

A DSS (Decision Support System) for Operations in a Container Terminal

Katta G. Murty^{1*}, Jiyin Liu², Yat-wah Wan², and Richard Linn³

¹ Department of IOE, University of Michigan,
Ann Arbor, Michigan 48109-2117, USA

² Department of IEEM, Hong Kong University of Science and Technology
Clear Water Bay, Kowloon, Hong Kong, China

³ Department of ISE, Florida International University,
Miami, Florida, USA

February 2002

revised January, May 2003

Abstract

We describe a variety of inter-related decisions made during daily operations at a container terminal. The ultimate goal of these decisions is to minimize the berthing time of vessels, the resources needed for handling the workload, the waiting time of customer trucks, and the congestion on the roads and at the storage blocks and docks inside the terminal; and to make the best use of the storage space. Given the scale and complexity of these decisions, it is essential to use decision support tools to make them. This paper reports on work on such decision support tools to develop a decision support system (DSS). We discuss the mathematical models and algorithms used in designing the DSS, the reasons for using these approaches, and some experimental results.

Key words: Decision support systems, models and algorithms, container terminals, approaches for handling uncertainty in decision making, port time (turnaround time) of vessels, quay crane rate, congestion, routing, reshuffling, storage space allocation, crane deployment, truck allocation to cranes, truck hiring plans.

*Author for correspondence. Department of IOE, University of Michigan, Ann Arbor, Michigan 48109-2117, USA. Phone: 734-763-3513; Fax: 734-764-3451; E-mail: katta_murty@umich.edu, Webpages: <http://www-personal.engin.umich.edu/~murty/>, <http://www.guthy.com/murty/index.html>

1 Introduction

Most of the cargo transported in ocean-going vessels around the world today can be classified into two types:

- bulk shipping of huge quantities of commodities like crude oil, coal, ore, grain, etc., which are shipped using specialized vessels called bulk carriers;
- containerized shipping in which a variety of goods are packed into standard-size steel containers that are shipped on vessels.

In this paper, we focus on containerized shipping. A **container terminal** (or **terminal** in short) in a port is the place where ocean-going vessels dock on berths and unload inbound (import) containers (empty or filled with cargo) and pick up outbound (export) containers. The terminals have storage yards for the temporary storage of these containers (Figure 1).

< INSERT Figure 1 around here. >

Containers, Storage Blocks, and Yard Cranes

Containers are steel boxes of dimensions (all measurements are in feet) $20 \times 8 \times 8.5$ or $20 \times 8 \times 9.5$ (called 20 ft. containers), or $40 \times 8 \times 8.5$ or $40 \times 8 \times 9.5$ (called 40 ft. containers), or specialized slightly larger size boxes (for example refrigerated containers for cargo that must be kept at specified cold temperatures during transit). In measuring terminal throughput and vessel capacity, etc., a unified unit, **TEU** (twenty foot equivalent unit), is commonly used, with each 40 ft. or larger container being counted as 2 TEUs.

The storage yard in a terminal is usually divided into rectangular shaped regions called **storage blocks** or **blocks**. A typical block has seven rows (or lanes) of spaces, six of which are used for storing containers in **stacks** or **columns**, and the seventh reserved for truck passing. Each row typically consists of over twenty 20 ft. container stacks stored lengthwise end to end. For storing a 40 ft. container stack, two 20 ft. stack spaces are used.

In each stack, containers are stored one on top of another. The placing of a container in a stack, or its retrieval from the stack, is carried out by huge cranes called yard cranes (**YCs**). The most commonly used yard cranes are **RTGCs (Rubber Tyred Gantry Cranes)** that move on rubber tyres. The RTGC stands on two rows of tyres and spans the seven rows of spaces of the block between the tyres (Figure 2). The bridge (top arm) of the RTGC has a spreader (container picking unit) that can travel across the width of the block between rows one to seven. The RTGC can move on its tyres along the length of the block. With these two motions, the RTGC can position its spreader to pick up or place down a container in any stack of the block, or on top of a truck in the truck passing row.

< INSERT Figure 2 around here. >

The height of an RTGC determines the height of each stack (i.e., the number of containers that can be stored vertically in a stack). Older models of RTGCs are five-level high RTGCs. This model can store only four containers in a stack, the 5th level is needed for container movement across the width of the block. Newer models are six-level high RTGCs. They can store five containers in a stack and use the sixth level for container movement.

Some blocks are served by fixed **RMGCs (Rail Mounted Gantry Cranes)** with 13 rows of spaces between their legs and a higher storage height (6 levels). The RMGCs are fixed to a block, but the RTGCs, which move on rubber-tyred wheels can be transferred from block to block offering greater flexibility. It is this flexibility for movement that makes the RTGCs the most commonly used container handling equipment in storage yard operations.

Outbound and Inbound Containers and Other Equipment Used in Terminal Operations

An **outbound container** is one that is being shipped by a customer of the terminal through this port to another destination port in the world. An **inbound container** is one that comes on a vessel from some other port in the world, to be unloaded in home port and kept in temporary storage until the customer for whom it is destined picks it up.

Customers bring outbound containers to the terminal, and take away inbound containers from the terminal, on their own trucks, which are called **XTs (External Trucks)**. Within the terminal itself, containers are moved using trucks known as **ITs (Internal Trucks or Stevedoring Tractors)**.

The unloading of containers from a vessel, or the loading of containers into a vessel, is carried out by huge cranes called **QCs (Quay Cranes)** (Figure 3).

< INSERT Figure 3 around here. >

The Hong Kong Port, the Setting for This Work

This paper describes the work being carried out to develop a DSS (computerized decision support system) for optimizing the operations of container terminals in Hong Kong.

Hong Kong is the busiest container port in the world. Container shipping is the lifeblood of Hong Kong's economy. Hong Kong is the principal entry port in the Southern China region, a region enjoying strong economic growth. The volume of containers transported through Hong Kong has been increasing by 10% yearly since 1986. The throughput in 2000 totaled 18.1 million TEUs. The throughput is estimated to reach 32.8 million TEUs by 2016 (Report of Hong Kong Port Development Board, 1998).

The intensity of container traffic in Hong Kong is estimated to be seven times that of New York, while the cramped space in Hong Kong container yards makes it very challenging to maintain high-quality service.

The container port in Hong Kong is currently run by four terminal operators, Hongkong International Terminals Limited, Modern Terminals Limited, CSX World Terminals Hong Kong Ltd., and COSCO-HIT Terminals (Hong Kong) Ltd. The daily operations of these companies are extremely complex. There are around 65 QCs, 200 RTGCs, 30 RMGCs, and 300 ITs congested into storage yards with the total area of 220 hectares in all these terminals put together. The storage yards are divided into blocks, each capable of storing about 600 TEUs if served by an RTGC and 1300 TEUs if served by an RMGC. The stacking density is about 720 TEUs per hectare, among the highest in the world. The 18 berths in Hong Kong add up to a 6-km long berth line. On average, all the terminals in Hong Kong together process 35 ocean-going vessels per day, with a container flow (both ways) of about 32,000 TEUs between the berths and the storage yard. There are close to 30,000 XTs bringing export containers to or picking up import containers from these terminals every day.

Given the scale and complexity of the Hong Kong container port, it is essential that computerized DSS tools be adopted. A DSS requires intelligent decision support models and algorithms and an efficient information system infrastructure to generate the data needed by the algorithms in a reliable fashion.

1.1 Hong Kong Terminal Operators' Experience with Decision Making in Terminal Operations

All the terminal operators have long experience in the field and are successful. They have invested huge sums in information infrastructures, though their investment in intelligent decision support systems remains relatively low. Most decision support functions in their existing information systems are based on experience and rules of thumb that evolved over decades. While such systems satisfy the current needs, the operators realize the necessity of introducing more intelligent decision support tools to meet future challenges and to deal with the complexity of operations more effectively. Modern container terminals in other parts of the world may also be facing similar situations. These situations have attracted operations researchers to study terminal operations and to develop decision support models and approaches to resolve operations problems. Some past studies are reviewed next.

1.2 Survey of Previous Work on Decision Making in Terminal Operations

Van Hee and Wijbrands [23] develop a decision support system for capacity planning of container terminals. Their system estimates the performance of a terminal under different operating capacities and operations rules. Their focus is more on the design phase of terminals, rather than on daily operations of terminals. For different aspects of storage and retrieval operations of containers in the yard, see [22] (space requirement), [21] (space requirement and congestion), [9] (space requirement and crane

capacity), [8] (re-marshaling strategy), and [28] (storage space allocation). The types of yard cranes adopted by terminals has led to different operational problems. RMGCs, moving only along a given block, generate mainly the container storage and retrieval problems. RTGCs, which have flexibility to move among blocks, add in a deployment problem ([27]). Straddle carriers, combining both stacking and transportation functions, further add in the routing problem that also exists for ITs ([10]). There are completely automated systems in which stacking cranes and automatic guided vehicles are computer controlled ([12]). Regarding quay-side problems, readers can refer to [1] and [25] for stowage of vessels, [7] and [19] for berth allocation, and [5] and [20] for the QC allocation problem. For a comprehensive literature review of operational problems in container terminals, see [13], [24], and [26].

Some of the work mentioned above may not be applicable to the terminals in Hong Kong. Terminals in different parts of the world may vary in size and may experience a variety of activity intensities. Such differences in characteristics induce differences in types of yard equipment and hence in storage and stacking strategies. An RMGC is a huge crane that accommodates blocks of more lanes and tiers than those of an RTGC, and hence it gives a higher storage capacity than does an RTGC for a comparable ground area. However, the retrieval time of containers, the complexity of operations, and the cost of the equipment increase with the size of the blocks. A straddle carrier combines the stacking and transportation functions in one. Its flexibility over an RTGC is offset by its lower storage capacity: a straddle carrier has a shorter stacking height and requires extra space between every lane of containers to accommodate its legs. A completely automated system is expensive and generally requires more area than that for an RTGC. Consequently, as Hong Kong operators accumulate experience, all of them have chosen the RTGC-block combination as their main operations mode, which is the mode we consider in this paper.

Our study is different from previous studies in how we handle uncertainty. We pinpoint the variation in the workload level over time, which seems to have been ignored in the existing literature. Refer to Section 2 for details.

The remainder of this paper is organized as follows. Section 2 describes how the randomness in the distribution of workload in the terminal over time effects the choice of models and algorithms for solving the decision problems in the terminal. Section 3 contains a brief introduction to the various yard operations in a terminal. Section 4 defines the key performance measures in evaluating the quality of work at a terminal. Section 5 describes the various decisions to be made in the daily operations and their relationships. Section 6 describes how the planning period for decision making has been determined. Sections 7 to 10 describe the decision support models and algorithms used for reaching decisions. Section 11 concludes the paper.

2 The Role of Uncertainty in Decision Making in Terminal Operations

A list of the most common decisions to be made in day-to-day operations in a container terminal is given later in Section 5. Among these, we have so far studied several resource allocation type decision problems that container operators in Hong Kong consider to be important. (These are explained in Section 5).

A critical factor for the success of a DSS is to ensure that the information infrastructure in the terminal can generate all the data needed by the algorithms implemented in the DSS.

The workload in the terminal in any specific time period can be measured by the number of containers processed during that period. This is the sum of four different quantities: Containers unloaded from vessels and stored in the storage yard; containers retrieved from the storage yard and loaded onto vessels; containers received from external customers and stored in the storage yard; and finally containers picked up by external customers from the storage yard. Each of these quantities for any future time period is subject to many uncertainties in terminal operations (arrival and departure times of vessels depend on uncertainties posed by weather; arrivals of customers' trucks at the terminal gates are subject to uncertain traffic conditions on Hong Kong roads, which face congestion at certain times).

Even after obtaining a reliable estimate of the total workload for a time period, we have noticed that a solution to a decision problem from a deterministic optimization model based on the total workload estimate performs well when implemented, only if this workload is evenly distributed over the time period. In reality, the workload in terminals quite often tends to be unevenly distributed over time (when a vessel's processing is nearing completion and its departure time is approaching, the workload undergoes a discrete change in a very short time interval). For this reason, we have found it necessary to develop approaches that react well to momentary changes in the workload level over time. Our choice of the decision problem model and the algorithm for solving it has to accommodate uncertainty in working conditions, which is always there.

The Special Nature of Terminal Work

All workloads in container terminals have the following special features:

- (1) As described above, the distribution of the workload over time tends to be uneven due to many uncertainties in weather, traffic conditions on roads, etc.
- (2) It is not always possible to control the order in which work comes, or the exact time of its arrival.
- (3) Work has to be carried out when it arrives; its processing cannot be delayed under ordinary circumstances.

It is thus essential to choose models for problems and algorithms that take all the above features into account and that are practically intuitive and easy to implement and maintain by the user's operations

and information system teams.

3 The Yard Operations

3.1 The Functions of a Container Terminal

A container terminal serves as an interface between ocean and land transportation. Its main functions are:

- to receive outbound containers from shippers for loading onto vessels and to unload inbound containers from vessels for picking up by consignees;
- temporary storage of containers between their passages on ocean and land transportation.

3.2 The Flows of Outbound and Inbound Containers

Outbound containers being brought in by customer XTs enter the terminal through the **TG (Terminal Gatehouse)** where each container and its documentation are checked. The TG then instructs the XT to proceed to the storage block where the container will be stored until the vessel into which it will be loaded arrives. The RTGC working at that block removes the container from the XT and puts it in its storage position. When the time to load comes, the RTGC retrieves the container from the stored position, puts it on an IT, which takes it to a QC for loading onto the vessel. The flow of outbound containers is represented in Figure 4. The flow of inbound containers has the reverse features as indicated in Figure 5.

< INSERT Figure 4 around here. >

< INSERT Figure 5 around here. >

3.3 How is a Containerized Vessel Organized ?

On a vessel, containers are stacked lengthwise along the length of the vessel. Along its length, the vessel is divided into segments, known as **hatches** or **holds** or **bays**. Each of these can accommodate one 40 ft or two 20 ft containers along its length. In each hatch, up to 20 containers may be stacked in a row across the width of a large vessel. The number of hatches in a vessel may be over 20 depending on its total length. Some of the big vessels may carry over 7000 TEUs. A vessel may call on 5 to 10 ports in a voyage. For quick unloading and loading, they usually assign containers to hatches according to ports of call. When the vessel is docked on the berth, several QCs work on it simultaneously, with each QC

working on a separate hatch. A QC itself is wider than a hatch. Different QCs cannot work on adjacent hatches at the same time.

A QC has four legs arranged in two rows of two legs each. The space between the two rows is divided into five truck lanes. While the QC is working on a vessel, it has a set of ITs serving it (either taking away the containers unloaded by the QC, or bringing containers to be loaded into the vessel by the QC). The ITs serving a QC always line up in one of the lanes between the two rows of legs of the QC. Each of these five lanes is assigned exclusively for the use of the ITs serving one of the QCs working on the vessel for the sake of safety and smooth operation. The number of QCs that can work on a vessel is usually limited by the number of these lanes. Normally 3 or 4 QCs work on a vessel simultaneously.

The unloading and loading sequence of containers is usually determined by a special algorithm to make sure that the docked vessel's balance is not affected while it is on the berth.

3.4 RTGC Operations

The RTGCs are expensive equipment (each costs about US\$850,000 at today's prices) whose proper utilization is very critical to the efficiency of container handling operations in the storage yard.

The RTGCs move containers from ITs or XTs and put them in their storage locations, or they retrieve stored containers and put them on ITs or XTs. ITs and XTs that arrive at a block to deliver or pick up a container queue in the truck passing lane of the block until the RTGC working in that block can serve them. Thus, if RTGCs are not efficient in their work, there may be truck congestion on the road near the block and inside the block itself. Also, if the RTGC holds up the ITs serving a QC working on a vessel, the QC may have to wait, resulting in a delay in unloading or loading the vessel.

There are two types of container retrieval operations of an RTGC:

- A **productive move**: When a container is moved directly from its storage location to an IT or XT waiting to pick it up, this is a productive move. For example, retrieving the top container, *A*, from the stack of four stored containers shown in Figure 6 is a productive move.
- An **unproductive or reshuffling move**: When the movement of a container, say *A*, becomes necessary to retrieve another container stored underneath it in the same stack, this is an unproductive move. In this reshuffling move, container *A* is placed in another stack in the same bay and may be put back in the original stack after the desired container is retrieved. For example, to retrieve container *C* in Figure 6, containers *A* and *B*, stored above *C*, have to be moved away first in reshuffling moves.

< INSERT Figure 6 around here. >

The number of reshuffling moves depends on the strategy used for allocating storage spaces to arriving containers.

Repositioning an RTGC from one block to another is slow. If two blocks are adjacent by width (as B_1 and B_2 in Figure 7) the RTGC can move between them without any turning motion. To move to an adjacent block by length (as B_1 and B_3 in Figure 7), the RTGC has to come to the road on one end of the block, make a 90° turn of its wheels, then move on the road parallel to the width line to the correct position for the adjacent block, make another 90° turn, and enter that block. These 90° turns take extra time and also obstruct traffic on the road for the time that the RTGC is on the road.

< INSERT Figure 7 around here. >

4 Key Performance Measures of a Container Terminal

Container terminals work under multiple operational objectives. The most critical performance measure for rating the terminals is the **ship turnaround time** (also called the **port time of the ship**), which is the average time the terminal takes to unload and load a docked vessel. This important objective function must be minimized.

Closely related to the ship turnaround time, another important measure is the average **QC rate**, which is the quay cranes' throughput measure during a period, given by

$$\text{QC rate} = \frac{\text{no. of containers unloaded, loaded}}{\text{total no. of QC hours of all QCs that worked}}.$$

In Hong Kong, as well as in some other places, terminals charge shipping companies for every container loaded into or unloaded from the vessel. Thus, the QC rate must be maximized.

Note that these two measures are used internally by a terminal to calibrate its own performance. In the same terminal, with the same equipment, as the QC rate goes up, the ship turnaround time goes down, and vice versa. However, if the terminal acquires new QCs and puts them in use, its ship turnaround time may go down even without the QC rate changing.

The performance of Hong Kong's terminals is shown in the following comparisons: In 2000, 6.6 million TEU were handled by two Hong Kong terminals, Hongkong International Terminals Limited (HIT) and COSCO-HIT, using 122 hectares of land. Around the same time (2001), Delta Terminal in the Netherlands, the busiest terminal in Europe, handled 2.5 million TEU, using 280 hectares of land, and the port of Long Beach, one of the busiest container ports in the United States, handled 4.6 million TEU, using 295 hectares of land.

On average, a QC in Hong Kong can handle over 30 containers per hour. Also, terminals in Hong Kong seem to be always ready to invest in acquiring adequate numbers of QCs to keep their ship

turnaround times short. This contributes to the belief that terminals in Hong Kong are among the most efficient in the world.

As we will discuss in Section 5, the daily operations of terminals require a lot of inter-related decisions. The very large number of decisions involved, the multi-objective nature of the problem, the uncertainty, and the complexity of the decisions make it impossible to derive answers to all the decisions by solving a single mathematical model. The only practical way to obtain reasonable answers to these decision problems is to study each of them separately in a hierarchical fashion. We found the technique called hierarchical decomposition with substitute objective functions for each stage developed in [18] very useful in handling these decision making problems. This is the basis for our DSS. Generally, the substitute objective functions are consistent with the key performance measure, i.e., they help reduce the ship turnaround time and increase the QC rate.

A Quantitative Measure of Congestion on the Roads Inside the Terminal

With so many trucks operating, the roads in the terminal may get congested. Congestion slows the trucks from carrying out their operation of transporting containers from one location to another. This has an undesirable effect on truck utilization and, more importantly, on the time taken to process the vessels. Hence, another important measure of performance is **congestion on the roads** in the terminal, which must be minimized.

Congestion is an easily recognized intuitive concept in city traffic. Usually in the street network of a city, there may be arcs of different capacities. The average speed and the standard deviations of speeds of vehicles on different arcs may be very different; and there may be different types of vehicles flowing on the arcs at the same time. That is why developing a good measure of congestion in city traffic is a difficult task.

The special nature of the road network inside a terminal: Inside a terminal, however, both the road network and the traffic on it have a special nature. All the roads have the same traffic capacity and the traffic on them consists of mostly trucks transporting containers. The average speeds and the standard deviations or range widths of speeds on all the roads are the same. It is thus comparatively easier to develop a specialized quantitative measure for congestion on the roads inside a terminal in terms of the decision variables that can be controlled in terminal operations.

We represent the gate, the berths, the blocks, and all road intersections as nodes in a network. A road segment is a portion of a road joining two such nodes. If the road is two-way, each direction of it is considered to be a separate road segment. Thus, each road segment can be traveled in only one direction and can be represented by a directed arc joining the two nodes on it. This leads to the representation of the road system inside the terminal as a directed network, which we denote by G .

We will measure flows on arcs in G in units of trucks traveling through those arcs per unit time. Let

s denote the number of arcs in G , and for $k = 1$ to s , let

f_k = flow (number of trucks traveling/unit time) on the k th arc.

As the number of containers handled by the terminal increases, flows on the arcs in G are expected to increase. We are interested in a measure of congestion in the whole road network, G , rather than on particular arcs. Define

$$\theta = \text{Max} \{f_k : k = 1, \dots, s\}, \mu = \text{Min} \{f_k : k = 1, \dots, s\}.$$

Since congestion is associated with high values of flows on arcs, given the special nature of our network G , θ is a natural measure of congestion in G . In reality, it is not only the value of θ , but the number of arcs with such high flow values that characterizes the quality of the traffic situation inside the terminal. Using only θ as the measure of congestion to minimize provides no control on how many arcs in G are subject to such high flow values.

For a given volume of containers handled, we can expect the best traffic situation to prevail if the flows along the various road segments are as close to each other as possible, because in this case the necessary traffic load is distributed among all the road segments as much as possible instead of being concentrated on just a few. Another measure of congestion in the road system inside the terminal is therefore $\theta - \mu$, and minimizing $\theta - \mu$ may lead to a better overall traffic situation than minimizing only θ .

We will use θ , $\theta - \mu$ as the measures of congestion to minimize. We will also consider the idea of adopting schemes that help distribute the traffic inside the terminal equally along the various road segments in order to alleviate the congestion.

Other Important Measures of Performance to Optimize

Some of the other measures of performance to be optimized are given below.

- The average waiting time of XTs that come to the terminal to deliver outbound containers or pickup inbound containers (to be minimized).
- The average waiting time of the ITs in queues at the QC and RTGC waiting to be serviced (to be minimized).
- The waiting time of the QC waiting for an IT (to be minimized).
- The volume of unproductive moves in the storage yard (to be minimized).
- The total number of ITs used in the various shifts each day (to be minimized).

Optimizing all these performance measures requires good resource allocation decisions, i.e., allocation and scheduling of berths and QCs to docked vessels, storage spaces to containers, RTGC deployment

among blocks, IT allocation to QC, and allocation of routes to traveling XTs and ITs, in order to achieve a smooth and orderly flow of containers between the gatehouse, the storage yard, and the quayside.

5 Decisions to be Made in Daily Operations

There are a variety of decisions to be made each day. These are listed below.

- D1. Allocation of berths to arriving vessels:** Allocate berths to arriving vessels to minimize ship waiting time, port cost, and the dissatisfaction felt by vessel's captains about berthing order; and to maximize utilization of berths and QCs.
- D2. Allocation of QCs to docked vessels:** Develop a plan to allocate and schedule QCs to work on docked vessels. This decision influences the turnaround time of the vessels and the throughput rate of the terminal.
- D3. Appointment times to XTs:** Ideally, all customers book the time to deliver outbound containers, or to come to pick up their inbound containers, by calling beforehand and making appointments. Develop simple online decision rules to generate these appointment times when they call, to help to minimize XT waiting time and congestion in the road network.
- D4. Route trucks:** Develop an algorithm to route XTs and ITs inside the terminal to minimize congestion on the roads.
- D5. Dispatch policy at the TG and the dock:** Develop an online procedure for dispatching XTs arriving at the TG with an outbound container and ITs carrying an inbound container at the berth to the blocks in the storage yard, to minimize congestion at the blocks and congestion on the roads.
- D6. Storage space assignment:** Assign storage space to arriving containers to minimize reshuffling volume and congestion on the roads.
- D7. RTGC deployment:** Determine how many RTGCs should work in each block and when to move an RTGC from one block to another. This decision influences the port time of vessels and the waiting times of QCs, XTs, and ITs.
- D8. IT allocation to QC:** Determine how many ITs to allocate to each working QC to minimize the vessel turnaround time and the waiting times of ITs and the QC.
- D9. Optimal IT hiring plans:** Estimate IT requirements in each half-hour interval of the day and develop a plan to hire ITs to meet these requirements, to minimize the total number of ITs hired each day and to maximize IT utilization.

The decisions pertaining to containers have to be made for each container as it goes through different stages on its path through the terminal. The operations of terminals are triggered by vessels. A terminal first receives the information on a vessel a few weeks before its actual arrival time. The terminal makes temporary allocations of berth (D1) and of QCs (D2) to the vessel based on its projected workload. XTs with appointment times (D3) start to bring in export containers for the vessel a few days before the actual vessel arrival. These containers are dispatched for storage in blocks (D5), depending on the situation in the yard at the time of their arrivals. A container allotted to a block is then assigned to a storage stack (D6) where it requires the minimum number of reshuffles.

The actual workload and the likely vessel arrival time are known about a day or so before the actual vessel arrival. The actual berth allocation (D1) and QC allocation (D2) are determined based on this information. With more updated total workload requirements, the terminal determines the total number of ITs (D9) to be hired on the following day and formulates the preliminary plan of assigning ITs to QC (D8). D5 also has an influence on D8 and D9.

During the actual operations, the terminal makes real-time decisions on the allocation of ITs to QC (D8), the movement of yard cranes among blocks (D7), and the routing of ITs between QC and blocks (D4). All the other planned decisions as described above are also subject to changes according to real-time activities at the terminal.

The development of a DSS to make all these decisions optimally is an ongoing effort. So far, we have investigated strategies for making decisions D4 to D9 because these decisions involve resource planning issues in the storage yard, which are issues of primary interest to the terminals in Hong Kong.

6 The Planning Horizon and Input Data for Decision Making

Container terminals work three 8-hour shifts around the clock, every day. It is convenient to make the planning horizon for decision making equal to the time interval of a shift, or less.

There are two important considerations for selecting the appropriate length of the planning period. One is that it should be possible to estimate the total workload in the period with reasonable accuracy. Another is that this total workload should be distributed as uniformly as possible over time during this period. Operators have found that a four-hour period (i.e., half of a shift) comes closest to meeting both these considerations adequately. With shorter time periods, the accuracy of the estimate of total workloads is poor. With longer time periods, the distribution of the workload over time tends to be uneven. Hence, terminals have found it convenient to organize their work with a four-hour period as the planning period for decision making. A day is divided into six planning periods: 00:00 – 4:00, 4:00 – 8:00, 8:00 – 12:00, 12:00 – 16:00, 16:00 – 20:00, 20:00 – 24:00. Activities spanning across more than one time period have their associated quantities, such as workload and equipment demand, shared across the time periods according to the amount of time that the activity requires. Towards the end of

each period, the terminals use the latest information available for the following period to formulate a plan for operations in that period.

The terminals in Hong Kong have advanced information systems that keep real-time information on container flows and the status of various resources in them. This information is not only essential for tracking the containers for security and customs reasons, it also provides reliable input data for decision making. While the data needed for each decision problem will be mentioned in the section for that decision, here we give a brief overview of the types of data and their sources.

There are two types of input data for the decisions: static and dynamic. Static data are the system parameters that (generally) hold constant throughout any planning horizon, e.g., the road network in the yard, the number and capacities of the blocks, the numbers of quay cranes and yard cranes, etc. They are readily accessible from database records whenever needed. Dynamic data reflect the status (values) of variables that change with operations, e.g., the current positions of equipment, the number of containers stored in each block, and the storage position of each container, etc. These are all updated in real time and their current values are available from the information system.

Changes to the dynamic data are triggered by the arrivals and departures of containers into/from the terminal. On the gate side, the times of container arrivals and departures are determined when the customers make appointments for dropping off or picking up the containers. On the quayside, long before a vessel arrives, information on the estimated arrival time, the number of containers to be unloaded at a Hong Kong terminal and their exact positions on the vessel is sent electronically to the terminal by the carrier. The carrier also gives the terminal details on the containers to be loaded into the vessel and the hatch into which each of them should go, based on its destination port, without restricting the specific position in the hatch for every individual container. Based on all the carriers' information, the terminals operations planner makes an unloading/loading work plan (the sequences of unloading and loading containers), using a special-purpose software. The vessel's arrival time and this unloading/loading plan determine the arrival times of inbound containers from the vessel and the departure times of the outbound containers to the vessel. With modern communication tools, fairly accurate vessel-related data are available in advance for planning decisions. The data on the gate side are not always known in advance precisely because some customers do not make appointments before dropping off or picking up containers. For planning purposes, we estimate the number of such container arrivals and departures in future periods based on the distributions of past data.

7 Storage Space Assignment, Dispatching Policy at the TG and the Berth, and the Routing of Trucks in the Storage Yard

Since storage spaces are sources and destinations for truck travel within the terminal, the three decisions on storage space assignment, the policy to be followed at the TG and the berth for dispatching XTs and

ITs, and the routing of trucks are highly inter-related, and we study them together.

The planning period for these decisions is a four-hour period (the beginning or the ending half of a shift) as discussed in Section 6. The input data needed for the models consists of: the road network, G (discussed in Section 4), the appointment time database (a list of customers who have been given appointments during the planning horizon and a list of the outbound containers they are expected to drop off, or a list of the inbound containers in storage they are expected to pick up), the vessel schedule data (the arrival time, workload), the vessel unloading and loading plans, and other data as listed below.

- $b =$ Number of berths in the terminal.
- $m =$ Number of blocks in the yard.
- $k_i =$ Storage capacity of block i (in number of containers), $i = 1$ to m .
- $ES_i(p) =$ Number of outbound containers stored in block i to be retrieved during period p for loading onto vessels, $i = 1$ to m .
- $ES_i^d(p) =$ Number of containers among $ES_i(p)$ that have to go to berth d , $d = 1$ to b .
- $ES(p) = \sum_{i=1}^m ES_i(p)$, total number of stored outbound containers to be retrieved during period p for loading onto vessels.
- $IS_i(p) =$ Number of inbound containers in storage in block i expected to be retrieved during period p for pickup by customers, $i = 1$ to m .
- $IS(p) = \sum_{i=1}^m IS_i(p) =$ total number of inbound containers to be picked up by customers during period p .
- $IM(p) =$ Total number of inbound containers expected to be unloaded from vessels during period p .
- $IM^d(p) =$ Number of inbound containers to be unloaded from vessels docked at berth d , $d = 1$ to b . We have $\sum_{d=1}^b IM^d(p) = IM(p)$.
- $EX(p) =$ Number of outbound containers expected to be brought in for storage during period p .
- $XIT_i(t) =$ Number of XTs and ITs waiting in block i for receiving service from the RTGC of block i , at the time of observation t .

7.1 Storage Space Allocation Decisions for the Next Period, $p + 1$

Let p be the current period. We discuss procedures for making storage allocation decisions for containers arriving in the next period, $p + 1$.

Each of the $IM(p + 1) + EX(p + 1)$ new containers arriving at the terminal during period $p + 1$ has to be assigned a specific storage space in the yard for storage. Ideally, the assignment policy could have an impact on the congestion in the road network, at the blocks, and on the total volume of reshuffling, all of which must be minimized.

There are a large number of container storage positions in the yard, some of which are already

occupied by containers in storage at the beginning of this period. Determining a specific open storage position for each of the newly arriving $IM(p+1) + EX(p+1)$ containers leads to a huge mathematical model that will be impractical to solve. In [14], this storage space assignment decision has been divided into three stages.

Stage 1: Block Assignment: Determine the **container quota** for block i , which is the number of containers among the arriving $EX(p+1) + IM(p+1)$ that will be stored in block i . This determination leads to the following decision variables for this stage:

$$x_i(p+1) = \text{Container quota number for block } i \text{ during period } (p+1) = \text{the total number of outbound and inbound containers arriving at the terminal in this period to be stored in block } i, i = 1 \text{ to } m.$$

Stage 1 determines only the container quota numbers for the blocks, not the identity of which containers will be stored in each block. This will be determined by the dispatching policy discussed in Stage 2.

Stage 2: Dispatching policy at the TG and the docks: Determine a policy that TG and dock supervisors will follow to dispatch XTs and ITs to the blocks in the storage yard, so that in the end each block has as many containers as determined in the quota in Stage 1, while minimizing congestion at the blocks and on the roads.

The reason for a good dispatching policy can be explained this way. Suppose the container quota for block 1 is 13. If the TG sends 13 of the arriving XTs consecutively one after the other to block 1, then serious congestion will take place at block 1. The dispatching policy should ensure that consecutive trucks in the arriving stream are dispatched to different blocks that are widely dispersed in the storage yard, so that each block has adequate time to process the truck reaching it before the next truck sent to this block arrives there. Also, the dispatching policy should distribute these trucks in all directions to ensure that the truck traffic on the roads is evenly distributed in all directions.

Stage 3: Storage Position Assignment: For each container arriving at block i for storage, determine the optimal available position in the block for storing it to minimize the incidence of reshuffling that may arise while retrieving the container later.

7.2 A Multicommodity Network Flow Approach to the Stage 1 Storage Problem (Block Assignment)

Given $EX(p+1), IM^d(p+1), ES_i^d(p+1), IS_i(p+1)$, the problem of determining the values of the decision variables, $x_i(p+1)$ for $i = 1$ to m , to minimize congestion on the road network (measured either by θ , or $\theta - \mu$) has been formulated in [14] as a multicommodity network flow problem on the network G . This problem can easily be solved, since it is a linear program, by using a commercial software package

like CPLEX. This problem is solely based on the estimated total workload during period $(p + 1)$. We found that the solution to this problem works well in practice as long as the workload is distributed more or less uniformly over time. The solution is poor during periods in which the workload distribution over time is uneven.

In [14], an alternative approach to this problem that adapts quite well to momentary changes in the workload level is developed. We discuss that approach next.

7.3 A Fill-Ratio Equalization Approach to the Stage 1 Storage Problem (Block Assignment)

The alternative approach developed in [14] for block assignment is based on the following hypothesis.

Hypothesis: Assuming that the dispatching policy developed in Stage 2 disperses consecutive trucks in the arriving stream all around the storage yard, traffic congestion in the terminal will be at its minimum (i.e., the traffic volume will tend to be equally distributed on all the road segments and the blocks in the terminal) if the fill-ratios of all the blocks are maintained to be nearly equal.

Applying this hypothesis leads to the following very simple procedure for determining the Stage 1 decision variables, $x_i(p + 1)$ for $i = 1$ to m for next period $p + 1$. Here,

- $x_i(p)$ is the container quota numbers for block i in period p ,
- $F_i(p)$ is the fill-ratio of block i at the beginning of period p ,
- $\rho(p)$ is the fill ratio for the whole storage yard (including all the blocks) at the beginning of period p ,
- $G_i(p)$ is the number of containers stored in block i at the beginning of period p ; for $i = 1$ to m .

Procedure for Stage 1 (Container Quota Number Determination) for Period $p + 1$

Here is a summary to help us understand the calculations in the procedure. Step 1 uses the data from the current period, p , to compute the fill-ratios of various blocks at the beginning of the next period $p + 1$. Step 2 uses these, the number of new containers arriving, and the number of stored containers expected to be retrieved during period $p + 1$ to compute the container quota numbers for the various blocks in this period to equalize the fill-ratios in the various blocks as much as possible by the end of the period (i.e., by the beginning of period $p + 2$).

Step 1. Compute the fill-ratios of the blocks at the beginning of period $p + 1$:

$$F_i(p + 1) = (G_i(p + 1)/k_i) = (F_i(p)k_i - ES_i(p) - IS_i(p) + x_i(p))/k_i.$$

Step 2. Computing the container quota numbers $x_i(p+1)$ for blocks in period $p+1$:

The fill ratio in the whole storage yard at beginning of period $p+2$ (i.e., the end of period $p+1$) will be

$$\rho(p+2) = \frac{\sum_{i=1}^m G_i(p+1) - (ES(p+1) + IS(p+1)) + EX(p+1) + IM(p+1)}{\sum_{i=1}^m k_i}.$$

Therefore, $\rho(p+2)$ can be computed from the available data at the beginning of period $p+1$. The fill-ratio equalization approach determines the container quota numbers for blocks $x_i(p+1)$ so that the fill ratios for all the blocks come as close to $\rho(p+2)$ as possible by the end of period $p+1$. This leads to the following linear programming (LP) model (the variables in this LP are $x_i(p+1), u_i^+, u_i^-$; all other symbols denote known data elements):

$$\begin{aligned} & \text{Minimize} && \sum_{i=1}^m (u_i^+ + u_i^-) \\ & \text{subject to} && \\ & [G_i(p+1) - (ES_i(p+1) + IS_i(p+1))] + x_i(p+1) + u_i^+ - u_i^- = k_i \rho(p+2), \quad \text{all } i \\ & \sum_{i=1}^m x_i(p+1) = EX(p+1) + IM(p+1) \\ & \text{all} && x_i(p+1), u_i^+, u_i^- \geq 0. \end{aligned}$$

The optimal solution of this LP model leads to the following scheme:

Let $c(p+1) = EX(p+1) + IM(p+1)$ initially. This is the number of new containers arriving in period $p+1$ to be allotted to the blocks.

Rearrange the blocks in increasing order of $G_i(p+1) - (ES_i(p+1) + IS_i(p+1))$, i.e., after rearrangement, this quantity will be increasing in the order $i = 1$ to m . In this order $i = 1$ to m , carry out

1. $x_i(p+1) = \min\{\max\{0, [k_i \rho(p+2)] - [G_i(p+1) - (ES_i(p+1) + IS_i(p+1))]\}, \text{current value of } c(p+1)\}$.
2. New value of $c(p+1) = \max\{0, \text{current value of } c(p+1) - x_i(p+1)\}$.
3. If $i < m$, increment i by one and go back to 1.

This provides the broad outline of the procedure for determining $x_i(p+1)$. Container terminals do make heuristic modifications to this general procedure according to their preferences and any special circumstances.

We provide a numerical example to illustrate Step 2. In this example, we consider a storage yard consisting of only nine blocks. For blocks $i = 1$ to 9, data on $k_i, A_i = G_i(p+1) - (ES_i(p+1) + IS_i(p+1))$ are given in Table 1.

In this numerical example, $EX(p+1) + IM(p+1) = 1040$. The fill ratio in the whole storage yard at the end of period $p+1$ will be

$$\rho(p+2) = (2570 + 1040)/4800 = 0.752.$$

The blocks are already in increasing order of $A_i, i = 1$ to 9. The computed values of the quota numbers, $x_i(p+1)$, are given in Table 1.

This algorithm assigns 1040 containers arriving for storage in period $p+1$ to blocks $i = 1$ to 9 in that order, bringing the expected number of stored containers in this block at the end of period $p+1$ to $k_i\rho(p+2)$, until all these containers are assigned. Notice that the above procedure only determines the values of the decision variables $x_i(p+1), i = 1$ to m ; it does not assign each specific arriving container to a storage block. Once the period begins, arriving containers are assigned to blocks by the following online procedure that helps to even out the traffic to the blocks and along the roads.

< INSERT Table 1 around here. >

7.4 Online Dispatching Procedure for Assigning Containers to Storage Blocks During a Period (Stage 2)

Sending too many XTs or ITs to a block in a short period of time creates congestion at that block. To avoid this, the following **truck dispatching policy** is used at the TG and the berths.

For outbound and inbound containers: When a new container arrives at time point t during the period, either at the TG or from a QC at a berth, the values of $XIT_i(t)$ for those blocks with the current value of $x_i(p+1) > 0$ at that moment are examined, and the container is assigned to the block with minimum $XIT_i(t)$. If there is a tie, the container is assigned to the block with a higher $x_i(p+1)$. Any further tie is broken arbitrarily. After the container is assigned, the value of $x_i(p+1)$ is updated for that i , i.e., the new value of that $x_i(p+1) =$ the current value of that $x_i(p+1) - 1$ for the assigned block i . The values of $XIT_i(t)$ are updated in real time.

The terminals also make other heuristic modifications to this online block assignment procedure for storage of arriving containers according to their preferences.

To implement this dispatching policy, it is necessary to set up a system that monitors how many trucks are waiting in each block and to convey this information to the TG and to the berth. In fact, each terminal operator in Hong Kong has a centralized information system with user terminals at the TG and inside the operator cabinet on each YC and QC. Every container moving between a truck and a QC and between a truck and a block is monitored and its location is fed back to the system. Hence, the real-time number of trucks waiting in and moving towards a block is readily available from the system. The values of $x_i(p+1)$ in the system are also updated in real time. Therefore, it is convenient to implement and

embed the truck dispatching policy in the information system. When a container arrives, the operator at the TG or QC simply needs to inform the system about its arrival; the system will use the dispatching policy to decide to which storage block the container should go based on the real-time information and convey the decision to the operator who then directs the XT or IT driver accordingly.

7.5 Allocating Storage Spaces to Containers Within Blocks, the Stage 3 Problem

Since we cannot control the order in which containers arrive for storage in a block and cannot hold arrived containers before storage, only online algorithms assigning positions to containers as they arrive in the block are suitable for implementation.

The assignment of a storage position to a container arriving in this block mainly impacts the total amount of reshuffling in this block. These assignments have to be made with the objective of minimizing the total amount of reshuffling in this block.

A block consists of a number of stacks for storing containers. Some of these stacks may already have some containers stored in them. Others may be empty.

For each arriving container, we can make an estimate of the time when it will be retrieved from storage.

For Inbound Containers in Storage: Using the data on appointment times given to customers, an estimate of the retrieval time of inbound containers in storage can be obtained.

For Outbound Containers in Storage: The expected time to retrieve an outbound container in storage to load it into its vessel is obtained using the data on vessel arrival times and the unloading and loading sequences for them.

The storage position for an arriving container is determined to minimize the number of reshuffling moves that it may create. With tens of thousands of storage location decisions to make in a day, the DSS uses a simple decision tool, the **reshuffle index**. Define the reshuffle index of an arriving container, A , for a stack in the block that has space for storing it to be

$$\gamma = \begin{cases} 0, & \text{if the stack is empty when } A \text{ arrives for storage,} \\ \text{the number of stored containers in the stack with estimated retrieval times before} \\ \text{that of } A, & \text{otherwise.} \end{cases}$$

The reshuffle index measures the number of times that container A may have to be reshuffled if it is put in storage in this stack at this time. It does not consider the possible effect on reshuffling of containers coming at later times.

The problem of storing the containers assigned to this block with the objective function of minimizing the reshuffle indices of all the containers in their storage positions is analogous to a bin packing problem

with the stacks playing the role of bins that can hold 5 (or 4, depending on the height of the RTGC serving this block) containers. The main difference between this problem and a traditional bin packing problem is that the order of arrivals of containers is dynamic and uncontrollable, and that the contribution of a container to the objective function varies with the stack in which it is stored. Under this analogy, we implement the solution to this problem corresponding to the best fit solution in bin packing. This leads to the online procedure discussed below.

7.6 Online Procedure for Assigning Storage Positions to Containers Arriving for Storage in a Block

When the container arrives, find the stack in the block with an empty space that has the smallest reshuffle index for it among all such stacks, breaking ties in favor of the stack with the maximum number of containers stored already. Store the container in the top position of that stack.

This procedure has found acceptance from the terminals as it is easy to implement with their information infrastructures.

Several other strategies for this decision have been proposed and analyzed in [26].

8 Optimal Deployment of RTGCs Among the Blocks

The RTGCs are a critical resource whose performance in the storage yard influences the waiting times of the XTs, ITs, and QCs and, in turn, the port time of vessels. As the workload in the various storage blocks changes over time, deployment of RTGCs among storage blocks in order to provide more RTGCs to blocks with heavier workloads is an extremely important issue in terminal operations management.

The planning horizon for the RTGC deployment problem is also a four-hour period (the beginning or ending half of a shift) as discussed in Section 6.

Because of the limited size of a block, there can be at most two RTGCs working in a block at any time. One of the big terminals in Hong Kong has 70 storage blocks that need RTGCs and, on a typical day, has around 95 RTGCs in working order to serve these blocks. In a typical planning period, all the 70 blocks will have some activity that requires an RTGC. It is thus reasonable to assume that every block needs at least one RTGC in every planning period for at least part of the period. Some blocks with a lot of activities may actually need two RTGCs to support the work in them. Since there are not enough RTGCs to station two in every block that needs them, it is necessary to move them from blocks in which the workload is low to those where it is high.

Since RTGC movement from one block to another is slow and since this movement results in not only the loss of productive time of the moving RTGC, but also obstructs traffic on the roads between the blocks, the terminals have adopted the policy that an RTGC can move from one block to another

at most once in each period. Such a move can occur at any time within the period.

As a period is four hours long, the capacity of an RTGC in a period can be stated as 240 RTGC minutes. From observations in the storage yard, we determine

$\tau =$ the average time in minutes that it takes an RTGC to either stack a container in a column, or to retrieve the stored top container from a column.

$\eta =$ reshuffling frequency per productive move. This is the number of containers reshuffled on an average per container retrieved from storage, or the average number of unproductive reshuffling moves per productive move. In terminals in Hong Kong, η has been observed to be 1.17 reshuffling moves for each container retrieved.

Given these observations, and the expected number of containers to be stacked in and retrieved from a block during a period, it is possible to measure the RTGC workload in that block in that period in terms of RTGC minutes. The RTGC deployment models use this workload data for deployment decisions.

It is very important that the RTGC workload in each block in a period be finished in that period itself, otherwise this may result in excessive XT waiting time or delay vessel departure times. An unfinished RTGC workload will be carried over to the next period. This will have to be finished during the first part of the next period. We call the unfinished RTGC workload in a block in a period the **delayed RTGC workload**. The aim of RTGC deployment is to minimize the total delayed RTGC workload in each period.

Besides the data elements τ and η mentioned above, RTGC deployment models for a period, p , need $ES_i(p)$, $IS_i(p)$ defined in Section 7, $x_i(p)$ defined and computed in Sections 7.1 and 7.3, and the following data:

$m =$ Number of blocks in the yard.

$n =$ Total number of working RTGCs in the yard.

$r_i(p) =$ Number of RTGCs stationed in block i at the beginning of period p , $i = 1$ to m .

$c_{ij} =$ Time in minutes that it takes an RTGC to move from block i to block j ; for $i, j = 1, \dots, m$, and $i \neq j$. c_{ij} is set to ∞ or a large positive number if the terminal wants to forbid the movement of an RTGC from block i to block j for any reason.

From this data, we can compute the expected RTGC workload in block i in period p , measured in RTGC minutes, to be

$$w_i(p) = (ES_i(p) + IS_i(p))(1 + \eta)\tau + x_i(p)\tau.$$

However, what makes the RTGC deployment problem difficult is the following important fact.

Fact 1 about RTGC workloads: If all the work of an RTGC in a period is available to be carried

out at the beginning of the period, then the RTGC can quickly finish it and move. Unfortunately, the RTGC workload in a period is usually not all available at the beginning of the period. The need for an RTGC occurs only when the trucks arrive in its block for its service. Thus, the work to be carried out by an RTGC is spread over the period as trucks arrive.

In [14], the RTGC deployment problem for a single period and for minimizing the total delayed workload has been modeled as an MIP (Mixed Integer Programming) problem that can be solved quite efficiently by commercial software packages like CPLEX. [26] and [27] modeled the same problem for six periods (a whole day) as a much larger MIP and developed a special approach for solving the problem efficiently based on Lagrangian relaxation.

One shortcoming of all these models is that they ignore Fact 1 mentioned above.

If there is full-time work for an RTGC in the block in which it is stationed at the beginning of period p , that RTGC should continue to stay in the same block during the entire period. We therefore first classify all RTGCs in their current positions at the beginning of period p into:

- eligible RTGCs** those that can be moved from their current blocks in period p ;
- ineligible RTGCs** those that will be kept in their current blocks in period p .

Unfortunately, it is not a simple matter of comparing the workload, $w_i(p)$, in block i with the available $240r_i(p)$ RTGC minutes from the $r_i(p)$ RTGCs stationed in this block at the beginning of period p to determine whether some of these RTGCs can be considered eligible, because of Fact 1. For this reason, crane supervisors make decisions on which RTGCs are eligible to move, based on the following considerations.

- (a) If a block has two RTGCs stationed in it at the beginning of the period and after some time in the period if the remaining workload in the block is low enough that only one RTGC can handle it, then one of these RTGCs is eligible to move from this block at that time.
- (b) If there is no remaining workload in a block after some time in this period, then all the RTGCs in this block are eligible to move from this block at that time.
- (c) An RTGC can move easily between a pair of adjacent blocks that share a common width line. If (B_1, B_2) is such a pair, and if the expected arrival pattern of trucks in these two blocks during a period is such that k ($= 1, \text{ or } 2, \text{ or } 3$) RTGCs in these two blocks together can handle this workload by moving between these two blocks freely, then any RTGC in these two blocks put together beyond k is eligible to move at the beginning of this period.

Crane supervisors use many such considerations to determine which RTGC is eligible to move from its initial block in this period and the time at which it can move.

Also, block j in period p is classified as a **sink block** if $r_j(p) = 1$ or 0 and the expected workload

in it exceeds the capacity of the RTGCs. The expected number of additional RTGCs it needs in this period, δ_j , is

$$\begin{aligned} \delta_j &= 1, & \text{if } r_j(p) = 1, & \text{ because there can be at most two RTGCs in a block at any} \\ & & & \text{time,} \\ \delta_j &= 1 \text{ or } 2, & \text{if } r_j(p) = 0, & \text{ determined based on the estimated workload in that block.} \end{aligned}$$

Let n_e be the total number of eligible RTGCs in period p ; number them as RTGC 1 to RTGC n_e . Let m_s be the number of sink blocks in period p ; number them as blocks 1 to m_s . Define the decision variable, x_{ij} , for $i = 1$ to n_e , $j = 1$ to m_s by

$$x_{ij} = \begin{cases} 1, & \text{if RTGC } i \text{ moves to block } j \text{ in period } p, \\ 0, & \text{otherwise.} \end{cases}$$

Denote $\beta(i)$ as the block in which eligible RTGC i stays at the beginning of period p , then $\sum_{i=1}^{n_e} \sum_{j=1}^{m_s} c_{\beta(i)j} x_{ij}$ measures the RTGC productive time lost in the moves. This is like a substitute objective function for the total delayed RTGC workload. Thus, the following transportation problem is a suitable single period model for RTGC deployment in that period when $n_e \geq \sum_j \delta_j$.

$$\begin{aligned} \text{Minimize} \quad & \sum_{i=1}^{n_e} \sum_{j=1}^{m_s} c_{\beta(i)j} x_{ij} \\ \text{subject to} \quad & \sum_{j=1}^{m_s} x_{ij} \leq 1, & i = 1, \dots, n_e \\ & \sum_{i=1}^{n_e} x_{ij} = \delta_j, & j = 1, \dots, m_s \\ & x_{ij} = 0 \text{ or } 1, & i = 1, \dots, n_e; j = 1, \dots, m_s. \end{aligned}$$

When $n_e < \sum_j \delta_j$, a similar transportation problem can be formulated in which the RTGC constraints hold as equality and the sink block constraints hold as less than or equal to. Any of these transportation problems can be easily solved to optimality. But the strategy of using the more easily computed Vogel solution ([15]) for it as a near optimum is simpler to implement by the terminals.

For a numerical example, suppose that, in a certain period, there are nine eligible RTGCs (RTGC1 to RTGC9) and seven sink blocks (B1 to B7), where sink block 2 needs 2 RTGCs in this period, and the rest need 1 RTGC. Data on the travel time in minutes of the eligible RTGCs to the sink blocks is given in Table 2. In this numerical example, the Vogel solution with a total cost of 115 minutes is in fact an optimum solution of the transportation model. The model has several alternative integer optimal solutions. One of them is given by: $x_{1,1} = x_{2,2} = x_{3,3} = x_{4,2} = x_{5,4} = x_{6,5} = x_{7,6} = x_{9,7} = 1$, and all others by $x_{ij} = 0$.

< INSERT Table 2 around here. >

9 Optimal Allocation of ITs to QC

The workload of a QC working on a hatch of a vessel can be estimated fairly accurately based on the stowage situation of the inbound containers in the hatch and by the stowage plan of the outbound containers that go into that hatch. When a QC is unloading containers from a docked vessel, a certain number of ITs are allotted to it. These ITs queue up under the QC. The QC unloads a container from the vessel and puts it on the IT at the head of the queue. That IT then takes the container to the storage block where it is to be stored and queues up in the truck passing lane of the block for the YC (i.e., RTGC or RMGC) to take the container from its top and store it. Then, the IT returns to the queue under the QC and the same process repeats until the unloading activity is completed.

When the QC is loading containers onto the vessel, the same process goes on in reverse with the ITs bringing containers from the storage block to the QC. Consider an IT fleet moving containers between a QC and the storage yard. Let

n_a = the number of ITs allocated to the QC.

If n_a is too small, the QC may have long idle time waiting for the ITs. If n_a is too large, then the waiting time of the ITs in the queues at the QC and in the storage yard may be high. In order to optimize the twin objectives of minimizing the QC idle time and the total number of IT hours used to unload and load a vessel, it is necessary to find the optimum value for n_a , which may depend on the position of the QC and on the location of the storage block to which the ITs have to travel. However, the storage block to which the IT has to travel is determined by the online dispatching procedure discussed in Section 7.4, which tends to select a storage block with uniform probability among all those in the storage yard.

The expected driving time of an IT, between a QC position on the dock and a storage block in the yard, has been found to vary between 30 sec to 3 min at one of the terminals. These times change slightly from season to season (differences exist between rainy and non-rainy days), with an average of about 1.5 min. For the purposes of determining the optimum value of n_a at this terminal, we consider a hypothetical (QC, storage block) pair with IT driving times and crane processing times being random variables whose distributions are found empirically from container terminal data. For the travel time distributions, we average out the spatial effect of the location of the QCs and blocks by taking data from equipment dispersed all over the yard. The input data needed to make this decision are those that compile the empirical distributions of the following random variables.

- x_a, y_a = travel times of an IT from a QC position on the dock to a storage block and back.
- z_a = the QC processing time for a single container.
- w_a = the YC serving time for an IT in the storage block.

The QC on the dock, the YC in the storage block, and the ITs traveling back and forth between

them form a queueing network. The mathematical analysis of this queueing network has been carried out in [17], under the assumption that these distributions are negative exponential. However, observations made on these random variables indicate that negative exponential is not a good fit for their distributions. Analysis of the queueing network under general distributions by mathematical techniques is very difficult. We therefore estimate the optimal values of the decision variables, n_a , by running simulation experiments. As an example, we provide the results from two such experiments in Tables 3 and 4. The number of containers to be processed (inbound containers to be unloaded and outbound containers to be loaded) is mentioned in the heading of the table. All times are in minutes. The entries in the tables shown are:

$t_Q =$	estimated QC idle time in minutes,
$t_I =$	estimated idle time of all ITs in minutes,
penalty =	penalty of QC, IT idle times under the assumption that one minute of QC [IT] idle time has a penalty of 50 units [1 unit],
processing duration =	time in minutes to process all the containers.

All the operators in Hong Kong consider QC time much more valuable than IT time. The ratio 50 to 1 has frequently been considered as the penalty for idleness of these resources.

< INSERT Table 3 around here. >

< INSERT Table 4 around here. >

In both experiments, it can be seen that $n_a = 4$ ITs allocated to the QC yielded the lowest penalty estimate and very nearly the shortest processing duration for all the containers.

The simulation experiments also give the estimated expected time in minutes to process a hatch, and the standard deviation of this time as functions of h , where h is the number of containers to be processed in this hatch. These quantities corresponding to the optimum value of n_a are used for making decisions on optimum IT hiring plans over the day as discussed in the next section.

In all such simulation runs carried out for different seasons, etc., the optimum value for n_a turned out to be 4 or 5. In practice, the terminals have decided to form a single **pool** of ITs serving all the working QCs, instead of having a dedicated set of ITs serving each QC working on a hatch separately. They try to make sure that at each point of time, the number of ITs in the pool is not less than 4.5 (number of working QCs at that time). Also, each IT in the pool returning to the dock after finishing its work in the storage yard is dispatched to join the queue of a QC with the smallest number of ITs in it.

10 IT Hiring Over the Day

Terminals work around the clock every day, and the demand for ITs varies over the day, triggered by the schedules of the vessels arriving or leaving that day. Here, we discuss how to prepare a profile of the number of ITs needed to help the QC while they are working on vessels, in half-hour intervals. Using this information, we develop a model to determine the optimal number of ITs to hire at various times of the day to meet the requirements of the QCs, while minimizing the total number of ITs hired during the day. Throughout our discussion, we divide the day into 48 half-hour intervals: 0:00 – 0:30, 0:30 – 1:00, etc. These half-hour intervals are numbered 1 to 48.

The input data for these decisions are: vessel arrival and departure times during the day, vessel berthing information, the number of QCs allocated to vessels, the workloads of QCs (which are determined from the assignment of hatches to QCs), and the functions that give, respectively, the mean and the standard deviation of the total time to complete a given amount of the workload.

To Estimate the Total Processing Time for a Hatch

ITs carrying inbound containers from the dock are dispatched to a storage block in the yard using the dispatching policy discussed in Section 7.4, which tends to select that storage block uniformly among all blocks in the yard. For this reason, in Section 9, we discussed that the optimum number of ITs to allocate to the QC working on a hatch with h containers to be processed, the mean and standard deviation of the processing time (in minutes) for this hatch could be estimated using simulation on a hypothetical (QC, storage block) pair for which the IT driving distance between them is that of a random such pair in the yard.

Let $\mu(h), \sigma(h)$ denote the estimates of the mean and standard deviation of the hatch processing time with different values of h (= the number of containers to be processed in the hatch) obtained in the simulation, corresponding to the optimum number of ITs to allocate to the QC. Values of $\mu(h), \sigma(h)$ at one of the terminals are shown in Table 5.

Plots of $\mu(h), \sigma(h)$ indicate that both these functions can be approximated very closely by affine functions of h . These affine functions of h for the data in Table 5, determined by the least-square fitting method, indicate that we can approximate

$$\mu(h) \text{ by } \hat{\mu}(h) = 8.28 + 1.79h, \quad \text{and } \sigma(h) \text{ by } \hat{\sigma}(h) = 1.31 + 0.019h.$$

< INSERT Table 5 around here. >

To Determine the IT Requirement Profile Over Time

We refer to the processing of a hatch as a **job**. The mean and standard deviation, $\hat{\mu}(h), \hat{\sigma}(h)$, of the

processing time (in minutes) for a job involving h containers are determined as affine functions of h as discussed above. For planning purposes, a safe practice is to allow $\hat{\mu}(h) + \lambda\hat{\sigma}(h)$ minutes for completing this job, where λ is given some value between 0.5 to 1 depending on the terminal's preference. From the vessel's schedule, and the unloading and loading sequences, we know which of these jobs will be carried out sequentially and which will be carried out concurrently. From all this information, we obtain a profile of the number of ITs needed for the processing of this vessel over time, during its processing interval.

Table 6 shows the information assembled in a numerical example for computing the IT requirement profile over time for processing a vessel during its processing duration. The workload in a hatch is the total number of containers to be unloaded from or loaded in it at this terminal. In this example, 13 hatches on the vessel, numbered 1 to 13, have positive workloads. When the vessel docks, three QCs, numbered 1, 2, and 3, are scheduled to work on it. QC1 will work on hatches 1 to 5 in that order, one after the other; QC2 on hatches 6 to 9, and QC3 on hatches 10 to 13. We plan to allow $\hat{\mu}(h) + \hat{\sigma}(h)$ minutes (rounded up to its ceiling) for a QC to process a hatch with workload h ; this is the planned processing duration in Table 6. Therefore, QCs 1, 2, and 3 take, respectively, 350, 362, and 336 minutes to finish their jobs on this vessel. Also, for each working QC, we plan to have 4.5 ITs helping it from the pool of ITs.

< INSERT Table 6 around here. >

Suppose that all 3 QCs start working on this vessel at 5 AM. Then, from 5 AM to 10:36 AM all three QCs will be working on this vessel. During this period, the IT requirement for this vessel will be 14 (rounding up 3×4.5). From 10:36 AM to 10:50 AM, only two QCs will be working on this vessel, creating a requirement of 9 ITs for this vessel; from 10:50 AM to 11:02 AM, the requirement is 5 ITs. The vessel's processing will be completed by 11:02 AM.

Adding up this information from all the vessels expected to be docked during the day, we get a profile of the number of ITs needed during the whole day. We then take the IT requirement in any half-hour interval to be the maximum requirement at any point of time in this interval. See Table 7 for an illustrative example of the IT requirement profile for processing vessels in a day.

< INSERT Table 7 around here. >

To Determine the Optimal IT Hiring

The most common plan for ITs in Hong Kong is to work for four hours, take an hour's meal break, and then work for three more hours. The model for determining the IT hiring plan is based on the assumption that this is the work plan for all the ITs hired. It is easy to modify the model to accommodate other work plans.

Let

$d_\ell =$ the requirement for ITs (i.e., the number of ITs needed to help the QCs working on docked vessels) during the ℓ th half-hour interval of the day, $\ell = 1$ to 48.

We assume that ITs can begin work at the terminal at the beginning of any of the 48 half-hour intervals. Define the decision variable

$x_\ell =$ the number of ITs beginning work at the beginning of the ℓ th half hour interval of the day, $\ell = 1$ to 48.

Since the workplan is for ITs to work for four hours, take a one-hour meal break, and then work for three more hours, the ITs that will be working in the ℓ th half-hour interval are those that started working at the beginning of the half-hour intervals in the set

$$S_\ell = \{\ell - 15 \text{ to } \ell - 10, \ell - 7 \text{ to } \ell\}.$$

Depending on the value of ℓ , some of the intervals in the set, S_ℓ , might belong to the previous day. This set, S_ℓ , depends on the work plan for ITs used at the terminal. If different workplans are used, define S_ℓ to be

$S_\ell =$ the set of half-hour intervals satisfying the property that all ITs, which started working at the beginning of one of these intervals, will be working during the ℓ th half-hour interval.

Then, the model for determining the decision variables, x_ℓ , to meet IT requirements using the minimum number of ITs is the following integer program

$$\begin{aligned} & \text{Minimize} && \sum_{\ell=1}^{48} x_\ell \\ & \text{subject to} && \sum_{j \in S_\ell} x_j \geq d_\ell, \quad \ell = 1, \dots, 48 \\ & && x_\ell \geq 0, \quad \text{and integer for all } \ell. \end{aligned}$$

This integer program can be solved quite efficiently using a commercial software package like CPLEX. Also, since the terminal uses ITs for many other activities like reshuffling/restacking of stored containers etc., we can also take the solution obtained by rounding up the optimum solution of the LP relaxation of this model, to implement and assign any spare time of ITs to these other activities. Table 8 gives the hiring plan of ITs for meeting the requirements shown in Table 7.

< INSERT Table 8 around here. >

11 Conclusion

The daily operations of a container terminal are complex and involve a variety of decisions to be made under conditions varying with time. We described these in detail and discussed the strategies being used to design a computerized DSS for making some of these decisions efficiently. This is part of an ongoing effort to keep the terminals in Hong Kong among the most competitive in the global transportation logistics industry.

Acknowledgments

This research was partially supported by the HKUST Research Infrastructure Grant RI95/96.EG04 funded by the University Grants Council of Hong Kong, and by the Service Support Fund SSF/066/96 of the Innovation and Technology Commission, Hong Kong SAR Government.

The present version of the paper has benefitted from the excellent comments of two anonymous referees. We are very grateful for their suggestions.

References

- [1] M. Avriel, M. Penn, N. Shpirer, and S. Witteboon, Stowage planning for container ships to reduce the number of shifts, *Annals of Operations Research*, 76, 55-71 (1998).
- [2] E.K. Bish, T.Y. Leong, C.L. Li, J.W.C. Ng, and D. Simchi-Levi, Analysis of a new vehicle scheduling and location problem, *Naval Research Logistics*, 48(5): 363-385 (2001).
- [3] C.-Y. Chen, and T.-W. Hsieh, A time-space network model for the berth allocation problem, 19th IFIP TC7 Conference on System Modeling and Optimization, July, Cambridge, UK. (1999).
- [4] R.K. Cheung, C.-L. Li, and W. Lin, Interblock Crane Deployment in Container Terminals. *Transportation Science*, 36(1), 79-83 (2002).
- [5] C.F. Daganzo, The crane scheduling problem, *Transportation Research: Part B Methodological* 23, 159-175 (1989).
- [6] A. Imai, K. Nagaiwa, and C.W. Tat, Efficient planning of berth allocation for container terminals in Asia, *Journal of Advanced Transportation* 31(1), 75-94 (1997).
- [7] A. Imai, E. Nishimurra, and S. Papadimitriou, The dynamic berth allocation problem for a container port, *Transportation Research: Part B Methodological*, 35, 401-417 (2001).
- [8] K.H. Kim, and J.W. Bae, Re-marshaling export containers in port container terminals, *Computers and Industrial Engineering*, 35(3-4), 655-658 (1998).

- [9] K.H. Kim, and H.B. Kim, 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 (1998).
- [10] E. Kozan and P. Preston, Genetic algorithms to schedule container transfers at multimodal terminals, *International Transactions in Operations Research*, 6, 311-329 (1999).
- [11] K.K. Lai, and K. Shih, A study of container berth allocation, *Journal of Advanced Transportation* 26(1), 45-60 (1992).
- [12] C.I. Liu, H. Jula, and P.A. Ioannou, Design, simulation, and evaluation of automated container terminals, *IEEE Transactions on Intelligent Transportation Systems*, 3(1), 12-26 (2002).
- [13] P. J. M. Meersmans and R. Dekker, Operations research supports container handling, Report EI 2001-22, Econometric Institute, Erasmus University Rotterdam, 2001.
- [14] K.G. Murty, Storage Yard Planning in Container Shipping Terminals, Technical Report, Dept. of IEEM, Hong Kong University of Science and Technology (1997).
- [15] K.G. Murty, *Operations research : deterministic optimization models* (Prentice Hall, Englewood Cliffs, 1995).
- [16] K.G. Murty, J. Liu, Y.-w. Wan, C. Zhang, M.C.L. Tsang, and R. Linn, DSS (Decision Support Systems) for Operations in a Container Shipping Terminal, *The Proceedings of the First Gulf Conference on Decision Support Systems*, Kuwait, 189-208 (6-8 November 2000).
- [17] K.G. Murty, M.C.L. Tsang, and Y.-w. Wan, A Newsboy Type Model for Scheduling Stevedoring Tractors At Container Terminals in Hong Kong, Technical Report, Dept. of IEEM, Hong Kong University of Science and Technology (1996).
- [18] K.G. Murty and P.A. Djang, The US Army National Guard's Mobile Simulators Location and Routing Problem, *Operations Research*, 47, 175-182 (1999).
- [19] E. Nishimura, A. Imai, and S. Papadimitriou, Berth allocation in the public berth system by genetic algorithms, *European Journal of Operational Research*, 131, 282-292 (2001).
- [20] R.I. Peterkofsky, and C.F. Daganzo, A branch and bound solution method for the crane scheduling problem, *Transportation Research B* 24(3), 159-172 (1990).
- [21] E.D Roux, Storage capacity for import containers at seaports, PhD thesis, University of California, Berkeley (1996).
- [22] M. Taleb-Ibrahimi, B.D. Castilho, and C.F. Daganzo, Storage space vs handling work in container terminals, *Transportation Research: Part B Methodological*, 27(1), 13-32 (1993).
- [23] K.M. Van Hee and R.J. Wijbrands, Decision support system for container terminal planning, *European Journal of Operations Research*, 34, 262-272 (1988).
- [24] I.F.A. Vis and R. De Koster, Transshipment of containers at a container terminal: an overview, *European Journal of Operational Research* (to appear).

- [25] I.D. Wilson, P.A. Roach, and J.A. Ware, Container stowage pre-planning: using search to generate solution, a case study, *Knowledge-Based Systems*, 14, 137-145 (2001).
- [26] C. Zhang, Resource Planning in Container Storage Yard, Ph.D. thesis, Dept. of IEEM, Hong Kong University of Science and Technology (2000).
- [27] C. Zhang, Y.-w. Wan, J. Liu, and R. Linn, Dynamic Crane Deployment in Container Storage Yards, *Transportation Research B Methodological*, 36(6), 537-555 (2002).
- [28] C. Zhang, J. Liu, Y.-w. Wan, K.G. Murty, and R.J. Linn, Storage space allocation in container terminals, accepted for publication in *Transportation Research: Part B Methodological* (2002).

Table 1. Illustration of Step 2

Data			Computed Results	
i	k_i	A_i	$\lceil k_i \rho(p+2) \rceil$	$x_i(p+1)$
1	600	100	452	352
2	600	120	452	332
3	550	150	414	264
4	550	300	414	92
5	500	325	376	0
6	500	350	376	0
7	500	375	376	0
8	500	400	376	0
9	500	450	376	0
$EX(p+1) + IM(p+1) = 1040$				
$A_i = G_i(p+1) - (ES_i(p+1) + IS_i(p+1))$				

Table 2. Travel time of eligible RTGCs to sink blocks (in minutes)

Eligible RTGC	Sink block						
	B1	B2	B3	B4	B5	B6	B7
RTGC 1	20	20	25	35	40	50	55
RTGC 2	25	10	20	30	35	45	50
RTGC 3	30	20	10	25	30	40	45
RTGC 4	35	25	20	20	25	35	40
RTGC 5	40	30	25	10	20	30	35
RTGC 6	45	35	30	20	10	25	30
RTGC 7	50	40	35	25	20	20	35
RTGC 8	50	40	35	25	20	20	35
RTGC 9	60	50	45	35	30	20	10
No. of eligible RTGC needed	1	2	1	1	1	1	1

Table 3. Experiment 1: h , No. of containers to be processed = 30

n_a	t_Q	t_I	Total penalty	For processing duration (minutes)	
				Average	Std. deviation
1	79.52	1.65	3977.58	130.62	5.61
2	25.00	24.01	1323.78	76.98	3.33
3	13.61	53.64	734.30	64.80	2.20
4	11.14	106.69	663.50	62.17	1.88
5	10.52	173.84	699.77	61.56	1.85
6	10.07	248.44	751.85	61.39	1.83
7	9.73	324.25	810.84	61.21	1.78
8	9.51	404.12	879.63	61.06	1.93

Table 4. Experiment 2: h , No. of containers to be processed = 60

n_a	t_Q	t_I	Total penalty	For processing duration (minutes)	
				Average	Std. deviation
1	156.73	2.32	7838.68	258.79	8.18
2	47.87	29.14	2422.69	149.94	4.73
3	20.92	67.72	1113.75	122.95	3.16
4	13.90	147.70	842.52	115.80	2.47
5	12.30	261.25	876.30	114.34	2.20
6	11.91	386.90	982.58	114.23	2.33
7	11.49	516.07	1090.81	114.04	2.24
8	11.57	648.00	1226.62	114.12	2.38

Table 5. The mean and standard deviation of the total processing time for jobs on a vessel (number of IT used = 4, optimum)⁺

<i>h</i> , no. of containers*	Expected value of total processing time	Standard deviation of total processing time
27	56.52	1.80
30	62.08	1.91
35	71.04	1.92
40	79.90	2.20
46	90.75	2.27
54	105.00	2.33
60	115.71	2.46

* This is the sum of inbound and outbound containers to process in this hatch.

+ This data corresponds to number of ITs used = 4, the one corresponding to the lowest total penalty in the simulation run.

Table 6. The planned processing duration for jobs on a vessel

Hatch no.	ID of QC	Workload	Planned processing duration (minutes)
1	1	32	68
2	1	26	57
3	1	36	75
4	1	32	68
5	1	40	82
6	2	56	111
7	2	39	81
8	2	27	59
9	2	56	111
10	3	25	55
11	3	35	73
12	3	46	93
13	3	58	115

Table 7. The profile of the number of ITs needed over a day

Interval	Optimal number of ITs	Interval	Optimal number of ITs
1 to 12	20	13	17
14 to 15	12	16	10
17	12	18 to 20	8
21 to 30	20	31	17
32	12	33 to 36	20
37	14	38	9
39 to 40	8	41 to 44	20
45	17	46	13
47	12	48	24

Table 8. The number of ITs hired in each 30-minute interval

Period	Number of ITs starting service	Period	Number of ITs starting service
1	20	3	6
6	1	7	5
8	6	9	2
18	1	19	2
20	1	21	2
22	1	23	5
24	6	25	3
27	2	33	6
36	4	41	14



Figure 1. The container storage yard in a terminal with containers stored in columns or stacks of four to six containers. RTGCs (Rubber Tyred Gantry Cranes), which help to stack containers and retrieve them, can also be seen among the containers. The storage yard is divided into rectangular areas called blocks.



Figure 2. An RTGC spans the width of a block with seven rows of spaces between its two legs. Six of these rows are used for container storage, the seventh (rightmost in the figure, a parked truck is in the row) is for truck passing.



Figure 3. A QC with an IT serving it. A vessel docked on the berth can be seen. The QC unloads inbound containers from the vessel and places them on ITs, which take them to the storage yard for storage until they are picked up by consignees. ITs also bring outbound containers from the storage yard for the QC to load onto the vessel.

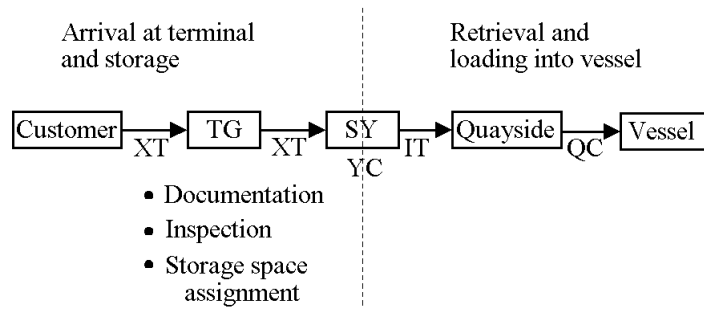


Figure 4. The flow of outbound containers. SY = Storage Yard. Underneath each location or operation, we list the equipment that handles the containers there.

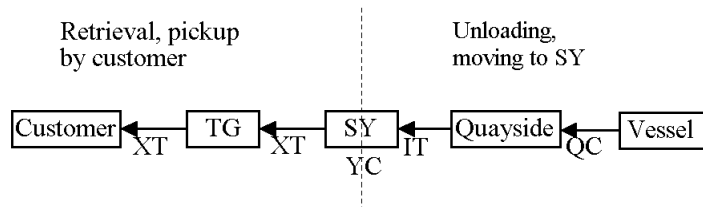


Figure 5. Flow of inbound containers.

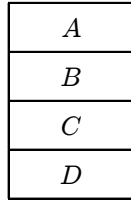


Figure 6. Four containers stored in a stack.

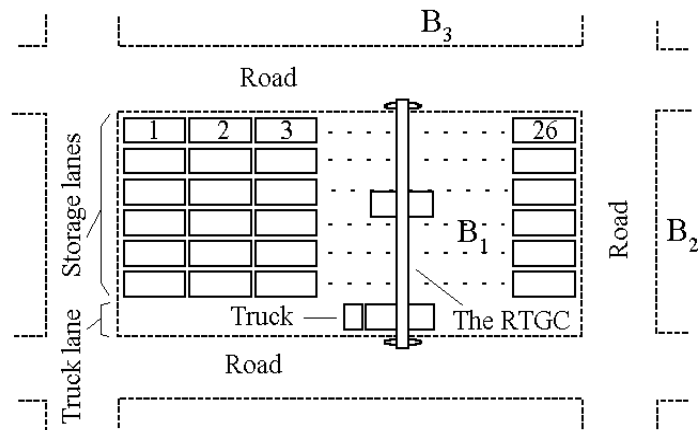


Figure 7. Top view of a block B_1 being served by an RTGC. Adjacent block B_2 shares a width line with B_1 ; adjacent block B_3 shares a length line with B_1 .