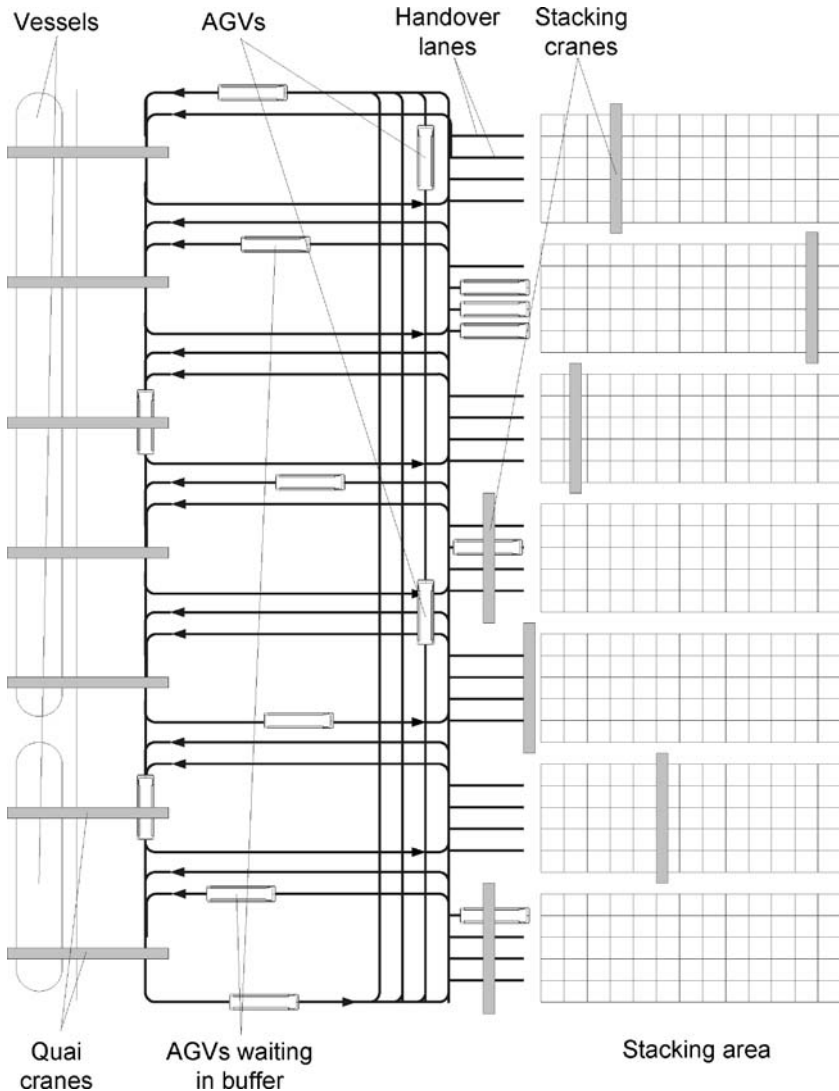


Fig. 1 Layout of the container terminal

computation times, and robustness. High productivity means that the number of containers transported per hour should be as high as possible. Short computation times are necessary to allow for real-time application within a terminal control system. Robustness means that the method should perform well in a rather unpredictable environment (which is typical in practice due to quay crane delays, inaccurate estimates for AGV travel times, manual interference, etc.).

The outline of the paper is as follows. We first describe a rather conventional approach to the AGV assignment problem, which is based on due times and an earliness–tardiness objective. This formulation will be solved both by a greedy heuristic [such simple methods are often used in practice and are also discussed in the scientific literature; see Bish et al. (2005)] and an optimal algorithm. Subsequently, we propose a new approach to the AGV assignment, which introduces the idea of inventory related to quay cranes. The motivation for this is to provide a problem formulation that avoids to employ time estimates as the latter are typically inaccurate on real world terminals. Our goal is to define a method that is more robust than a time-based one and, thus, leads to higher productivity. The approaches are then compared in a simulation study. We first point out how much the terminal productivity can be improved by using an optimal algorithm instead of a simple heuristic in the conventional time-based formulation. Then, we indicate the improvement that can be obtained from using the inventory-based formulation instead of the time-based one.

2 General problem description

We consider the problem of assigning jobs to AGVs. Each job corresponds to the transportation of a container from a pick-up location to a delivery location. An AGV can be assigned one job (and, thus, a single container) at a time. After completing a job, an AGV can start another job. A job consists of an empty drive from its last position to the pick-up location, a hand-over time at the pick-up location, a drive to the delivery location, and a hand-over time at the delivery location. Two types of processes are distinguished, namely, discharging and loading a vessel. For a job related to a discharging operation, the pick-up location is a quay crane and the delivery location is a stack. Analogously, for a job related to a loading operation, the pick-up location is a stack and the delivery location is a quay crane. For each job, the locations are fixed (specific quay crane or specific stack). Estimates of driving times between any two locations on the layout, as well as estimates of the hand-over times are assumed to be given (if needed by the actual solution approach).

Depending on the vessel's stowage plan and operational strategies, some container i may have to arrive at the quay crane before some container j when loading a vessel. That is, there may be precedence relations between some (but usually not all) of the jobs related to the same loading quay crane. There are no precedence relations between discharging jobs.

The problem essentially consists of a number of AGVs and a number of jobs. We consider n AGVs, namely, those which are currently available and those which will soon complete their current job (note that this means we have a look-ahead in the assignment process). For these AGVs, an estimated waiting time for availability is given. Without discussing the details in this paper, it should be mentioned that

determining the n AGVs to be considered is not based on a horizon but on conditions related to certain events in the progress of the current job (only the occurrence of these events allows for a relatively good estimation of the availability time). Due to the problem-inherent rolling planning horizon, only the n most urgent jobs are considered when computing for an assignment of jobs to AGVs.

The main goal when assigning jobs to AGVs is to maximize the waterside productivity, that is, the number of containers handled per hour by the quay cranes. This goal cannot be used directly as an objective function for the AGV assignment problem. In fact, different objective functions can be defined to achieve the productivity goal. Two such approaches will be discussed in the following sections. In general, one may achieve high productivity by employing goals such as minimization of the quay crane waiting times for AGV (when AGVs arrive too late), minimization of the AGV waiting times at quay cranes (when AGVs arrive too early), minimization of the empty travel times, and an even distribution of AGVs among the quay cranes. (Note that the loaded travel times cannot be influenced by assignment decisions because the pick-up and delivery locations of each job are fixed.)

The AGV assignment problem is embedded into an overall terminal control system. Whenever a certain event occurs, a new AGV assignment is calculated. The main event is the completion of a job. Thus, frequent re-planning is done. If the assignment procedure assigns a job to an AGV that is currently available, this assignment is fixed and the AGV starts this job. Otherwise, if the assignment procedure assigns a job to an AGV that is not yet available, the assignment is not fixed. In the latter case, the job and the AGV will be considered again when the assignment procedure is started after the next event. This way, the decision to actually execute a job is made as late as possible. This allows for decisions based on actual data, which is important as data are frequently changing in practice due to delays, etc. In fact, frequent changes in the data and the inaccuracy of time estimates (which are typical in practice) lead to a short planning horizon and to an assignment problem in which an AGV obtains only one job (instead of a scheduling problem with a sequence of jobs).

In Sections 3 and 4, we present two different formulations of the problem setting described above. Both approaches have essentially the same structure as they are both assignment problems with n jobs and n AGVs (i.e., each AGV must be assigned exactly one job and vice versa) and with an objective to minimize the total assignment costs. They differ only in the way of selecting the n jobs to be assigned and in the definition of the costs c_{ja} , which evaluate the assignment of an AGV a to a job j .

3 Due-time-based approach

3.1 Problem formulation

In this section, we provide a formulation of the AGV assignment problem that makes use of due times for the jobs. This approach is similar to the formulation of Hartmann (2004) and will be summarized briefly.

Each quay crane is associated with a sequence of either loading or discharging jobs. Considering the time the quay crane needs for loading or discharging one container, we can define a due time d_j for each job j . The due time reflects the time at which an AGV should arrive at a quay crane either empty (discharging operation) or with a container (loading operation). Note that a job always has a later due time than all of its predecessors.

As AGVs are unable to load and unload themselves, they should arrive at the quay cranes just in time. Early arrival implies that the quay crane is not yet ready and that the AGV has to wait, which is a waste of AGV capacity. Late arrival means that the quay crane has to wait for the AGV, which decreases its productivity. This leads to a traditional earliness–tardiness objective function. Moreover, one may wish to obtain short empty travel times (to save fuel costs and to save AGV capacity for future jobs). Thus, our objective function minimizes the weighted sum of earliness, tardiness and empty travel time.

For a more formal definition, let J be the set of the jobs to be assigned, and let α_E , α_T , and α_e be the weights for earliness, tardiness, and empty travel time, respectively. Moreover, let f_j^q be the estimated arrival time of job j at the quay crane resulting from the assignment, and let e_{ja} denote the empty travel time of job j when assigned to AGV a . Now the costs c_{ja} of assigning AGV a to job j are defined as

$$c_{ja} = \begin{cases} \alpha_E \cdot (d_j - f_j^q) + \alpha_e \cdot e_{ja} & \text{if } f_j^q < d_j \\ \alpha_T \cdot (f_j^q - d_j) + \alpha_e \cdot e_{ja} & \text{otherwise.} \end{cases} \quad (1)$$

Note that the due time d_j does not refer to the completion of the job but to the arrival time f_j^q at the quay crane. In case of a discharging job, the latter corresponds to the end of the drive to the pick-up location. Let us consider a discharging job j with assigned AGV a and waiting time for availability w_a of AGV a (we have $w_a = 0$ if AGV s is currently available). Then, we obtain $f_j^q = w_a + e_{ja}$ for discharging jobs. In case of a loading job, however, the due time refers to the end of the drive to the delivery location. Let h_{SC} be the estimated hand-over time at the stacking crane, and let t_{ja} be the estimated transportation time from the pick-up to the delivery location. Then, we have $f_j^q = w_a + e_{ja} + h_{SC} + t_{ja}$ for loading jobs.

We consider n jobs and n AGVs for the assignment problem. As outlined in Section 2, the n AGVs are those that are currently available and those which will soon complete their current job. The n jobs are given as the n most urgent jobs that are not yet in process, that is, the n jobs with the earliest due times among those jobs that have not yet been started.

3.2 Solution methods

To solve the due time based assignment problem, we employ two procedures. Both start by computing the set of jobs J and the set A of AGVs to be assigned, as described in the previous subsection.

The first approach is the Hungarian method of (Kuhn 1955) which is implemented as described in (Munkres 1957). This algorithm leads to an optimal assignment with respect to the due-time-based assignment costs given in Eq. 1.

The second approach is a simple greedy heuristic that will be used to provide benchmark results for the comparison. We employ a priority rule-based procedure similar to that of Hartmann (2004). The procedure repeatedly applies the following steps until each job has been assigned to an AGV, that is, until $J = \emptyset$ and $A = \emptyset$.

1. Select the job j to be assigned next as the most urgent job, that is, the job with the smallest due time $d_j = \min\{d_i \mid i \in J\}$.
2. Select the AGV a that leads to the smallest increase in the objective function, that is, the lowest possible costs $c_{ja} = \min\{c_{jb} \mid b \in A\}$ for job j .
3. Assign AGV a to job j .
4. Remove AGV a from A and job j from J , respectively.

3.3 Implications for stacking-crane decisions

The AGV assignment problem decides which empty AGV carries out which job, but it should not decide which container the AGV will actually receive. Consider two empty AGVs a and b with waiting times for availability $w_a < w_b$. Moreover, consider two jobs i and j with the same stack as pick-up location and with due times $d_i < d_j$. Let us assume that the AGV assignment decision was to assign job i to AGV a and job j to AGV b . It may happen that AGV b arrives at the stack before a (a may have been delayed due to congestion on the layout). Now the stacking crane should put container i on AGV b because container i is more urgent (note that one could say that AGVs a and b switch their jobs).

The stacking-crane decisions (i.e., which container is to be moved next) is based on various goals and requirements such as high waterside and landside productivity, short empty travel times, AGVs, short waiting times for external trucks, etc. Considering the interface to the AGVs, we assume that the stacking cranes make use of rules analogous to those employed for the AGVs when deciding which AGV should receive which container. This means that the stacking cranes prefer containers with earlier due times (in addition to their further goals). This does not have an impact on the AGV assignment problem itself, but is important when testing the AGV assignment approach in a simulation study as will be done in Section 5.

4 Inventory-based approach

4.1 Basic idea

At each quay crane, there is a waiting buffer for AGVs, that is, an area in which arriving AGVs have to wait until the quay crane is ready to serve them. This buffer can be seen as a storage. In this analogy, the quay cranes are customers which have to be supplied with goods. These goods correspond to AGVs. A loading quay crane requires AGVs with containers to be loaded, while a discharging quay crane requires empty AGVs on which a discharged container can be put. Like in

inventory management, the task is to make sure that no customer has to wait for goods, i.e., the inventory level should not be zero. On the other hand, the inventory level should not be too high. In our case, the latter is especially important because not only containers but also AGVs are tied up in stock. Hence, if queues become too long, there is a negative effect on the system's behavior because less transportation capacity is available.

Considering the AGV buffer as an inventory, we say that the inventory level of a quay crane is the number of AGVs in the buffer. Furthermore, the inventory level plus those AGVs on their way to the quay crane's buffer can be seen as the quay crane's net inventory level. To keep the analogy, we define a special net inventory level for our problem. We consider Q quay cranes $q = 1, \dots, Q$. For each quay crane q , the inventory level for assignment decisions ila_q is defined as the number of AGVs that are busy with a job of a quay crane q and have not yet reached q . Furthermore, we denote the set of AGVs belonging to ila_q as ILA_q , that is, we have $ila_q = |ILA_q|$. Note that for a loading quay crane q , ILA_q consists of the AGVs that are either waiting in the buffer at q , transporting a container towards q , waiting for a container for q at a stack, or driving to a stack where a container for q is to be picked up. For a discharging quay crane p , ILA_p contains those empty AGVs that are either waiting in the buffer at p or driving towards p (observe that AGVs transporting a container picked up at p do not belong to ILA_p).

Considering the analogy described above, the basic idea for assigning AGVs to jobs can be stated as follows: Whenever an AGV a should get a new job, assign a to the first unassigned job of the quay crane q whose buffer is most probably empty when a would arrive at q . According to the analogy to inventory management, we choose the quay crane q with the smallest ila_q . In other words, the next job of that quay crane q for which ila_q is minimal is the most urgent job. One may also say that quay crane q is the most urgent quay crane to receive an AGV. A methodology to assign jobs to AGVs, which is based on this basic idea, will be presented in Section 4.2.

There is another motivation for this idea: If we want to reduce waiting times of AGVs at quay cranes, we have to shorten the waiting queues. By sending the AGV to quay crane with lowest ila_q , we select the shortest expected waiting queue for the AGV to queue into.

However, the inventory levels ila_q , as described above, are not yet suitable for directly comparing the current needs of the quay cranes for further AGVs with each other. Obviously, the time an AGV needs to arrive at the quay crane is much longer for loading quay cranes than for discharging ones. In the former case, it includes driving to the stacking crane, waiting for service, and driving to the quay crane, while in the latter case there is just a direct drive to the quay crane. Naturally, to reach the same supply level for all quay cranes (or, in other words, the same productivity), the inventory level of loading quay cranes must be higher than that of discharging ones. Therefore, we introduce a parameter ϕ called phase factor by which the inventory level of loading quay cranes must be higher. We consider adapted inventory levels for loading quay cranes q by defining $ila'_q = ila_q/\phi$. The inventory levels of discharging quay cranes are not modified, that is, we set $ila'_p = ila_p$ for each discharging quay crane p . The urgency with which a quay

crane requires an AGV is now measured by inventory levels ila'_q for all quay cranes q .

So far, we have defined a quay crane q with inventory level ila'_q to be more urgent than that of a quay crane p if we have $ila'_q < ila'_p$. Finally, we consider quay cranes having the same inventory level, that is, $ila'_q = ila'_p$. To resolve such a tie, we define the quay crane for which the last AGV was started a longer time ago to be more urgent.

Note that ila'_q can further be modified to reflect operational issues in practice. One might wish to prioritize some quay crane q , e.g., if q has the longest remaining job list and must be accelerated to finish the vessel on time. This can be achieved by reducing ila'_q . This makes the jobs of quay crane q appear more urgent and, thus, leads to more AGVs for quay crane q . This should provide a higher productivity of q (although, of course, the productivity of the remaining quay cranes may decrease). This example shows the straightforward applicability of the inventory idea with respect to practical needs.

4.2 Assignment procedure

First, we determine all AGVs, say n , which are currently free or will be free within a short time as described in Section 2. Next, we find n jobs to be assigned to those available AGVs. At this point, we employ our basic idea as described in Section 4.1: The most urgent job is a job which belongs to the quay crane q , which has the lowest inventory level ila'_q . Among all those, we select a job all predecessors of which are already assigned to an AGV, are in transport, or are finished. By paying attention to the precedence relations when assigning AGVs to jobs, we reduce the risk of AGV waiting times at a quay crane that are caused by delayed predecessor containers. We then note the job just chosen as assigned, temporarily increase the corresponding ila'_q by one and, once again, determine the most urgent job based on the new inventory levels. This process loops until we have n jobs.

To assign the n jobs to the n AGVs, we create a standard linear assignment problem. The costs c_{ja} of assigning job j to AGV a consist of three components:

- An AGV a may have a current job that must be completed before it can start the next empty travel. The estimated waiting time for availability w_a obviously influences the duration until the next job j can be started as well as the duration until the AGV can arrive at the related quay crane. Note that w_a is zero if AGV a does not have a current job.
- According to the pick-up location of job j and the current position of AGV a , there is an expected empty travel time e_{ja} if j is assigned to a . This empty travel time affects the arrival of the AGV at the quay crane.
- We define $1 \leq o_j \leq n$ as the ordinal number of job j according to the order in which the jobs were chosen for assignment. That is, job j with $o_j = 1$ is the most urgent job with respect to the inventory levels ila'_q , job i with $o_i = 2$ is the second most urgent job and so on. (Note that o_j corresponds to the due time d_j in the due time based approach as both reflect the urgency of a job to receive an AGV.)

Now we define the cost as follows:

$$c_{ja} = (\lambda \cdot (n - o_j) + 1) \cdot (w_a + e_{ja})$$

The first part of the formula covers the job's urgency (λ is a weight to adjust the impact the job's urgency has on the costs). The urgency value of the least urgent job (ordinal number $o_j = n$) is 1. The next (more urgent) jobs have coefficients $1 + \lambda$, $1 + 2 \cdot \lambda$, $1 + 3 \cdot \lambda$ and so on. The second part of the formula reflects the estimated time that will pass until the container related to job j would be picked up if AGV a is assigned to job j . Multiplying both parts means that a very urgent job j (i.e., with a low o_j) and an AGV a that would need a long time to pick up the container related to job j have high assignment costs c_{ja} .

Having determined the costs c_{ja} , we solve the resulting assignment problem by the Hungarian method of (Kuhn 1955), designed as an executable in (Munkres 1957). This algorithm leads to an optimal assignment in terms of our objective to minimize the total assignment cost.

4.3 Implications for stacking-crane decisions

As already discussed in Section 3.3, stacking cranes are involved in the decision of which container to load on an AGV. Therefore, we describe a rule for loading containers which is, analogously to the assignment rule, based on net inventory levels.

We distinguish the loading decisions to be made when an AGV receives a container from a stacking crane, and those to be made when the AGV receives a container from a quay crane. In the latter case, the AGV simply receives an arbitrary container from the quay crane it is waiting at. In the former case, this decision is much more difficult: The stacking crane may have containers required by different quay cranes, thus, it has to decide which to pick first. To support the selection, we introduce a further inventory level. The inventory level for transport decisions ilt_q of a loading quay crane q is defined as the number of AGVs driving straight towards q after picking up a container for q at the stacking area. Additionally, we define the corresponding set of AGVs as ILT_q .

Then, we select the quay crane in a way similar to the assignment decision: We assume (see Section 3.3) a stacking crane to consider the quay crane q with the lowest ilt_q among all loading quay cranes having containers at the specific stacking crane as the most urgent quay crane. Again, we want to respect the precedence relations, namely only pick up containers whose predecessors are already picked up. However, it is possible that none of the containers to be loaded fulfills this precedence condition because we consider a subset of the containers. For example, it might occur that each container has at least one predecessor not picked up yet which stands at another stacking crane. Then, to prevent congestion as much as possible, we propose to start with strong requirement formulations and lower them step by step, if no container fulfills them. As soon as we find some containers, we select the one belonging to the most urgent quay crane.

Sending an AGV to a quay crane q with low ilt_q is motivated by reducing waiting times of quay cranes and AGVs. This idea directly corresponds to the one for selecting containers for the assignment process described in Section 4.2.

4.4 Enforcing dual cycles

An AGV's drive to the pick-up location is often necessary but worth avoiding if possible. It ties up the AGV capacity and, moreover, leads to more traffic in the terminal so the risk of congestion increases. Therefore, we provide a feature to be plugged into the decision process described so far.

A constellation of an AGV transporting a container to its destination and receiving a new job with a pick-up location equal to the previous job's delivery location is called a dual cycle. Dual cycles are possible only at stacks where quay cranes are either loading or discharging, which means they do not discharge a container immediately after loading another one in the same ship bay.

The assignment process described above arranges dual cycles only if there is a container with sufficient urgency at a stack where an available AGV is located. To suppress more empty drives, we take into account containers stored at a stack which would be ignored when creating the assignment problem in Section 4.2 because of a lack of urgency. Hence, we state an assignment rule as follows: If an AGV is available at a stack, it is assigned to the most urgent job located at this specific stack and whose predecessors already have been assigned or completed. As a result, we might assign a container which would not be considered by the basic method of Section 4.2 but offers a profitable dual cycle. This assignment process is executed right before the basic assignment process in Section 4.2. The jobs and AGVs assigned by this procedure are deleted from the corresponding sets. For the remaining AGVs, the assignment problem is created, solved, and evaluated as stated in Section 4.2.

Note that the AGV process in case of a dual cycle differs from the standard process only in that the empty travel to the pick-up location is actually a dummy drive-obviously, it takes no time because the last delivery location of the AGV corresponds to its next pick-up location. Afterwards, we decide which container to load on the AGV and select the most urgent one as described in Section 4.3. Therefore, we always arrange a dual cycle for the most urgent container of the specific stack (to be accurate, the AGV assignment procedure can only decide to leave the empty AGV at that stack, but we assume that the stacking crane scheduling selects the most urgent container with respect to the second inventory level ilt_q). Unfortunately, although this rule reduces empty travel times, it can also lead to undesirable effects which can be resolved as follows:

- As outlined in Section 4.1, we aim at inventory levels as similar as possible. By partially ignoring the urgency of jobs, we risk to disturb this balance. Therefore, we introduce two parameters $0 \leq \sigma, \tau \leq 1$ to prevent the balance from getting too much disturbed. Furthermore, we calculate the current minimum and maximum inventory levels among all quay cranes, that is, $ila_{all}^{min} = \min\{ila'_q \mid q = 1, \dots, Q\}$ and $ila_{all}^{max} = \max\{ila'_q \mid q = 1, \dots, Q\}$. Analogously, the current minimum and maximum inventory levels $ilt_{loading}^{min}$ and $ilt_{loading}^{max}$ among

the loading quay cranes are calculated. We employ them to formulate two conditions for a dual cycle concerning a specific candidate job j and its quay crane's q_j inventory levels ila'_{q_j} and ilt_{q_j} :

$$ila'_{q_j} \leq (1 - \tau) \cdot ila_{all}^{min} + \tau \cdot ila_{all}^{max} \quad (2)$$

$$ilt_{q_j} \leq (1 - \sigma) \cdot ilt_{loading}^{min} + \sigma \cdot ilt_{loading}^{max} \quad (3)$$

Following these conditions, we only choose a container for a dual cycle if it belongs to one of the more urgent quay cranes.

- Dual cycles only support loading quay cranes by more efficient use of AGVs (the AGV driving time for loading quay cranes is shortened on the average). Moreover, the dual cycle approach assigns AGVs to loading quay cranes that otherwise might have been assigned to discharging ones. Again, this disturbs the balance between loading and discharging quay cranes. Hence, we have to adapt the phase factor ϕ described in Section 4.1 to readjust that balance.

5 Simulation study

To compare and evaluate the two assignment approaches given in Sections 3 and 4, we developed a simulation model. In the following, we give some details of the simulation model, summarize the parameters employed, and, finally, discuss the results.

5.1 Model

According to our problem setting, we identify three substantial material flow components of the considered container terminal configuration (for a sketch of the terminal layout in the simulation model, we refer again to Fig. 1).

- Quay cranes load containers onto a vessel or discharge them from it. We can look at their life cycle as an endless loop of either waiting for AGVs or handling containers. When a quay crane holds a container to set down on an AGV or waits for a container to load on the vessel, it has to wait until an AGV arrives at the quay crane. After a quay crane's interaction with an AGV, it either transports the container onto the vessel (if loading) or picks the next container from it (if discharging). To characterize the quay crane's behaviour, we employ three distributions: hand-over time for AGVs to be loaded with discharged containers, hand-over time to get containers from AGVs to load them on a vessel, and the time the quay crane requires before it is ready for the next hand-over. The former two contain the processes of picking the container and lifting it up to a height that allows the AGV to leave (if loading) and putting the container down and releasing it (if discharging), respectively. The latter includes the container's travel to or from the vessel.

- Stacking cranes manage the stacking area and, therefore, receive containers from AGVs after they were discharged from vessels. Additionally, stacking cranes provide containers for AGVs to be loaded onto vessels. Both processes are modeled by distributions for the transfer times, that is, the times the AGVs have to wait at the stacks. As the behaviour of the stacking cranes is not modelled explicitly, these distributions implicitly contain all other activities such as shuffling containers and serving the landside.
- AGVs transport containers from quay cranes to the stacking area and vice versa. Their only activity to be modeled is driving. Therefore, a distribution for the driving time from each possible starting position to each possible destination position is registered in the model. These distributions cover interferences of AGVs on the layout, especially congestion. Moreover, for hand-over at the quay cranes and stacking cranes, an estimated availability time for an AGV is generated. Both the time at which the estimate is generated in advance and the error of the estimate (i.e., deviation from actual availability time) are controlled by distributions.

The simulation model has been implemented in Desmo-J, a discrete event-based simulation framework in Java (see Page et al. (2000)). A more detailed presentation of the simulation model can be found in Briskorn and Hartmann (2006).

5.2 Experimental design

To evaluate our approach, we compare four different methods to assign jobs to AGVs. First, we implemented the greedy heuristic described in Section 3, which we will refer to as “dueTimePrio.” Our own approach, which was described in Section 4, was realized both with (“invDualCycle”) and without forcing dual cycles (“inv”). Because we want to get results concerning the different methods to select containers for assignment, namely, the due-time-based rule and the inventory-based idea, we have to eliminate effects caused by different assignment methods. We achieve this by using the Hungarian method for assigning containers selected by the due-time idea in a fourth method, “dueTimeHung.”

We apply these approaches to scenarios that differ by the structure of the containers’ precedence relations. Varying this structure gives a hint about the capability of an approach because the structure defines the degrees of freedom which are left for it. Obviously, precedence relations between containers i and j can solely exist if i and j belong to the same quay crane. We considered five structures of precedence relations:

- The lowest requirement level is given in a scenario without precedence relations. The approaches can randomly choose containers to load or discharge when available.
- The strongest requirement level is given by “linear” precedence relations between the containers of each quay crane. Then, at each point of time there is just a single container for each quay crane, which can be loaded or discharged.

- In addition, we have three settings with partial precedence relations. They are different with respect to the number of precedence relations per job, leading to scenarios with “many,” “medium,” and “few” precedence relations per job.

In each scenario there are 20 stacking cranes and 40 AGVs. Ten quay cranes, of which five are loading and five are discharging, are randomly distributed on the 20 possible positions. We created 60 jobs per hour and quay crane. Note that this roughly corresponds to the maximum technical productivity of a quay crane. This way, the actual throughput results from the AGV dispatching strategy under consideration. The distributions for the material flow behavior mentioned in Section 5.1 were taken from statistics of the Container Terminal Altenwerder. The original statistics were modified for reasons of confidentiality, but the resulting distributions still allow for a realistic simulation.

For the simulation runs, we identify four goals resulting from the discussion in Section 2. We use them to compare the approaches:

- Increasing the container terminal’s waterside productivity, i.e., the number of containers loaded onto and discharged from vessels per hour, is the main goal of our approach.
- Waiting times of quay cranes increase the time of the vessels in port. Hence, we want to reduce them.
- Waiting times of AGVs tie up capacity without having any positive effect on the system’s productivity, so we want to reduce them.
- Empty travel times should be shortened because, like waiting times, they tie up capacity without supporting the main goal. Besides, they increase traffic on the AGV layout and, therefore, the probability of congestion.

We carried out two series of simulation runs. In preliminary experiments, we tested a broad variety of values for each parameter while fixing others. After evaluating these runs, we fixed all parameters to their best settings for further experiments. Tables 1 and 2 give the fixed values of essential parameters. Note that phase factor ϕ has to be adapted according to Section 4.4 when dual cycles are forced.

For each approach, we performed 100 simulation runs with a simulation time of 11 h per run, which were preceded by 2 h to get the system in balance and followed by 2 h to make sure containers were not running out in the period to be evaluated. Solely, the period of 11 h is evaluated by means of statistics.

Table 1 Parameters for due time approach

Parameter	Symbol	Value
Earliness weight	α_E	1
Tardiness weight	α_T	7.5
Empty driving weight	α_e	1

Table 2 Parameters for inventory approach

Parameter	Symbol	Value
Phase factor	ϕ	1.6
Cost step	λ	3
Dual cycle	τ	1
Dual cycle	σ	0.5

5.3 Comparison of the approaches

In the following, we present the results of the simulation runs, taking into account our four approaches and five different scenarios.

Table 3 gives an overview of the productivity resulting from the different approaches. The productivity is measured as the average number of containers loaded or discharged per hour and quay crane. Although we used neither the original approach employed at the Container Terminal Altenwerder nor the original statistics, we cannot give absolute productivity figures in this paper to avoid misinterpretations. Therefore, the results are given as relative figures. We selected “dueTimePrio,” the simplest method in our study, as a base and set its productivity index to 1.0 for each of the five scenarios. The productivity resulting from the other methods are given relative to those of “dueTimePrio” (e.g., 1.015 of “dueTimeHung” for the “medium” scenario indicates a productivity improvement of 1.5% over “dueTimePrio”).

One can observe that productivity using “dueTimeHung” is slightly higher in each scenario than when “dueTimePrio” is applied. Remember that these approaches only differ in the algorithm, not in how the most urgent jobs are determined or how job assignments are evaluated. The results show that the Hungarian method is better suited than the greedy heuristic, although the productivity is increased only by 1.0–1.8%. Furthermore, “inv” reaches a higher productivity than “dueTimeHung”. These two approaches make use of the same algorithm (i.e., the Hungarian method) but employ different problem formulations. Therefore, we can say that the inventory-based concept is more promising than the due-date approach. In particular, we can see that the improvement due to the inventory concept is higher than the improvement that can be obtained from using an optimal algorithm in the due-time-based model. When comparing “inv” and “invDualCycle”, we observe that using the option to enforce dual cycles in the inventory-based approach seems to be extremely promising. Also, note that the

Table 3 Quay crane productivity

Precedence relations	dueTimePrio	dueTimeHung	Inv	invDualCycle
Linear	1	1.010	1.050	1.049
Many	1	1.014	1.046	1.059
Medium	1	1.015	1.045	1.183
Few	1	1.014	1.075	1.229
Without	1	1.018	1.047	1.190

superiority of the dual cycle approach further increases, if there are less precedence relations.

Table 4 gives an impression of the influence the approaches have on the total empty travel time of AGVs. Again, the Hungarian method in “dueTimeHung” is superior to the simple priority rule in “dueTimePrio”. The inventory-based approach leads to smaller empty travel times than the due-time-based approach. Obviously, enforcing dual cycles strongly reduces empty driving times. The effect of dual cycles on the empty travel times increases with decreasing number of precedence relations. This is because less precedence relations make it more likely to fulfill the conditions for arranging dual cycles on a higher requirement level (see Section 4.3), which will reduce congestion in front of the quay crane.

Table 5 shows the waiting times of the AGVs in the buffer at the quay crane. Recall that AGVs have to wait in this buffer, if more AGVs than the quay crane can handle have been assigned to this quay crane, or if AGVs have to wait for delayed predecessors. We can see that the inventory-based approach reduces waiting times of AGVs significantly. If the dual cycle extension is considered, the waiting times of the AGVs are higher than otherwise. The latter results from the drawback discussed in Section 4.4: By enforcing dual cycles, we partially ignore the urgency of containers. Therefore, it becomes more likely that we send AGVs to quay cranes with higher ila_q . Hence, AGV queues get longer and waiting times in the buffer increase.

The waiting times of the quay cranes are given in Table 6. Again, the inventory-based idea leads to better results than the due-time approach. Enforcing dual cycles reduces the quay crane waiting times even further.

5.4 Impact of the look-ahead

As described in Section 2, the assignment procedures consider the AGVs that are currently free and those that will soon be available. This implies that we have a certain look-ahead which gives us more degrees of freedom for finding good assignments. On the other hand, considering AGVs that are not available yet means that we have to take estimated availability times into account, and the quality of such estimates is not so good in practice. Thus, to validate the look-ahead approach, we compare it with a version without look-ahead that considers only AGVs that are currently available.

The results can be found in Table 7. We compare the inventory-based approach with and without look-ahead and report relative productivity (with the version without look-ahead being the base). The version with look-ahead leads to 10%

Table 4 Empty travel times of AGVs

Precedence relations	dueTimePrio	dueTimeHung	Inv	invDualCycle
Linear	1	0.955	0.906	0.876
Many	1	0.951	0.906	0.814
Medium	1	0.943	0.924	0.580
Few	1	0.952	0.914	0.559
Without	1	0.950	0.919	0.529

Table 5 AGV waiting times in buffer at quay crane

Precedence relations	dueTimePrio	dueTimeHung	inv	invDualCycle
Linear	1	1.032	0.860	0.944
Many	1	1.027	0.881	1.036
Medium	1	1.075	0.666	0.696
Few	1	1.039	0.911	0.964
Without	1	1.007	0.601	1.052

higher productivity. This confirms that having more degrees of freedom for optimization is more important than the often low quality of the availability time estimates.

Finally, it should be mentioned that the inventory approach with look-ahead considers 4.6 AGVs on average when executing the assignment procedure. In the version without look-ahead, however, usually only one AGV is considered in cases of high workload (which are simulated in this case and which are most important in practice). This is because the assignment procedure is executed whenever an AGV has completed its last job (cf Section 2). In case of high workload, all other AGVs are busy, hence, that AGV is the only one free when the assignment procedure is executed.

5.5 Impact of precedence relations

Finally, we have a brief look at the impact of the precedence relations on the productivity, empty travel times, waiting times of AGVs in the quay crane buffer, and quay crane (QC) waiting times for AGVs. The results are displayed in Table 8. We consider only the greedy priority-rule-based heuristic for the due date approach (dueTimePrio), which has been the benchmark in our study. As in the previous tables, we give relative results. Here, we have selected the linear precedence relations as a basis for the comparison.

We observe a significant influence of the precedence relations' density on the results. In particular, having less precedence relations leads to higher productivity. If we have no precedence relations at all, the productivity (with the same heuristic) is 11.8% higher compared to the case of linear precedence relations. This is because less precedence relations make it less likely that an AGV has to wait for a delayed predecessor in the buffer at a loading quay crane. This is confirmed by Table 8,

Table 6 Quay crane waiting times for AGVs

Precedence relations	dueTimePrio	dueTimeHung	Inv	invDualCycle
Linear	1	1.032	0.860	0.944
Many	1	0.990	0.974	0.955
Medium	1	0.982	0.957	0.800
Few	1	0.990	0.962	0.837
Without	1	0.987	0.977	0.838

Table 7 Impact of look-ahead on productivity (method: inv)

Precedence relations	Only free AGVs	Free and soon available AGVs
Linear	1	1.090
Many	1	1.097
Medium	1	1.099
Few	1	1.095
Without	1	1.107

which shows that the AGV waiting times in the quay crane buffer decrease drastically when we have less precedence relations.

5.6 Computation times

We close this section with a brief look at the times required to compute one assignment. In both the due time and the inventory approach, the average computation time for one execution of the Hungarian method has been below 0.001 s. The maximum computation time for one execution has been 0.016 s. The experiments were carried out on an Athlon XP 2200+ computer with 512 MB RAM. These computation times show that the approaches presented in this paper are well suited for application in a terminal control system, which requires decisions in real time.

6 Conclusions and outlook

In this paper, we proposed an approach to schedule container transports between quay cranes and the stacking area. We captured the problem of assigning transportation jobs to AGVs by introducing a concept related to inventory management. The essential idea is to assign an AGV to a job that belongs to a quay crane to which a relatively small number of AGVs is currently assigned. This problem formulation was compared to a more traditional formulation that is based on due times for the jobs and an earliness–tardiness objective. Both formulations differ only in how the jobs to be considered are determined and in the way the assignment costs of jobs to AGVs are calculated, but not in the underlying mathematical structure.

Table 8 Impact of precedence relations (method: dueTimePrio)

Precedence relations	Productivity	Empty travel	AGV waiting	QC waiting
Linear	1	1	1	1
Many	1.035	1.000	0.744	0.982
Medium	1.065	0.981	0.370	0.973
Few	1.071	0.993	0.256	0.964
Without	1.118	0.989	0.242	0.943

In a simulation study, we found that the problem formulation has an impact on the resulting terminal productivity. Even when both problem formulations are solved with the same algorithm (the well-known Hungarian method), the inventory-based concept outperformed the due-time-based approach with respect to waterside productivity (although only by a few percent). At first glance, the due-time approach seems to allow for more precise scheduling because it accurately plans events and durations on the terminal. However, our results indicate that the bad time estimates, which are common in practice (and which were considered in our simulation model in a realistic way), lead to suboptimal decisions in the due-time approach and, thus, to lower productivity. The inventory-based approach avoids the use of estimated times to a large extent. Hence, it appears to be more robust and, thus, better suited for application in practice. Besides, it leads to a simpler terminal control system because frequent updates of times are not necessary.

Additionally, we introduced a feature to enforce dual cycles of AGVs at stacks (that is, a stacking crane unloads a container from the AGV and puts another on the AGV). This allows reduction of the empty travel times of the AGVs and, as shown by our results, leads to higher waterside productivity.

Furthermore, we analyzed the impact of the precedence relations both on the productivity and on the performance of the different approaches. Less precedence relations between containers to be loaded onto vessels lead to higher productivity. This is due to more degrees of freedom for the AGVs, that is, in case of fewer precedence relations, AGVs can directly proceed to the quay crane without having to wait for a delayed predecessor to pass. Moreover, the additional productivity gain of the dual cycle extension increased with a decreasing number of precedence relations.

Considering the good results of the inventory-based concept for AGV dispatching, an objective of further research should be the application of this approach to other types of equipment for container handling. In particular, inventory-based optimization would be promising for stacking cranes and straddle carriers. In both cases, the inventory idea would have to be adapted to reflect the specific requirements of those types of equipment.

References

- Bae JW, Kim KH (2000) A pooled dispatching strategy for automated guided vehicles in port container terminals. *Int J Manag Sci* 6:47–70
- Baker C (1999) Altenwerder—the details. *Port Dev Int* (07/08):24–25
- Bish EK, Chen FY, Leong YT, Nelson BL, Ng JWC, Simchi-Levy D (2005) Dispatching vehicles in a mega container terminal. *OR Spectrum* 27:491–506
- Böse J, Reiners T, Steenken D, Voß S (2000) Vehicle dispatching at seaport container terminals using evolutionary algorithms. In: Sprague RH (ed) *Proceedings of the 33rd Annual Hawaii International Conference on System Sciences*. IEEE, Piscataway, pp 377–388
- Briskorn D, Hartmann S (2006) Simulating dispatching strategies for automated container. *GOR Proceedings 2005* (in press)
- Daganzo CF (1989) The crane scheduling problem. *Transp Res B* 23:159–175

- de Castilho B, Daganzo CF (1993) Handling strategies for import containers at marine terminals. *Transp Res B* 27:151–166
- de Koster RBM, Le-Anh T, van der Meer JR (2004) Testing and classifying vehicle dispatching rules in three real-world settings. *J Oper Manag* 22:369–386
- Grunow M, Günther H-O, Lehmann M (2004) Dispatching multi-load AGVs in highly automated seaport container terminals. *OR Spectrum* 26:211–235
- Guan Y, Cheung RK (2004) The berth allocation problem: models and solution methods. *OR Spectrum* 26:75–92
- Hartmann S (2004) A general framework for scheduling equipment and manpower at container terminals. *OR Spectrum* 26:51–74
- Hwang H, Kim SH (1998) Development of dispatching rules for automated guided vehicle systems. *J Manuf Syst* 17:137–143
- Imai A, Nagaiwa K, Tat CW (1997) Efficient planning of berth allocation for container terminals in Asia. *J Adv Transp* 31:75–94
- Imai A, Nishimura E, Papadimitriou S (2001) The dynamic berth allocation problem for a container port. *Transp Res B* 35:401–417
- Kim KH, Kim HB (1999a) Segregating space allocation models for container inventories in port container terminals. *Int J Prod Econ* 59:415–423
- Kim KY, Kim KH (1999b) A routing algorithm for a single straddle carrier to load export containers onto a container ship. *Int J Prod Econ* 59:425–433
- Kim KH, Park YM, Ryu K-R (2000) Deriving decision rules to locate export containers in container yards. *Eur J Oper Res* 124:89–101
- Kim KH, Won SH, Lim JK, Takahashi T (2001) A simulation-based test-bed for a control software in automated container terminals. *Proceedings of the international conference on computers and industrial engineering, Montreal, Canada*, pp 239–243
- Kuhn HW (1955) The Hungarian method for the assignment problem. *Nav Res Logist Q* 2:83–97
- Lim A (1998) The berth planning problem. *Oper Res Lett* 22:105–110
- Meersmans PJM, Dekker R (2001) Operations research supports container handling. Technical report EI 2001-22. Econometric Institute, Erasmus University, Rotterdam
- Meersmans PJM, Wagelmans APM (2001) Effective algorithms for integrated scheduling of handling equipment at automated container terminals. Technical report EI 2001-19. Econometric Institute, Erasmus University, Rotterdam
- Munkres JR (1957) Algorithms for the assignment and transportation problems. *J Soc Ind Appl Math* 5:32–38
- Page B, Lechler T, Claassen S (eds) (2000) *Objektorientierte simulation in Java mit dem Framework DESMO-J*. BoD GmbH, Norderstedt
- Park Y-M, Kim KH (2003) A scheduling method for berth and quai cranes. *OR Spectrum* 25: 1–23
- Peterkofsky RI, Daganzo CF (1990) A branch and bound solution method for the crane scheduling problem. *Transp Res B* 24:159–172
- Steenken D, Henning A, Freigang S, Voß S (1993) Routing of straddle carriers at a container terminal with the special aspect of internal moves. *OR Spectrum* 15:167–172
- Steenken D, Voß S, Stahlbock R (2004) Container terminal operations and operations research—a classification and literature review. *OR Spectrum* 26:3–49
- Taleb-Ibrahimi M, de Castilho B, Daganzo CF (1993) Storage space vs. handling work in container terminals. *Transp Res B* 27:13–32
- Vis IFA (2006) Survey of research in the design and control of automated guided vehicle systems. *Eur J Oper Res* 170:677–709
- Vis IFA, de Koster R (2003) Transshipment of containers at a container terminal: an overview. *Eur J Oper Res* 147:1–16
- Vis IFA, Harika I (2004) A comparison of vehicle types for automated container terminals. *OR Spectrum* 26:117–143
- Zhang C, Liu J, Wan Y-W, Murty KG, Linn RJ (2001) Storage space allocation in container terminals. Technical report, Hong Kong University of Science and Technology