An online storage strategy with dynamic bay reservations for container terminals

Andre Voß*, Gabriel Guckenbiehl*, Holger Schütt† and Tobias Buer*

*Computational Logistics, University of Bremen
Bibliothekstr. 1, 28359 Bremen, Germany
Email: andre.voss@uni-bremen.de, guckenbiehl@uni-bremen.de, tobias.buer@uni-bremen.de

†ISL - Institute of Shipping Economics and Logistics and ISL Applications GmbH
Barkhausenstraße 2, 27568 Bremerhaven, Germany
Email: schuett@isl.org

***Abstract*—Deciding where to store containers has a high impact on the performance of a container terminal. The problem includes incomplete planning information. We propose a two-stage strategy which decides online where to store import, export, and transshipment containers. On the first stage, a block is chosen based on the workload of the gantry cranes at the blocks as well as either a transport distance or a storage capacity measure. On the second stage, bay and slot are determined. Bays are reserved dynamically so that containers with the same destination vessel are aggregated in related bays. We follow the well-known segregation strategy, but in a successive dynamic fashion to deal with incomplete planning data. Six variants of the strategies are compared to each other by means of a study based on a commercial discrete-event terminal simulation software. Computational results suggest that the workload-based component of the storage strategy is sensitive to parameter tuning. However, the bay reservation strategy is in particular effective for terminals with many transshipment containers as it is able to reduce the number of inefficient container reshuffle moves by up to sixty percent.**

I. Introduction

Container terminals in sea ports are an important interface among world-wide and inter-modal transport chains. They connect sea-side transport via vessels and land-side transport via rail and road transportation [1]. Dispatchers of container terminals face a variety of planning problems [2], [3]. Such operational problems often are of a combinatorial nature and characterized by uncertainty. A popular way to address these difficulties has been the use of forecasting software based on simulation [4]. Based on the findings of simulation studies, terminal operations can be improved. This not only includes the actual container movements studied in this paper but also further decisions like, e.g., the number of staff and terminal equipment to employ.

One of the core problems in a container terminal is to decide where to store incoming containers. This decision includes the choice of a container block as well as the position of a container within the block. Inferior decisions may lead to additional crane movements like unnecessary reshuffling of containers which may lead to higher waiting times of yard trucks and quay cranes. Inferior storage decisions may ultimately increase the turnaround time of the handled vessels which indicates poor terminal performance. In contrast, smooth terminal operations based on smart container storage decisions can contribute to improve the overall performance of a container terminal.

Storage decisions are generally required to be made quickly. They are highly interdependent with other decisions in a container terminal. Therefore, simulation systems often cannot use sophisticated (offline) optimization models known in the literature. Planning data required for these models is often unknown: for example, because the data is unavailable at the required quality level or the available data is incomplete or uncertain; both situations may be caused by badly integrated information systems. In summary, there are many barriers in practice for modeling and solving container storage problems as well-formed combinatorial optimization problems.

We address this problem by proposing an online container storage strategy for simulation-based planning of container terminals. We evaluated the strategy by means of a computational study which simulates container handling in a terminal within a period of 24 hours (3 work-shifts) where up to 3,000 import, export, and transshipment containers are handled. Several performance indicators are used to evaluate the efficiency and effectiveness of the developed strategy in comparison to a number of other approaches.

After this introduction the literature on offline and online planning is reviewed in Section II. The online problem of assigning containers to storage locations is introduced in Section III. A heuristic container storage strategy to deal with this problem is presented in Section IV. The pros and cons of this approach are evaluated by means of a computational simulation study in Section V. Section VI concludes the paper.

II. Literature review

A. *Offline optimization*

For a general overview of storage processes in container terminals the reader is referred to [5]. In the field of storage strategies [6] was among the first to develop general mathematical expressions for measuring the expected handling effort to retrieve containers from stacks. Two basic storage strategies (balancing stacking height and segregating container groups according to arrival time) were developed.

Relationships between the number of reshuffles, available storage space and traffic demand were analyzed in [7]. The authors also proposed a segregated storage strategy that gathers containers with identical attribute values in groups. For each group a certain storage area in the yard is reserved and only containers of the same group are allowed to be stored in this reserved area. Such a segregated strategy can be used to minimize the handling effort to store and retrieve containers but may lead to an inefficient space utilization, see e.g. [8]. A less restrictive segregated storage strategy is called the consignment strategy. Here containers are grouped in regard of their destination vessels, their contents, and loading time and stored in dedicated storage areas. This strategy is used in [9] to reduce reshuffles in storage blocks. Furthermore, traffic congestion is reduced due to splitting the storage blocks into smaller sub-blocks. For each destination vessel a number of sub-blocks is preassigned for storing export containers. The number of yard cranes to be deployed in each storage block and the amount of containers to be assigned to each sub-block are predetermined for the planning horizon. This approach was extended by [10], [11]. The static reservation of sub-blocks for container groups is relaxed in a way that neighbored sub-blocks can share storage space with each other. In [12] it is stated that among the storage strategy, there are several other factors that have influence on the number of reshuffling moves, e.g., the container density within the yard, the available information, and the stacking height.

Saanen and Dekker [13] classify storage strategies in the following categories:

- Dedicated vs. non-dedicated: Only containers with similar attributes are allowed to share a stack/block/user-defined range in the yard vs. containers with different attributes can share the same storage stack.
- Consolidated vs. dispersed: Containers with the same destination vessel are clustered in the yard vs. they are dispersed.
- Housekeeping vs. immediate final grounding: The container will be moved at least one more time before loaded onto the vessel vs. the container is retrieved from its storage slot only when it is leaving the terminal.
- Discharge-optimized grounding vs. loading-optimized grounding: Efficiency maximization of storing activities vs. efficiency maximization of retrieving activities.

This categorization is, for example, used in [14] with the goal to find the optimal yard planning strategy to define a weekly yard plan. To find an optimal plan for operations with pre-known input data can be defined as an optimization approach in a static and offline environment. There are several further papers which propose optimization models and heuristics to find good storage space assignment plans for defined planning horizons. The main objective is usually to minimize the turnaround time of vessels at a container terminal, see e.g. [15]–[18]. However, in these papers no individual container moves are considered. The optimization models are solved and discussed at a medium detailed level and, therefore, are difficult to use in a terminal simulation system.

B. Online optimization

Besides offline approaches, there are online optimization methods that focus on the dynamic character of the problem. In this field of research simulation is often used. An extensive literature review on simulation modeling in ports is given by [4]. A convenient approach is the development and evaluation of discrete-event simulation programs.

Simulation can be an appropriate tool to improve the decisions in ports, evaluate performance measures and to reveal possible bottlenecks, e.g. [19]–[22]. Moreover, some research has been done to identify the effects of different storage strategies on performance measures using simulation tools. For example, [23] evaluated a hierarchical storage policy for import containers: The first level concerns about choosing the optimal storage block and in the second level the exact slot in the chosen block is determined. However, as in the segregated strategy, the stacks are preassigned which leads to inefficient usage of storage space. In [24] the importance of the choice of the storage location for export containers in a vessel-to-vessel transshipment terminal is pointed out. The goal is to maximize the gross crane rate, i.e., the average number of lifts per gantry crane per hour. A modified segregated storage strategy is used: Only containers of the same group are allowed to be stored in the same stack. If no suitable slot can be found to store a newly arrived container, a new stack has to be reserved for the container group. So the stacks are assigned in a more dynamic fashion. The storage block from which a new stack is assigned to the group is determined with the help of an amount of penalty terms, regarding the distances and the congestion at the yard. The results of the simulation experiments show that the performance using the random allocation strategy of containers is not much worse than the results using an advanced location assignment. Opposing to this [25] developed strategies in an automated container terminal on a container-individual level and tested them using simulation. It was stated that category-based stacking delivers better results regarding the amount of reshuffling than a random stacking approach. However, the effects of the storage strategies on the retrieving processes of the stored containers and thus on the overall terminal performance are not considered in these works.

Dynamic bay and stack reservation strategies are studied in [26]. When storage space is exhausted, containers are stored outside the terminal taking into account a penalty term. Storage strategies are based on groups of containers; house keeping moves are not studied. However, housekeeping in combination with storage strategies is studied in [27].

In this paper we consider the entire container flows inside a terminal and evaluate the effects of different storage strategies on the performance of a transshipment container terminal. We use the commercial discrete-event simulation software CHESSCON. Our strategy is based on a consignment approach which is adjusted for a dynamic environment.

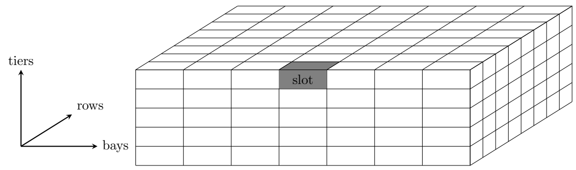

Figure 1. Organisation of a block; number of bays, rows, and tiers varies according to the used equipment.

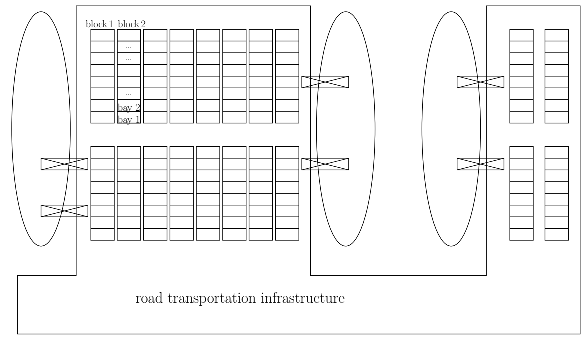

Figure 2. Outline of a terminal layout with three quays and fourteen blocks

III. Problem description

In a container terminal there are basically five operation areas: berth, quay, storage yard, road transport infrastructure, and gate (for trucks or trains). Operational decisions are manifold and often interdependent with respect to the performance of the terminal [28]. In this paper the problem is to decide where to store incoming containers in the container yard without knowing the handling sequence a priori. To deal with this unknown or unsure information we use an online optimization approach. We refer to the way the storage slot for a container is chosen as a *storage strategy*.

All storage and retrieval decisions are considered as *online decisions*. The input parameters of the problem are *unknown* or *incomplete* prior to planning. For example, the arrival time of containers might be stochastic. Stochastic arrival times usually occur when the containers arrive by truck, a vessel is discharged by multiple quay cranes or there are difficulties with the individual container, like damages or the occurrence of random inspections. The planning period is approximately 24 hours and thereby corresponds to approximately three work shifts. By planning multiple shifts we can take interdependencies between shifts into account which can contribute to improve forecasts.

With respect to the planning situation at hand, we consider three types of containers: as *transshipment containers* we denote the set C^T of containers that arrive seaside and leave seaside *during* the planning period. An *import container* has arrived seaside at the terminal before the planning period or is going to arrive seaside during the planning period. It is going to leave the terminal landside during or after the planning period. The set of import containers is denoted as C^I. An *export container* has arrived landside at the terminal before the planning period or is going to arrive landside during the planning period. It is going to leave the terminal seaside during or after the planning period. The set of export containers is denoted as C^E. The sets C^I, C^E, and C^T are pairwise disjoint. The set C of all containers is $C := C^I \cup C^E \cup C^T$. Only general purpose containers of two sizes are considered, namely containers with a length of 20-feet and 40-feet. Other container types, like open top containers, tank containers, or refrigerated containers are not considered.

We assume a container terminal that stores containers in blocks. As Figure 1 shows, blocks are organized in bays, rows, and tiers. In order to load and unload containers in a block, a gantry crane (GC) is used. During the planning period, one or two GC's are permanently assigned to a block. A reassignment of rubber tyred GC's to blocks during the planning period is not considered. The horizontal transport of containers within the terminal, e.g. from quay to block or vice versa, is done by a yard truck (YT). YT's are homogeneous with respect to operational characteristics such as speed or transport capacity. At seaside a traditional berth layout is assumed, see Figure 2, where quay cranes (QC's) operate at only one side of a vessel. All QC's operate with the same speed and in single cycle mode. Using a single cycle mode implies that a QC has to unload the import containers of a vessel first and subsequently has to load the export containers on the vessel.

In many terminals it is common to select a subset of YT's and combine them in a pool. To each QC a pool is assigned in order to simplify work routines. For example, transporting containers to or from a QC is always and only done by a truck from the assigned pool. A truck outside of the assigned pool is never used, even if it is idle. We assume such a pooling is given and unchanged during the planning period. Furthermore, a number of vessels and external trucks (ET's) arrive during the planning period in a stochastic manner; they deliver, demand, or deliver and demand containers to and from the terminal. In the transshipment case, we assume a time interval which is reasonable large.

Container handling processes which are relevant for the problem at hand are as follows. When a container *arrives seaside*, it is unloaded from the vessel by a QC. The QC lifts the container from the vessel and puts it on a YT. Timing is important, if the YT is not already waiting for the container, the QC cannot drop the container and has to wait until a YT arrives. This slows down the processing time of the vessel. Therefore a waiting QC is considered worse than a waiting YT. After the unloading process is finished, the YT transports the container to a block where it is stored. After arriving at the block, the YT waits until the respective GC is ready to handle the container. When a container *arrives landside* at the gate on an ET, the ET is assigned a block to drop off the container. At the block, the ET has to wait until the GC is ready to handle the container. We assume that a container is always stored in the yard. That is, we do not consider direct transportation from the quay to the gate or vice versa. Also, direct transportation from quay to quay in the transshipment case is forbidden.

For containers that leave the terminal seaside or landside, the handling is essentially reversed. Corresponding to the unloading at a QC, a YT has to stand by so that a GC can remove a container from a block. As containers are stacked and only one container can be handled at a time, it may be necessary to reshuffle containers within a bay or a block so that the requested container can be removed. A reshuffling move is intrinsically inefficient. They increase the workload of the GC's and the waiting time of the YT's. However, avoiding reshuffling moves at all cost may not be suitable for all situations. If, for example, reshuffling is performed during times of low crane utilization and with foresight to prepare the loading of a soon arriving vessel, it may increase the productivity of a terminal. This approach is known as housekeeping or container pre-marshalling [27], [29]. Housekeeping is not the focus of this study and, therefore, has been implemented in straight way, see V-B.

The goal of a storage strategy is to smoothen and to speed up the operations at a terminal. Criteria to measure the performance of a terminal are: vessel turnaround time, the number of reshuffles, and the simulated time. The informative value of the latter criterion may be limited because vessels berth and are processed continuously and a cut in time may be arbitrary. To handle this in our tests, we assume a planning horizon of approximately 24 hours and schedule the last arrival of one or more vessels at 12 hours after the start of planning; because a vessel requires an empty berth before the handling operations can start, the scheduled arrival time may differ from the actual start of the handling operations. We measure the simulation time until the handling of the last arrived vessel is finished. In the next section the proposed storage strategy is described.

IV. Two-stage container storage strategy

A. Online optimization and application area

Loading and unloading of containers from vessels, storing containers and transporting containers within a terminal (horizontally or vertically) are interdependent problems equipped with high uncertainty. Uncertainty affects arrival and departure times of containers at the terminal, e.g. due to delays of vessels or trucks. It also affects transportation time within a terminal and other events which contradict planning, e.g. disturbances or inadequate precision of planning data. Therefore, we propose a container storage strategy based on online planning, i.e., we assume that planning data is incomplete prior to the start of planning.

The storage strategy should support the decision makers in improving the overall performance of the terminal. It is a non-trivial task to measure terminal performance by a single criterion. Good performance is characterized by short turnaround times of vessels, a small number of crane movements (i.e., presumed ineffective movements are avoided), a high lift rate of the quay cranes, little congestion in the yard, or low transport distances within a terminal. The proposed strategy has these criteria in mind without explicitly stating one as the most important. Moreover, it is possible that the strategy will *not be directly* used by a human decision maker but rather *indirectly* in a terminal simulation system. In that case, the goal might be even more nebulous. A terminal simulation system forecasts a series of diverse decisions. The human decision makers using such a simulation system might be more interested in the staff and equipment to employ in future shifts and less in actual slots where containers are stored. Nevertheless, these deduced decisions – which are out of the scope of this paper – can and should be improved by more efficient container storage strategies.

We introduce a two-stage storage strategy. The strategy is used to decide in which slot a given container $c \in C$ should be stored. On the first stage *(block assignment)*, a suitable block b^* is chosen from the set B of all blocks. On the second stage *(bay assignment)*, a bay in b^* is chosen according to a dynamic reservation mechanism.

B. Workload-based block assignment on the first stage

Decisions are mainly based on two types of first-in-first-out queues. The *container queue* Q^C manages all jobs that require horizontal transportation of containers via trucks. It is implemented as a sequence of containers. In particular, Q^C contains containers that arrive seaside or landside at the terminal and have to be stored in the yard. Containers that are already stored in the yard and have to be delivered to quay or to an external truck are also managed in Q^C. Furthermore, there is a *truck queue* Q_b^T for each block $b \in B$. A truck queue Q_b^T manages trucks that are scheduled to be processed via a GC associated with block $b, b \in B$.

The idea of the *block assignment strategy* is to balance the number of trucks waiting at the blocks to be processed. Another approach in an offline-environment is presented in [30]. Our thoughts and observations are as follows. Usually, QC's work faster than GC's. Both the number of QC's per vessel and the number of GC's per block are limited due to the working principles of the cranes and consequential physical restrictions. A well chosen balance between the number of QC's and GC's is very helpful for smooth terminal operations, but this action alone is not sufficient. If too many containers flow from a QC to a single block, congestion may be caused at this block. That is why it is necessary to use multiple blocks for a single QC. Basically, we measure the workload of cranes at a block $b \in B$ simply as the length (i.e., number of elements) of the truck queue Q_b^T. A truck queue contains trucks that are assigned to pickup or deliver a container at a given block. This includes trucks that (a) are currently processed at the block, (b) actually wait at the block to be processed, and (c) are currently driving within the terminal to the block.

The block assignment procedure is presented in Figure 3. Let c be the next container in Q^C. Now, the block b where to store c is determined. Let B^q be a set of blocks with a truck queue length smaller-or-equal than $q \in \mathbb{N}$. The parameter q for the queue length is static and a priori given.

$$B^q := \{b | b \in B \text{ and } |Q_b^T| \leq q\}$$

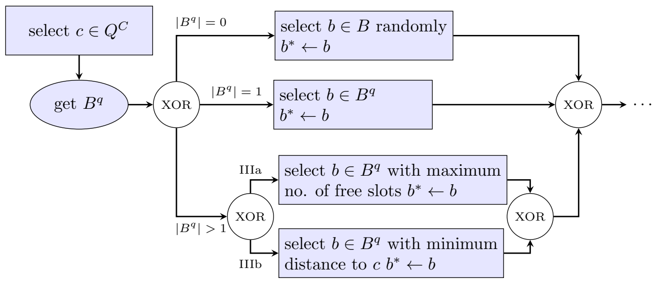

Figure 3. Procedure of block assignment strategy (first stage)

Depending on the cardinality of B^q, i.e., number of blocks with a waiting queue less than q, there are three cases:

$$|B^q| = 0 \quad \text{(I)},$$
$$|B^q| = 1 \quad \text{(II)},$$
$$|B^q| > 1 \quad \text{(III)}.$$

In case (I), no block meets the waiting queue length criterion and that is why simply a random block b^* from the set of all blocks B is chosen. In case (II), only a single block meets the waiting queue criterion and is, therefore, chosen as b^*. In case (III), there are multiple blocks that meet the waiting queue length criterion. To break ties, either the free capacity of a block (i.e., number of free slots) or the distance of the block to the current location of container c is used. In case (IIIa) we choose the block b^* with the largest number of free slots. In the other case (IIIb) we choose the block b^* with minimum distance to the location of c whereby distance is measured as the shortest path connection on the road network. After the block b^* is chosen, the first stage of the strategy is complete.

C. Dynamic bay reservation mechanism on the second stage

On the second stage, the *bay assignment* procedure chooses a bay in the selected block b^* where the container is stored. We use a dynamic bay reservation mechanism. Let bay i of a block $b \in B$ be denoted as b_i with $1 \leq i \leq n(b)$, where $n(b)$ is the number of bays in block b. A random bay b_i^* for storing container c is chosen if (a) in block b^* either no bay is reserved, yet, or (b) the reserved bay has no storage capacity left. Then b_i^* is reserved. This means that all future containers stored in block b^* which are loaded on the same vessel as c are also stored in bay b_i^* until it is full. Containers that are shipped via a different vessel than c are not allowed to be stored in bay b_i^*. The only exception is, when the storage space is depleted on the logical level but not on the physical level (because empty slots of a bay are reserved for different types of containers) then the reservation maybe bypassed. In case c leaves the terminal landside, the interpretation is analogously: only containers that leave landside as well are allowed to be stored in the same bay. The slot in the bay is selected in a random fashion. However, the aim is to place containers next to each other for reasons of stability such that gaps or isolated towers are avoided. The reservation of bay b_i^* for containers with the same destination vessel is canceled, after the last container with this destination from b_i^* has been removed.

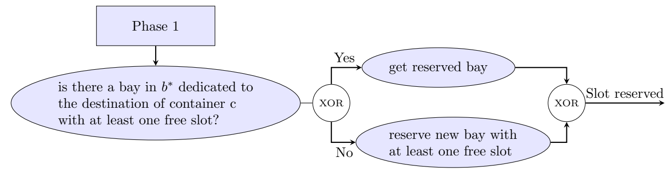

Figure 4. Procedure of bay reservation and selection (second stage)

Reservation of a bay works independent of those which are stored prior to the start of planning. This may appear counterintuitive but it should enable a higher degree of flexibility and smooth the transition between planning periods. Storage space expands and shrinks dynamically. On the one hand, this should balance the usage of the available storage space. On the other hand and in contrast to the strategy of reserving all bays of a block for containers which are loaded on the same vessel, this should also distribute the workload of the yard cranes for container retrieval and the associated truck queues. Because it is possible to reserve a bay that already contains containers stored prior to the planning period, the goal to extend the planning horizon to multiple shifts should also be supported.

V. Results of a simulation study

A. Terminal layout and test instances

For our simulation study we use a single basic terminal layout for testing. The terminal layout is inspired (but simplified) by the Piraeus Container Terminal in Greece. Figure 2 shows a sketch which is, however, not true to scale. There are three quays where vessels berth. Currently, there work only two QC's at each berth at the same time; the number of QC's per berth has to be increased in order to obtain more realistic results. Nevertheless, all strategies are tested under the same test setting. Each QC is assigned a pool of five to eight YT's which deal exclusively with containers handled by this QC. Reallocation to other QC's during the simulated time is not considered. The terminal is connected to the hinterland by a gate where external trucks enter and leave the terminal. Transport by train is not included in the layout. The storage yard is separated into 20 blocks, eight deployed by rail mounted gantry cranes (RMG's) and twelve handled by rubber-tyred gantry cranes (RTG's). The potential capacity of a block is between 1,782 and 3,240 TEU and is served by one RMG or up to two RTG's. The total capacity of the terminal used in the numerical experiments is 50,114 TEU. Due to properties of the yard layout the number of storable 40-feet containers is 24,526, slightly smaller than half of 50,144. RTG-Blocks can be stacked with containers in eight rows and six tiers. RMG-Blocks have seven rows and five tiers. In both block types, an additional row is reserved for restacking processes.

Based on this terminal layout seven test instances I_A to I_G have been generated. They differ in the following characteristics, see Table I: The number of vessels #VB which berth in the terminal at the beginning of the planning horizon $(t = 0)$ and are ready for processing. The number of vessels

Table I
PROPERTIES OF THE USED TEST INSTANCES

Property	Instance						
	I_A	I_B	I_C	I_D	I_E	I_F	I_G
#VB	2	2	2	3	3	0	0
#VL	3	3	3	3	6	0	0
#VO	2	2	2	3	0	7	7
#CY (in 1,000)	10	20	10	10	10	10	10
#CH (in 1,000)	1.5	1.5	1.5	.5	2.0	1.5	3.0
#CL (in 1,000)	0.5	0.5	0.5	0.0	0.0	0.5	0.5
#CT (in 1,000)	0.5	0.5	1.0	0.5	2.0	0.0	0.0

#VL that arrive at the latest possible time is indicated by #VL. For all instances, the latest arrival time of a vessel in the planning horizon is set to $t = 720$ minutes. For instance I_E the value of #VL is 6, however, in the given layout there are only three berthing places available. The vessels that can't be berthed directly, have to wait for the berth to be available again and then are processed sequentially. The exact vessel arrival sequence as well as the total simulation time is thereby hard to forecast. The number of other vessels that arrive between $0 < t < 720$ is given by #VO. The total number of vessels processed is #VB + #VO + #VL. The number of containers stored in the yard at $t = 0$ is given by #CY. The number of containers which have to be handled during the planning period is #CH. This number includes all import, export, and transshipment containers that arrive or leave the terminal during the planning period. The number of containers that arrive or leave the terminal landside by an external truck is given by #CL. The number of transshipment containers is indicated by #CT. Note, here transshipment containers refers only to those containers that arrive and leave the terminal via ship *during* the planning horizon. For all instances, the share of 40-feet (20-feet, respectively) containers among all containers (#CY + #CH + #CL) is 60 percent (40 percent, respectively).

B. Tested storage strategies

All in all, six two-level container storage strategies are tested. A two-stage strategy is denoted as S^1_2 where the superscript 1 denotes the mechanism on the first stage, and the subscript 2 gives the mechanism on the second stage. On the *first stage*, a container is assigned to a block when it arrives at the terminal. There are three options: choose a pseudo random block S^r, choose the block with minimum transport distance to the current position of the container S^d, or choose the block with the minimum load ratio, aiming at balancing the used storage capacities of all blocks S^b. Strategy S^r is denoted *pseudo random* because the blocks are chosen randomly but no block is chosen for the $n+1th$ time until every other block has been chosen at least n times. For the strategies S^d and S^b a virtual waiting queue with length q limits the number of trucks waiting at the blocks. A block can only be selected to store a container if the number of YT's waiting at the block is smaller than q at the moment of the storage assignment decision. Regarding import and transshipment containers, the storage decision on the first level is limited further to the number of blocks that are assigned to the quay at which the vessel is berthed. This limits the maximal transport distances to a reasonable amount. On the *second level*, a container is assigned to a slot in the block. For this, there are two options: random slot selection S_r and dynamic slot selection S_d, see Section IV. Hence three strategies with a dynamic slot selection (S^b_d, S^d_d, S^r_d) and three strategies with a random slot selection are compared (S^d_r, S^b_r, S^r_r).

A reshuffling of containers is necessary when the required container is not stored at the top of the specific stack. If the block contains a compatible reserved bay with free slots then the container to be reshuffled is moved there. Otherwise crane movements are minimized by choosing the nearest free slot as reshuffle destination.

Table II
RESULTS ON INSTANCE I_A FOR DIFFERENT TRUCK QUEUE LENGTHS q

Strategy	q	Sim. time (hh:mm)	Reshuffles (%)
S^d	1	20:42	20.6
	2	21:49	22.6
	3	22:03	23.2
	4	23:22	23.4
	5	23:34	23.2
S^b	1	18:32	20.0
	2	20:21	20.6
	3	20:18	20.1
	4	21:00	19.4
	5	21:37	20.9

C. Determining a threshold q for the waiting queue length

The storage strategy uses the workload at a block as a main decision criterion for distributing containers throughout the terminal. The workload at block $b \in B$ is measured by the length of its truck waiting queue Q^T_b. We first test which value for the threshold parameter q provides good results. We use the instance I_A for testing because it is a kind of average instance in our set of instances. The strategies S^b and S^d are tested for $q = 1, ..., 5$. The results are shown in Table II. The table shows the simulated time after the last container has been handled in column three. Column four shows the ratio of unproductive reshuffle moves. Regarding both criteria, strategy S^d performs best for $q = 1$. Strategy S^b performs best for $q = 1$ with respect to the simulated time criterion and ranks second for the number of reshuffles. Therefore, we set the threshold parameter $q = 1$ for the overall strategy comparison.

D. Comparison of the strategies

The six storage strategies are compared by means of the seven test instances. Each instance is solved five times with each of the strategies. The averaged values over three performance criteria are shown in Table III. The length of the simulated time is given in the format hh:mm. Furthermore, the ratio of the number of reshuffle moves of all container moves in percent (RES (%)) as well as the required computing time (CPU (s)) in seconds are given.

Table III
RESULTS INSTANCES

	Criteria	S_r^d	S_d^d	S_r^b	S_d^b	S_r^r	S_d^r
I_A	Sim. time	20:42	18:51	18:32	17:48	18:03	16:57
	RES (%)	20.6	9.9	20.0	9.1	19.4	7.4
	CPU (s)	56	79	69	81	51	79
I_B	Sim. time	18:05	17:42	17:57	17:07	17:14	16:52
	RES (%)	13.3	7.4	14.5	7.2	12.3	6.4
	CPU (s)	53	67	51	63	49	62
I_C	Sim. time	28:37	25:52	24:28	22:28	22:22	21:09
	RES (%)	27.6	16.1	26.2	15.7	25.6	14.3
	CPU (s)	79	102	68	101	66	100
I_D	Sim. time	14:53	14:48	14:30	14:13	14:17	14:11
	RES (%)	25.1	13.3	22.4	8.7	22.5	8.9
	CPU (s)	30	53	29	45	27	43
I_E	Sim. time	35:43	31:05	32:51	29:31	29:16	27:24
	RES (%)	29.5	16.4	28.9	14.7	27.4	11.3
	CPU (s)	108	157	113	156	103	156
I_F	Sim. time	12:53	12:54	12:50	12:50	12:55	12:54
	RES (%)	4.7	4.8	3.5	3.4	3.9	3.8
	CPU (s)	39	60	38	62	37	59
I_G	Sim. time	20:15	20:14	20:15	20:18	20:25	20:36
	RES (%)	9.8	10.0	9.0	9.1	9.8	9.6
	CPU (s)	77	116	72	119	72	112

All tested strategies require a computing time of less than 160 seconds on average. This is the total computing time, that includes all the expensive calculations of CHESSCON to simulate the terminal as well as the calculation of the storage strategies and the communication of both programs via an interface. The required time is low and is reasonable for practical applications involving a human decision maker. Another general observation with respect to CPU time is that for most cases the first-level random block assignment strategy S^r is the fastest, thereafter the balanced block strategy S^b followed closely by the distance-based strategy S^d. Higher impact on the CPU time appears to make the second-level strategy. The dynamic bay reservation mechanism (S_d) requires in all cases more computing time than the randomized slot choice mechanism (S_r). This is obvious due to more costly computations. Nevertheless, in cases CPU time is a bottleneck, tuning of the strategies should be performed on the second-stage.

A comparison between instance I_A and I_B reveals that the strategies perform better when the container terminal is more occupied, i.e., the yard is more filled. This phenomenon can be explained by two opposed effects: First the retrieving activity is more difficult for containers that are stored in lower tiers. However, in the instance generation the containers that are to be handled during the simulation were placed in the storages after the initial containers in the yard (#CY) were created. Hence, no initial container is stacked on an import or export container that is moved during the planning period and thus, the negative influence of a fuller yard on the retrieving difficulty is nullified in this case. On the other hand a filled yard has a positive effect on containers that are stored and retrieved during the planning horizon because it reduces the number of available slots. In a fuller yard the stacking of *new* containers on top of each other is less probable. Therefore, storing and retrieving containers within the planning horizon becomes easier. This may also explain the lower CPU time for a fuller yard. Decisions are calculated faster because the number of feasible decisions is smaller. However, we expect that the positive effect of a fuller yard will not remain if the simulated time is increased. One reason is that the complete set of stored containers will have to be exchanged in a longer time horizon and the above mentioned positive effects only take place if a large number of containers is not moved. This result indicates that on the one hand the way how the test is performed and the instances are created should be reviewed again. On the other hand, it confirms the idea of studying a longer planning period of multiple work shifts in order to make those effects visible and deal with them in the future.

The results of test instances I_C, I_D, and I_E show that the second-level dynamic bay reservation mechanism always outperforms the corresponding random slot assignment, both in simulation time and the percentage of reshuffles. The average reduction in the share of reshuffles is about 50 % in some cases even 60 %. This is very promising. Moreover, storing and retrieval activities can be executed with less delay caused by occupied equipment. Consequently, the simulation time is reduced significantly. The scheduled container moves are processed faster and the average turnaround time of the vessels increases.

In most of the instances, i.e., I_A to I_E, the productivity of the container terminal is increased by means of the dynamic bay reservation procedure compared to a random slot allocation. All in all, the strategy S_d^r (random block and dynamic slot assignment) can be identified as the best performing strategy on the given test instances. However, these instances have in common that they all contain transshipment activities. The performance indicators of test instances I_F and I_G show the shortcoming of the dynamic strategy. In the case with a lower workload at instance I_F there is not much variation in the results regarding the different storage strategies. Only the minimization of transport distance causes a little more reshuffling amount than the other approaches. Besides from the calculation time (which is higher for the dynamic slot assignment) there is no significant difference between a random slot assignment and the dynamic variant. For a high workload in a no transshipment environment, like instance I_G, the results indicate that the dynamic bay reservation mechanism even performs worse than a random approach. It is important to notice that the structure of the dynamic slot assignment is built to sort containers into the same bay according to their destination vessel. As shown in the experiments this has a positive effect on the handling efficiency of transshipment containers. On the other hand, import containers that leave by truck are also sorted into the same bay which appears to be suboptimal and probably leaves room for future improvement. To sum up, at this point, the proposed dynamic bay reservation mechanism can be recommended for terminals with a high amount of transshipment activities.

VI. Conclusion

A two-stage storage strategy for container terminals has been proposed. It is an online optimization strategy that works with incomplete information. On the first stage the strategy chooses a block were a given container should be stored. This decision is based on the length of truck queues waiting at the blocks as well as either a distance measure or a balancing of block capacity. On the second stage a dynamic capacity reservation mechanism based on bays is introduced. Six strategy variants have been tested by means of a simulation study enabled by the commercial terminal simulation software CHESSCON. Based on the criteria simulated time and unnecessary reshuffles, the dynamic bay reservation mechanism used on the second stage provides very promising results as it is able to reduce the number of reshuffles by up to 60 %. However, the simulation study also opened up opportunities for future research. It could be meaningful to choose and vary the truck queue length parameter q self-adaptively. A storage strategy should be more fine-grained when dealing with containers that leave via external trucks. Finally, to get closer to reality, longer planning horizons of multiple work shifts should include further decisions like the available handling equipment to be variably usable during the planning horizon.

Acknowledgment

The junior research group on Computational Logistics is funded by the University of Bremen in line with the Excellence Initiative of German federal and state governments. The authors were supported by the Federal Ministry for Economic Affairs and Energy due to a resolution of the German Bundestag (project "simTOS", grant KF3352901). The authors would like to thank Andreas Dressler and Horst-Dieter Kassl for their support in developing a data interface to the terminal simulation software CHESSCON provided by ISL Applications GmbH.

References

[1] N. K. Tran, H.-D. Haasis, and T. Buer, "Container shipping route design incorporating the costs of shipping, inland/feed transport and CO_2 emission," *Maritime Economics & Logistics*, 2016, in press.

[2] R. Stahlbock and S. Voß, "Operations research at container terminals: a literature update," *OR Spectrum*, vol. 30, no. 1, pp. 1–52, 2008.

[3] L. Heilig and S. Voß, "Inter-terminal transportation: an annotated bibliography and research agenda," *Flexible Services and Manufacturing Journal*, 2016.

[4] B. Dragović, E. Tzannatos, and N. K. Park, "Simulation modelling in ports and container terminals: literature overview and analysis by research field, application area and tool," *Flexible Services and Manufacturing Journal*, pp. 1–31, 2016.

[5] H. J. Carlo, I. F. Vis, and K. J. Roodbergen, "Storage yard operations in container terminals: Literature overview, trends, and research directions," *European Journal of Operational Research*, vol. 235, no. 2, pp. 412–430, 2014.

[6] B. de Castillo and C. F. Daganzo, "Handling strategies for import containers at marine terminals," *Transportation Research Part B: Methodological*, vol. 27, no. 2, pp. 151–166, 1993.

[7] M. Taleb-Ibrahimi, B. de Castilho, and C. F. Daganzo, "Storage space vs handling work in container terminals," *Transportation Research Part B: Methodological*, vol. 27, no. 1, pp. 13–32, 1993.

[8] K. H. Kim and H. B. Kim, "Segregating space allocation models for container inventories in port container terminals," *International Journal of Production Economics*, vol. 59, no. 1, pp. 415–423, 1999.

[9] L. H. Lee, E. P. Chew, K. C. Tan, and Y. Han, "An optimization model for storage yard management in transshipment hubs," *OR Spectrum*, vol. 28, no. 4, pp. 539–561, 2006.

[10] Y. Han, L. H. Lee, E. P. Chew, and K. C. Tan, "A yard storage strategy for minimizing traffic congestion in a marine container transshipment hub," *OR Spectrum*, vol. 30, no. 4, pp. 697–720, 2008.

[11] X. Jiang, L. H. Lee, E. P. Chew, Y. Han, and K. C. Tan, "A container yard storage strategy for improving land utilization and operation efficiency in a transshipment hub port," *European Journal of Operational Research*, vol. 221, no. 1, pp. 64–73, 2012.

[12] T. Chen, "Yard operations in the container terminal-a study in the 'unproductive moves'," *Maritime Policy & Management*, vol. 26, no. 1, pp. 27–38, 1999.

[13] Y. Saanen and R. Dekker, "Intelligent stacking as way out of congested yards? Part 1," *Port Technology International*, vol. 31, pp. 87–92, 2007.

[14] L. P. Ku, L. H. Lee, E. P. Chew, and K. C. Tan, "An optimisation framework for yard planning in a container terminal: case with automated rail-mounted gantry cranes," *OR Spectrum*, vol. 32, no. 3, pp. 519–541, 2010.

[15] E. Nishimura, A. Imai, G. K. Janssens, and S. Papadimitriou, "Container storage and transshipment marine terminals," *Transportation Research Part E: Logistics and Transportation Review*, vol. 45, no. 5, pp. 771–786, 2009.

[16] K. H. Kim and K. T. Park, "A note on a dynamic space-allocation method for outbound containers," *European Journal of Operational Research*, vol. 148, no. 1, pp. 92–101, 2003.

[17] P. Preston and E. Kozan, "An approach to determine storage locations of containers at seaport terminals," *Computers & Operations Research*, vol. 28, no. 10, pp. 983–995, 2001.

[18] E. Kozan and P. Preston, "Mathematical modelling of container transfers and storage locations at seaport terminals," *OR Spectrum*, vol. 28, no. 4, pp. 519–537, 2006.

[19] R. Stahlbock and S. Voß, "Efficiency considerations for sequencing and scheduling of double-rail-mounted gantry cranes at maritime container terminals," *International Journal of Shipping and Transport Logistics*, vol. 2, no. 1, p. 95, 2010.

[20] J. Klaws, R. Stahlbock, and S. Voß, "Container terminal yard operations – simulation of a side-loaded container block served by triple rail mounted gantry cranes," in *Lecture Notes in Computer Science*. Springer Science+Business Media, 2011, pp. 243–255.

[21] F. Longo, A. Huerta, and L. Nicoletti, "Performance analysis of a southern mediterranean seaport via discrete-event simulation," *Strojniški vestnik-Journal of Mechanical Engineering*, vol. 59, no. 9, pp. 517–525, 2013.

[22] M. Kondratyev, "An object-oriented approach to port activity simulation," *International Journal of Simulation and Process Modelling*, vol. 10, no. 1, pp. 1–9, 2015.

[23] E. Guldogan, "Simulation-based analysis for hierarchical storage assignment policies in a container terminal," *Simulation*, vol. 87, no. 6, pp. 523–537, 2011.

[24] M. E. Petering, "Real-time container storage location assignment at an RTG-based seaport container transshipment terminal: problem description, control system, simulation model, and penalty scheme experimentation," *Flexible Services and Manufacturing Journal*, vol. 27, no. 2-3, pp. 351–381, 2015.

[25] R. Dekker, P. Voogd, and E. van Asperen, "Advanced methods for container stacking," *OR Spectrum*, vol. 28, no. 4, pp. 563–586, 2006.

[26] Y. Ma and K. H. Kim, "A comparative analysis: various storage rules in container yards and their performances," *Industrial Engineering and Management Systems*, vol. 11, no. 3, pp. 276–287, 2012.

[27] J.-F. Cordeau, P. Legato, R. M. Mazza, and R. Trunfio, "Simulation-based optimization for housekeeping in a container transshipment terminal," *Computers & Operations Research*, vol. 53, pp. 81 – 95, 2015.

[28] K. G. Murty, J. Liu, Y.-w. Wan, and R. Linn, "A decision support system for operations in a container terminal," *Decision Support Systems*, vol. 39, no. 3, pp. 309–332, 2005.

[29] P. Legato, R. M. Mazza, and R. Trunfio, "Medcenter container terminal SpA uses simulation in housekeeping operations," *Interfaces*, vol. 43, no. 4, pp. 313–324, 2013.

[30] M. Barketau, E. Pesch, and Y. Shafransky, "Minimizing maximum weight of subsets of a maximum matching in a bipartite graph," *Discrete Applied Mathematics*, vol. 196, pp. 4 – 19, 2015.