



PERGAMON

Transportation Research Part B 37 (2003) 883–903

TRANSPORTATION
RESEARCH
PART B

www.elsevier.com/locate/trb

Storage space allocation in container terminals

Chuqian Zhang ^a, Jiying Liu ^a, Yat-wah Wan ^{a,*}, Katta G. Murty ^b,
Richard J. Linn ^c

^a *Department of Industrial Engineering and Engineering Management, The Hong Kong University of Science and Technology, Kowloon, Hong Kong*

^b *Department of Industrial and Operations Engineering, University of Michigan, Ann Arbor, Michigan 48109-2117, USA*

^c *Department of Industrial and Systems Engineering, Florida International University, Miami, FL 33199, USA*

Received 12 October 2001; received in revised form 27 November 2002; accepted 2 December 2002

Abstract

Container terminals are essential intermodal interfaces in the global transportation network. Efficient container handling at terminals is important in reducing transportation costs and keeping shipping schedules. In this paper, we study the storage space allocation problem in the storage yards of terminals. This problem is related to all the resources in terminal operations, including quay cranes, yard cranes, storage space, and internal trucks. We solve the problem using a rolling-horizon approach. For each planning horizon, the problem is decomposed into two levels and each level is formulated as a mathematical programming model. At the first level, the total number of containers to be placed in each storage block in each time period of the planning horizon is set to balance two types of workloads among blocks. The second level determines the number of containers associated with each vessel that constitutes the total number of containers in each block in each period, in order to minimize the total distance to transport the containers between their storage blocks and the vessel berthing locations. Numerical runs show that with short computation time the method significantly reduces the workload imbalance in the yard, avoiding possible bottlenecks in terminal operations.

© 2003 Elsevier Ltd. All rights reserved.

Keywords: Container terminal; Storage space allocation; Hierarchical approach; Mathematical programming

* Corresponding author. Tel.: +852-2358-7100; fax: +852-2358-0062.

E-mail address: ieywan@ust.hk (Y.-w. Wan).

1. Introduction

A container terminal is an intermodal interface that usually connects container vessels on sea with trucks on land. Other than providing loading and unloading services for the container carriers, a container terminal also serves as a temporary storage space for containers that are between two journeys on carriers. For container terminals in hub ports, their container throughputs can be higher than the total throughput of a reasonably busy trade port. For example, in 2000, two terminal operators in Hong Kong, Hongkong International Terminals Limited (HIT) and COSCO-HIT, handled 6.6 million Twenty-foot Equivalent Units (TEUs).

The achievements of these Hong Kong terminal operators are clearer when the productivity figures are put in the correct perspective. An annual throughput of 6.6 million TEUs means that on average there are no less than 20,000 trucks passing through the gates of the terminals, whose total area is only of 122 hectares. The busiest terminal in Europe, the Delta Terminal in the Netherlands, handled 2.5 millions TEUs (in 2001) with an area of 280 hectares, and one of the busiest container ports in the United States, the port of Long Beach, handled 4.6 million TEUs with an area of 295 hectares. The Hong Kong container terminals have a significantly higher throughput per area when compared to their counterparts in Europe and America. To achieve a high productivity in a relatively small space, the Hong Kong terminals employ larger number of yard cranes than their counterparts. There are 167 yard cranes (rubber tyred gantry cranes (RTGCs) and rail mounted gantry cranes) in the two Hong Kong terminal operators, while there are around 50 of those in each of the Delta Terminal and the port of Long Beach. (The comparison may not be comprehensive, because there is an automatic guided vehicle system in Delta, and there are chassis and train systems in Long Beach. Nevertheless, the statistics generally reflect the characteristics of terminals in different regions.) These statistics show that terminals in Hong Kong work in a dense, intensive environment such that the efficiency in operations is vital.

Fig. 1 is a schematic diagram showing the typical core operations of a container terminal in Hong Kong. Readers interested in realistic glimpses of daily operations of container terminals are

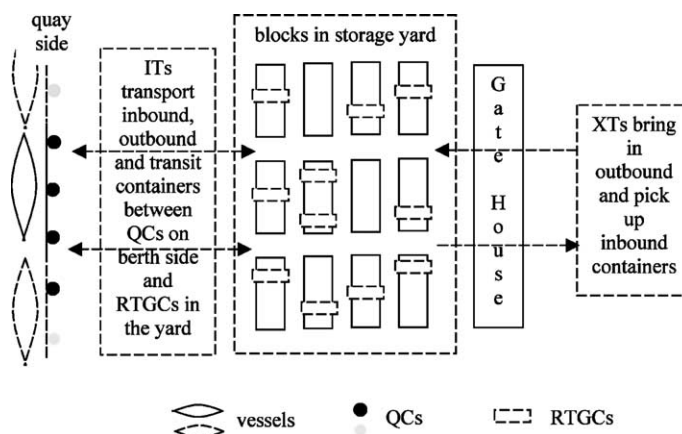


Fig. 1. A schematic diagram of a container terminal.

invited to visit the web sites of major terminal operators in Hong Kong, such as Hongkong International Terminals Limited (http://www.hph.com.hk/business/ports/hong_kong/hit.htm) and Modern Terminals Limited (<http://www.mtl.com.hk/main.htm>).

Based on the type of container handling operations, a container terminal can be roughly divided into two main areas, the quayside and the storage yard. The quayside is where vessels are berthed. Quay cranes (QCs) discharge inbound (I/B) and transit containers from and load outbound (O/B) and transit containers to vessels. The storage yard is typically made up of blocks of containers. Each block consists of a mass of containers, usually placed in six lanes side by side, with each lane including 20 or more container stacks that are of four to five tiers of containers. The number of lanes and the height of the container stacks depend on the height and the span of the cranes that are used to stack containers in a block. The six-lane, four-tier stack size is the typical size for RTGCs, whose gantries span across the lanes of a block. Internal trucks (ITs) provide transportation of containers between the QCs and the storage blocks. External trucks (XTs) bring O/B containers from customers into the yard and pick up I/B containers from the yard and deliver them to customers. RTGCs, or yard cranes in general, are used to handle the containers in storage blocks. They load containers from trucks (ITs or XTs) and stack them onto blocks, and retrieve containers from blocks and load them onto trucks.

The container flows in a terminal are triggered by the vessel arrival process. Each vessel arrives at a designated time with specified numbers of containers to be discharged and loaded at the terminal. The berth allocation, where the vessel is to be berthed, and the stowage plan, the sequences for discharging and loading the containers, of the vessel are determined well before the vessel's arrival. Containers to be handled in the yard can be classified into the following four types according to their status at different handling stages.

- (a) Vessel discharge (VSDS) containers: I/B and transit containers on vessels before they are unloaded and allocated to the yard.
- (b) Container yard pickup (CYPI) containers: I/B containers already in the yard waiting for picking up by customers.
- (c) Container yard grounding (CYGD) containers: O/B containers before they are brought in and stored in the yard.
- (d) Vessel loading (VSLD) containers: O/B and transit containers already in the yard waiting for loading to vessels.

These types of containers are measured in terms of the number of containers. This unit is used throughout when we discuss the storage space allocation problem of containers.

Since the arrivals of VSDS and the departures of VSLD containers are directly triggered by vessel schedules, the time epochs to handle these containers (by RTGCs or QCs) are known in advance. On the other hand, we can only determine from historical data the distributions of the time epochs to handle CYGD and CYPI containers. There is a free storage period for these containers, usually lasting for several days before the arrivals and after the departures of the corresponding vessels. This makes the time epochs to work on CYGD and CYPI containers imprecise.

There are various interrelated performance indicators of a container terminal, measuring the productivity and the utilization of every type of resource, and various aspects of customer

satisfaction. Some performance indicators are taken as the objectives, and the two commonest objectives of Hong Kong terminals are:

- (a) to minimize the (average) vessel berthing time, which is a measure of the service of a terminal to ship liners,
- (b) to maximize the (average) throughput of QCs, which is a measure of the productivity of a terminal.

These two objectives link directly with the income of Hong Kong container terminals, which charge ship liners for *each* move of a QC. Other performance indicators, such as the turnaround times of XTs, are noted by container terminals. The values of these indicators reflect whether a container terminal is in a healthy operating condition. In general, within a reasonable range, indicators are compatible with the two commonest objectives. For example, given the small yard area and the high throughput of Hong Kong container terminals, a short berthing time or high QC throughput generally come along with short XT turnaround times.

There are many different decisions involved in operating container terminals and all these decisions affect each other. For example, decisions about the storage of containers in the yard directly affect the workloads of the yard cranes in the blocks and the traveling distances of the ITs and indirectly affect the efficiency of QCs. All these decisions are also related to the berth allocation of vessels. Given the multi-criterion nature, the complexity of operations, and the size of the entire operations management problem, it is impossible to make the optimal decisions that will achieve the overall objectives. Logically, the hierarchical approach is adopted to break the whole problem into smaller sequential problems. The input to a problem is actually the output of its immediate predecessor, and is treated as a known quantity after the preceding problem is solved. Fig. 2 gives a typical hierarchical structure of operational decisions in a container terminal. See Zhang (2000) and Murty et al. (2000) for a more detailed description of these problems.

In this paper we will concentrate on the storage space allocation problem. We will decide in which blocks to place the VSIDS and CYGD containers of each vessel. Our formulation and its

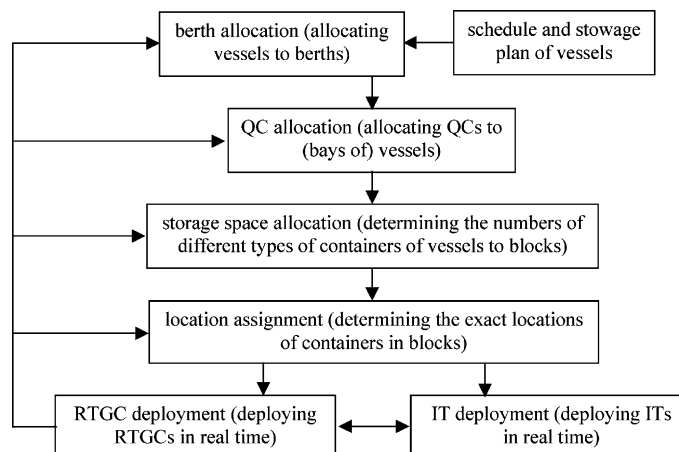


Fig. 2. Hierarchical structure of operational decisions in a container terminal.

solution take care of I/B, O/B and transit containers. Note that after being placed in the yard, I/B VSDS containers will turn into CYPI containers, and transit VSDS containers and CYGD containers will turn into VSLD containers. Decisions about space allocation not only directly affect the workloads of yard cranes for storing the VSDS and CYGD containers, but also affect the workloads of yard cranes for retrieving the CYPI and VSLD containers in later periods.

In the rest of the paper, we first briefly review relevant literature in Section 2. Then a detailed problem description is given and a solution approach is outlined in Section 3. With this approach the problem is modeled and solved at two levels, Sections 4 and 5 for the first and the second levels, respectively. A numerical study comparing the proposed approach with the current practice in Hong Kong is reported in Section 6. Section 7 concludes the paper.

2. Literature review

Terminal operators need to consider both the design and the operational issues of the container yards. At the design stage, the storage space capacity of the yard is determined by considering primarily the tradeoffs between the set up and the operation stages (e.g., the land and construction costs versus operational efficiency). We study an operational problem in this paper: the space allocation decisions for a given terminal, where the amount of storage space is fixed.

Various aspects of the operational problems in Fig. 2 have been studied. See, for examples, Zhang et al. (2002) for crane deployment, Imai et al. (2001) for berth allocation, and Zhang (2000) for a summary of various problems. Here we concentrate on the storage space allocation of containers.

Schematically, the yard operations of a container terminal are the storage and retrieval of containers, which can be streamlined by automatic material handling equipment. Indeed the Delta Terminal in the Netherlands is equipped with automatic guided vehicles and automated stacker cranes. However, with more intensive operations in a smaller area, there is no plan for the container hub terminals in Hong Kong to widely adopt such systems.

There are theoretical studies showing the advantages of using the automated storage/retrieval system (AS/RS) for the storage of container terminals (e.g., Liu et al., 2002). While the storing and retrieving in an AS/RS is similar to that in a container terminal, their operations bear one key difference. In an AS/RS, the storing and retrieving times of storage cells are linked together by the limited resources, such as stacker cranes for lifting and automated guided vehicles for transporting. In a container terminal, the stacking of containers adds in a new dimension: the storage of a container affects the retrieval times of containers already in a stack, and the retrieval of containers also affects future container retrieval times (when containers on top of a retrieved container are put on top of other containers). Given this difference and that terminals of our interest do not operate with AS/RS, we will not further investigate the effect of AS/RS for terminals. Interested readers can read, for examples, Liu et al. (2002) for the application of automated material handling systems in container terminals and Thonemann and Brandeau (1998) for determining optimal location in an AS/RS.

The export (O/B) and import (I/B) containers have different characteristics. Export containers are usually carried by trucks to terminals and their arrival times are often uncontrollable. For export containers of the same destination and similar weight (i.e., they are of the same *group*),

their storage locations on vessels are interchangeable. Any stacking arrangements of such containers on the container yard are equivalent, because they take the same retrieval effort. The storage locations of groups have been the key point of interest for export containers. Contrarily, import containers come in batches on vessels at predictable times, and leave in a random order on trucks. The expected number of moves to retrieve them has received attention. For terminals with a large area, which is *not* applicable to Hong Kong, they tend to separate the yards into two parts, one for export and the other for import containers. Most previous research studies the operations of the two areas separately as well.

Given the vessel arrival schedules and the workload, Taleb-Ibrahimi et al. (1993) find the aggregate space requirements for two storage strategies of export containers. They conclude that the strategy of having a buffer space and marshaling containers not in their permanent positions virtually eliminate wasted space. In addition to space, Taleb-Ibrahimi (1993) considers also the amount of handling equipment for a given amount of workload under different container arrival and departure patterns. Kim and Bae (1998) discuss how to re-marshall export containers in the storage yard from any given configuration to the best configuration for loading on vessels. They use the hierarchical approach to break the problem into three sub-problems and solve each of them as an optimization problem.

Roux (1996) derives analytical expressions to estimate the minimum storage capacity for a given throughput requirement for import containers under the constraint of infrequent congestion. Kim and Kim (1998) determine the best combination of space and the number of yard cranes that minimizes the total cost of space, yard cranes, and ITs. Castilho and Daganzo (1993) compare the amount of handling work required for retrieving import containers under two strategies, storing to reduce the difference in stack heights, and segregating according to the arrival times. They find that the segregation strategy works better for dwell times of low variability. Kim (1997) derives expressions to estimate the amount of moves to retrieve one container and hence to clear an arbitrary heap. Regression equations and tables are developed to estimate the expected number of re-handles easily and quickly. See Holguin-Veras (1996) for the pricing and optimal amount of space for containers of different priorities.

The above papers consider the space requirement for planning, the amount of (re-)handling work of containers, or both the space requirement and the handling work of containers. None of them consider the storage location of containers in the operational stage. In this aspect, Kim et al. (2000) use dynamic programming and a decision tree heuristic to determine the storage locations of export containers, aiming at reducing the total number of re-handling. Bruzzone and Signorile (1998) combine simulation and genetic algorithms to determine the storage clusters of containers (and the berth allocation) of vessels.

Due to the limited storage space and the very high throughput, container terminals in Hong Kong (and in some other Asian ports) have much higher space utilization than ports in other parts of the world. Hong Kong terminals mix inbound, outbound, and transit containers at the block level. While they tend to separate containers going to vessels and to customers at the stack level, they do not fix the purpose of a stack. Whenever a stack becomes empty, it may be used to store any type of containers no matter what type of containers was previously stored. In addition, there is no buffer area for incoming containers. Containers are placed directly in storage blocks upon their arrivals (though marshaling is possible when there is time). This policy can save space and reduce re-handling efforts under high space utilization. However, it complicates the problem

because the allocation of import and export containers has to be considered at the same time for each block. This paper considers the space allocation problem in storage yards under this complex storage policy.

3. Problem description and solution approach

3.1. Problem description

From the hierarchical approach, we assume that the berth allocation problem and the QC allocation problem have been solved (Fig. 2). Our problem is to determine the numbers of VSDS and CYGD containers of each vessel stored in each block. The pre-requisite is to determine the (expected) workload requirements (in terms of number of CYGD, VSDS, VSLD, and CYPI) for each time period of the yard.

Fig. 3 presents a typical pattern of the variation with time for the numbers of I/B and O/B containers in the yard of a vessel. The duration of unloading and loading of a vessel, (t_1, t_3) , can be assumed to be a known quantity for a given vessel (with the pre-specified workload). Transit containers are discharged from one vessel and later re-loaded onto other vessels (without leaving the terminal). The patterns in time intervals, (t_0, t_1) and (t_2, t_4) , which are the accumulation of O/B and the dissipation of I/B containers, respectively, are random. Fortunately, over time, these patterns of accumulation and dissipation are stable. Consequently, for any vessel, given the workload associated with it, we can deduce the (expected) workload of the vessel for any time period. By superimposing the workloads of all vessels, we can find all types of workloads for any time period for the whole terminal.

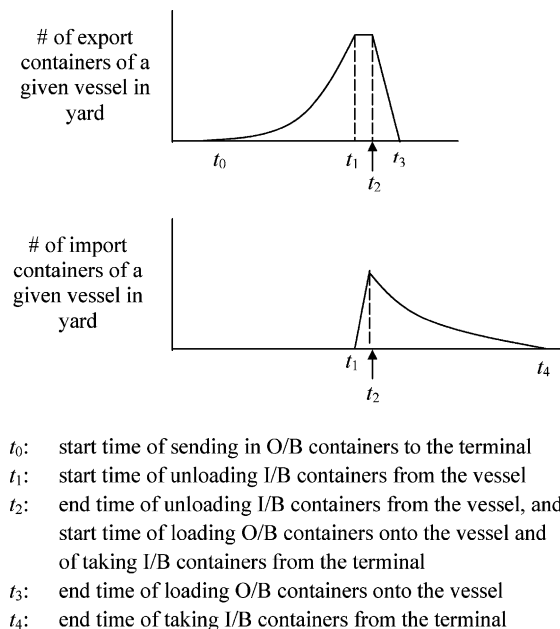


Fig. 3. The numbers of inbound and outbound containers in the yard associated with a vessel versus time.

Given the total workload requirement of the whole container terminal, we have a better idea of the demand for RTGCs, QCs, XTs and ITs over time. Clearly, it would be still too complex to consider the interaction of the storage blocks, the travel of ITs, and the availability of RTGCs and QCs all together. Instead, we further break down the storage space allocation problem into two levels. For each level, we select an objective function in line with the overall objective of minimizing the vessel berthing time and maximizing the QC throughput rate.

At the first level, to minimize vessel berthing times, we balance the workload of RTGCs and QCs for vessels. With workloads of a vessel dispersing in different blocks, the yard cranes in the blocks serve as parallel servers processing jobs for the vessel, and the deberthing time of the vessel is the maximal processing time of these parallel servers. Balancing the workload of parallel servers generally works well to minimize the completion times of vessels. Similar results on the RTGC deployment problem confirm that balancing workloads of blocks reduces delay in container handling (Zhang et al., 2002).

There are several aspects of balancing at the first level. It is natural to balance the total number of containers handled among different blocks, which equalizes the workload of RTGCs. However, purely doing so ignores the key that VSDS and VSLD containers are related to the on-time departures of vessels. We have to balance them and also highlight their effect as compared to that of the total workload. See Section 4 for our choice of an objective function that considers these two types of balancing.

The second level determines the number of containers associated with each vessel that constitutes the total number of containers in each block in each period, in order to minimize the total distance to transport the containers between their storage blocks and the vessel berthing locations.

Within a block, the exact location of a container can be assigned to shorten the handling time by minimizing re-shuffling. This is the decision about storage location, which is a problem at a lower level (of the hierarchy), and is not discussed in this paper. In a similar fashion, we do not explicitly consider the effect of destinations of the containers. These are decisions about the stowage plan of vessels, which is a problem at a higher level.

3.2. Implementation details of the solution approach

A container terminal operates around the clock each day for 365 days a year. This forces us to choose a fixed planning horizon and run our method with the rolling-horizon approach: At each planning epoch, we plan for a fixed horizon in immediate future and execute the plan accordingly up to the next planning epoch; then we formulate a new plan based on the latest information; this pattern goes on continually (Fig. 4).

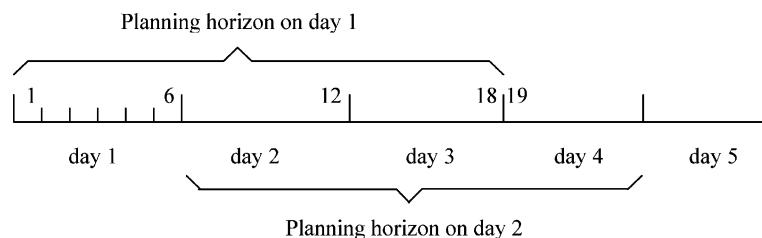


Fig. 4. Rolling of the planning horizon.

A short planning horizon means less computational burden but also less predictive power about the future, while a long planning horizon may be computationally unfeasible and may include too much uncertain information. By judging the effect of the planning horizon on the complexity of the problem, the feasibility of computation, and the validity of the data, we settle on a planning horizon of three days, with each day being divided into six 4-hour periods. At (the beginning of) day 1, a storage space allocation plan is formed for the 18 periods in days 1–3. Only the first day of the plan is executed and a new three-day plan is formed at the end of the first day (beginning of the second day) based on the latest information. This goes on for every day.

The maximum dwell times of both I/B and O/B containers approximately equal the maximal free storage period, which is beyond the planning horizon. Thus, there are containers with unknown departure times at the moment of planning or containers with known departure times beyond the planning horizon. Their workloads do not occur in the planning horizon and consequently such containers are not directly included in the storage allocation model. To account for their possible effect in future, these containers are distributed to blocks in proportion to their available storage capacities at the beginning of the planning horizon so as to balance the block densities. Such an approximation has a marginal effect on the overall performance: The majority of containers of a vessel are accumulated and dissipated in the planning horizon (within three days before or after the vessel's berthing) and, in any case, most containers are allocated under known information, since only the first day of the three-day plan is implemented.

The rolling-horizon approach and the two-level solution method are summarized in Fig. 5. For each planning horizon, the sub-problems at the two levels are solved using mathematical programming models. The details of the two models will be presented in Sections 4 and 5, respectively.

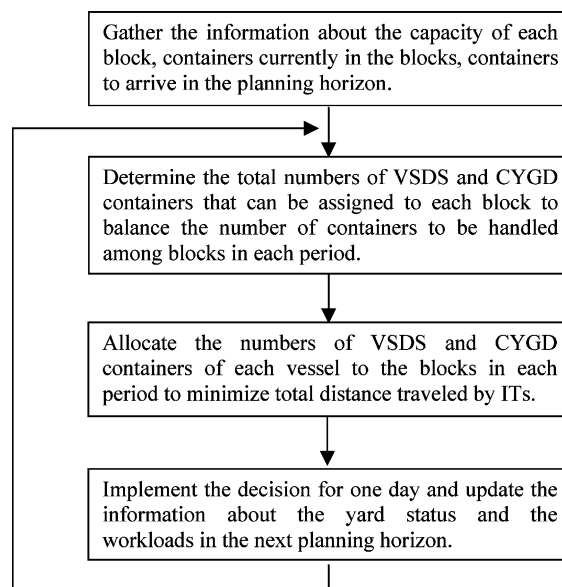


Fig. 5. The framework of the solution methodology.

4. Assignment of total numbers of containers to blocks

In this section, we formulate the first-level problem as an integer programming model. Basically, we want to determine the numbers of VSDS and CYGD containers stored in each block for each planning period. In our formulation, we assume that there is enough resource to handle the workload. This is close to reality for terminals in Hong Kong, where in general there is a sufficient number of yard cranes. As a bi-product, in Hong Kong, the movements of yard cranes among blocks usually take place after major work shifts, which is not as at real time as possibly in some overseas container terminals (Zhang et al., 2002). Consequently, we ignore the movement of yard cranes in our model.

Our model assumes that containers are of one size. The workloads are counted in terms of the number of containers. For consistency the storage capacities are also measured in numbers of containers. While there exist containers of different sizes (as well as different types, such as regular and refrigerated containers), our method is not directly affected. In general, containers of different sizes are normally not mixed in blocks, and the size of containers stored in a block is only changed very infrequently. Consequently, the storage of containers of different sizes can be decoupled into separate problems.

4.1. Notation

We first present the notation for data and then for decision variables. The *data known at the beginning of a planning horizon* are:

B	the total number of blocks in the yard;
T	the total number of planning periods in a planning horizon, $T = 18$;
C_i	the storage capacity of block i , $1 \leq i \leq B$;
V_{i0}	the initial inventory of block i , i.e., the number of containers in block i at the beginning of the planning horizon, $1 \leq i \leq B$;
P_{it}^0	the expected number of initial CYPI containers stored in block i to be picked up in period t , $1 \leq t \leq T$, $1 \leq i \leq B$;
L_{it}^0	the expected number of initial VSLD containers stored in block i to be loaded onto vessels in period t , $1 \leq t \leq T$, $1 \leq i \leq B$;
\tilde{G}_{tk}	the expected total number of CYGD containers that arrive at the container terminal in period t and to be loaded onto the vessels in period $t + k$, $1 \leq t \leq T$, $0 \leq k \leq T - t$;
\tilde{D}_{tk}	the expected number of I/B VSDS containers that are discharged from vessels in period t and to be picked up by customers in period $t + k$, $1 \leq t \leq T$, $0 \leq k \leq T - t$;
\tilde{R}_{tk}	the expected number of transit containers that are discharged from vessels in period t and to be loaded onto other vessels in period $t + k$, $1 \leq t \leq T$, $0 \leq k \leq T - t$;
α_{it}	the expected number of CYGD containers arriving at the container terminal in period t , allocated to block i (determined by the proportional method) and to be loaded onto vessels in periods beyond the planning horizon, $1 \leq t \leq T$, $1 \leq i \leq B$;
β_{it}	the expected number of I/B VSDS containers discharged from vessels in period t , allocated to block i (determined by the proportional method), with an unknown pickup time or a pickup time beyond the planning horizon, $1 \leq t \leq T$, $1 \leq i \leq B$;

γ_{it} the expected number of transit containers discharged from vessels in period t , allocated to block i (determined by the proportional method), with an unknown loading time or a loading time beyond the planning horizon, $1 \leq t \leq T$, $1 \leq i \leq B$.

Both \tilde{G}_{tk} and α_{it} are the expected number of CYGD containers found from the workloads of vessels and the patterns that CYGD containers are brought into the terminal yard. Containers counted in \tilde{G}_{tk} have their vessel loading times within the current planning horizon and they will be converted to VSLD containers accordingly; containers counted in α_{it} have their vessel loading times beyond the current planning horizon and they affect the initial condition of the next planning epoch. Similarly, \tilde{D}_{tk} and β_{it} are the expected number of VSDS containers such that the pickup times of \tilde{D}_{tk} are within and of β_{it} are beyond the planning horizon; \tilde{R}_{tk} and γ_{it} are the expected number of transit containers such that the loading times of \tilde{R}_{tk} are within and of γ_{it} are beyond the planning horizon.

The *decision variables* of the model are:

- G_{itk} the number of CYGD containers with full information stored in block i such that they arrive at the terminal in period t and to be loaded onto vessels in period $t + k$, $1 \leq i \leq B$, $1 \leq t \leq T$, $0 \leq k \leq T - t$;
- G_{it} the total number of CYGD containers (with full or partial information) stored in block i that arrive at the terminal in period t , $1 \leq i \leq B$, $1 \leq t \leq T$;
- D_{itk} the number of I/B VSDS containers with full information stored in block i that are discharged from vessels in period t and to be picked up in period $t + k$, $1 \leq i \leq B$, $1 \leq t \leq T$, $0 \leq k \leq T - t$;
- D_{it} the total number of VSDS (inbound and transit) containers (with full or partial information) stored in block i that are discharged from vessels during period t , $1 \leq i \leq B$, $1 \leq t \leq T$;
- R_{itk} the number of transit containers with full information stored in block i that are discharged from vessels in period t and to be loaded onto other vessels in period $t + k$, $1 \leq i \leq B$, $1 \leq t \leq T$, $0 \leq k \leq T - t$;
- L_{it} the total number of VSLD (outbound and transit) containers stored in block i that are loaded onto vessels in period t , $1 \leq i \leq B$, $1 \leq t \leq T$;
- P_{it} the total number of CYPI containers stored in block i that are picked up by customers in period t , $1 \leq i \leq B$, $1 \leq t \leq T$;
- V_{it} the inventory of block i at the end of period t , $1 \leq i \leq B$, $1 \leq t \leq T$.

4.2. The objective function

Section 3.1 explains the rationale to balance the vessel related (loading and discharging) containers and the total number of containers among blocks in each period. This objective is:

$$\begin{aligned} \text{Minimize} \quad & \sum_{t=1}^T \{w_1 [\max_{\{i\}} (D_{it} + L_{it}) - \min_{\{i\}} (D_{it} + L_{it})] \\ & + w_2 [\max_{\{i\}} (D_{it} + L_{it} + G_{it} + P_{it}) - \min_{\{i\}} (D_{it} + L_{it} + G_{it} + P_{it})]\}. \end{aligned} \quad (1)$$

In this function, $(D_{it} + L_{it})$ is the expected total number of vessel related containers that need to be handled in block i during period t and $(D_{it} + L_{it} + G_{it} + P_{it})$ is the expected total number of containers to be handled in block i during period t . Therefore the two terms of (1) measure the imbalances of the vessel related containers and of the total number of containers in the blocks in each planning period, respectively. w_1 and w_2 , the weights of the two terms in (1), are adjusted according to the relative importance of the vessel related containers within the total number of containers as interpreted by a terminal. Theoretically, it is possible to set $(w_1, w_2) = (1, 0)$ or $= (0, 1)$, dependent on whether the vessel related containers or the total number of containers are of utmost importance. In general, both w_1 and w_2 are strictly positive in practice, and are tuned according to the needs of a container terminal.

4.3. The constraints

The following constraints are introduced in the model to ensure the practical feasibility of the solution.

(a) Container flow conservation constraints

$$\tilde{D}_{tk} = \sum_{i=1}^B D_{itk}, \quad t = 1, 2, \dots, T; \quad k = 0, 1, \dots, T - t. \quad (2)$$

$$\tilde{G}_{tk} = \sum_{i=1}^B G_{itk}, \quad t = 1, 2, \dots, T; \quad k = 0, 1, \dots, T - t. \quad (3)$$

$$\tilde{R}_{tk} = \sum_{i=1}^B R_{itk}, \quad t = 1, 2, \dots, T; \quad k = 0, 1, \dots, T - t. \quad (4)$$

$$D_{it} = \sum_{k=0}^{T-t} (D_{itk} + R_{itk}) + \beta_{it} + \gamma_{it}, \quad i = 1, 2, \dots, B; \quad t = 1, 2, \dots, T. \quad (5)$$

$$G_{it} = \sum_{k=0}^{T-t} G_{itk} + \alpha_{it}, \quad i = 1, 2, \dots, B; \quad t = 1, 2, \dots, T. \quad (6)$$

Constraint (2) ensures that the expected total number of I/B VSDS containers with full information waiting for the allocation, \tilde{D}_{tk} , is the sum of these containers assigned to all the blocks. Constraints (3) and (4) have a similar meaning but for CYGD and transit VSDS containers, respectively.

Constraint (5) ensures that the expected total number of VSDS containers allocated to block i during period t , D_{it} , is the sum of the total number of VSDS (I/B and transit) containers with full information, $\sum_{k=0}^{T-t} (D_{itk} + R_{itk})$, and of those containers with unknown departure times at the planning horizon, $\beta_{it} + \gamma_{it}$. Constraint (6) is a similar balance of flow for CYGD containers.

(b) Constraints on CYPI and VSLD containers

$$L_{it} = L_{it}^0 + \sum_{k=0}^{t-1} (G_{i(t-k)k} + R_{i(t-k)k}), \quad i = 1, 2, \dots, B; \quad t = 1, 2, \dots, T. \quad (7)$$

$$P_{it} = P_{it}^0 + \sum_{k=0}^{t-1} D_{i(t-k)k}, \quad i = 1, 2, \dots, B; \quad t = 1, 2, \dots, T. \quad (8)$$

Constraint (7) indicates that the number of VSLD containers handled in block i during period t , L_{it} , consists of two parts. The first part is the VSLD containers initially stored in block i to be loaded onto the vessels in period t in the current planning horizon, L_{it}^0 . The second part is the containers transferred from the corresponding CYGD and transit containers that arrived in the planning horizon, $\sum_{k=0}^{t-1} (G_{i(t-k)k} + R_{i(t-k)k})$. Constraint (8) has a similar meaning but for CYPI containers.

(c) Block density constraints

$$V_{it} = V_{i(t-1)} + [(G_{it} + D_{it}) - (P_{it} + L_{it})], \quad i = 1, 2, \dots, B; \quad t = 1, 2, \dots, T. \quad (9)$$

$$V_{it} \leq \eta C_i, \quad i = 1, 2, \dots, B; \quad t = 1, 2, \dots, T, \quad (10)$$

where η is the allowable density for each block.

Constraint (9) represents the updating of inventory, V_{it} , from period to period. Constraint (10) ensures that the inventory of each block in each planning period will not exceed the allowable block density.

(d) Integer conditions

$$\text{All variables take up non-negative integer values.} \quad (11)$$

4.4. Conversion to a linear model

The above model is non-linear because of the objective function. To convert it to a linear model, we define:

$$A_t = \max_{\{i\}} (D_{it} + L_{it}), \quad B_t = \min_{\{i\}} (D_{it} + L_{it}),$$

$$M_t = \max_{\{i\}} (D_{it} + L_{it} + G_{it} + P_{it}), \quad N_t = \min_{\{i\}} (D_{it} + L_{it} + G_{it} + P_{it}).$$

Then the model can be rewritten as the linear integer programming model below.

$$(\text{LIP}) \text{ Minimize } \sum_{t=1}^T [w_1(A_t - B_t) + w_2(M_t - N_t)] \quad (12)$$

Subject to (2)–(11) and

$$D_{it} + L_{it} \leq A_t, \quad i = 1, 2, \dots, B; \quad t = 1, 2, \dots, T. \quad (13)$$

$$D_{it} + L_{it} \geq B_t, \quad i = 1, 2, \dots, B; \quad t = 1, 2, \dots, T. \quad (14)$$

$$G_{it} + D_{it} + L_{it} + P_{it} \leq M_t, \quad i = 1, 2, \dots, B; \quad t = 1, 2, \dots, T. \quad (15)$$

$$G_{it} + D_{it} + L_{it} + P_{it} \geq N_t, \quad i = 1, 2, \dots, B; \quad t = 1, 2, \dots, T. \quad (16)$$

The additional constraints (13)–(16) reflect the definitions of the new variables A , B , M and N in the model. With these definitions, it is clear that the new linear objective function (12) is equivalent to the original non-linear function (1).

5. Allocation of containers of each vessel to blocks

The first level determines the total number of VSDS and the total number of CYGD containers that can be assigned to each block in each planning period. The second level determines the number of containers associated with each vessel that constitutes the total number of containers in each block. The objective is to minimize the total container moving cost. The moving cost is measured by the total distance traveled by ITs between the berthing places of vessels and the storage blocks.

After solving the problem at the first level, the numbers of VSDS and CYGD containers to be placed to each block in each planning period, $D_{itk} + R_{itk}$ and G_{itk} , are fixed. Therefore, the second-level decisions can be made for VSDS and CYGD containers separately in each planning period to decide vessel identifications of containers. Due to the similarities between the allocation problems for CYGD and VSDS containers, one set of notation and a common mathematical model is proposed for both VSDS and CYGD containers with different parameters and decision variables.

5.1. Notation

Table 1 presents the notation for VSDS and CYGD containers. Similarly, the decision variables in the model for the VSDS and CYGD containers are defined in Table 2.

5.2. Model structure

The problem can be solved period by period using the same model. For each period, $t = 1, 2, \dots, T$, the problem can be viewed as a transportation problem as shown in Fig. 6.

Table 1
Notation for parameters in the level-two problem

	Meaning for VSDS containers	Meaning for CYGD containers
d_{ij}	Distance between storage block i and the berthing place of vessel j	Distance between storage block i and the berthing place of vessel j
N_{jt}	The total number of VSDS containers discharged from vessel j in period t	The total number of CYGD containers arrived in period t to be loaded onto vessel j
S_t	The total number of vessels berthed in the terminal in period t for discharging	The total number of vessels onto which the CYGD containers arrived in period t to be loaded
U_{itk}	$D_{itk} + R_{itk}$	G_{itk}

Table 2
Definition of decision variables in the level-two problem

	Meaning for VSDS containers	Meaning for CYGD containers
X_{ijtk}	The number of VSDS containers discharged from vessel j in period t to be picked up by customers or loaded onto other vessels in period $t + k$ that can be stored in block i	The number of CYGD containers arrived at the terminal in period t to be loaded onto vessel j in period $t + k$ that can be stored in block i

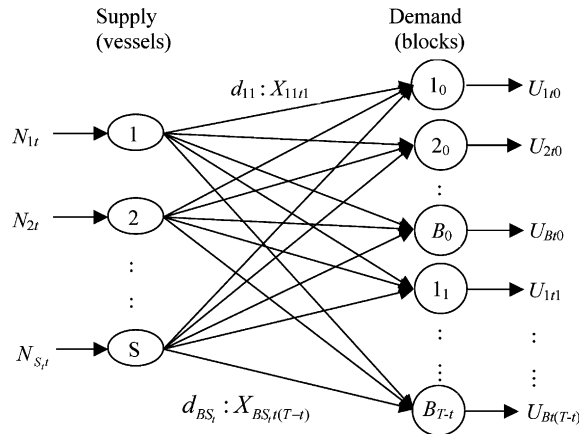


Fig. 6. Second-level problem representation.

There are two classes of nodes or constraints in this transportation model. The supply nodes represent the vessels and the demand nodes represent the blocks. There are totally S_t supply nodes in period t . Each storage block is split into $T-t$ demand nodes with demands U_{itk} ($k = 1, 2, \dots, T-t$), respectively. The total number of demand nodes in the network is then $B(T-t)$. Connecting each supply node, j , to each demand node, i , is an arc with a container flow of X_{ijtk} and a cost coefficient of d_{ij} .

5.3. Model formulation

Mathematically, the transportation model can be presented as follows.

$$\text{Minimize} \quad \sum_{i=1}^B \sum_{j=1}^{S_t} \sum_{k=0}^{T-t} d_{ij} X_{ijtk} \quad (17)$$

$$\text{Subject to} \quad \sum_{j=1}^{S_t} X_{ijtk} = U_{itk}, \quad i = 1, 2, \dots, B; \quad k = 0, 1, \dots, T-t. \quad (18)$$

$$\sum_{i=1}^B \sum_{k=0}^{T-t} X_{ijtk} = N_{jt}, \quad j = 1, 2, \dots, S_t. \quad (19)$$

$$X_{ijtk} \geq 0, \quad i = 1, 2, \dots, B; \quad j = 1, 2, \dots, S_t; \quad k = 0, 1, \dots, T-t. \quad (20)$$

The objective function (17) is to minimize the total IT traveling distance. Constraint (18) represents the capacity constraint of each demand node in a period while constraint (19) is the vessel discharge quantity constraint of each supply node. Constraint (20) is the non-negativity constraint. Since all N_{jt} and U_{itk} are integers, integer solutions of X_{ijtk} are guaranteed by the transportation model structure.

6. Numerical study

In this section, the proposed storage block assignment method, model (LIP), is tested using practical data generated from a typical container terminal in Hong Kong for a total of 180 days with 6 periods in each day. The assignment results of the proposed approach are compared with those of the current practice. Given the result of the first level, the second-level transportation models can be solved easily. Since the terminals do not keep the traveling distances of ITs, our comparison will be based on the workload imbalance only.

6.1. Input data

In this experiment, the storage yard has 10 blocks with different capacities. Handled by RTGCs, each bay of a block is 6-stack, 5-tier. As one stack in each bay is usually reserved for the temporary storage of re-shuffled containers in that bay (when digging out a required container at the bottom of a stack if necessary), the stack cannot be used for storage. Based on this consideration, the actual capacity for use is $\eta C_i = (5/6)C_i$. Table 3 shows the nominal and actual storage capacity for each of the 10 blocks.

The following container flow data are needed for each planning horizon: L_{it}^0 , P_{it}^0 , \tilde{D}_{tk} , \tilde{G}_{tk} , β_{it} , α_{it} , and V_{i0} . There is no transit container in this example. In the experiment these data are generated based on the distributions of the real container flows at Hong Kong container terminals. The process is as follows: (1) the arrival and departure times of each container in the 10 blocks are collected for three months; (2) based on these data, the distributions of numbers of discharge and grounding container arrivals in a period and the distributions of the number of periods for which a container stays in the yard are obtained; (3) container flow data used in the experiment are generated using these distributions.

6.2. Implementation of the proposed approach

The procedure for allocating storage blocks to VSDS and CYGD containers arriving in each period of each planning horizon during the 180 days is as follows. It is consistent with the flow chart in Fig. 5.

For $h = 1$ to 180

{

1. Read the input data (stated in Section 6.1) of the current planning horizon h .
2. Solve the integer programming model (11)–(15) to obtain the numbers of I/B and O/B containers allocated to each block.

Table 3
Capacity of each block

	1	2	3	4	5	6	7	8	9	10
C_i	210	1230	750	630	540	1230	480	630	900	1260
ηC_i	175	1025	625	525	450	1025	400	525	750	1050

3. Solve the transportation model (16)–(19) to determine the numbers of I/B and O/B containers of each vessel allocated to each block.
4. Update the container information:
 - a. Update the initial inventory for the next planning horizon.
 - b. Update L_{it}^0 and P_{it}^0 .

}

The program is coded in C++ and run on a Pentium III computer. The integer programming model is solved using CPLEX 7.0, a commercial software package.

6.3. Results

The weights used in model (2)–(16) are set as $w_1 = w_2 = 0.5$. Because of the large size of the integer programming model (totally there are 4320 integer variables), there is no guarantee that the model can be solved to optimality in a short time. Therefore, we set CPLEX to emphasize on integer feasibility and let it stop after a feasible integer solution was found. The computational results of the integer programming model for the 180 planning horizons are summarized in Table 4.

The table reflects that the solutions obtained are near-optimal. The relative gap to optimal, i.e., the relative gap between the upper and lower bounds of the objective value, has an average value of only 1.84% and a maximum of 6.58%. For some planning horizons the solutions are indeed optimal. Fig. 7 shows the relative gap for all the planning horizons tested. The average

Table 4
Computational results

	Relative gap to lower bound (%)	Computation time (s)	Imbalance in vessel related containers	Imbalance in total no. of containers
Average	1.84	109.64	7.25	1.45
S.D.	1.29	77.71	2.94	1.49
Maximum	6.58	541.91	18	13
Minimum	0.00	16.46	0	0

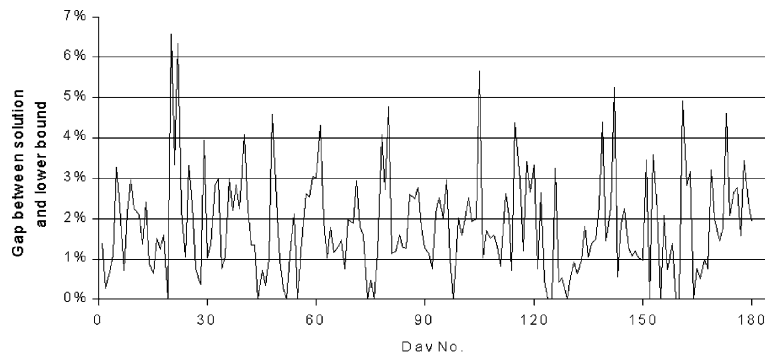


Fig. 7. Relative gap between the solution and lower bound.

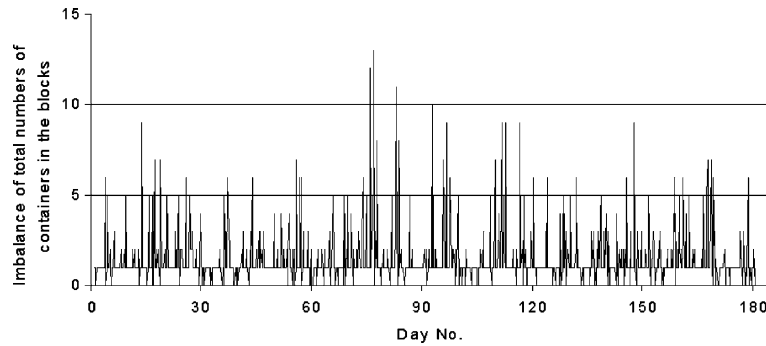


Fig. 8. Results of the proposed approach: imbalance of total numbers of containers.

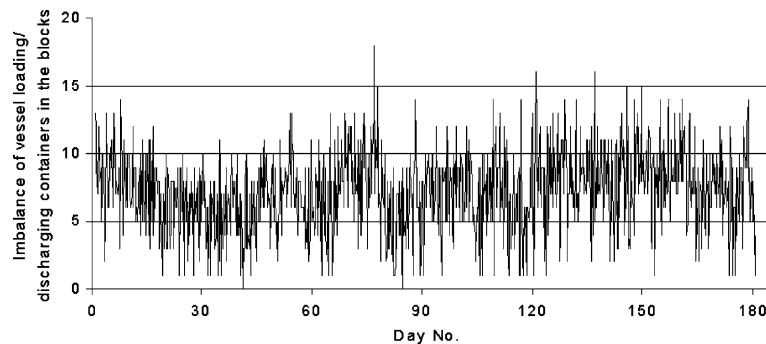


Fig. 9. Results of the proposed approach: imbalance of vessel handling containers.

computation time for a planning horizon is about 110 s, which is very short as compared to the planning cycle. Even the maximum of around 9 min serves the daily planning purpose well. In practice, a container terminal can enhance the performance of our method by pre-setting a long computation time (e.g., 30 min) that is conformable with the daily planning requirement.

The performance measures of the allocation are the imbalance of the total number of containers among the 10 blocks, $\max_{\{i\}}(D_{it} + G_{it} + L_{it} + P_{it}) - \min_{\{i\}}(D_{it} + G_{it} + L_{it} + P_{it})$, and the imbalance of the vessel related containers, $\max_{\{i\}}(D_{it} + L_{it}) - \min_{\{i\}}(D_{it} + L_{it})$, in each period t . Figs. 8 and 9 show the resulting two performance measures of the proposed approach, respectively, over the 180 days tested. A summary of the measures is included in Table 4. The average imbalance of the total number of containers in the yard in each period is only 1.45, representing a 96.58% improvement over the average figure in the current real operation. The average imbalance of the number of vessel related containers in each period is 7.25, representing a 87.75% improvement over that of the current real operation. These results demonstrate the practical usefulness of the model and the effectiveness of the approach.

6.4. Further experiments and discussion

To test the robustness of our method, we carried out further experiments with different parameter settings. To ensure meaningful comparisons, we used the same 180-day data for these

Table 5
Result summary of further computational experiments

Changed settings for new run	Relative gap to lower bound (%)	Computation time (s)	Imbalance in vessel related containers	Imbalance in total no. of containers
$w_1 = 1, w_2 = 0$	0.32	0.89	4.29	19.58
$w_1 = 0, w_2 = 1$	9.23	11.81	15.31	1.37
$w_1 = 1, w_2 = 0.5$	1.26	107.42	6.19	2.94
5-min run	1.24	260.34	7.17	1.46

experiments. The results are summarized in Table 5. In the above numerical runs, we set to 0.5 both w_1 the weight of the imbalance of vessel related containers, and w_2 the weight of the imbalance of total number of containers. In the first two new experiments, we set one of the weights to 1 and another to 0, i.e., we minimize only one of the imbalances and ignore the other. From the results (the first two rows of Table 5), we can see that the computation times are shorter to obtain an integer solution. Although the resulting relative gaps for two settings are different, one being very small and the other one being quite large, the performance measures in terms of imbalance are similar in the following sense: the imbalance of the quantity with $w_i = 1$ becomes slightly better but the other imbalance with $w_j = 0$ becomes much worse, $i, j = 1, 2, i \neq j$, as compared to the original result with $w_1 = w_2 = 0.5$. This shows that combining the two types of imbalances in the objective function is a better strategy than objective functions with only one type of imbalance. Since the imbalance in vessel related workload is larger than that in the total workload in the original result, we change the weights to $w_1 = 1$ and $w_2 = 0.5$ in the third new experiment. This leads to a little reduction of the imbalance in vessel related workload and a little increase of the imbalance in total workload (see the third row of Table 5). The computation time is almost the same as the original model. This result indicates that the model is not very sensitive to the values of the weights. In the fourth new experiment, we allow the program to run for 5 min or, in cases that an integer solution has not been obtained within this time, until the first integer solution is found, for each planning horizon. As the optimal solution is reached before the 5-min limit for some planning horizons, the average computation time is less than 5 min, but more than double of that of the original experiment. With this much longer time, only a very small improvement is achieved for both the relative gap and the imbalances (see the last row of Table 5). In fact the small relative gap achieved in the original experiment indicates that there is little room for improvement even we allow the program run to optimal. In summary, these experiments show that the proposed method is very effective and efficient for the container yard storage allocation problem and is quite robust to parameter settings and computation time.

7. Conclusions

In this paper, we have studied the storage space allocation problem in the storage yards of container terminals. This problem is related to all the resources in the terminal operations, including QCs, yard cranes, storage space, and ITs. We considered a complex situation in which inbound, outbound and transit containers are mixed in the storage blocks in the yard. This is

consistent with the practice in Hong Kong, the busiest container port of the world. We proposed a rolling-horizon approach to solve the problem. For each planning horizon, the problem is decomposed into two levels and each level is formulated as a mathematical programming model. The solution to the first level determines the total number of containers to be placed in each storage block in each time period so as to balance the workloads among blocks in each period. The solution to the second level determines the number of containers associated with each vessel that makes up the total number allocated to each block in each period, in order to minimize the total distance to transport the containers between their storage blocks and the vessel berthing locations. Our numerical runs showed that with short computation time, the method significantly reduces the workload imbalance in the yard, avoiding possible bottlenecks in terminal operations. In future, we may try different hierarchical breakdowns of the space allocation problem so as to improve the solution procedure. With this problem solved, we will start to study the location assignment problem of containers, a lower level problem in the hierarchy.

Acknowledgements

We thank three referees for their constructive comments that improve the content and the presentation of the paper. The research was supported in part by the Research Infrastructure Grant (RI95/96.EG04) of the Hong Kong University of Science and Technology and by the Service Support Fund (SSF/066/96) of the Industrial Department, the Government of the Hong Kong SAR.

References

- Bruzzzone, A., Signorile, R., 1998. Simulation and genetic algorithms for ship planning and shipyard layout. *Simulation* 71 (2), 74–83.
- Castilho, B.D., Daganzo, C.F., 1993. Handling strategies for import containers at marine terminals. *Transportation Research* 27 (2), 151–166.
- Holguin-Veras, J., 1996. Priority systems for marine inter-modal containers. Ph.D. Thesis, The University of Texas, Austin.
- Imai, A., Nishimura, E., Papadimitriou, S., 2001. The dynamic berth allocation problem for a container port. *Transportation Research Part B* 35 (4), 401–417.
- Kim, K.H., 1997. Evaluation of the number of rehandles in container yards. *Computers and Industrial Engineering* 32 (4), 701–711.
- Kim, K.H., Bae, J.W., 1998. Re-marshaling export containers in port container terminals. *Computers and Industrial Engineering* 35 (3–4), 655–658.
- Kim, K.H., Kim, H.B., 1998. The optimal determination of the space requirement and the number of transfer cranes for import containers. *Computers and Industrial Engineering* 35 (3–4), 427–430.
- Kim, K.H., Park, Y.M., Ryu, K.-R., 2000. Deriving decision rules to locate export containers in container yards. *European Journal of Operational Research* 124, 89–101.
- Liu, C.I., Jula, H., Ioannou, P.A., 2002. Design, simulation, evaluation of automated container terminals. *IEEE Transactions on Intelligent Transportation Systems* 3 (1), 12–26.
- Murty, K.G., Liu, J., Wan, Y.-w., Zhang, C., Tsang, M., Linn, R., 2000. DSS (decision support systems) for operations in a container shipping terminal. In: *Proceedings of the First Gulf Conference on Decision Support Systems*, Kuwait, 6–8 November, 2000. pp. 189–208.

- Roux, E.D., 1996. Storage capacity for import containers at seaports. Ph.D. Thesis, University of California, Berkeley.
- Taleb-Ibrahimi, M., 1993. Modeling and analysis of container storage in ports. Ph.D. Thesis, University of California, Berkeley.
- Taleb-Ibrahimi, M., Castilho, B.D., Daganzo, C.F., 1993. Storage space vs. handling work in container terminals. *Transportation Research B* 27 (1), 13–32.
- Thonemann, U.W., Brandeau, M.L., 1998. Optimal storage assignment policies for automated storage and retrieval systems with stochastic demands. *Management Science* 44 (1), 142–148.
- Zhang, C., 2000. Resource planning in container storage yards. Ph.D. Thesis, The Hong Kong University of Science and Technology.
- Zhang, C., Wan, Y.-w., Liu, J., Linn, R., 2002. Dynamic crane deployment in container storage yards. *Transportation Research B* 36 (6), 537–555.