

Evaluating impact of truck announcements on container stacking efficiency

Eelco van Asperen · Bram Borgman ·
Rommert Dekker

© The Author(s) 2011. This article is published with open access at Springerlink.com

Abstract Container stacking rules are an important factor in container terminal efficiency. We build on prior research and use a discrete-event simulation model to evaluate the impact of a truck announcement system on the performance of online container stacking rules. The information that is contained in the announcement, i.e., the expected departure time for an import container, can be used to schedule pre-emptive remarkshalling moves. These moves can then be performed when the workload is low in order to decrease the export time and the crane workload at peak times.

Keywords Container stacking · Marine terminals · Container rehandling · Truck appointment system · Simulation

1 Introduction

Marine container terminals play a central role in the worldwide distribution of goods that is essential to global supply chains. These container terminals link sea transport via container ships to land transport via trains, barges, and trucks. One of the main

E. van Asperen (✉)

Center for Maritime Economics and Logistics, Erasmus University Rotterdam,
Burg. Oudlaan 50, 3062 PA Rotterdam, The Netherlands
e-mail: vanasperen@ese.eur.nl

B. Borgman

Quintiq, Bruistensingel 500, 5232 AH Hertogenbosch, The Netherlands
e-mail: borgman.bram@gmail.com

R. Dekker

Econometric Institute, Erasmus University Rotterdam, Burg. Oudlaan 50,
3062 PA Rotterdam, The Netherlands
e-mail: rdekker@ese.eur.nl

problems in container terminals concerns the stacking of containers. Although it is also one of the main advantages of containers, viz. that they can be stacked on top of each other, additional work is required if the bottom container is needed. In that case the top containers have to be moved to another place. These unproductive moves are called reshuffles. They are undesirable as they take additional handling time by the scarce resources.

Accordingly, every terminal needs a stacking strategy. The main objectives of such a strategy are (1) the efficient use of storage space, (2) limiting transportation time from quay to stack and beyond, (and vice versa), and (3) the avoidance of reshuffles. Of course, the importance of each criterion varies from terminal to terminal. Ports like Singapore and Hong Kong have limited land space, so they need to use the available storage space as efficient as possible. Note also that these objectives are conflicting: you cannot maximize them all. For example, the third objective would be optimized by having stacks of only one container high; however, this would lead to very inefficient use of storage space and long travel times.

One of the main problems facing a marine container terminal operator is a lack of detailed knowledge regarding the departure time and mode of the containers. While some information may be available at the time of arrival, the mode may change (as containers are rerouted by the shipper) and the departure time is typically a rough estimate for non-scheduled transport modes such as trucks. In this paper we investigate the impact of more accurate departure time information for truck exports (containers that leave the container terminal by truck) that becomes available after the container has been stacked.

This paper builds on our prior work regarding container terminal stacking (Dekker et al. 2006; van Asperen 2009; Borgman et al. 2010). In these studies we focus on the short term decision to allocate an incoming container to a stacking position. In this paper, we investigate a number of scenarios to see whether the more precise information on the time a container that is going to be picked up by truck can have an influence on the time it takes to remove the container from the stack. While some studies have focused on possible improvements at the entry of the terminal (the gate area), we limit our scope to the effects on the stacking yard and the potential of the additional information contained in the detailed departure times for truck pick-ups.

We consider an import container terminal with a high uncertainty regarding the departure time of the containers, such as terminals in the Hamburg–Le Havre range in Europe and the USA. Moreover, we consider online stacking rules, which do not require extensive computations and can be used in many types of stacks and for large numbers of containers. The type of container terminal we consider uses a high degree of automation. This typically implies a tight coupling of the various activities and a reduced flexibility to handle unexpected circumstances. Also, the terminal under consideration has both high yard utilization and high berth utilization. This reduces the potential to optimize the storage yard by performing extensive housekeeping moves at times of low utilization. It is therefore relevant to research whether the more accurate departure time information allows for a reduced exit time for containers from the stack.

After this introduction, we will first discuss the available academic literature before we explain how we study container stacking with a detailed simulation model. In the section on experimental setup, we describe the settings used for the simulation experiments. Next, we define the main performance indicators used to evaluate a range of stacking algorithms. We then discuss the results of the simulation experiments and formulate our conclusions.

2 Literature review

The academic literature on stacking problems is scant but growing, perhaps because the problem does not easily lend itself to analytical solutions (Dekker et al. 2006). In an overview paper on operations research at container terminals, Stahlbock and Voss (2008) looked at a number of aspects of container terminal operations. Among the topics surveyed were stowage planning, berth allocation, crane optimization, terminal transport optimization, and storage and stacking logistics. Their work is an extension of an earlier overview (Steenken et al. 2004), which also contains a paragraph on how stacking is done in practice. Stacking problems exist at three time levels, viz. strategic/design, tactical and operational. At the strategic/design level one chooses the stacking equipment (automated or manual, layout and maximum stacking height). At the tactical level one chooses a stacking and equipment operation strategy. At the operational level one determines which container to stack where and one schedules the equipment. This paper focuses on the last level. Caserta et al. (2011) give a nice overview of different circumstances in stacking. We concentrate in this paper on import containers, which are brought by ship and are transferred in the terminal to truck, rail or barge. We especially concentrate on the exchange to trucks, which is in most terminals the largest category. Other stacking papers concentrate on export containers, which are brought by truck to a terminal and leave by ship. For these containers one has to retrieve them from a stack. When the ship has arrived one has made a loading plan which specifies the exit time. Accordingly one has more information than for import containers. For the latter category few papers exist. We like to mention Dekker et al. (2006) and Borgman et al. (2010), who compare several stacking rules in a simulation study for an automated container terminal. This paper builds forth on their results. A very recent paper on stacking import containers is from Park et al. (2011), who use a dynamic updating policy in their simulation to determine the best stacking position of an incoming container based on several criteria, like travel time, crane utilisation, etc. They do not model the residence time for import containers however, and also do not consider truck announcements.

The use of Truck Appointment Systems (TAS) for external trucks at container terminals has gained some industry attention but the academic literature is still limited. Some authors refer to “Vehicle Appointment System” or “Container Scheduling System”. We have not encountered any papers on container stacking problems that explicitly address the issue of (truck) announcements.

The use of TA systems can be related to solving terminal gate congestion, to improve the turnaround time for external trucks, to reduce the environmental impact

of waiting trucks, and to improve stacking operations. The first reported case of a TAS is for the Hong Kong International Terminal (Murty et al. 2005), where it was aimed at making the most of the limited available space in the terminal. The introduction in 2002 of the Lowenthal Bill in California (USA) imposed limits of the waiting times for trucks within a terminal in order to reduce the environmental impact. The use of TAS was proposed as one of the possible solutions (Guiliano and O'Brien 2007). Plata Peredo (2008) describes the use of TAS to minimize terminal handling time for trucks and to reduce the cost for trucking companies. The results of a TAS implementation in Vancouver, Canada were disappointing (Morais and Lord 2006) as the truck waiting time and congestion were not reduced as expected. The inability of truckers to plan trips was cited as a major cause of this result.

Following a period of heavy congestion for trucks at one of the main container terminals in the port of Rotterdam, a number of studies have looked at the potential of TA systems to reduce this congestion. Van der Heide and van Vliet (2009) give an overview of the IT architecture of a TAS and discuss the potential impact for a non-automated container terminal that uses straddle-carriers. Vlugt (2009) evaluated the potential of a TAS for the automated ECT Delta terminal in the port of Rotterdam. The results showed that the decrease of the expected average truck turnaround time for trucks with an appointment is countered by an increased turnaround time for trucks without an appointment. The net result was that the average turnaround time did not decrease.

So far, the reported results are disappointing with regard to reducing gate congestion and truck turnaround time. In this paper, however, we focus on the possible impact of a truck appointment system on the performance of the (un)stacking operations in the terminal yard, i.e., we are more interested in the impact on the internal terminal operations. As the focus is on the terminal yard, we are not concerned with gate congestion and environmental impacts. We are more interested in the additional information a TAS may generate with regard to the actual time of departure of a stacked container and use that information in the stacking by doing necessary reshuffles ahead of the collection. This is called intra-bay premarshalling by Caserta et al. (2011). They list some papers who address this problem, but mainly in case of export containers with perfect information. These papers also focus on the crane scheduling. In our case new containers come in, with limited information, at the same time as the remmarshalling operations, which complicates the problem and these approaches are not possible. Finally, we would like to state that the term “truck announcement” indicates that the actual coordination between trucking company and terminal operator is outside the scope of this research.

3 Basic concepts

We want to investigate the impact of having more information about truck arrivals on container stacking efficiency. As trucks are typically the least controlled transport modes that are served by marine container terminals, having more information could be beneficial. In this paper, we do not use a detailed allocation

mechanism for trucks (i.e., the time of arrival is not determined by a slot allocation mechanism). Instead, we consider the situation in which the terminal operator is informed of the arrival time a certain number of hours beforehand.

In our previous research (Dekker et al. 2006; Borgman et al. 2010) we have looked at the performance of a number of container stacking algorithms. The main question we now face is how each stacking rule will benefit from advanced knowledge of truck arrivals. Incoming containers that have to be stored in the stack do not pose problems as long as the information on the departure time and mode are available. This is however not always the case. Information may not be available or it may be incorrect. If we consider the different transport modes a container terminal has to deal with, then road transport by truck is the least coordinated. The most disruptive influence of these uncoordinated truck arrivals occurs when a truck wants to pick-up a container. This container has to be retrieved (“unstacked”) from the yard, potentially causing one or more unproductive moves (reshuffles) in order to access the container. With advance information on the precise time and mode of departure we may be able to perform the reshuffle operations using idle time of the automated stacking crane (ASC) between the moment the departure time information becomes available and the actual departure time. While this in itself will not reduce the number of unproductive moves, we may be able to perform these unproductive moves at a time when they do not impact the performance of the system, i.e., during idle time of the stacking crane.

If we consider a single lane of the type of container terminal under investigation, we see that there is a single rail-mounted stacking crane that has to perform all the stacking moves for that lane. In this front-end interchange design there is an interchange point between the horizontal transport (AGV, straddle carrier, or yard truck) and vertical transport (by the automated stacking crane) at either end of the lane. We can distinguish between containers that are moving into the lane (i.e., that are being stacked) and containers that are moving out of the lane. Containers can enter and leave the lane at two sides: at the quay side (for containers that are coming from or going to ships) and at the land side (for all other modes of transport). Figure 1 provides a schematic overview of the terminal layout. The layout of this terminal has the stacking lanes perpendicular to the quay. Each lane has a length, a width, and a height.

The overall approach of the experiments in this paper is focused on the operational decisions that have to be made by terminal operators. Specifically, we take the arrivals and departures that are specified as part of the generator output (v.i.) and perform these operations. There is no global optimization or explicit planning; the operations are performed one at a time, i.e. in a greedy fashion, whenever a container arrives or has to leave the stack. We do not consider future events such as other incoming containers. The pre-arrival information with regard to the truck arrivals is used as follows:

1. The container is locked, i.e. no container may be placed on top of it;
2. If the ASC has an empty work queue, then it checks whether the locked container is on top of the pile. If it is not, the ASC tries to reshuffle the containers on top of the locked container in order to free it for easy retrieval. Next, the container is restacked using the stacking algorithm.

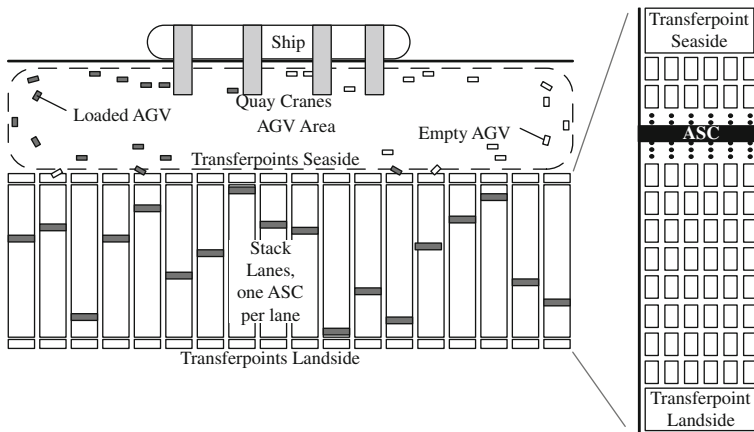


Fig. 1 Terminal layout

In other words, if the crane is going to be idle between the moment we receive the announcement and the arrival time of the truck, we may be able to perform the required reshuffles without interfering with other jobs for the crane.

4 Simulation model

The simulation model that was developed for the experiments in this and previous papers consists of two major components: a generator and a simulator. Both programmes have been implemented in the Java programming language; the simulator uses the SSJ discrete-event simulation library (L'Ecuyer and Buist 2005).

The generator program creates arrival and departure times of the containers. The generator is based on the same data as the generator in Dekker et al. (2006), including sailing schedules and a modal-split matrix. For the experiments in this paper, the modal-split matrix has been updated to reflect the recent increase in truck volume in comparison with the scenarios studies in Dekker et al. (2006). The output of the generator is a file that contains the ship arrivals, details of the containers to be discharged and loaded, and the specification of the destination of each container. The departure time is specified as the planned (a.k.a. expected) departure time and the actual (a.k.a. real) departure time. The destination can be another deep-sea vessel or (for import containers) a short-sea vessel, barge, or truck. For each container the location of the individual container within a ship is specified. The generator takes the detailed quay crane sequences for loading and discharging into account.

The simulator program reads the output of the generator and performs the stacking algorithms. The main task of the simulator is to keep track of the state of the stacking area (i.e., track the details of all the containers stored there) and to implement the moves specified by the output of the generator. The stack itself and the automated stacking cranes are simulated in great detail while components such as the horizontal transport by straddle carriers (on the land side) and AGV's (on the

sea side) are simulated in less detail. The core of the simulator itself is deterministic: the stochastic components are in the generator and, optionally, in the stacking algorithm. This setup facilitates a comparison of stacking algorithms as any changes in the statistical output of the simulator must be caused by the stacking algorithm. This approach can be considered a way to implement common random numbers. More details of the generator and simulator are described in Borgman (2009).

5 Experimental setup

The experiments in this paper all use the following configuration. Experiments are run for a 15 week period, of which 3 weeks are used for warm-up (to initialize the stack). We have used the generator program, discussed in Sect. 4, to generate an arrivals file containing 139,736 arriving containers during the 12 weeks of simulation (45.2% 20ft, 48.4% 40ft, and 6.4% 45ft). There are 19,516 containers that leave the terminal by truck. Some stacking rules have a stochastic component (such as selecting a position at random). As in our previous experiments we have used ten replications to get statistically robust results (we also tested with 30 replications but the reduction in output variability did not justify the additional computation time). These replications are used to compute the 95%-confidence intervals of the means.

We assume that there are sufficient AGVs and straddle carriers to ensure that these resources do not act as bottlenecks. The basic configuration for the stacking area was adapted from the earlier work by the same authors (Borgman et al. 2010), but resized to accommodate the increased number of containers and the various container sizes. The stack configuration is now 19 lanes, each consisting of 6 segments, which in turn have 55 positions of 20 ft. Each segment is dedicated to one size of container (20/40/45ft) in proportion with the number of containers for each size. Mixing container sizes within a segment fragments the available space, which makes stacking 40 and 45 ft containers very difficult. The length of the lanes was increased because the average size of a container is higher and the average dwell time is longer in comparison with our prior experiments. Keeping the lanes short would thus result in less containers per lane, and hence less work for the ASCs. The increase in the number of lanes was still required to accommodate the increase in the absolute number of containers. Each lane still has a single ASC per lane and its operation is modelled in detail, including travelling and hoisting, hence it can be a bottleneck. At the arrival or departure time of a container, an ASC job is created to move to the container and bring it to the specified transfer point, after which it continues with other jobs. The maximum stacking height is three containers. The average residence time of a container is 4.6 days with a standard deviation of 1.8 days and the average utilization of the yard is 67%.

6 Experiments

In this section we will present our experiments with a number of stacking rules. Before we discuss the stacking rules we should first consider how we are going to

evaluate the stacking rules. We measure the performance of a stacking algorithm with the following statistics:

6.1 Exit time (ETQ and ETL)

The exit time is the time (in hours) between the moment the operation to remove the container from the stack is planned (this is the time listed in the generator file) and the moment the container is ready for onward transport (to the quay or to a truck/train/barge). This time is measured for each side (quay-side and land-side) of the stack and will be listed as ETQ and ETL respectively. The exit time is the main performance indicator for a stacking algorithm. It is negatively influenced by stacking further away from the exit point and by any reshuffles that are needed to retrieve the container. In case of truck announcements we may do reshuffles before the planned time, but the ETQ or ETL is measured only from the planned time on. When a container enters the stack, the time it takes to perform this operation is determined by the workload of the ASC (how many jobs are in the current job queue) and the time it takes the ASC to move the container to its position. There are no reshuffles when containers are stored in the stack; reshuffles only occur when a container has to leave the stack. As we consider these two statistics the main performance indicators, we also report the 90% percentile of the mean.

6.2 Crane workload (ASC)

The automated stacking cranes are critical components for the overall performance so we measure the percentage of time that the ASC's are busy. (The crane workload will be denoted as ASC in the results.)

6.3 Reshuffles (RDC and ROC)

For the unproductive reshuffle moves at the departure moments, we measure the number of reshuffles (denoted as RDC) as a percentage of the total number of container movements. To get an indication of the number of reshuffles that happen per move, we also measure the reshuffle occasions (again as a percentage of the total number of container movements, denoted as ROC); a single reshuffle occasion implies one or more reshuffles. Reshuffles created by truck announcements are not included in RDC or ROC. These numbers are not absolute indicators of performance as the time of the reshuffle is not reported. A reshuffle that occurs when the workload is low has less impact on the overall performance than a reshuffle during a peak workload, for example when handling a very large vessel.

6.4 TAS impact (DART)

To measure the impact of the truck announcements, we track the number of containers that can be accessed directly (i.e., without reshuffles) when a truck comes to pick them up. This Direct Access when Retrieved for Trucks (DART) is a

measure of the potential impact of the announcements and is expressed as a percentage of the total number of truck pick-ups.

The stacking algorithms are described in more in detail in Dekker et al. (2006) and Borgman et al. (2010). A short description for each of the algorithms is listed in Table 1. We first test the impact of truck announcement with the reference algorithms (random stacking and levelling). To test the potential benefit of having this more accurate information on the container departure time, we have run both algorithms without the truck announcement and with announcement a number of hours before the departure time (ranging from 0.5 to 24 h). The actual announcement times are not fixed constants but drawn from a uniform distribution around the stated mean value (for example, for the 4 h announcements, the interval is between three and 5 h). The results are listed in Table 2.

Table 1 Stacking algorithms

RS	Random Stacking: The new container is placed at a randomly chosen allowed location, with every allowed location having an equal probability of being chosen. This algorithm is also applied for reshuffling, with the difference that we then only want to search the lane the container is in
LEV	Leveling: The idea is to fill lanes in layers, so that all empty ground positions are filled with containers first, before containers are stacked upon others. The stacking lane is filled from the transfer point quayside on
RS-DT	The Random-Stacking with Departure Times algorithm searches for a random pile where the top container's departure time is after the new container's departure time. The departure time may be the real one or an expectation (which may be different). This is tried for 5,000 times, next a location is chosen randomly
RS-DTC	This is the same algorithm as RS-DT but instead of the real departure time, it uses a small number (5) of classes of departure times. The five departure time classes are defined using the 20/40/60/80-percentiles of the dwell times. These classes are used to model the impact of having less precise information than the real departure times
LDT	The Leveling with Departure Times algorithm modifies the basic leveling algorithm to only stack a container on top of a container with a later departure time
LDT-DTC	Leveling with Departure Time Classes uses a small number of departure time classes rather than the actual departure time (again with five classes and using the same definition as RS-DTC)
TPRL	The Transfer Point Random Level algorithm picks a level (height) at random and then looks for a location that is closest to the transfer point
TVR	The Travelling Distance versus Reshuffling algorithm balances the two components. The TVR class of algorithms feature a parameter p that modifies the relative weight of reshuffles versus the travelling distance. For positive values of p , the algorithm will prefer the ASC to travel longer; for negative values of p , the algorithm will prefer to travel shorter and accept a higher probability of future reshuffles
TVR-DTC	With this algorithm we modify the TVR algorithm to use a small number of classes rather than the actual departure time (with five classes, identical to the approach used for RS-DTC and LDT-DTC)
TVR-DTC-MD	This modification of the TVR-DTC algorithm minimizes the difference in the departure time class. Higher classes represent longer dwell times. The algorithm searches for a location that has a departure time class higher than the departure time class of the incoming container and picks the location for which the difference between the two classes is smallest

Table 2 Results for the reference stacking algorithms

Stacking algorithm	TAS (h)	ROC (%)	RDC (%)	ASC (%)	ETQ (h)	ETL (h)	90% ETQ (h)	90% ETL (h)	DAR T (%)
RS	–	64.99	97.66	53.79	0.39	0.33	0.92	0.77	34.56
RS	0.5	58.75*	88.27*	54.96*	0.38*	0.29*	0.88	0.72*	80.67
RS	1	58.26*	87.56*	55.13*	0.37*	0.29*	0.86*	0.70*	84.20
RS	4	57.18*	85.87*	55.42*	0.35*	0.27*	0.80*	0.61*	92.90
RS	12	56.91*	85.49*	55.60*	0.34*	0.25*	0.74*	0.57*	97.60
RS	24	57.31*	86.13*	55.69*	0.35*	0.26*	0.76*	0.57*	98.72
LEV	–	56.65	66.57	50.03	0.38	0.33	0.91	0.75	44.09
LEV	0.5	51.33*	60.65*	52.00*	0.48*	0.40*	1.22*	1.02*	82.78
LEV	1	50.73*	59.86*	51.69*	0.40	0.33	0.94	0.78	86.96
LEV	4	49.84*	58.75*	51.73*	0.36	0.29	0.76	0.60	94.44
LEV	12	49.83*	58.72*	51.87*	0.35	0.29	0.74	0.59*	97.77
LEV	24	50.30*	59.40*	51.87*	0.33	0.27*	0.70*	0.55*	99.10

In Table 2 we have listed the results of the two reference algorithms, random stacking and leveling. The statistically significant results for the TAS (in comparison with the results without the TAS, at the 95% level) have been marked with an asterisk.

We can see that the truck announcement has an impact on the exit times for the random stacking algorithm. Even an announcement just before the pick-up time is useful. There appears to be little value in announcements of more than 12 h in advance; with the implementation of TAS chosen for this model, the announcement information is used immediately to schedule housekeeping moves and then the container that was announced will not be eligible for incoming containers to be stacked on top of it, reducing the options available for both regular stacking and housekeeping moves. For the leveling algorithm, the results are less clear cut. For the very short announcements with a mean of 0.5 h, the exit time increases. This can happen because the ASC can still be in the process of performing the preemptive reshuffles as well as the restacking when the truck arrives to pick up the container. For earlier announcements, the results are better, until a level of 24 h is reached upon which the results deteriorate.

The percentage of TAS-based reshuffles is clearly related to the length of the TAS period and we can see that even a modest 4-h preannouncement already has significant benefits for the reference algorithms. Notice that the ROC and RDC values are lower in case of preannouncements because most of the reshuffles that would otherwise be caused by containers picked up by trucks are now counted as TAS reshuffles. Very early announcements lead to a significant reduction in the available stacking options for incoming containers as more and more positions are locked. This will eventually lead to more reshuffles.

In Table 3, we show the results of all our other stacking algorithms, tested with no TAS and a TAS announcement time with an average of 12 h, which proved to have the best performance in most cases (for brevity we have not included the full

Table 3 Result of the experiments for all stacking algorithms

Stacking algorithm	TAS	ROC (%)	RDC (%)	ASC (%)	ETQ (h)	ETL (h)	90% ETQ (h)	90% ETL (h)	DART (%)
LEV	0	56.65	66.57	50.03	0.38	0.33	0.91	0.75	44.09
LEV	12	49.83*	58.72*	51.87*	0.35	0.29	0.74	0.59*	97.77
RS	0	64.99	97.66	53.79	0.39	0.33	0.92	0.77	34.56
RS	12	56.91*	85.49*	55.60*	0.34*	0.25*	0.74*	0.57*	97.60
LDT (exp)	0	13.03	15.65	42.33	0.19	0.18	0.32	0.29	78.17
LDT (exp)	12	10.90*	12.77*	44.60*	0.19	0.18	0.32	0.28	96.94
RS-DT (exp)	0	26.73	38.62	45.94	0.23	0.21	0.45	0.38	74.20
RS-DT (exp)	12	23.30*	33.54*	47.65*	0.22*	0.18*	0.42*	0.33*	99.61
RS-DT (real)	0	23.14	34.41	45.57	0.21	0.19	0.41	0.34	78.31
RS-DT (real)	12	19.77*	29.69*	47.38*	0.21*	0.18*	0.39*	0.31*	99.91
LDT-DTC (exp)	0	27.86	34.07	45.26	0.26	0.23	0.52	0.41	74.01
LDT-DTC (exp)	12	24.75*	30.37*	48.18*	0.24	0.20	0.44	0.34*	98.38
RS-DTC (exp)	0	54.76	78.97	51.02	0.34	0.29	0.79	0.64	45.51
RS-DTC (exp)	12	48.88*	70.33*	52.85*	0.31*	0.24*	0.68*	0.52*	97.97
RS-DTC (real)	0	55.09	79.40	51.05	0.34	0.29	0.81	0.66	45.60
RS-DTC (real)	12	49.16*	70.68*	52.85*	0.31*	0.24*	0.69*	0.53*	97.94
TPRL	0	55.17	83.81	47.48	0.29	0.25	0.58	0.47	49.65
TPRL	12	49.22*	74.73*	49.30*	0.28	0.22*	0.51	0.37*	97.32
TVR ($p = 0.03$)	0	54.68	77.01	46.77	0.26	0.23	0.52	0.40	58.62
TVR ($p = 0.03$)	12	49.20*	69.73*	48.50*	0.25	0.20	0.44	0.33	98.41
TVR-DTC ($p = 0.03$)	0	19.25	26.84	42.57	0.18	0.17	0.31	0.27	85.76
TVR-DTC ($p = 0.03$)	12	17.65*	24.73*	44.31*	0.18	0.16	0.30	0.25*	99.60
TVR-DTC-MD ($p = 0.03$)	0	15.52	20.88	42.09	0.18	0.17	0.30	0.26	90.82
TVR-DTC-MD ($p = 0.03$)	12	14.79*	19.76*	43.70*	0.18	0.16	0.29	0.25	99.47

Significant differences between experiments are indicated by a * sign (at the 95% level)

list of experiments that were performed for this analysis). The experiments marked with ‘real’ and ‘exp’ indicate the use of the actual and the expected time of departure in the stacking algorithm, respectively. The expected departure time is a way to model that departure time information may be unavailable or inaccurate. For large vessels, the expected departure time is set to the middle of their planned berth time. For all other modalities, we add the average dwell time to the arrival time. This results in an average difference of 4 h between the expected and actual departure time, with a standard deviation of 5.5 h.

Within the 12 h period, a large proportion of the TAS-based housekeeping moves can be performed as indicated by the DART-% column: many more containers can be retrieved without reshuffles when the truck arrives. From the table, we can see that for most algorithms, a TAS improves performance but not significantly. We

observe that the stacking algorithms that do benefit from the TAS information are the algorithms that include a random component (RS, RS-DT, RS-DTC, TPRL). This indicates that TAS is a potentially useful feature in situations where departure time information is either not available or very unreliable.

From Table 3 we can further make the following observations. The LEV and RS are benchmark strategies and perform indeed poorly compared to the other strategies. Yet we can use the truck announcements to improve their performance significantly. The LDT algorithm uses an expected departure time and performs better than the variant with departure time classes. Although for both cases truck announcements reduce the ROC value, the impact on the ETL value is marginal. For the random stack based algorithms, RS-DT, RS-DTC we do see a significant reduction in reshuffles and the ETL decreases significantly only if its initial value is bad compared to the lowest observed value of 0.18. The same is also true for the travelling-time based stacking algorithms, TPRL and TVR. In all cases, if we observe a significant drop in ETL, we also see a significant drop in ETQ. This is interesting as it shows that shifting some of the ASC workload in time due to the TAS information also has a beneficial effect on the exit time for containers that depart on the other side of the stack.

The preannouncement information allows housekeeping moves to be shifted in time. Since the stacking crane is automated and does not require a human operator, there should not be significant additional cost when some of these housekeeping moves are done for example at night. In this way, some of the peak workload for the ASC as caused by truck arrivals during the morning and afternoon rush hours can be shifted to off-peak times. The DART% column indicates that the 12 h preannouncement period allows for most of the housekeeping moves to be performed before the exit time.

7 Conclusion

In this paper, we experimented with a Truck Announcement System to try to improve container stacking efficiency. The departure announcement for containers that will be picked up by truck provides three main benefits. First, the stacking algorithms can use this more accurate departure time information to select better stacking positions in the event that a container will depart very shortly after it has been stored in the stack. Second, the algorithm can try to avoid storing containers on top of the stacked container in order to improve the exit time of containers for which accurate departure time information is available. Third, the stacking algorithm can try to improve the position of a container once this information is available by performing housekeeping moves ahead of the actual departure.

For several strategies, we found that an average announcement time of 0.5–24 h significantly improves the efficiency. An average announcement time of 4 h is already sufficient to allow a significant proportion of the housekeeping moves (over 90% of truck pick-up containers can then be accessed directly) to be done ahead of the actual departure. This provides an opportunity to shift these unproductive moves from peak hours to off-peak hours. An announcement that arrives 12 h before the actual pick-up

would for example enable a terminal operator to perform most of these operations during the night shift prior to the day shift. As we have focused on a terminal with an automated stacking crane, this could be largely automated without a high cost in terms of labour. The benefits of TAS mainly apply to the more basic strategies. More elaborate strategies, using more information about the containers, and resulting in a low percentage of reshuffles, did not significantly benefit from the announcements as these strategies already use departure time information. Our experiments show that more accurate departure time information for trucks does not enhance the performance of these strategies. This indicates that our prior stacking strategies are quite efficient as adding more information does not improve their results.

Overall, we infer from these experiments that it would be more beneficial to improve the available information at the time of stacking (i.e., when the container is stored in the stacking yard) than to attempt to fix poor stacking decisions using the departure time information from the trucks. It is easier to make better stacking decisions with one of the algorithms that explicitly use departure time information. Even the algorithms that use approximations of this departure time information (the DTC-class of algorithms) have a better performance than the more basic stacking algorithms that are enhanced with TAS information.

Acknowledgments The authors would like to thank the referees for their useful comments.

Open Access This article is distributed under the terms of the Creative Commons Attribution Non-commercial License which permits any noncommercial use, distribution, and reproduction in any medium, provided the original author(s) and source are credited.

References

- Borgman B (2009) Improving container stacking efficiency using simulation. Master's thesis, Erasmus University Rotterdam, Erasmus School of Economics. <http://hdl.handle.net/2105/5040>. Economics & Informatics programme
- Borgman B, van Asperen E, Dekker R (2010) Online rules for container stacking. *OR Spectrum*, 32(3):687–716. doi:10.1007/s00291-010-0205-4. (Open Access)
- Caserta M, Schwarze S, Voss S (2011) Container rehandling at maritime container terminals. *Handbook of terminal planning. Operations research/computer science interfaces series 49(4)*:247–269. doi:10.1007/978-1-4419-8408-1_13
- Dekker R, Voogd P, van Asperen E (2006) Advanced methods for container stacking. *OR Spectrum* 28:563–586
- Guiliano G, O'Brien T (2007) Reducing port-related truck emissions: The terminal gate appointment system at the ports of Los Angeles and Long Beach. *Transp Res Part D* 12(7):460–473
- L'Ecuyer P, Buist E (2005) Simulation in Java with SSJ. In: Kuhl ME, Steiger NM, Armstrong FB, Joines JA (eds), *Proceedings of the 2005 Winter Simulation Conference*, pp 611–620
- Morais P, Lord E (2006) Terminal appointment system study. TP14570E
- Murty KG, Wan Y-w, Liu J, Tseng MM, Leung EL, Lai K-k, Chiu HWC (2005) Hongkong international terminals gains elastic capacity using a data-intensive decision-support system. *Interfaces* 35(1): 61–75
- Park T, Choe R, Hun Kim Y, Ryel Ryu K (2011) Dynamic adjustment of container stacking policy in an automated container terminal. *Int J Prod Econ* (in Press)
- Plata Peredo IE (2008) Fluidity at the terminal gate: truck appointment systems as remedy. Master's thesis, Maritime Economics and Logistics, Erasmus University Rotterdam, School of Economics

- Stahlbock R, Voss S (2008) Operations research at container terminals: a literature update. *OR Spectrum* 30:1–52
- Steenken D, Voss S, Stahlbock R (2004) Container terminal operation and operations research a classification and literature review. *OR Spectrum* 26:3–49
- van Asperen E (2009) Essays on Port, Container, and Bulk Chemical Logistics Optimization. PhD Thesis, Erasmus Research Institute In Management, Erasmus University Rotterdam. ISBN 978-90-5892-222-9
- van der Heide S, van Vliet H (2009) Truck appointment systems, Seminar Report Economics and ICT, Erasmus University Rotterdam, School of Economics
- Vlugt D (2009) Truck congestion at and in the terminal; a simulation study. Bachelor thesis Econometrics and Operations Research, Erasmus University Rotterdam, School of Economics

Author Biographies

Eelco van Asperen is a lecturer and deputy director at the Center for Maritime Economics and Logistics, Erasmus University Rotterdam, The Netherlands. He graduated in Business Computer Science at Erasmus University Rotterdam in 1993. He worked in IT and then as an assistant professor at the Department of Computer Science and the Econometric Institute at Erasmus University Rotterdam. In 2009 he obtained his Ph.D. in Economics at the Erasmus University Rotterdam. His research focuses on simulation with applications in logistics. His web address is <http://people.few.eur.nl/vanasperen/>.

Bram Borgman is a consultant at Quintiq. He obtained his M.Sc. degree in Economics & Informatics from the Erasmus University Rotterdam and also worked as a lecturer there.

Rommert Dekker is a full professor in operations research at the Econometric Institute of Erasmus University Rotterdam. He obtained his Ph.D. in operations research at the State University of Leiden, and his M.Sc. degree in industrial engineering from Twente University of Technology. He worked with Shell for 7 years on reliability and refinery logistics. His current research interests are: maintenance and logistics (inventory control, spare parts, ports, containers and reverse logistics). He has applied simulation models in various logistical problems. His web address is <http://people.few.eur.nl/rdekker/>.