EFFICIENT YARD STORAGE
IN TRANSSHIPMENT CONTAINER HUB PORTS

HAN YONGBIN

NATIONAL UNIVERSITY OF SINGAPORE

2007

EFFICIENT YARD STORAGE IN TRANSSHIPMENT CONTAINER HUB PORTS

HAN YONGBIN

(M. Eng., Tsinghua University)

A THESIS SUBMITTED

FOR THE DEGREE OF DOCTOR OF PHILOSOPHY

DEPARTMENT OF INDUSTRIAL AND SYSTEMS ENGINEERING

NATIONAL UNIVERSITY OF SINGAPORE

2007

Acknowledgements

This thesis would never have been written without the support of the people who have enriched me through wisdom, friendship and love in many ways.

I would like to express my deepest appreciation to my three supervisors: A/Prof. LEE Loo Hay, A/Prof. CHEW Ek Peng, and A/Prof. TAN Kok Choon. They have continued to provide much invaluable guidance and encouragement throughout the whole course of my research.

Gratitude also goes to all other faculty members in the Department of Industrial and Systems Engineering for their kind attention and help in my research.

I am also grateful to the fellow students in the Department of Industrial and Systems Engineering. Particularly, I would like to thank Ms. WANG Qian for her help to conduct the simulation project on docking station problem.

Last, but not the least, I would like to thank my wife for her continuous support and encouragement, my dearest daughter who makes my life full of expectations, and her four grand-parents for their wholehearted help.

HAN YONGBIN

Table of Contents

Summary

In the past two decades, much research effort has been spent on studying various operations management problems in order to help container terminals handle the continuous growing container traffic more efficiently and cost effectively. However, most previous works do not sufficiently address the particular needs of major container transshipment hubs. These works tend to focus on some generic terminal where import (unloading) and export (loading) activities can be handled separately. In contrast, this thesis aims to study a critical operations management problem, which is efficient yard storage, in a mega-transshipment hub port where unloading and loading activities are very often both heavy and concentrated.

Export and transshipment containers depart in large batches at designated time when the vessel comes. Hence the port operator uses the consignment strategy to group export and transshipment containers to dedicated sub-blocks to reduce the number of reshuffles, hence to reduce the vessel turnaround time. In order to handle the potential traffic congestion of prime movers, a high-low workload balancing protocol is proposed. However, the port operator does not have any formal planning tool to solve this yard template problem and the decisions are based on intuition and past experiences. Hence a mathematical model is developed, which is able to provide a holistic and systematic way to address this problem. The model cannot be solved to optimality by CPLEX because of the problem structure and scale. To solve the formulated model, the yard allocation problem is solved by a proposed heuristic algorithm, the sequential method, assuming that the yard template is given. Based on this, an iterative improving solution method is developed to solve the yard template problem. Computational experiments show that the proposed method can generate excellent results within a reasonable time, even for extreme cases. This is the first study to address the yard template problem with the consignment strategy and high-low workload balancing protocol for a transshipment hub.

In contrast, import containers arrive at the storage yard in large batches and in a predicted fashion, but depart one by one in an unpredictable order. Therefore, import containers are usually stored in separate blocks from export and transshipment containers so as to facilitate the ease of customer retrieval. In order to manage the competing demands for yard cranes in the import blocks, a docking station concept is proposed to change the current horizontal layout for import container blocks to a vertical layout. With the docking station concept, internal prime movers and external trucks are segregated, which allows the port operator the flexibility of assigning yard crane service priority to internal prime movers and hence the ship turnaround time can be reduced when required. To verify the effectiveness of the docking station concept, two simulation models for the base layout and the proposed perpendicular layout are built respectively. Simulation results show that the cycle time of internal prime movers can be reduced when priority is given to them, but the required service level for external trucks needs to be slightly lowered because the yard crane service capacity decreases as a result of the extra movement of yard cranes. However, a new method of operations in docking station is proposed to reduce the yard crane's effective traveling distance per handling, with which the internal prime movers'

cycle time could be significantly reduced while the current service requirement for external trucks can also be met.

Although the yard template problem and the docking station problem are actual problems raised from a leading transshipment port in the South-East Asia, the methodology including the strategies used, the model formulations, and the solution methods can be used for any transshipment hub where transshipment of containers is the major activity and the yard activity is heavy.

List of Tables

List of Figures

List of Abbreviations

AGV:	Automated Guided Vehicle
ALV:	Automated Lifting Vehicle
ASC:	Automated Stacking Crane
CI:	Confidence Interval
FCFS:	First Come First Served
FCL:	Full Container Load
GA:	Genetic Algorithm
IYT:	Initial Yard Template
LB:	Lower Bound
MILP:	Mixed Integer Linear Programming
MIP:	Mixed Integer Programming
PM:	Prime Mover
QC:	Quay Crane
BBP:	Big-block Problem
LBP:	Lower Bound Problem
RMG:	Rail Mounted Gantry
RTG:	Rubber Tyred Gantry
YAP:	Yard Allocation Problem
SA:	Simulated Annealing
Std. Dev.	Standard Deviation

TEU:	Twenty-feet Equivalent Unit
TRACES:	Traffic Control Engineering System
W/O:	Without
YC:	Yard Crane
YCS:	Yard Crane Shift
YTP:	Yard Template Problem

List of Notations

A_j	the smallest number of sub-blocks that should be reserved for Vessel j, $1 \le j \le J$. It is necessary to ensure that the total number of sub-blocks in the storage yard is not less than the summation of all the A_j, $1 \le j \le J$.
B	the number of big-blocks under consideration, $B = K/2$.
B_k	the set of sub-blocks that belong to Block k, $1 \le k \le K$.
C_k	the maximum number of yard cranes allowed to reside in Block k at any one time, $1 \le k \le K$.
CC	the capacity of each yard crane in terms of container moves per shift, which is 100 in this thesis according to the current practice in the studied port.
CS	the space capacity of each sub-block in terms of TEUs, which is 240 (5 tiers×6 lanes×8 slots) in this thesis.
d_{kt}	the number of yard cranes allocated to Block k for unloading in Shift t, $1 \le k \le K$, $1 \le t \le T$.
d_{ktr}	the number of yard cranes allocated to Block k for unloading in Shift t for Column r, $1 \le k \le K$, $1 \le t \le T$, $r \ge 1$.
E_j	the maximum number of shifts allowed to load the containers for Vessel j, $1 \le j \le J$.

F_{ij}	= 1, if Sub-block i is reserved for Vessel j, $1 \le i \le I$, $1 \le j \le J$. = 0, otherwise.
G_i	the maximum number of shifts allowed to load the containers for Sub-block i, $1 \le i \le I$. $G_i = \sum_{j=1}^{J} F_{ij} E_j$.
h_{it}	= 1, if the total workload that are allocated to Sub-block i for unloading in Shift t is high, that is, $HL \le \sum_{j=1}^{J} \left(x_{ijt} + y_{ijt} \right) \le HU$ or $HL \le x_{it}+y_{it} \le HU$, $1 \le i \le I$, $1 \le t \le T$. = 0, if the total workload that are allocated to Sub-block i for unloading in Shift t is low, that is, $LL \le \sum_{j=1}^{J} \left(x_{ijt} + y_{ijt} \right) \le LU$ or $LL \le x_{it}+y_{it} \le LU$, $1 \le i \le I$, $1 \le t \le T$.
h_{itr}	= 1, if the workload that are allocated to Sub-block i for unloading in Shift t for Column r is high, i.e., $HL \le x_{itr}+y_{itr} \le HU$, $1 \le i \le I$, $1 \le t \le T$, $r \ge 1$. = 0, if the workload that are allocated to Sub-block i for unloading in Shift t for Column r is low, i.e., $LL \le x_{itr}+y_{itr} \le LU$, $1 \le i \le I$, $1 \le t \le T$, $r \ge 1$.
HL	the lowest value that a high workload can take.
HU	the highest value that a high workload can take.
I	the number of sub-blocks under consideration.
J	the number of vessels under consideration in the planning horizon.
K	the number of blocks under consideration.

L_i	the set of shifts in which Sub-block i is in the loading process, $1 \le i \le I$.
LL	the lowest value that a low workload can take.
LU	the highest value that a low workload can take.
M	a sufficiently large positive value.
N_i	the set of sub-blocks that are neighbors of Sub-block i, $1 \le i \le I$.
n_{tr}	the total number of yard cranes required in Shift t for Column r, $1 \le t \le T$, $r \ge 1$.
NL_{kt}	the number of sub-blocks in the loading process in Block k in Shift t, $1 \le k \le K$, $1 \le t \le T$.
P	the number of patterns for big-block.
P_{ps}	whether the Sub-block s is high workload for pattern p, $1 \le p \le P$, $1 \le s \le S$.
Q_b	the number of sub-blocks in Big-block b, $1 \le b \le B$.
$R_{ii'}$	$= 1$, if Sub-block i is a neighbor of Sub-block i', $1 \le i \le I$, $1 \le i' \le I$.
	$= 0$, otherwise.
S	the number of sub-blocks in one big-block, $S = 10$ in this model.
T	the number of shifts under consideration in the planning horizon.
V_j	the set of sub-blocks that are reserved for Vessel j, $1 \le j \le J$.
VL_{jt}	$= 1$, if Vessel j is in the loading process in Shift t, $1 \le j \le J$, $1 \le t \le T$.

	= 0, otherwise.
w_{tr}	= 1, if Column r is selected for Shift t, $1 \leq t \leq T$, $r \geq 1$.
	= 0, otherwise.
WX_{jt}	the number of 20-feet containers arriving at the terminal in Shift t and will be loaded onto Vessel j finally. It is given and input to the model, $1 \leq j \leq J$, $1 \leq t \leq T$.
WY_{jt}	the number of 40-feet containers arriving at the terminal in Shift t and will be loaded onto Vessel j finally. It is given and input to the model, $1 \leq j \leq J$, $1 \leq t \leq T$.
x_{it}	the number of 20-feet containers that are allocated to Sub-block i for unloading in Shift t, $1 \leq i \leq I$, $1 \leq t \leq T$.
x_{ijt}	the number of 20-feet containers that are allocated to Sub-block i for unloading in Shift t if Sub-block i is reserved for Vessel j, $1 \leq i \leq I$, $1 \leq j \leq J$, $1 \leq t \leq T$.
	= 0, if Sub-block i is not reserved for Vessel j.
x_{itr}	the number of 20-feet containers that are allocated to Sub-block i for unloading in Shift t for Column r, $1 \leq i \leq I$, $1 \leq t \leq T$, $r \geq 1$.
y_{it}	the number of 40-feet containers that are allocated to Sub-block i for unloading in Shift t, $1 \leq i \leq I$, $1 \leq t \leq T$.
y_{ijt}	the number of 40-feet containers that are allocated to Sub-block i for unloading in Shift t if Sub-block i is reserved for Vessel j, $1 \leq i \leq I$, $1 \leq j \leq J$, $1 \leq t \leq T$.

	= 0, if Sub-block i is not reserved for Vessel j.
y_{itr}	the number of 40-feet containers that are allocated to Sub-block i for unloading in Shift t for Column r, $1 \leq i \leq I$, $1 \leq t \leq T$, $r \geq 1$.
z_{ij}	= 1, if Sub-block i is reserved for Vessel j, $1 \leq i \leq I$, $1 \leq j \leq J$. = 0, otherwise.
z_{btp}	whether big-block Pattern p is selected for Big-block b during Shift t, $1 \leq b \leq B$, $1 \leq t \leq T$, $1 \leq p \leq P$.

1 Introduction and Overview

Container traffic has been growing steadily and this trend is expected to continue (see Figure 1.1). To handle the increasing volume of containers, container vessels are becoming larger in size. This will result in a longer processing time to turn around the vessels. Therefore, the optimal management of logistic activities at container terminals is needed to improve the performance of container terminals. This is crucial to guarantee that the terminal system can react in the most cost-effective way to meet the continuous growth of container traffic.

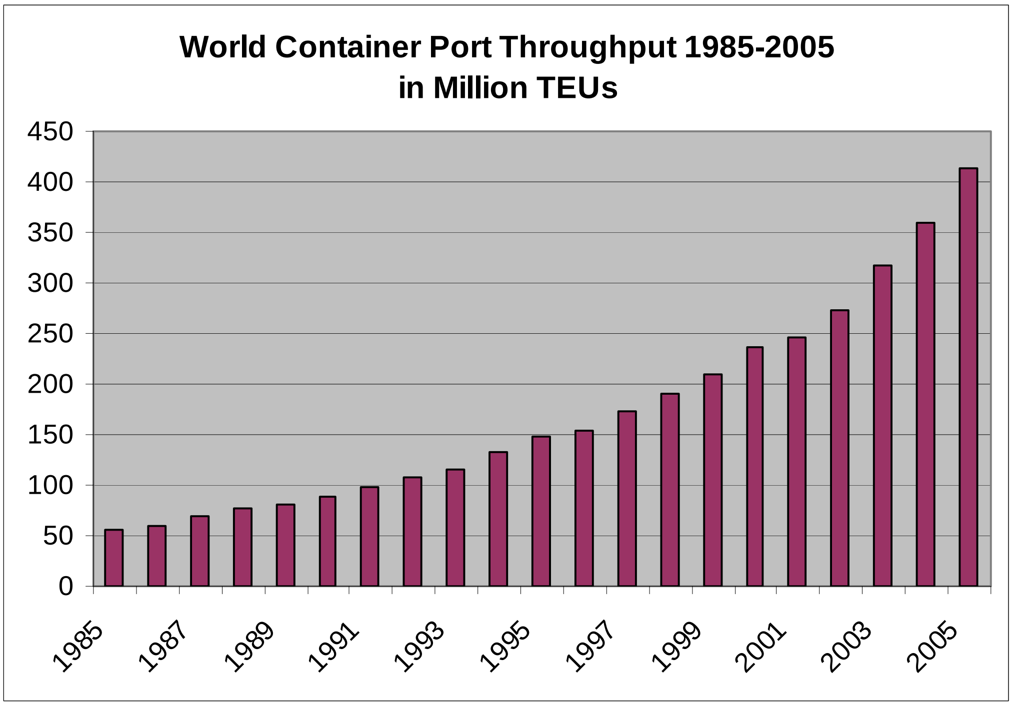

Figure 1.1 World container port throughput 1985-2005[1]

---

[1] From www.mainport-hamburg.de

1.1 Introduction

Containers have been designed for easy and fast handling of freight so that the contents do not have to be unpacked at each point of transfer. Consequently, high productivity can be achieved. Besides, the metal boxes provide protections against weather and pilferage. The dimensions of containers for maritime purpose have been standardized. The term twenty-feet-equivalent-unit (TEU) is used to refer to a container with a length of twenty feet. A container with a length of forty feet is expressed by 2 TEUs. Additional properties of containers may be specified whenever appropriate (e.g., the weight class of a container, the necessity of special handling for reefer containers or oversized containers).

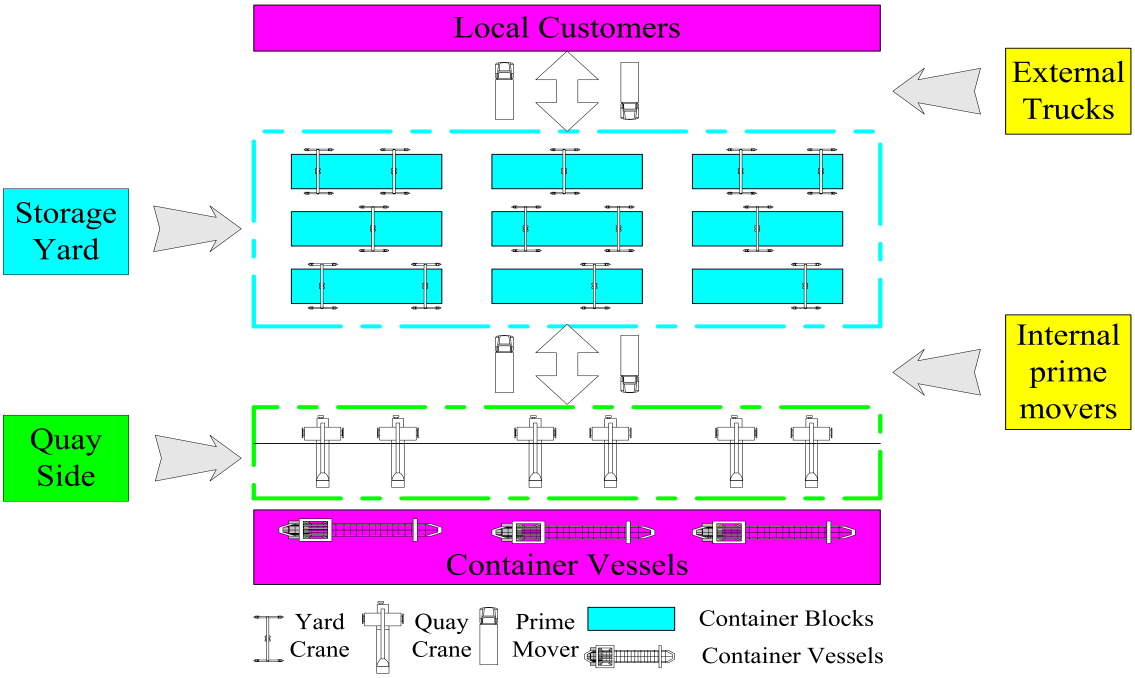

Figure 1.2 A schematic diagram of a container terminal

A container terminal is a place where containers are loaded (unloaded) onto (from) container vessels. Based on the types of container handling operations, a container terminal can be roughly divided into two main areas, the quayside for berthing vessels and the storage yard for holding containers (As shown in Figure 1.2). The quayside is made up of several berths for vessels to moor. The vessels moored at berths are served by quay cranes (QCs) which load and unload containers. The storage yard is used to temporarily store containers until they are picked up by external trucks or loaded onto destination vessels. A large-scale storage yard is typically divided into several storage areas called blocks. In each block, containers are stored side by side and one on top of another. A typical container block, as shown in Figure 1.3, may have up to 12 lanes of containers in width, more than 20 containers in length, and up to 7 containers in height. The width and length of a container block depend on the width and height of the yard cranes used. Yard cranes lift containers from vehicles and store them at storage locations, or retrieve containers from their storage locations and put them on the vehicles. The transport of containers between the quayside and the storage yard is carried out by vehicles such as prime movers or straddle carriers; while the transport of containers between the storage yard and local customers is carried out by external trucks, rail or barge.

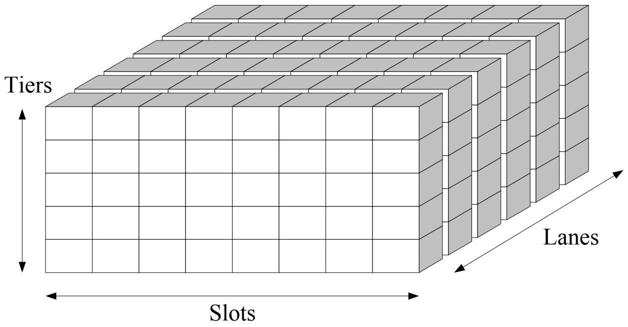

Figure 1.3 A typical block of containers

A schematic diagram of the typical processes in a container terminal is shown in Figure 1.4 (Vis et al. 2003). Container activities can be categorized into three types: import, export, and transshipment activities. For export activities, the containers are brought in by shippers and will be stored at their designated locations in the storage yard. When it is time to load the containers, they are retrieved from the stored locations and transported by vehicles to the quayside. The quay cranes then remove the containers from the vehicles and load them onto the vessels. The processes for import activities are similar but they are done in the reverse order. For transshipment activities, the processes are a little different. The transshipment containers will be stored in the storage yard after they are unloaded from the vessel, and will be finally loaded onto other vessels. In this thesis, our study is focused on the storage yard management in transshipment hubs where transshipment of containers is the major activity and the yard activity is heavy.

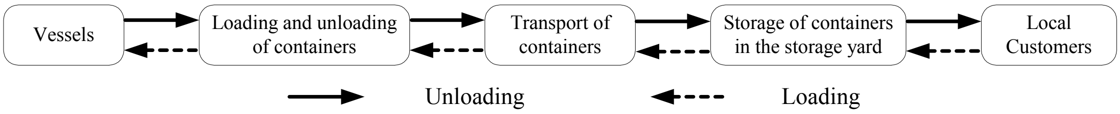

Figure 1.4 A flow diagram demonstrating the interaction between container terminal processes

The storage yard plays an important role in transshipment hubs where most containers unloaded from one vessel will be stored in the storage yard and will be eventually loaded onto other vessels. This means the loading and unloading activities are both concentrated and need to be considered at the same time for a good planning. Therefore, the planning problem for transshipment hubs is much more challenging compared to port planning for general terminals. A lot of studies have been done on the storage yard management problem, but all the studies are based on general terminals which emphasize on import and export activities. In this thesis, the storage yard management problem particularly for a transshipment hub is studied.

1.2 Organization of the Thesis

This thesis consists of seven chapters. The rest of this thesis is organized as follows, which is shown in Figure 1.5.

Chapter 2 introduces related works dealing with port operations including capacity planning, berth allocation, quay crane assignment, ship stowage, yard configuration, yard allocation, yard crane deployment, prime mover deployment, inter-terminal operations, outside terminal operations, and integrated terminal study, etc.

In Chapter 3, the formulated mixed integer linear programming model for the yard template problem is presented, in which export and transshipment containers are stored in dedicated sub-blocks and a high-low workload balancing protocol is incorporated.

Chapter 4 describes the yard allocation problem and the proposed heuristics to solve it. Numerical experiments on the yard allocation problem are conducted and computational results are presented in this chapter.

In Chapter 5, the yard template model is solved by an iterative improving solution procedure based on the sequential method proposed in Chapter 4. Numerical experiments and extreme case experiments are conducted to test the effectiveness and robustness of the proposed solution procedure.

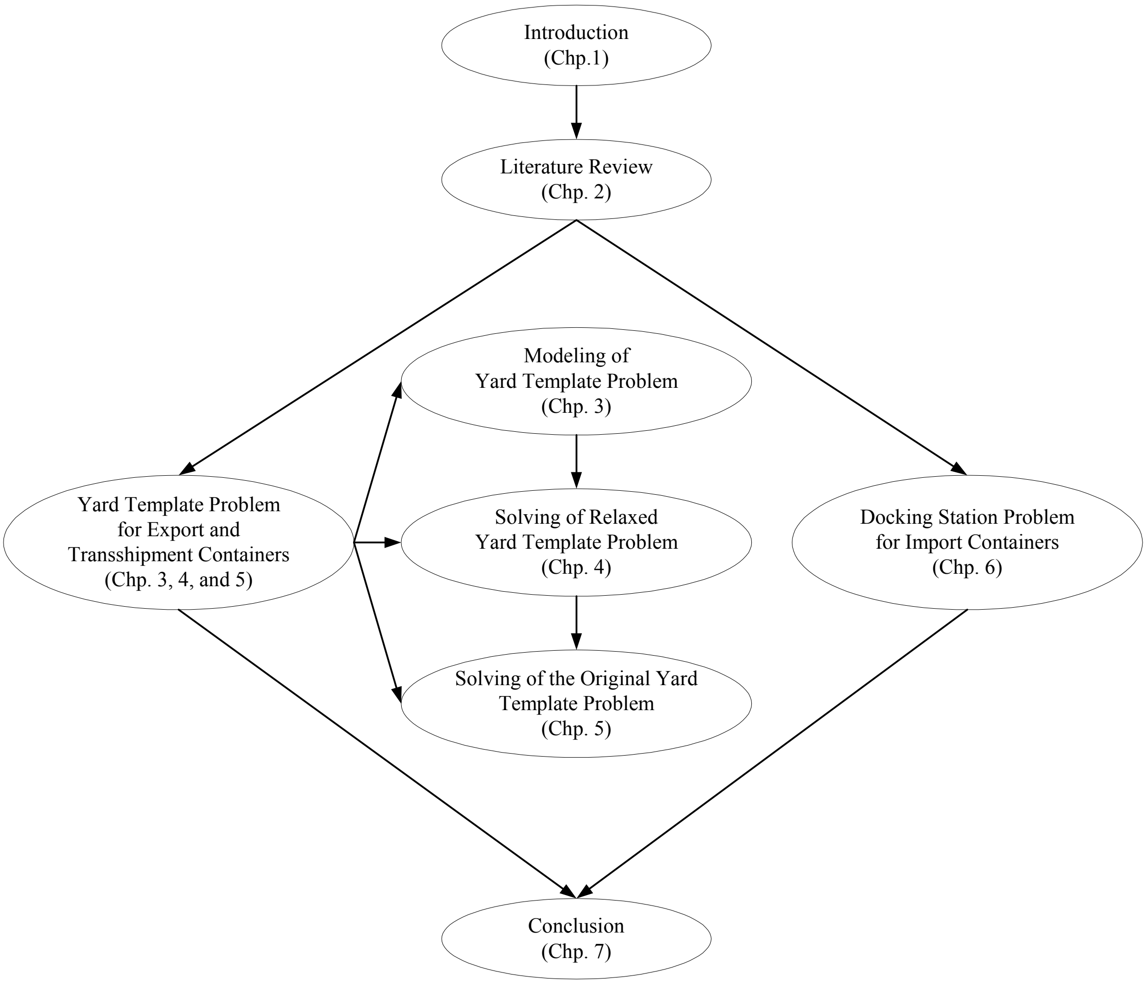

Figure 1.5 The structure of the thesis

In Chapter 6, the docking station concept for import container blocks is studied by discrete event simulation. Two simulation models for the base layout and the proposed layout are built. Simulation runs are conducted to test the efficiency and effectiveness of the proposed layout. In addition, a recommended improvement on the operations in the import container blocks is presented in this chapter.

Finally, in Chapter 7, the findings from previous chapters are consolidated and issues for future research are discussed.

2 Literature Review

There are lots of decisions when operating container terminals and all these decisions are interrelated to some extent. For example, how to allocate the containers in the storage yard directly affects the workload of the yard cranes in the blocks and the traveling distance of internal prime movers or external trucks, and indirectly affects the turnaround time of vessels and the productivity of quay cranes. There are various interrelated performance indicators of a container terminal, measuring the productivity and utilization of every type of resource, and various aspects of customer satisfaction. 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 satisfy the overall objectives. Logically, the hierarchical approach is adopted to treat the whole problem as several smaller sequential problems. The input to a problem is actually the output of higher level decisions, and is treated as a known value after the higher level decisions are solved.

The decisions can be divided into three levels, i.e. the strategy level, the tactical level, and the operational level. At the strategy level, it is decided, for example, which yard layout should be used? Strategy level decisions usually cover a long time horizon, say several years. At the tactical level, it is decided, for example, how import and export containers should be stored; should they be mixed in the same block or separately stored in different blocks? Capacity planning problem, for example, at the tactical level addresses how many quay cranes, yard cranes, and prime movers should be used? The time horizon for tactical level decisions covers from days to months. The detailed daily problems belong to the operational level, for example, which way a vehicle should go to deliver a container to the storage location in the yard?

In this chapter, a detailed literature review is presented according to the processes in container terminals. For each type of process, different levels of decisions, i.e. the strategy level, the tactical level, and the operational level are discussed. For general port operations, only the references are provided; while most related studies will be talked in more details. Literature reviews on port operations can also be found in Vis and de Koster (2003) and Steenken et al. (2004).

2.1 Berthing Activities

When a ship arrives at the terminal, it has to find a place to moor. The berth (place for ship to moor) together with several quay cranes will be assigned to the ship.

2.1.1 Berth Capacity Planning Problem

The number of berths that should be available at the quayside is one of the strategic decisions. The berth capacity planning problem was studied in Edmond and Maggs (1978), Agerschou et al. (1983), Bruzzone and Signorile (1998), Lim (1998), Moon (2000), Legato and Mazza (2001), and Nam et al. (2002).

2.1.2 Berth Allocation Problem

One of the decisions at the operational level is the allocation of a berth to the ship. The berth allocation problem was studied in Lai and Shih (1992), Imai et al. (1997), Imai et al. (2001), Nishimura et al. (2001), Guan et al. (2002), Park and Kim (2002), Imai et al. (2003), Kim and Moon (2003), Park and Kim (2003), Guan and Cheung (2004), and Moorthy and Teo (2006).

2.1.3 Quay Crane Scheduling Problem

Another decision at the operational level is the allocation of quay cranes to the container ships. The quay crane scheduling problem was studied in Daganzo (1989), Peterkofsky and Daganzo (1990), Zaffalon et al. (1998), and Murty et al. (2006).

2.2 Loading and Unloading of Containers

2.2.1 Ship Stowage Problem

To ensure fast and efficient transshipment of containers, a good distribution of containers over the ship is necessary. In other words, stowage planning is needed at the operational level. The ship stowage problem was studied in Shields (1984), Avriel and Penn (1993), Avriel et al. (1998), Wilson and Roach (1999), Avriel et al. (2000), Wilson and Roach (2000), Steenken et al. (2001), Wilson et al. (2001), Dubrovsky et al. (2002), Kang and Kim (2002), Roach and Wilson (2002), and Giemsch and Jellinghaus (2003).

2.2.2 Load and Unload Sequencing Problem

According to the stowage plan, there is usually a loading list for each assigned quay crane. The load sequencing problem was studied in Gambardella et al. (2001), Haghani and Kaisar (2001), and Kim et al. (2004). An unloading plan, which indicates which container should be unloaded and in which area it is situated in the ship, is given before the arrival of the ship. Within a defined area the quay crane driver can freely determine the order in which the containers are unloaded. The unload sequencing problem was studied in Gambardella et al. (2001).

2.3 Transport of Containers

When the container terminal is designed, the type of material handling equipment that carries out the transport of containers between the quayside and the storage yard should be determined at the strategy level. Vehicles like forklift trucks, yard trucks or straddle carriers can be used at a manned terminal; while at an automated terminal, Automated Guided Vehicles (AGVs) are the commonest equipment. Different types of container transport equipment were studied in Baker (1998), Asef-Vaziri et al. (2003a, 2003b), Vis and Harika (2004), Yang et al. (2004) and Duinkerken et al. (2006).

2.3.1 Fleet Sizing Problem

One of the decisions at the tactical level is the determination of the necessary number of transport vehicles. The fleet sizing problem was studied in Steenken (1992), Vis et al. (2001), and Koo et al. (2004).

2.3.2 Vehicle Routing and Dispatching Problem

A decision at the operational level is to determine which vehicle transports which container by which route. The general vehicle routing problem was studied in Bish et al. (2001), Narasimhan and Palekar (2002), Li and Vairaktarakis (2004), and Bish et al. (2006).

Straddle carriers are alternative vehicles for the transport, retrieval and storage of containers. Thus the routing of straddle carriers has got much attention from the researchers. The routing problem of straddle carriers was studied in Steenken (1992), Steenken et al. (1993), Kim and Kim (1999b, 1999c), and Böse et al. (2000).

Recently, more container terminals utilize automated transporters, like AGVs. Therefore the research on the dispatching of AGVs becomes important. The AGV dispatching problem was studied in Evers and Koppers (1996), Chen (1998) , Zaffalon et al. (1998), Duinkerken et al. (1999), Kim and Bae (1999), Gademann and van de Velde (2000), Reveliotis (2000), van der Meer (2000), Bish et al. (2001), Chan (2001), Leong (2001), van der Heijden et al. (2002), Lim et al. (2003), Moorthy et al. (2003), Schneidereit (2003), Grunow et al. (2004), Liu et al. (2004), Nishimura et al. (2005), Briskorn et al. (2006), Lehmann et al. (2006), and Grunow et al. (2006).

The amount of delay time of external trucks for receiving and delivery operations is the most important performance measure for the customer service level. The external truck sequencing problem was studied in Kim et al. (2003).

2.4 Storage of Containers in the Yard

Containers can be stored on a chassis or directly on the ground. Containers stored on a chassis are individually accessible but need a lot of storage space; while containers stored on the ground can save storage space at the expense of accessibility. Nowadays ground stacking is much more common because the land is becoming scarce as a result of the growing container volume.

One of the decisions at the strategy level is the determination of the material handling equipment that carries out the container storage and retrieval operations. Equipment like yard cranes, forklift trucks, reach stackers, and straddle carriers can be chosen. In automated terminals, Automated Stacking Cranes (ASCs) are commonly used. Adopting automated handling systems will increase capital burdens on port operators and does not always guarantee increased productivity. This general aspect of automated systems is somewhat dependent on terminal characteristics such as labor costs. Nam and Ha (2001) discussed the determination of container handling systems, particularly with respect to the port in Korea.

2.4.1 Yard Layout Problem

As a consequence of the growing container traffic, the storage yard is becoming scarce. A good yard layout is desired for other related operations. The yard layout problem was studied in Agerschou et al. (1983), Ballis and Abacoumkin (1996), and Bruzzone and Signoriler (1998).

2.4.2 Capacity Planning Problem

One of the decisions at the tactical level is the determination of the necessary number of transfer cranes or the storage space required. The yard capacity planning problem was studied in Kim and Kim (1998), and Kim and Kim (2002).

2.4.3 Storage Allocation Problem

The efficiency of stacking depends greatly on the strategies of allocating storage space to arriving containers. When a container needs to be retrieved, those containers that are stored on top of the requested container should be moved first. The move of those containers on top of the requested container is unproductive and is called reshuffle. The container reshuffles should be reduced as much as possible in order to increase the productivity of the yard cranes. To significantly eliminate the unproductive reshuffles, Chung et al. (1988) proposed the use of buffer space to increase the utilization of the material handling equipment and reduce the total container loading time. A simulation model was built to compare the system with buffers and the current non-buffer system. The simulation results showed that the system with buffer could significantly reduce the number of reshuffles and the total container loading time. However, extra yard cranes were needed for double-handling and extra storage space was needed to serve as the buffer. Sculli and Hui (1988) developed a simulation model to study the stacking of containers with the same dimensions. Simulation results showed that the number of different types of containers had the largest impact on the measure of performance selected. The authors failed to locate any references that were directly relevant. Many more stacking policies could be explored combined with different patterns of arrival and demand for containers, and also different stack dimensions. Kim (1997) proposed a methodology to estimate the expected number of reshuffles to pick up an arbitrary container and the total number of reshuffles to pick up all the containers in a block for a given initial stacking configuration. He found that the height and width of the container blocks were the key factors which determine the average number of reshuffles to pick up a container. However, the analysis of reshuffles was restricted to a single bay and he assumed that every rehandled container was moved to a different slot in the same bay. Gambardella et al. (1998) built a decision support system for the storage allocation problem. An integer linear programming model was formulated to get the optimal solution. In addition, a process-oriented discrete event simulation model was developed to check the validity and robustness of the policy obtained from the mathematical programming model. Their study was not generic but a case study restricted to Contship La Spezia Container Terminal. Holguín-Versa and Jara-Díaz (1996) studied the storage allocation problem with priority service. The intrinsic and logistic cargo value was taken into account in the formulated model, which extended the classical price differentiation theory, i.e. the inverse elasticity rule, in various directions. However, it was only a research tool and could not be used as a decision aid. Chen (1999) identified several major factors that influenced operational efficiency and caused unproductive reshuffles in terminal operations. He concluded that higher container stacking had a serious impact on the number of reshuffles and the major impact was on the delivery operation. Kozan and Preston (1999) studied the storage policy in the storage yard. They concluded that containers storing in the closest rows to the berth was better than random storage policy. Another finding was that decreasing the maximum height of container blocks could dramatically reduce the transfer time. The analysis was under the assumption that ships were equally distributed to the berths and the same level of service would be provided by the operators. In Chen et al. (2000) the storage space allocation problem was examined with a time-space network in which the storage locations were allocated to containers in advance. The objective was to re-use the storage space in different time spans. Real world case was used to evaluate the model and computer graphics was used to display the output of the model for quick response to operators. However, they did not consider the uncertainty in the vessel arriving time. Preston and Kozan (2001) developed a container allocation model to minimize the turnaround time of container vessels. Genetic algorithm was used to solve the model. The results for different resource levels and a comparison with the current practice for the studied port were presented. However, they did not differentiate the velocity of the transporters for different types of machines and containers. Zhang et al. (2003) studied the storage space allocation problem in the storage yard by a rolling-horizon approach. The problem was solved by two stages. At the first stage the total number of containers to be stored in each container block in each shift was determined to balance the workload among blocks. Based on the result of the first stage problem, the number of containers associated with each vessel was determined to minimize the total traveling distance at the second stage. One of the assumptions in the model was that there were always sufficient yard cranes to handle the workload, which might not be realistic in many container terminals. Murty et al. (2005) developed an online dispatching procedure for assigning containers to storage locations. To reduce traffic congestion of prime movers, a fill ratio equalization approach was used to allocate containers to the storage locations, which was based on the hypothesis that traffic congestion in the terminal would be at its minimum if the fill ratios of all the blocks were maintained to be nearly equal. Dekker et al. (2006) studied the storage allocation problem for an automated container terminal. Several variants of consignment strategy were discussed. The consignment strategy is to store the same group of containers together in order to reduce the number of reshuffles because the sequence in which the containers from the same group are retrieved does not matter. However, they did not consider the availability of AGVs and assumed that there were sufficient AGVs to handle all the workload. Kozan and Preston (2006) presented an iterative search algorithm for the integrated container-transfer and container-allocation model to determine the optimal storage strategy and corresponding handling schedule. Different resource levels were analyzed and a comparison with current practice at the studied port was done as well.

Lots of researchers studied the storage allocation problem for different categories of containers separately: import, export, transshipment, and empty containers. In de Castilho and Daganzo (1993), they presented methods for measuring the amount of handling effort required when two basic strategies were adopted to store import containers. One strategy tried to keep all the container blocks the same size and the other segregated containers according to the arrival time. These two strategies were compared in idealized situation only. Kim and Kim (1999a) studied the import container allocation problem where the arrival rate of import containers was constant, cyclic, and dynamic. A mathematical model with the objective of minimizing the total number of reshuffles was developed. Solution procedure and experiment results were provided for illustration. However, they did not consider the situation that some containers stayed in the storage yard after the free time limit. Taleb-Ibrahimi et al. (1993) described handling and storage strategies for export containers and quantified their performance according to the amount of space and number of handling moves required. The minimum storage space was determined for a given traffic and storage strategy, which could be of use for long-term planning and short-term operations. However, they did not consider the availability of yard cranes and yard tractors and the uncertainty in the vessel arriving times. In order to speed up the loading operation of export containers, Kim and Bae (1998) discussed how to reshuffle export containers in container terminals. A methodology was proposed to convert the current yard layout into a desirable layout by moving the fewest number of containers in the shortest traveling distance. More efficient algorithm was needed to solve the problem within a reasonable time period. Kim et al. (2000) proposed a methodology to determine the storage location of an arriving export container by considering its weight. A dynamic programming model was formulated to determine the storage location to minimize the number of reshuffles before loading. A solution procedure was also developed to obtain a decision tree for making real time decisions. The model was based on the assumption that containers were classified into several per-determined weight groups, but in practice, this might not be available before the arrival of the containers. In Kim and Park (2003), the storage space allocation problem for export containers was studied. A mixed integer linear programming model was formulated for the transfer system. Two heuristics that were based on the duration-of-stay of containers and the sub-gradient optimization technique were suggested to solve the MIP model. However, they did not consider the uncertainty in the amount of containers for every vessel over the planning horizon. Lee et al. (2006) studied the storage allocation problem in transshipment hubs. The consignment strategy, in which containers to the same destination vessel were stored in the same storage locations, was used to reduce the number of reshuffles. A high-low workload balancing protocol was used to reduce potential traffic congestion of prime movers. Two heuristics were proposed to solve the formulated model and experiments were conducted to evaluate the two heuristics proposed.

The world trades are typically imbalanced in terms of the number of export and import containers. Consequently, the relocation of empty containers has become one of the major problems faced by linear operators and port operators. Empty containers are usually stored separately from full containers. Empty container management involves dispatching empty containers in response to export customers and repositioning empty containers to storage depots or ports for future demand. Crainic et al. (1993) studied the empty container allocation problem and identified its basic structure and main characteristics. Two dynamic deterministic formulations for the single and multi-commodity cases were introduced, which could offer a general modeling framework for empty container allocation problem. There were a lot of assumptions for simplification as it was the first study for empty container reallocation. Shen and Khoong (1995) developed a decision support system to solve a large-scale multi-period distribution problem for empty containers. A network optimization model was built for the empty container positioning across ports. Constraint relaxation techniques were incorporated into the decision support system to minimize the perturbation to the existing decisions. They could extend their work by considering the related business processes, like loaded container movement, as well. Cheung et al. (1998) studied the dynamic empty container allocation problem to meet the customer's demand over time. A two-stage stochastic network model was formulated. A stochastic quasi-gradient method and a stochastic hybrid approximation procedure were applied to solve the problem. Some variations of the proposed methods were proposed to solve it within a shorter time period as well. The type of containers was not differentiated in the model and the demand had to be met in time, which is not true in practice for many container terminals.

2.4.4 Transfer Crane Deployment Problem

Transfer cranes take care of the storage and retrieval of containers. One of the decisions at the operational level is the deployment of transfer cranes. The transfer crane deployment problem was studied in Muller (1995), Kim and Kim (1997), Kim and Kim (1999d), Lin (2000), Moorthy and Hock-Guan, (2000), Chung et al. (2002), Zhang et al. (2002), Eisenberg (2003), Kim and Kim (2003), and Linn et al. (2003).

2.5 Inter-terminal Operations

The sea container terminal may be connected with another type of transport terminal, for example, rail station. The transport of containers between the sea terminal and other type of terminal is the main activity for inter-terminal operations. Inter-terminal transport becomes more important due to the steady growth of container traffic. Inter-terminal transport can be carried out by vehicles like multi-trailer system, automated guided vehicles. Multi-trailer system is a system which uses a truck that pulls a train consisting of several trailers, each with a capacity of 2 TEUs. The inter-terminal operations were studied in Duinkerken et al. (1996), Kurstjens (1996), Ottjes et al. (1996), Kozan (1997), Mastrolilli et al. (1998), Powell and Carvalho (1998), Ottjes et al. (1999), Kozan (2000), Alicke (2002), and Ottjes et al. (2002).

2.6 Outside Terminal Operations

Import containers will be collected by the consignee companies and export containers will be brought in by the shipper companies. The transport of containers between the shipping companies and the port usually is carried out by external trucks. An operational problem at the shipper side is the container packaging problem. The container packaging problem was studied in Chen et al. (1995), Davies and Bischoff (1999), Scheithauer (1999), and Eley (2003).

2.7 Integrated Terminal Study

In previous sections, problems for individual process in terminal operations are discussed. With these studies only optimal solution for part of the whole terminal can be obtained. It is necessary to conduct an integrated study to achieve the overall objective.

Most of the researchers used simulation as the methodology to study a whole terminal, for example, Gibson et al. (1992), Koh et al. (1996), Charnes et al. (1996), Konings (1996), Kozan (1997b), Nevins et al. (1998), Yun and Choi (1999), Duinkerken et al. (2000), Duinkerken et al. (2001), Duinkerken et al. (2002), Kia et al. (2002), Liu et al. (2002), Hartmann (2004), and Ottjes et al. (2006).

Some researchers developed decision support systems for a whole terminal, for example, van Hee and Wijbrands (1988b), van Hee et al. (1988a), and Murty et al. (2006).

Additional references dealing with the management of a whole container terminal are, e.g., Leeper (1988), Hayuth (1994), Mosca et al. (1994), Ramani (1996), Hulten (1997), Merkuryev et al. (1998), Thiers and Janssens (1998), Rizzoli et al. (1999), Veeke and Ottjes (1999), Rebollo et al. (2000), Saanen (2000), Bish (2001), Carrascosa et al. (2001), Meersmans and Wagelmans (2001a, 2001b), Henesey et al. (2002), Kim et al. (2002), Meersmans (2002), Shabayek and Yeung (2002), Veeke and Ottjes (2002), Mattfeld (2003), Yun and Choi (2003), and Hartmann (2004).

From the literature it can be seen that various problems associated with terminal operations have been addressed. These papers do not sufficiently address the particular needs of transshipment hubs, but more on general terminals which emphasize on import and export activities. For transshipment hubs, loading and unloading activities are both concentrated and need to be considered at the same time. This makes the planning problem much more challenging compared to port planning for general terminals where loading and unloading activities can be considered independently by having different dedicated storage areas for import and export activities. In this thesis, the yard management problem for a transshipment hub is studied, where transshipment of containers is the major activity and the yard activity is heavy. More specifically, a yard template problem is studied for export and transshipment containers, and a simulation study is conducted on the docking station concept for import container blocks.

Export and transshipment containers depart in large batches at designated time when the vessel comes. Currently, the operator in the studied port uses consignment strategy to group export and transshipment containers to dedicated sub-blocks to reduce the number of reshuffles, hence to reduce the vessel turnaround time. In order to handle the potential traffic congestion of prime movers, a high-low workload balancing protocol is proposed. However, they do not have any formal planning tool and the decisions are based on intuition and past experiences. Hence a tool is developed, which is able to provide a holistic and systematic way to address the problem which takes into consideration of port operator’s actual requirements. The tool is based on an MIP formulation and is particularly useful for transshipment hubs. So far this is the first study to address the yard template problem with the consignment strategy and high-low workload balancing protocol for a transshipment hub.

In contrast, import containers arrive at the storage yard in large batches and in a predicted fashion, but depart one by one in an unpredictable order. Therefore, import containers are usually stored in separate blocks from export and transshipment containers so as to facilitate the ease of customer retrieval. In order to manage the competing demands for yard cranes in import blocks, a docking station concept is proposed to change the current horizontal layout (i.e., yard blocks are parallel to the wharf) for import container blocks to a vertical layout (i.e., yard blocks are perpendicular to the wharf). With the docking station concept, internal prime movers and external trucks are segregated, which allows the port operator the flexibility of assigning yard crane service priority to internal prime movers and hence the ship turnaround time can be reduced when required. To verify the effectiveness of the docking station concept, two simulation models for the base layout and the proposed perpendicular layout are built respectively.

The details of the yard template problem for export and transshipment containers and the simulation study on the docking station concept for import containers are discussed in the following chapters.

3 Formulating the Yard Template Problem for Export and Transshipment Containers

3.1 Problem Definition

One of the most important performance measures of container terminals is the vessel turnaround time, which should be kept at the minimum level. Transshipment hub ports usually handle a high volume of containers and most containers unloaded from one vessel will be eventually loaded onto other vessels in the port. The loading and unloading activities are both concentrated and they need to be considered at the same time. To reduce the number of reshuffles, which will help to reduce the vessel turnaround time, the port operator uses the consignment strategy to store export and transshipment containers going to the same destination vessel together at dedicated storage areas. Import containers have different characteristics so that they are stored in separate blocks from export and transshipment containers.

The consignment strategy is based on the storage yard configuration. For planning purpose, the studied terminal is divided into sections, and vessels are assigned to sections rather than the exact berth locations. This is due to the uncertainties in their arrival times and by doing so, it provides more flexibility during the operation. Due to these reasons, when the container space allocation is conducted within a section, the planned berth of a vessel is not considered. To manage the yard allocation process more efficiently, the port operator organizes each section of the storage yard into several blocks as shown in Figure

3.1. The depth of each block is 6 lanes, and the length of each block is 40 slots (each slot can store one 20-feet container). The stacking height is 5 containers (which is called tier in this thesis). Every block is further divided into 5 sub-blocks, where the length of each sub-block is 8 slots. The basic unit for the yard storage allocation process is at the sub-block level, i.e., for the consignment strategy, the incoming containers that are going to the same destination vessel are assigned to the same sub-blocks. Currently, the size of a sub-block is determined based on the practice in the studied port. If the sub-block is too big, one yard crane cannot handle all the containers over a shift. On the other hand, if the sub-block is too small, the benefit from concentrating containers in a location will be lost as the yard crane may need to move to other sub-blocks to get the jobs which will result in loss of productivity due to traveling.

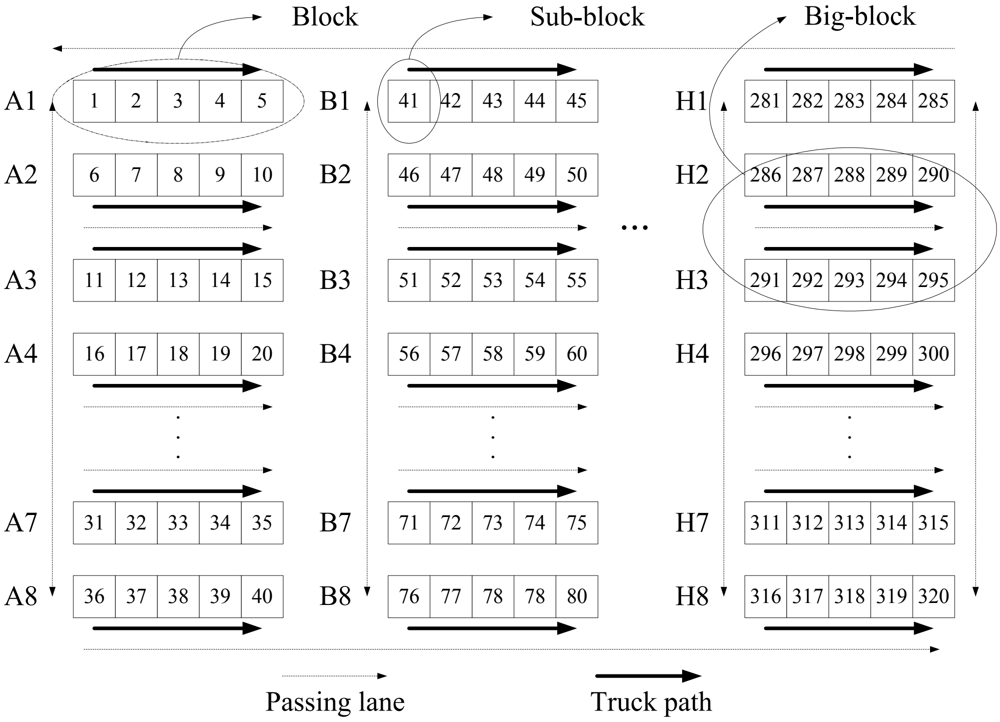

Figure 3.1 The storage yard configuration in the studied port

Another problem in the studied port is the traffic congestion of prime movers, which is one of the direct results of high volume of containers. As shown in Figure 3.1, there is a dedicated lane for the movement of containers by prime movers (the “truck path”) and a separate “passing” lane strictly to allow trucks to pass each other when required. The passing lane is narrow and it is shared between two neighboring container blocks. Traffic congestion may happen when too much workload needs to be handled within a small area at the same time. For example, if there are lots of container movements in Sub-blocks 7 and 12 (see Figure 3.1), there will be many prime movers waiting or moving nearby. This will result in traffic congestion. Similarly, if the workload in Sub-block 6 is heavy, the prime movers waiting at Sub-block 6 may block other prime movers from going to Sub-block 7 since they share the same truck path.

To ensure a smooth flow of traffic, the port operator has imposed the following restrictions during the planning stage:

- When a sub-block is in the loading process, its neighboring sub-blocks should not have any loading or unloading activities.
- There should not be two or more neighboring sub-blocks which are having heavy unloading activities in one shift.

To incorporate these restrictions into the mathematical model, the high-low workload balancing protocol is introduced. On one hand, high workload needs to be concentrated at the sub-block so that the yard crane will be heavily utilized and it does not need to waste time to travel between sub-blocks to perform work. On the other hand, low workload is used to separate heavy workload in order to reduce the potential traffic congestion. The ranges of high workload and low workload do not overlap. For example, the range of high workload is set between 50 and 100 containers per shift, while the low workload is set between 0 and 20 containers per shift by the operator of the studied port.

To capture the possible traffic congestion in the storage yard, the neighborhood structure between the sub-blocks is defined as follows. A sub-block is a neighbor of another sub-block if these two sub-blocks share the same truck path or passing lane directly. For example, in Figure 3.1, Sub-block 7 is a neighbor of Sub-blocks 6 and 12. Sub-block 7 is not a neighbor of Sub-block 2 because they do not share any truck path or passing lane. Sub-block 7 is not a neighbor of Sub-block 9 or 13 because they only share the truck path or passing lane indirectly. To incorporate the neighborhood structure into the mathematical model, a vicinity matrix is introduced, in which a value of 1 means the sub-blocks are neighbors of each other and 0 means they are not. More specifically,

$R_{ii'}$ = 1, if Sub-block i is a neighbor of Sub-block i'.

= 0, otherwise.

Table 3.1 shows part of the vicinity matrix for the yard shown in Figure 3.1.

Table 3.1 Part of the vicinity matrix for the yard configuration shown in Figure 3.1

$R_{ii'}$	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2		0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3			0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4				0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6		0	1	0	0	0	1	0	0	0	0	0	0	0	0	0
7			0	1	0	0	0	1	0	0	0	0	0	0	0	0
8				0	1	0	0	0	1	0	0	0	0	0	0	0
9					0	1	0	0	0	1	0	0	0	0	0	0
10						0	0	0	0	0	1	0	0	0	0	0
11							0	1	0	0	0	0	0	0	0	0
12								0	1	0	0	0	0	0	0	0
13									0	1	0	0	0	0	0	0
14										0	1	0	0	0	0	0
15											0	0	0	0	0	0
16												0	1	0	0	0
17													0	1	0	0
18														0	1	0
19															0	1
20																0

For the yard template problem, the set of sub-blocks reserved for each vessel should be determined first. Given these sets of reserved sub-blocks, the number of export and transshipment containers and the smallest number of yard cranes to deploy in each shift will be determined. The total loading activities in each sub-block can be derived from above decisions (to simplify the discussion, in the subsequent sections, “containers” are referred to as both export and transshipment containers unless specified otherwise). Currently, the port operator does not have any formal planning model to solve the yard template problem, and the decisions are based on intuition and past experiences. As a remedy means, a mathematical model that incorporates the concepts discussed above is developed. The detailed model is presented in the next section.

3.2 Model Development

In this section, the yard template problem is formulated as a mixed-integer linear programming model.

3.2.1 Model Assumptions

The following assumptions are made when developing the model for the yard template problem.

- The incoming containers are grouped according to their destination vessel only. Assigning a specific yard-bay to a container group (with the same destination port, the same size or weight) is out of the scope of this thesis.
- The yard template is static which means the assignment of a sub-block to a vessel is fixed and it does not change from one shift to another.
- A sub-block can be assigned to only one vessel during the whole planning horizon.
- The time span of the model is 7 days with 3 planning periods in each day. Each planning period corresponds to an 8-hour working shift.
- The amount of containers arriving in each shift is assumed to be given and will repeat weekly (this implies the planning period can be wrapped around). The actual number can vary, but for planning purpose it can be assumed to be deterministic and an input to the model.

- At any given time, if a sub-block is in the loading process, a dedicated yard crane will be assigned to that sub-block since loading activities have higher priority. However, a sub-block with high unloading workload can share a yard crane with its neighbors with low unloading workload. Also, several sub-blocks with low unloading workload can share a yard crane.

- Two different types of containers are handled in container terminals. They are 20-feet containers and 40-feet containers. As a sub-block consists of a few lanes, it is possible to have a mixture of different types of containers in one sub-block.

- All containers that arrive in a given shift will be stored in a sub-block until they are loaded onto the destination vessel. The loading activities in any sub-block have to be completed within a few shifts which is required by the operator of the studied port.

- A yard crane assigned to a particular block should work until the end of the shift.

3.2.2 Notations

The model parameters are as follows:

I	the number of sub-blocks under consideration.
J	the number of vessels under consideration in the planning horizon.
K	the number of blocks under consideration.
T	the number of shifts under consideration in the planning horizon.

A_j	the smallest number of sub-blocks that should be reserved for Vessel j, $1 \le j \le J$. It is necessary to ensure that the total number of sub-blocks in the storage yard is not less than the summation of all the A_j, $1 \le j \le J$.
B_k	the set of sub-blocks that belong to Block k, $1 \le k \le K$.
C_k	the maximum number of yard cranes allowed to reside in Block k at any one time, $1 \le k \le K$.
E_j	the maximum number of shifts allowed to load the containers for Vessel j, $1 \le j \le J$.
N_i	the set of sub-blocks that are neighbors of Sub-block i, $1 \le i \le I$.
WX_{jt}	the number of 20-feet containers arriving at the terminal in Shift t and will be loaded onto Vessel j finally. It is given and input to the model, $1 \le j \le J$, $1 \le t \le T$.
WY_{jt}	the number of 40-feet containers arriving at the terminal in Shift t and will be loaded onto Vessel j finally. It is given and input to the model, $1 \le j \le J$, $1 \le t \le T$.
VL_{jt}	= 1, if Vessel j is in the loading process in Shift t, $1 \le j \le J$, $1 \le t \le T$. = 0, otherwise.
CS	the space capacity of each sub-block in terms of TEUs, which is 240 (5 tiers×6 lanes×8 slots) in this thesis.
CC	the capacity of each yard crane in terms of container moves per shift, which is 100 in this thesis according to the current practice in the studied port.

HL the lowest value that a high workload can take.

HU the highest value that a high workload can take.

LL the lowest value that a low workload can take.

LU the highest value that a low workload can take.

M a sufficiently large positive value.

Note: Subscript i is for sub-block, j for vessel, k for block, and t for shift.

The decision variables are as follows:

d_{kt} the number of yard cranes allocated to Block k for unloading in Shift t, $1 \leq k \leq K$, $1 \leq t \leq T$.

h_{it} = 1, if the total workload allocated to Sub-block i for unloading in Shift t is high, that is, $HL \leq \sum_{j=1}^{J} \left(x_{ijt} + y_{ijt}\right) \leq HU$, $1 \leq i \leq I$, $1 \leq t \leq T$.

= 0, if the total workload allocated to Sub-block i for unloading in Shift t is low, that is, $LL \leq \sum_{j=1}^{J} \left(x_{ijt} + y_{ijt}\right) \leq LU$, $1 \leq i \leq I$, $1 \leq t \leq T$.

x_{ijt} the number of 20-feet containers that are allocated to Sub-block i for unloading in Shift t if Sub-block i is reserved for Vessel j, $1 \leq i \leq I$, $1 \leq j \leq J$, $1 \leq t \leq T$.

= 0, if Sub-block i is not reserved for Vessel j.

y_{ijt} the number of 40-feet containers that are allocated to Sub-block i for unloading in Shift t if Sub-block i is reserved for Vessel j, $1 \leq i \leq I$, $1 \leq j \leq J$, $1 \leq t \leq T$.

= 0, if Sub-block i is not reserved for Vessel j.

z_{ij} = 1, if Sub-block i is reserved for Vessel j, $1 \leq i \leq I$, $1 \leq j \leq J$.

= 0, otherwise.

3.2.3 Model Formulation

In the proposed model, the total number of yard crane shifts required to handle all the workload should be minimized. Each yard crane shift corresponds to one active yard crane in one shift. The problem here is not to determine the minimum total number of yard cranes to carry out the work for the entire port during the whole planning horizon. Instead, the smallest number of yard cranes to be assigned in each shift will be determined so that the operating cost can be reduced by putting less yard cranes to use. Yard cranes are very expensive equipment and should be utilized highly. In addition, the operating cost for yard crane is high. If there is not a good yard template or the containers are not allocated properly, it may result in requiring more yard cranes to handle the same amount of workload in each shift. YTP is used to denote the model for the yard template problem. The model is formulated as follows.

$$\text{(YTP)} \quad Min\ w = \sum_{k=1}^{K}\sum_{t=1}^{T} d_{kt} + \sum_{t}^{T}\sum_{i}^{I}\sum_{j}^{J} z_{ij} VL_{jt} \tag{3.1}$$

Subject to:

$$\sum_{j=1}^{J} z_{ij} \le 1 \qquad \forall\ 1 \le i \le I \tag{3.2}$$

$$\sum_{i=1}^{I} z_{ij} \ge A_j \qquad \forall\ 1 \le j \le J \tag{3.3}$$

$$\sum_{i=1}^{I} x_{ijt} = WX_{jt} \qquad \forall\ 1 \le j \le J,\ 1 \le t \le T \tag{3.4}$$

$$\sum_{i=1}^{I} y_{ijt} = WY_{jt} \qquad \forall\ 1 \le j \le J,\ 1 \le t \le T \tag{3.5}$$

$$x_{ijt} + y_{ijt} \le z_{ij} CC \qquad \forall\ 1 \le i \le I,\ 1 \le j \le J,\ 1 \le t \le T \tag{3.6}$$

$$\sum_{t=1}^{T} \sum_{j=1}^{J} \left(x_{ijt} + 2y_{ijt} \right) \le CS \qquad \forall\ 1 \le i \le I \tag{3.7}$$

$$\sum_{t=1}^{T} \sum_{j=1}^{J} \left(x_{ijt} + y_{ijt} \right) \le \sum_{j=1}^{J} z_{ij} E_j CC \qquad \forall\ 1 \le i \le I \tag{3.8}$$

$$\sum_{i \in B_k} \sum_{j=1}^{J} \left(x_{ijt} + y_{ijt} \right) \le d_{kt} CC \qquad \forall\ 1 \le k \le K,\ 1 \le t \le T \tag{3.9}$$

$$HL + (LL - HL)(1 - h_{it}) \le \sum_{j=1}^{J} \left(x_{ijt} + y_{ijt} \right) \le LU + (HU - LU) h_{it}$$

$$\forall\ 1 \le i \le I,\ 1 \le t \le T \tag{3.10}$$

$$\sum_{i' \in N_i} \sum_{j=1}^{J} \left(x_{i'jt} + y_{i'jt} \right) \le M \left(1 - \sum_{j=1}^{J} z_{ij} VL_{jt} \right) \qquad \forall\ 1 \le i \le I,\ 1 \le t \le T \tag{3.11}$$

$$\sum_{i' \in N_i \cup \{i\}} h_{i't} \le 1 \qquad \forall\ 1 \le i \le I,\ 1 \le t \le T \tag{3.12}$$

$$d_{kt} + \sum_{i \in B_k} \sum_{j=1}^{J} z_{ij} VL_{jt} \le C_k \qquad \forall\ 1 \le k \le K,\ 1 \le t \le T \tag{3.13}$$

$$x_{ijt} \ge 0\,,\ y_{ijt} \ge 0 \qquad \forall\ 1 \le i \le I,\ 1 \le j \le J,\ 1 \le t \le T \tag{3.14}$$

$$h_{it} \in \{ 0,1 \} \qquad \forall \ 1 \le i \le I, 1 \le t \le T \qquad (3.15)$$

$$z_{ij} \in \{ 0,1 \} \qquad \forall \ 1 \le i \le I, 1 \le j \le J \qquad (3.16)$$

$$d_{kt} \in \{ \text{Positive Integer} \} \qquad \forall \ 1 \le k \le K, 1 \le t \le T \qquad (3.17)$$

In the objective function, the first part is the total number of yard crane shifts exclusively for unloading; while the second part is the total number of yard crane shifts exclusively for loading.

Constraint (3.2) ensures that a sub-block can be reserved for at most one vessel during the whole planning horizon. In other words, a sub-block cannot be shared by two vessels and no change on the reservation can be made once the reservation is made. Constraint (3.3) ensures that sufficient sub-blocks are reserved for each vessel.

Constraints (3.4) and (3.5) ensure that all the workload arriving in each shift for each vessel will be allocated to corresponding storage locations. Constraint (3.6) ensures that an incoming container can only be allocated to a sub-block that is reserved for the destination vessel of that container.

Constraint (3.7) ensures the space capacity restriction of each sub-block. Constraint (3.8) ensures that the containers in each sub-block should be loaded onto the destination vessel within a certain time span. Different vessels may have different loading time requirements. Constraint (3.9) ensures that the yard cranes allocated to each block for unloading can handle all the unloading workload in each shift.

To make full use of yard cranes, workload allocated to each sub-block in each shift should be either high or low. In this model, Constraint (3.10) is used to ensure this restriction. Constraint (3.11) ensures that all the neighbors of a sub-block in the loading process cannot accept any workload in that shift. Constraint (3.12) ensures that high unloading workload cannot be allocated to two sub-blocks that are neighbor of each other in the same shift.

As a result of the limitation of the length of the chassis trailer and due to safety consideration, each block can hold at most a certain number of yard cranes at any one time. Constraint (3.13) ensures this restriction. In addition, one yard crane is required for each sub-block in the loading process, and hence the number of sub-blocks in the loading process is exactly equal to the number of yard cranes assigned to that block for loading. Constraints (3.14), (3.15), (3.16), and (3.17) are non-negative and integer restrictions.

The model is not easy to solve because there are too many integer and binary variables. In addition, the MIP structure is poor because of the high-low workload Constraints (3.10), (3.11), and (3.12). For example, the optimal solution to the LP relaxation of the model is seldom close to the feasible solution of the original model because it is easy to get a solution in which the workload assigned to some sub-block is neither high nor low. The model is solved with CPLEX 8.1 and implemented into C++ programming. The program run out of memory and the model can not be populated into CPLEX model as a result of too many variables. To tackle this problem, the problem is relaxed assuming that the set of sub-blocks reserved for each vessel is given and is treated as input to the model in the next chapter.

4 Formulating and Solving the Yard Allocation Problem

The model formulation for the yard template problem in Chapter 3 has too many variables and constraints and cannot be solved by CPLEX directly. Hence, in order to solve the yard template problem, first the set of sub-blocks reserved for each vessel is assumed to be given and is treated as input to the model. Given this set of sub-blocks reserved for each vessel, the number of export and transshipment containers and the smallest number of yard cranes to deploy in each shift will be determined. This is called the yard allocation problem and is solved in this chapter.

4.1 Model Development

4.1.1 Notations

The model parameters are as follows:

I	the number of sub-blocks under consideration.
J	the number of vessels under consideration in the planning horizon.
K	the number of blocks under consideration.
T	the number of shifts under consideration in the planning horizon.
B_k	the set of sub-blocks that belong to Block k, $1 \le k \le K$.

C_k	the maximum number of yard cranes allowed to reside in Block k at any one time, $1 \le k \le K$.
F_{ij}	$= 1$, if Sub-block i is reserved for Vessel j, $1 \le i \le I$, $1 \le j \le J$. $= 0$, otherwise.
E_j	the maximum number of shifts allowed to load the containers for Vessel j, $1 \le j \le J$.
G_i	the maximum number of shifts allowed to load the containers for Sub-block i, $1 \le i \le I$. $G_i = \sum_{j=1}^{J} F_{ij} E_j$.
L_i	the set of shifts in which Sub-block i is in the loading process, $1 \le i \le I$.
N_i	the set of sub-blocks that are neighbors of Sub-block i, $1 \le i \le I$.
V_j	the set of sub-blocks that are reserved for Vessel j, $1 \le j \le J$.
WX_{jt}	the number of 20-feet containers arriving at the terminal in Shift t and will be loaded onto Vessel j finally. It is given and input to the model, $1 \le j \le J$, $1 \le t \le T$.
WY_{jt}	the number of 40-feet containers arriving at the terminal in Shift t and will be loaded onto Vessel j finally. It is given and input to the model, $1 \le j \le J$, $1 \le t \le T$.
NL_{kt}	the number of sub-blocks in the loading process in Block k in Shift t, $1 \le k \le K$, $1 \le t \le T$.
CS	the space capacity of each sub-block in terms of TEUs, which is 240 (5 tiers×6 lanes×8 slots) in this thesis.

CC	the capacity of each yard crane in terms of container moves per shift, which is 100 in this thesis according to the current practice in the studied port.
HL	the lowest value that a high workload can take.
HU	the highest value that a high workload can take.
LL	the lowest value that a low workload can take.
LU	the highest value that a low workload can take.
M	a sufficiently large positive value.

Note: Subscript i is for sub-block, j for vessel, k for block, t for shift.

The decision variables are as follows:

d_{kt}	the number of yard cranes allocated to Block k for unloading in Shift t, $1 \leq k \leq K$, $1 \leq t \leq T$.
h_{it}	= 1, if the total workload allocated to Sub-block i for unloading in Shift t is high, that is, $HL \leq x_{it}+y_{it} \leq HU$, $1 \leq i \leq I$, $1 \leq t \leq T$. = 0, if the total workload allocated to Sub-block i for unloading in Shift t is low, that is, $LL \leq x_{it}+y_{it} \leq LU$, $1 \leq i \leq I$, $1 \leq t \leq T$.
x_{it}	the number of 20-feet containers that are allocated to Sub-block i for unloading in Shift t, $1 \leq i \leq I$, $1 \leq t \leq T$.

y_{it} the number of 40-feet containers that are allocated to Sub-block i for unloading in Shift t, $1 \le i \le I$, $1 \le t \le T$.

4.1.2 Model Formulation

The objective of the yard allocation problem is as same as that of the yard template problem, which is to minimize the total number of yard crane shifts required to handle all the workload. However, for the yard allocation problem the set of sub-blocks reserved for each vessel is given. This means the total number of yard crane shifts for loading is fixed because a dedicated yard crane will be assigned to each sub-block in the loading process. Hence, only the yard crane shifts for unloading will be minimized in the yard allocation problem. YAP is used to denote the model for the yard allocation problem. The model is formulated as follows.

$$\text{(YAP)}\ \ Min\ w = \sum_{k=1}^{K} \sum_{t=1}^{T} d_{kt} \tag{4.1}$$

Subject to:

$$\sum_{i \in V_j} x_{it} = WX_{jt} \qquad \forall\ 1 \le j \le J,\ 1 \le t \le T \tag{4.2}$$

$$\sum_{i \in V_j} y_{it} = WY_{jt} \qquad \forall\ 1 \le j \le J,\ 1 \le t \le T \tag{4.3}$$

$$\sum_{t=1}^{T} \left(x_{it} + 2y_{it} \right) \le CS \qquad \forall\ 1 \le i \le I \tag{4.4}$$

$$\sum_{t=1}^{T} \left(x_{it} + y_{it} \right) \le G_i CC \qquad \forall\ 1 \le i \le I \tag{4.5}$$

$$\sum_{i \in B_k} \left(x_{it} + y_{it} \right) \le d_{kt} CC \qquad \forall\ 1 \le k \le K,\ 1 \le t \le T \tag{4.6}$$

$$HL + (LL - HL)(1 - h_{it}) \le x_{it} + y_{it} \le LU + (HU - LU)h_{it}$$

$$\forall\ 1 \le i \le I,\ 1 \le t \le T \qquad (4.7)$$

$$x_{i't} = 0\,,\ y_{i't} = 0 \qquad \forall\ i' \in N_i,\ t \in L_i,\ 1 \le i \le I \qquad (4.8)$$

$$\sum_{i' \in N_i \cup \{i\}} h_{i't} \le 1 \qquad \forall\ 1 \le i \le I,\ 1 \le t \le T \qquad (4.9)$$

$$d_{kt} + NL_{kt} \le C_k \qquad \forall\ 1 \le k \le K,\ 1 \le t \le T \qquad (4.10)$$

$$x_{it} \ge 0\,,\ y_{it} \ge 0 \qquad \forall\ 1 \le i \le I,\ 1 \le t \le T \qquad (4.11)$$

$$h_{it} \in \{\ 0, 1\ \} \qquad \forall\ 1 \le i \le I,\ 1 \le t \le T \qquad (4.12)$$

$$d_{kt} \in \{Positive\ Integer\} \qquad \forall\ 1 \le i \le I,\ 1 \le k \le K,\ 1 \le t \le T \qquad (4.13)$$

Constraints (4.2) and (4.3) ensure that all the workload arriving at the terminal in each shift for each vessel will be allocated to corresponding storage locations. Constraint (4.4) ensures the space capacity restriction of each sub-block. Constraint (4.5) ensures that the containers in each sub-block should be loaded onto the destination vessel within a certain time span. Different vessels may have different loading time requirements. Constraint (4.6) ensures that the yard cranes allocated to each block for unloading can handle all the unloading workload in each shift.

To make full use of yard cranes, workload allocated to each sub-block in each shift should be either high or low. In this model, Constraint (4.7) is used to ensure this restriction. Constraint (4.8) ensures that all the neighbors of a sub-block in the loading process cannot accept any workload in that shift. Constraint (4.9) ensures that high unloading workload cannot be allocated to two sub-blocks that are neighbor of each other in the same shift.

As a result of the limitation of the length of the chassis trailer and due to safety consideration, each block can hold at most a certain number of yard cranes at any one time. Constraint (4.10) ensures this restriction. In addition, one yard crane is required for each sub-block in the loading process, and hence the number of sub-blocks in the loading process is exactly equal to the number of yard cranes assigned to that block for loading. Constraints (4.11), (4.12), and (4.13) are non-negative and integer restrictions.

The model for the yard allocation problem has much less decision variables and constraints, which may make it easier to solve. In the next section the yard allocation problem is solved with CPLEX 8.1.

4.2 Numerical Experiments

In this section, the model for the yard allocation problem is first tested using two sets of input data for the simplified small-scale problem. And then it is used to solve a large-scale problem close to a moderate terminal.

4.2.1 Small-scale Problem Experiment

For the small-scale problem, there are 8 blocks arrayed in 4 rows and 2 columns in the storage yard which is shown in Figure 4.1. A vicinity matrix can be determined easily from the yard configuration.

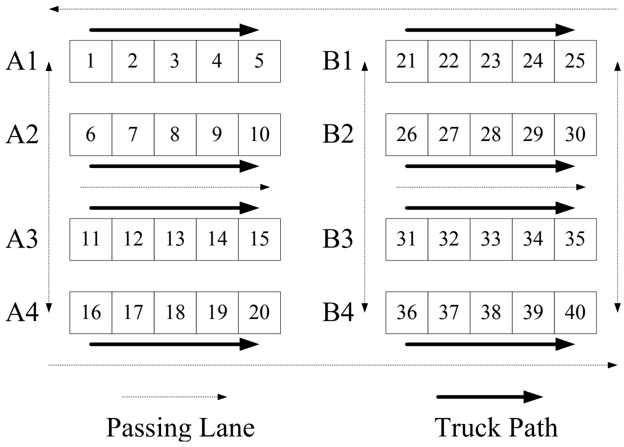

Figure 4.1 Yard configuration for the small-scale problem

It is assumed that there is exactly one vessel being loaded in each shift. The lowest and highest value that a high workload can take are 100 and 200, respectively; and those for a low workload are 0 and 100, respectively. The capacity of one yard crane is 200 container moves per shift. The maximum number of yard cranes that can reside in each block is 2 at any one time. Only 5 shifts are considered in the model. The parameters are drawn from the real practice with some minor changes to make the model feasible.

The model for the yard allocation problem is implemented in C++ and run on a Pentium IV computer (CPU: 2.4GHz, Memory: 512M). The mixed-integer programming model is solved using CPLEX 8.1 with concert technology and C++ optimization modeling library and interface.

The computational results are summarized in Tables 4.1 and 4.2. Two sets of input data are used to conduct the small-scale problem experiments. The utilization is defined as the ratio of the total storage space occupied by all the unloading containers to the total storage space in the storage yard.

Table 4.1 Results of the YAP model for the small-scale problem (Case 1)

Utilization	Computation Time (s)	Solution Status	Objective Value (# of YCSs)
0.05	1	Optimal	10
0.10	1	Optimal	10
0.15	1	Optimal	10
0.20	1	Optimal	10
0.25	1	Optimal	10
0.30	274	Optimal	15
0.35	340	Optimal	15
0.40	1	Optimal	15
0.45	34196.03	Out of memory	20 (LB[*]=17.56, Gap[^]=13.9%)
0.50	3833	Optimal	20
0.55	110	Optimal	21
0.60	13304	Optimal	25
0.65	114	Optimal	25
0.70	6	Optimal	25
0.75	11602	Optimal	30
0.80	6	Optimal	30
0.85	328	Optimal	35
0.90	16	Optimal	35
0.95	1	Infeasible	-
1.00	1	Infeasible	-

[*] LB: lower bound.

[^] Gap = (20-17.56)/17.56

As shown in Tables 4.1 and 4.2, for relatively low utilization scenarios (with utilization less than 0.45) an optimal solution can be obtained easily. For moderate utilization scenarios (with utilization between 0.45 and 0.60) it takes a longer time to solve as there are too many choices to allocate containers to their storage locations. In addition, too many choices can result in a huge branch and bound tree, which may cause the computer to run out of memory (for example, Case 1 with utilization of 0.45). For high utilization scenarios (with utilization greater than 0.90 for Case 1 and 0.70 for Case 2), the problem is prone to be infeasible as the capacity constraints cannot be satisfied.

Table 4.2 Results of the YAP model for the small-scale problem (Case 2)

Utilization	Computation Time (s)	Solution Status	Objective Value (# of YCSs)
0.05	1	Optimal	10
0.10	1	Optimal	10
0.15	1	Optimal	10
0.20	1	Optimal	11
0.25	1	Optimal	12
0.30	1	Optimal	13
0.35	2	Optimal	15
0.40	91	Optimal	17
0.45	1	Optimal	18
0.50	936	Optimal	21
0.55	643	Optimal	23
0.60	106	Optimal	23
0.65	36	Optimal	27
0.70	26	Optimal	28
0.75	1	Infeasible	-
0.80	1	Infeasible	-
0.85	1	Infeasible	-
0.90	1	Infeasible	-
0.95	1	Infeasible	-
1.00	1	Infeasible	-

4.2.2 Large-scale Problem Experiment

For the large-scale problem, there are 64 blocks arrayed in 8 rows and 8 columns in the storage yard, which is shown in Figure 3.1. The time horizon is 7 days with 3 shifts in each day. The scale of the problem (around 5 million TEUs) is comparable to a section of the studied terminal.

The lowest and highest value that a high workload can take are 50 and 100, respectively; and those for low workload are 0 and 20, respectively. The capacity of one yard crane is 100 container moves per shift. The maximum number of yard cranes that can reside in each block is 2 at any one time. The parameters are fairly reflective on any terminal with high traffic intensity. Due to the confidentiality of the data, instead of using the actual data, the data is generated based on the feature of the pattern provided by the operator in the studied port. If actual data for other transshipment ports are available, they can also be plugged into the model to solve the problem.

The large-scale problem is also implemented in C++ and run on the same computer as that for the small-scale problem. The computational results are summarized in Table 4.3.

Table 4.3 Results of the YAP model for the large-scale problem

Utilization	Computation Time (s)	Solution Status	Result Details
0.05	30346.80	Out of memory	104, Gap^=15.9% (LB*=89.758)
0.10	24640.89	Out of memory	149, Gap=56.4% (LB=95.257)
0.15	26680.84	Out of memory	188, Gap=57.1% (LB=119.68)
0.20	25721.92	Out of memory	217, Gap=56.5% (LB=138.68)
0.25	64013.97	Out of memory	221, Gap=49.9% (LB=147.47)
0.30	1	Infeasible	-
0.35	1	Infeasible	-
0.40	11060.56	Out of memory	285, Gap=25.3% (LB=227.50)
0.45	11037.89	Out of memory	315, Gap=23.2% (LB=255.58)
0.50	13642.97	Out of memory	352, Gap=23.3% (LB=285.53)
0.55	16114.41	Out of memory	364, Gap=16.3% (LB=313.04)
0.60	25609.03	Out of memory	396, Gap=15.7% (LB=342.27)
0.65	23034.78	Out of memory	415, Gap=12.1% (LB=370.17)
0.70	26500.50	Out of memory	453, Gap=13.6% (LB=398.84)
0.75	58407.64	Out of memory	486, Gap=13.3% (LB=429.09)
0.80	220047.20	Out of memory	524, Gap=14.5% (LB=457.53)
0.85	196890.58	Out of memory	557, Gap=14.7% (LB=485.74)
0.90	554958.83	Out of memory	605, Gap=17.5% (LB=514.69)
0.95	684439.48	Out of memory	623, Gap=14.7% (LB=543.12)
1.00	845611.17	Out of memory	647, Gap=13.7% (LB=568.83)

* LB: Lower bound

^ Gap = (104-89.758)/89.758

As shown in Table 4.3, the large-scale problem cannot be solved to optimality within a reasonable time and always terminates as a result of insufficient memory. The results before running out of memory are presented. In addition, for scenarios with utilization of 0.3 and 0.35 it turns out to be infeasible. This is due to Constraint (4.7), which restricts the workload allocated to each sub-block in each shift to be either high or low. For some input data it is impossible to satisfy these constraints. For example, it is impossible to assign 45 containers to two sub-blocks in one shift.

For such a large-scale problem, the MIP model (in total there are 7,392 integer variables and 24,370 constraints) is too complex to solve to optimality within a reasonable time. Therefore heuristic algorithms should be developed to find a satisfactory solution to meet the requirement of the port operator. To evaluate the performances of the heuristics, it is necessary to find a lower bound.

4.3 Finding a Lower Bound

One possible way to find a lower bound of the yard allocation problem is to solve each shift independently. Under this assumption, Constraints (4.4) and (4.5) can be removed from the formulation. LBP is used to denote the model to find a lower bound for the yard allocation problem, which is shown as follows.

$$\text{(LBP)}\ Min\ w = \sum_{k=1}^{K} d_{kt} \tag{4.14}$$

Subject to:

$$\sum_{i \in V_j} x_{it} = WX_{jt} \qquad \forall\ 1 \le j \le J \qquad (4.15)$$

$$\sum_{i \in V_j} y_{it} = WY_{jt} \qquad \forall\ 1 \le j \le J \qquad (4.16)$$

$$\sum_{i \in B_k} \left(x_{it} + y_{it} \right) \le d_{kt} CC \qquad \forall\ 1 \le k \le K \qquad (4.17)$$

$$HL + (LL - HL)(1 - h_{it}) \le x_{it} + y_{it} \le LU + (HU - LU) h_{it}$$

$$\forall\ 1 \le i \le I \qquad (4.18)$$

$$x_{i't} = 0\,, y_{i't} = 0 \qquad \forall\ i' \in N_i,\ t \in L_i,\ 1 \le i \le I \qquad (4.19)$$

$$\sum_{i' \in N_i \cup \{i\}} h_{i't} \le 1 \qquad \forall\ 1 \le i \le I \qquad (4.20)$$

$$d_{kt} + NL_{kt} \le C_k \qquad \forall\ 1 \le k \le K \qquad (4.21)$$

$$x_{it} \ge 0\ \ y_{it} \ge 0 \qquad \forall\ 1 \le i \le I \qquad (4.22)$$

$$h_{it} \in \{\ 0, 1\} \qquad \forall\ 1 \le i \le I \qquad (4.23)$$

$$d_{kt} \in \{\ Positive\ Integer\ \} \qquad \forall\ 1 \le k \le K \qquad (4.24)$$

The objective of the model is to minimize the total number of yard cranes used in the current Shift *t*. All constraints ensure the same restrictions as the model YTP but only for the current Shift *t*.

The LBP model is also implemented in C++ and run on the same computer as that for the yard allocation model YAP. The same input data as those for both the small-scale problems and the large-scale problem are used. The computational results are presented in Tables 4.4, 4.5 and 4.6.

Table 4.4 Results of the LBP model for the small-scale problem (Case 1)

Utilization	Results of YAP	Lower Bound from LBP	
		Computation Time (s)	Objective Value (YCSs)
0.05	10	1	10
0.10	10	1	10
0.15	10	1	10
0.20	10	1	10
0.25	10	1	10
0.30	15	1	15
0.35	15	1	15
0.40	15	1	15
0.45	20 (LB* =17.56, Gap^ =13.9%)	1	20
0.50	20	1	20
0.55	21	1	20
0.60	25	1	25
0.65	25	1	25
0.70	25	1	25
0.75	30	1	30
0.80	30	1	30
0.85	35	1	35
0.90	35	1	35
0.95	Infeasible	-	-
1.00	Infeasible	-	-

* LB: lower bound

^ Gap = (20-17.56)/17.56

Table 4.5 Results of the LBP model for the small-scale problem (Case 2)

Utilization	Results of YAP	Lower Bound from LBP	
		Computation Time (s)	Objective Value (YCSs)
0.05	10	1	10
0.10	10	1	10
0.15	10	1	10
0.20	11	1	11
0.25	12	1	12
0.30	13	1	13
0.35	15	1	15
0.40	17	1	17
0.45	18	1	18
0.50	21	1	21
0.55	23	1	23
0.60	23	1	23

0.65	27	1	26
0.70	28	1	28
0.75	Infeasible	-	-
0.80	Infeasible	-	-
0.85	Infeasible	-	-
0.90	Infeasible	-	-
0.95	Infeasible	-	-
1.00	Infeasible	-	-

As shown in Tables 4.4 and 4.5, the model LBP for the small-scale problem gives good lower bound for most scenarios. Most of them have the same objective value as the optimal solution. As the model LBP for each shift can be solved efficiently, the lower bound for the yard allocation model YAP can be obtained within a short time period.

Table 4.6 Results of the LBP model for the large-scale problem

Utilization	Lower bound from YAP before MR*		Lower bound from LBP	
	Lower bound	Computation time (s)	Lower bound	Computation time (s)
0.05	89.758	30346.80	102	21
0.10	95.257	24640.89	138	744
0.15	119.68	26680.84	174	5217
0.20	138.68	25721.92	196	8706
0.25	147.47	64013.97	196	19603
0.30	Not found	1	Not found	6229
0.35	Not found	1	Not found	1297
0.40	227.50	11060.56	239	337
0.45	255.58	11037.89	266	297
0.50	285.53	13642.97	296	523
0.55	313.04	16114.41	323	134
0.60	342.27	25609.03	353	56
0.65	370.17	23034.78	381	32
0.70	398.84	26500.50	409	27
0.75	429.09	58407.64	438	152
0.80	457.53	220047.20	468	66
0.85	485.74	196890.58	495	27
0.90	514.69	554958.83	524	53
0.95	543.12	684439.48	556	25
1.00	568.83	845611.17	575	36

[*] MR: Memory running out.

For the large-scale YAP model, the optimal solution cannot be obtained. Therefore, the best lower bound before the program run out of memory is presented. As shown in Table 4.6, the lower bound obtained from the model LBP is always better than the best lower bound obtained from the model YAP before the program run out of memory. For scenarios with utilization of 0.3 and 0.35, some single-shift model is infeasible so the lower bound cannot be found also. This is because the input data for these scenarios, which is as same as the model YAP, cannot satisfy Constraint (4.18).

4.4 Solution Procedures

The yard allocation problem is intractable when the size of the problem becomes large. In this section, several heuristic algorithms that may find a feasible solution close to the optimal solution within a reasonable time are proposed. A new formulation of the model is also proposed to solve this problem.

4.4.1 The Sequential Method

From Section 4.3 it can be seen that single shift model can be solved effectively. Inspired by this finding, the proposed model can be solved one shift at a time, which is called as the sequential method (denoted as Algorithm SQM). The difference between LBP and SQM is that in the sequential method, the linking Constraints (4.4) and (4.5) are considered so as to ensure that the solution is feasible. Specifically, a sequence of shifts should be picked first, and based on this sequence the model is solved shift by shift. Note that after solving the model for each shift, the remaining capacity of each sub-block should be updated. Therefore, Constraints (4.4) and (4.5) can be modified as follows

$$x_{it} + 2y_{it} \le CS - \sum\nolimits_{\tau \in \Gamma_t} (x_{i,\tau} + 2y_{i,\tau}) \qquad \forall\ 1 \le i \le I \qquad (4.25)$$

$$x_{it} + y_{it} \le G_i CC - \sum\nolimits_{\tau \in \Gamma_t} (x_{i,\tau} + y_{i,\tau}) \qquad \forall\ 1 \le i \le I \qquad (4.26)$$

where Γ_t is the set which consists of all those shifts that come before the current Shift t in a given sequence. Constraints (4.25) and (4.26) ensure the capacity restriction of each sub-block in terms of storage space and yard cranes, respectively. Hence, Algorithm SQM is as same as the LBP model except that the two additional Constraints (4.25) and (4.26) are added. The sequential method may be effective as it has much less decision variables and constraints than the model for the yard allocation problem.

It is also noted that the sequence of shifts chosen need not be in chronological order since the demand repeats weekly and the only linking constraints are the space restriction and crane capacity restriction. It does not matter which shift to choose first, as it will not violate any of these constraints. On the other hand, different solutions are obtained from different shift orderings which implies that the sequence is important. However as there are *T!* different sequences that can be used and it is time consuming to enumerate all the sequences, a subset of the sequences are proposed, which is as follows: Select any shift in the planning horizon as the start shift of the sequence. Then starting from this shift, the sequence is formed by including the shifts according to the order of time. When the end of planning horizon is reached, the sequence is wrapped around to the beginning of the planning horizon and it is continued until it reaches the shift immediately before the start shift.

The sequential method is implemented in C++ and run on the same computer as that for the yard allocation problem. It uses the same input data as those for the yard allocation problem. The computational results are presented in Tables 4.7, 4.8, and 4.9.

Table 4.7 Results of Algorithm SQM for the small-scale problem (Case 1)

Utilization	Results of YAP (YCSs)	Results of SQM (YCSs)
0.05	10	10
0.10	10	10
0.15	10	10
0.20	10	10
0.25	10	10
0.30	15	15
0.35	15	15
0.40	15	15
0.45	20 (LB=17.56, Gap=13.9%)	20
0.50	20	20
0.55	21	21
0.60	25	25
0.65	25	26
0.70	25	Not Found
0.75	30	Not Found
0.80	30	Not Found
0.85	35	Not Found
0.90	35	Not Found
0.95	Infeasible	Infeasible
1.00	Infeasible	Infeasible

Table 4.8 Results of Algorithm SQM for the small-scale problem (Case 2)

Utilization	Results of RTYP (YCSs)	Results of SQM (YCSs)
0.05	10	10
0.10	10	10
0.15	10	10

0.20	11	11
0.25	12	12
0.30	13	13
0.35	15	15
0.40	17	17
0.45	18	18
0.50	21	21
0.55	23	23
0.60	23	24
0.65	27	Not Found
0.70	28	Not Found
0.75	Infeasible	Infeasible
0.80	Infeasible	Infeasible
0.85	Infeasible	Infeasible
0.90	Infeasible	Infeasible
0.95	Infeasible	Infeasible
1.00	Infeasible	Infeasible

Tables 4.7 and 4.8 present the results of the sequential method for the small-scale problems. For relatively low utilization scenarios, the obtained feasible solutions are close to the optimal solutions. For some scenarios, the objective value of the feasible solution is exactly as same as that of the optimal solution. For high utilization scenarios the sequential method may not find any feasible solution. It is because the sequential method is a greedy algorithm and Constraints (4.25) and (4.26) cannot be satisfied for high utilization scenarios.

Table 4.9 Results of Algorithm SQM for the large-scale problem

Utilization	Lower bound from LBP (YCSs)	Results of SQM (YCSs)
0.05	102	102
0.10	138	138
0.15	174	174
0.20	196	196
0.25	196	197
0.30	Infeasible	Infeasible
0.35	Infeasible	Infeasible

0.40	239	239
0.45	266	266
0.50	296	296
0.55	323	323
0.60	353	353
0.65	381	381
0.70	409	409
0.75	438	438
0.80	468	472
0.85	495	Not Found
0.90	524	Not Found
0.95	556	Not Found
1.00	575	Not Found

Table 4.9 presents the results of the sequential method for the large-scale problem. For most relatively low and moderate utilization scenarios (with utilization up to 0.80), the sequential method can give a near-optimal or optimal solution. However, for high utilization scenarios (with utilization greater than 0.8), the sequential method does not work well. For scenarios with utilization of 0.3 and 0.35, some single-shift model is infeasible so the sequential method cannot find a feasible solution. This is because the input data for these scenarios, which is as same as model YAP, cannot satisfy Constraint (4.18).

In order to handle those scenarios for which the sequential method cannot find any feasible solution or can only find near optimal solution, two other heuristic algorithms are proposed in the following two sections.

4.4.2 The Column Generation Method

One heuristic algorithm is the column generation method (denoted as Algorithm CGM). To be consistent, the same notations are used as those defined in the yard allocation model

YAP. In addition, some new notations are defined to represent the column coefficients and new decision variables.

In the column generation model, a column represents a storage allocation in a particular shift. Specifically, the column coefficients represent the workload assigned to each sub-block, the number of yard cranes allocated to each block for unloading, the total number of yard cranes required for all the blocks and the high-low workload pattern for each sub-block in that shift and they are listed below.

x_{itr}	the number of 20-feet containers that are allocated to Sub-block i for unloading in Shift t for Column r, $1 \le i \le I$, $1 \le t \le T$, $r \ge 1$.
y_{itr}	the number of 40-feet containers that are allocated to Sub-block i for unloading in Shift t for Column r, $1 \le i \le I$, $1 \le t \le T$, $r \ge 1$.
h_{itr}	= 1, if the workload that are allocated to Sub-block i in Shift t for Column r is high, i.e., $HL \le x_{itr}+y_{itr} \le HU$, $1 \le i \le I$, $1 \le t \le T$, $r \ge 1$. = 0, if the workload that are allocated to Sub-block i in Shift t for Column r is low, i.e., $LL \le x_{itr}+y_{itr} \le LU$, $1 \le i \le I$, $1 \le t \le T$, $r \ge 1$.
d_{ktr}	the number of yard cranes that are allocated to Block k for unloading in Shift t for Column r, $1 \le k \le K$, $1 \le t \le T$, $r \ge 1$.
n_{tr}	the total number of yard cranes required in Shift t for Column r, $1 \le t \le T$, $r \ge 1$.

In the column generation model, decision variable w_{tr} is used to represent whether column r is selected for Shift t. It should be binary in the original MIP master problem, while it should be continuous between 0 and 1 in the relaxed master problem.

w_{tr} = 1, if Column r is selected for Shift t, $1 \le t \le T$, $r \ge 1$.

= 0, otherwise.

In the master problem, the total number of yard crane shifts required to handle all the workload in the whole planning horizon should be minimized.

$$\text{(CGM)}\ Min\ w = \sum_{t=1}^{T} \sum_{r} n_{tr} w_{tr} \qquad (4.27)$$

Subject to:

$$\sum_{r} w_{tr} = 1 \qquad \forall\, 1 \le t \le T \qquad (4.28)$$

$$\sum_{t=1}^{T} \sum_{r} (x_{itr} + 2y_{itr}) w_{tr} \le CS \qquad \forall\, 1 \le i \le I \qquad (4.29)$$

$$\sum_{t=1}^{T} \sum_{r} (x_{itr} + y_{itr}) w_{tr} \le G_i CC \qquad \forall\, 1 \le i \le I \qquad (4.30)$$

$$w_{tr} \in \{0, 1\} \qquad \forall\, 1 \le t \le T,\ \forall\, r \qquad (4.31)$$

Constraint (4.28) ensures that one and only one column should be selected for each shift. Constraints (4.29) and (4.30) ensure the capacity restriction of each sub-block in terms of storage space and yard cranes respectively. Constraint (4.31) is integer restriction.

To solve the column generation model CGM, the master problem should be feasible and all the variables should be continuous, which are necessary to obtain the dual price of each constraint. Therefore, in the relaxed master problem, w_{tr} is a continuous variable assuming a value between 0 and 1. A feasible solution to the original problem is added as the initial columns. This can ensure the feasibility of the relaxed master problem.

After obtaining the dual price of each constraint from the relaxed master problem, they can be used to price out the new columns. A new column with the most negative objective function coefficient in master problem can be found from the pricing problem for each shift. After adding the new columns, the master problem can be solved again to obtain the updated dual price of each constraint.

The pricing problem is defined as follows.

$$Min\ z = n_{tr} - \pi_t - \sum_{i=1}^{I}(x_{itr} + 2y_{itr})\sigma_i - \sum_{i=1}^{I}(x_{itr} + y_{itr})\delta_i \tag{4.32}$$

π_t, σ_i, and δ_i are dual prices for Constraints (4.28), (4.29) and (4.30), respectively. The objective function of the pricing problem is to find the most negative objective function coefficient in the master problem.

Subject to:

$$\sum_{i \in V_j} x_{itr} = WX_{jt} \qquad \forall\ 1 \le j \le J \tag{4.33}$$

$$\sum_{i \in V_j} y_{itr} = WY_{jt} \qquad \forall\, 1 \le j \le J \tag{4.34}$$

$$\sum_{i \in B_k} \left(x_{itr} + y_{itr}\right) \le d_{ktr} CC \qquad \forall\, 1 \le k \le K \tag{4.35}$$

$$HL + (LL - HL)(1 - h_{itr}) \le x_{itr} + y_{itr} \le LU + (HU - LU) h_{itr}$$

$$\forall\, 1 \le i \le I \tag{4.36}$$

$$x_{i'tr} = 0\,, y_{i'tr} = 0 \qquad \forall\, 1 \le i \le I, i' \in N_i, t \in L_i \tag{4.37}$$

$$\sum_{i' \in N_i \cup \{i\}} h_{i'tr} \le 1 \qquad \forall\, 1 \le i \le I \tag{4.38}$$

$$d_{ktr} + NL_{kt} \le C_k \qquad \forall\, 1 \le k \le K \tag{4.39}$$

$$\sum_{k=1}^{K} d_{ktr} = n_{tr} \tag{4.40}$$

$$x_{itr} \ge 0\,, y_{itr} \ge 0 \qquad \forall\, 1 \le i \le I, 1 \le t \le T \tag{4.41}$$

$$h_{itr} = 0 \; or \; 1 \qquad \forall\, 1 \le i \le I, 1 \le t \le T \tag{4.42}$$

$$n_{tr} \in \{\; Positive\ Integer \;\} \qquad \forall\, 1 \le t \le T \tag{4.43}$$

All constraints ensure the same restrictions as the yard allocation model YAP for the current Shift *t*. Constraints (4.4) and (4.5) are removed from the formulation as only one shift is considered at one time.

To accelerate the column generation procedure, the results obtained from model LBP and Algorithm SQM are treated as alternative columns. By this means, the quality of the solution obtained should be not worse than that from Algorithm SQM. In conjunction with the initial feasible solution, the column generation method may improve the quality of the results obtained from the sequential method.

The column generation method is implemented in C++ and run on the same computer as that for the yard allocation model YAP. It uses the same input data as those for the yard allocation model YAP. The computational results are presented in Tables 4.10, 4.11, and 4.12.

Tables 4.10 and 4.11 present the results of the column generation method for the small-scale problems. For relatively low utilization scenarios the column generation method can yield near-optimal or optimal solutions as the results of the model SQM are considered as alternative columns. For high utilization scenarios, the solution quality is not good as there is a big gap between the solution obtained and the lower bound. Gap in the 4th column means the ratio of the difference between the lower bound and the solution from CGM model to the lower bound.

Table 4.10 Results of Algorithm CGM for the small-scale problem (Case 1)

Utilization	Results of YAP	Results of SQM	Results of CGM	Gap
0.05	10	10	10	0
0.10	10	10	10	0
0.15	10	10	10	0
0.20	10	10	10	0
0.25	10	10	10	0
0.30	15	15	15	0
0.35	15	15	15	0
0.40	15	15	15	0
0.45	20 (LB=17.56, Gap=13.9%)	20	20	0
0.50	20	20	20	0
0.55	21	21	21	0
0.60	25	25	25	0
0.65	25	26	26	4%
0.70	25	Not Found	38	52%
0.75	30	Not Found	40	33.3%
0.80	30	Not Found	40	33.3%
0.85	35	Not Found	40	14.3%

0.90	35	Not Found	40	14.3%
0.95	Infeasible	Infeasible	Infeasible	Infeasible
1.00	Infeasible	Infeasible	Infeasible	Infeasible

Table 4.11 Results of Algorithm CGM for the small-scale problem (Case 2)

Utilization	Results of RTYP	Results of SQM	Results of CGM	Gap
0.05	10	10	10	0
0.10	10	10	10	0
0.15	10	10	10	0
0.20	11	11	11	0
0.25	12	12	12	0
0.30	13	13	13	0
0.35	15	15	15	0
0.40	17	17	17	0
0.45	18	18	18	0
0.50	21	21	21	0
0.55	23	23	23	0
0.60	23	24	24	4.3%
0.65	27	Not Found	40	48.1%
0.70	28	Not Found	40	42.8%
0.75	Infeasible	Infeasible	Infeasible	Infeasible
0.80	Infeasible	Infeasible	Infeasible	Infeasible
0.85	Infeasible	Infeasible	Infeasible	Infeasible
0.90	Infeasible	Infeasible	Infeasible	Infeasible
0.95	Infeasible	Infeasible	Infeasible	Infeasible
1.00	Infeasible	Infeasible	Infeasible	Infeasible

Table 4.12 Results of Algorithm CGM for the large-scale problem

Utilization	Lower bound from LBP	Results of SQM	Results of CGM	Gap
0.05	102	102	102	0
0.10	138	138	138	0
0.15	174	174	174	0
0.20	196	196	196	0
0.25	196	197	197	0.5%
0.30	Infeasible	Infeasible	Infeasible	-
0.35	Infeasible	Infeasible	Infeasible	-
0.40	239	239	239	0
0.45	266	266	266	0
0.50	296	296	296	0

0.55	323	323	323	0
0.60	353	353	353	0
0.65	381	381	381	0
0.70	409	409	409	0
0.75	438	438	438	0
0.80	468	472	468	0
0.85	495	Not Found	2048	313.7%
0.90	524	Not Found	2048	290.8%
0.95	556	Not Found	2048	268.3%
1.00	575	Not Found	2048	256.2%

Table 4.12 compares the results of the sequential method and the column generation method for the large-scale problem. For most relatively low and moderate utilization scenarios (with utilization up to 0.80 except 0.3 and 0.35), the column generation method can give a near-optimal or optimal solution. This is because the column generation method uses the results of the sequential method as alternative columns. However, for high utilization scenarios, the column generation method can only give a feasible solution which has a big gap with the lower bound.

In the experiments conducted, only for those scenarios in which the sequential method can get the feasible solution the column generation method gives a slightly better solution than that of the sequential method. This means that the column generation model CGM actually cannot improve the solution quality effectively. Although the relaxed master problem can be solved to optimality easily, the columns generated cannot improve the quality of solution to the MIP version master problem. For each single shift, there are too many optimal solutions as there are too many possible allocations with the same number of cranes. Once the master problem finds some linear combination of the columns that can satisfy the linking constraints, the column generation procedure will stop right away. Consequently, the solutions for different shifts will not be explored enough.

4.4.3 The Simulated Annealing Algorithm

Another heuristic algorithm is a simulated annealing based heuristic. To describe the simulated annealing algorithm (denoted as Algorithm SA), an encoding and decoding method of solutions must be defined first. A solution is encoded by a sequence of integer numbers representing the high-low value of all the sub-blocks in each shift together with the number of yard cranes deployed to each block in each shift. This means different solutions may be encoded into the same sequence. For the decoding process, the values for all the integer variables are obtained from the sequence. The number of containers assigned to each sub-block can be obtained by plugging all the integer values into the model for the yard allocation problem. That is to say, the decoding process is to solve one linear programming model, which can be done quite efficiently.

In the simulated annealing method, the quality of the solution depends on the control parameters and the schedule of the temperature. In typical implementations, the simulated annealing approach involves a pair of nested loops and several additional parameters, a cooling rate, $0 < r < 1$, and a temperature length, R (see the algorithm below). The following describes the procedure for obtaining a container assignment schedule by using the simulated annealing method:

Algorithm SA:

Step 1: Obtain an initial solution S either by the sequential method or by CPLEX directly. Let $T = T_0$.

Step 2: Repeat the following steps until one of the stopping conditions becomes true.

Step 2.1: Perform the following loop R times.

Step 2.1.1: Pick a random neighbor S' of S.

Step 2.1.2: Check whether S' is feasible, if yes, go to Step 2.1.3, else go to Step 2.1.1

Step 2.1.3: Let $\delta = \text{cost}(S') - \text{cost}(S)$. The cost is evaluated by Equation (4.1).

Step 2.1.4: If $\delta < 0$ (downhill move), set $S = S'$.

Step 2.1.5: If $\delta \geq 0$ (uphill move), generate random number, x, from the interval, (0, 1); if $x < \exp(-\delta / T)$, which is the Boltzmann factor, then set $S = S'$.

Step 2.2: Set $T = rT$ (Reduce the temperature).

The simulated annealing based solution procedure is described in more detail in the following paragraphs. A flowchart for this is shown in Figure 4.2.

- The initial solution: It is expected that the better the initial solution, the better the final solution. So the best solution that can be found is used as the initial feasible solution, which is the solution from the sequential method because it is near optimal. If the sequential method cannot find any feasible solution, the solution obtained from the YAP model solving by CPLEX after a certain time period can be used. Usually the quality of solutions obtained by CPLEX increases rapidly at the beginning, and increases slowly when it is approaching the optimal solution. Therefore, a good solution can be obtained instead of spending too much time to find a slightly better solution.

- Neighborhood: Two types of neighborhood structures are defined. One is to change the high-low status of one randomly selected sub-block. The other is to increase or decrease the number of yard cranes deployed to one randomly selected block in one shift by one.
- The initial temperature and the cooling rate: The selection of the initial temperature and the cooling rate influences the quality of the solutions obtained through the simulated annealing algorithm. In general, a slower schedule (i.e., one that starts from a higher temperature, has a larger cooling rate and has a larger temperature length) will lead to better solutions. On the other hand, slower schedules tend to consume more computational time. This study attempts to select a set of cooling parameters that are expected to produce good, though not necessarily optimal, solutions within a reasonable time. The problems are solved with three different initial temperatures: 10, 50, and 200^{o}C. They are also solved with three different cooling rates: 0.5, 0.9, and 0.99. Thus, each problem is solved 9 times.

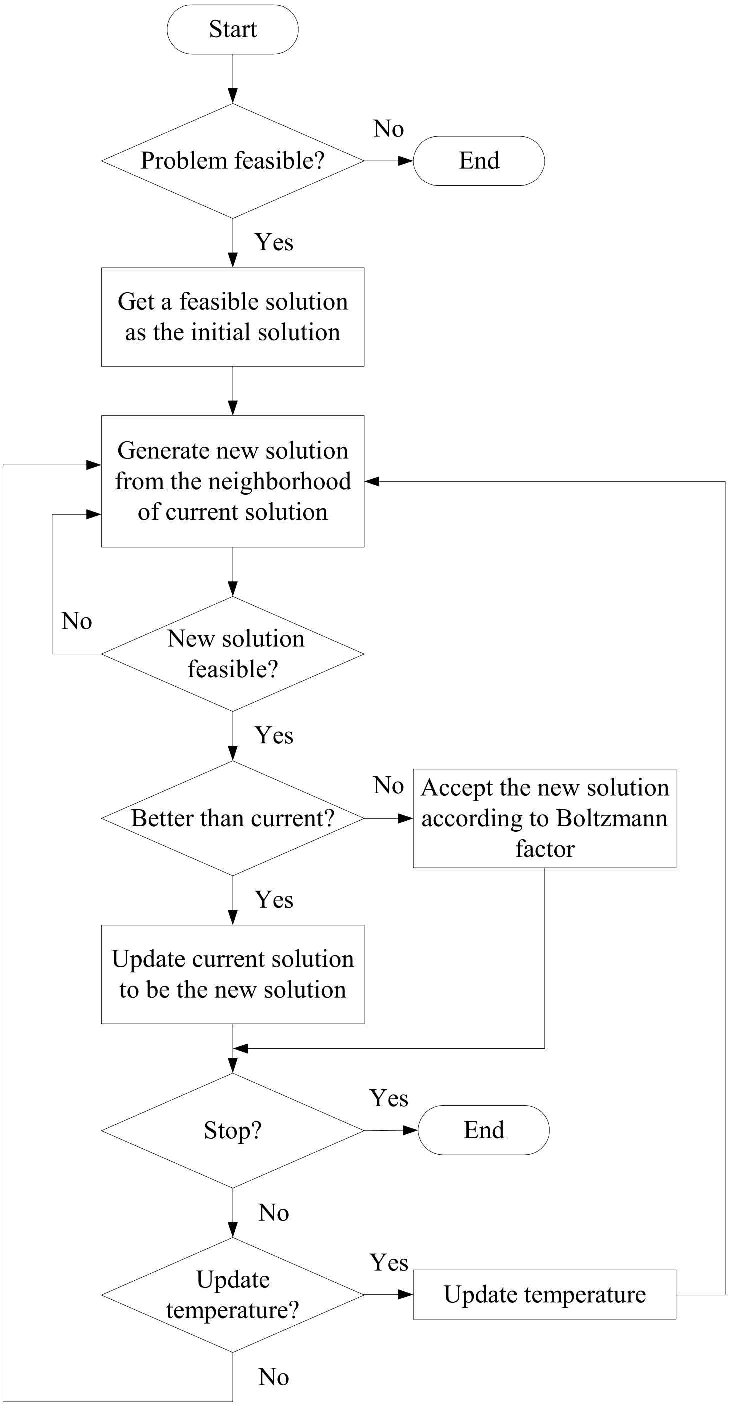

Figure 4.2 Flowchart for Algorithm SA

- The length of temperature: The length of temperature is set *TI*+2*TK*, which represents the size of a neighborhood of a solution.

- The stopping criterion: When one of the following three conditions is met, the iterations are stopped:
    - A solution whose objective value is equal to the lower bound found.
    - Temperature becomes less than 1.
    - The best value of the objective function has not been changed during the previous five consecutive external loops.

The computational results for the large-scale problem by the simulated annealing algorithm are presented in Table 4.13. For those scenarios that the sequential method can find the optimal solution, there is no need to test it by Algorithm SA. Only those scenarios, for which the sequential method can only find a feasible solution or cannot get any feasible solution, are tested. As mentioned above, if the sequential method can find a feasible solution, this feasible solution will be used as the initial solution; if it cannot find a feasible solution, an initial feasible solution will be generated by setting a time limit on the solution procedure for the yard allocation problem by CPLEX directly. Only the best results from all these 9 combinations of parameter settings are presented.

As shown in Table 4.13, the improvement on the quality of the solutions is not significant. For most cases, the procedure is stuck in the searching of a feasible neighboring solution, which means that most of the neighbors of one solution are infeasible. This makes the solution procedure lack efficiency. Therefore, after a long time period, the solution quality improves little only.

Table 4.13 Results of Algorithm SA for the large-scale problem

Utilization	Lower bound from LBP	Results of SQM	Results of YAP within a time limit	Best result from SA
0.05	102	102	-	-
0.10	138	138	-	-
0.15	174	174	-	-
0.20	196	196	-	-
0.25	196	197	-	197
0.30	Infeasible	Infeasible	-	Infeasible
0.35	Infeasible	Infeasible	-	Infeasible
0.40	239	239	-	-
0.45	266	266	-	-
0.50	296	296	-	-
0.55	323	323	-	-
0.60	353	353	-	-
0.65	381	381	-	-
0.70	409	409	-	-
0.75	438	438	-	-
0.80	468	472	-	472
0.85	495	Not Found	560	558
0.90	524	Not Found	624	624
0.95	556	Not Found	631	631
1.00	575	Not Found	655	655

To overcome the shortcoming of Algorithm SA, the neighborhood structure is changed intelligently. For example, when changing the workload status of one sub-block from low to high, the other sub-blocks in the same block, the other sub-blocks reserved for the same vessel, and the same sub-block in different shifts are also changed accordingly. Making corresponding changes on those sub-blocks can potentially increase the chance to find a feasible neighbor. For example, if one Sub-block *i* is randomly selected to change the workload status from low to high and this sub-block has been assigned high workload in another two shifts, this neighbor is infeasible since the space capacity Constraint (4.4) cannot be satisfied. If the workload in one of the two shifts is changed from high to low, the resulting solution is probably feasible. Based on this change on the neighborhood structure, the same input data are used to conduct the experiments and the results are presented in Table 4.14.

Table 4.14 Results of the improved Algorithm SA for the large-scale problem

Utilization	Results of SQM	Results of YAP within a time limit	Best result from SA without changes	Best result from SA with changes
0.05	102	-	-	-
0.10	138	-	-	-
0.15	174	-	-	-
0.20	196	-	-	-
0.25	197	-	197	197
0.30	Infeasible	-	Infeasible	Infeasible
0.35	Infeasible	-	Infeasible	Infeasible
0.40	239	-	-	-
0.45	266	-	-	-
0.50	296	-	-	-
0.55	323	-	-	-
0.60	353	-	-	-
0.65	381	-	-	-
0.70	409	-	-	-
0.75	438	-	-	-
0.80	472	-	472	472
0.85	Not Found	560	558	557
0.90	Not Found	624	624	624
0.95	Not Found	631	631	630
1.00	Not Found	655	655	655

As shown in Table 4.14, with the new neighborhood structure, Algorithm SA obtains a slightly better solution than that with the original neighborhood structure. But the quality of the solutions is still not satisfactory. Because the new neighborhood structure only increases the chance to find a feasible neighbor, but due to the problem structure itself, it is still not easy to find a feasible solution by Algorithm SA. The long solution time makes this method unfit for this problem.

4.4.4 The Big-block Formulation

According to the definition of neighborhood structure, only the sub-blocks sharing the same truck path or passing lane will affect each other directly. Hence, the concept of big-block is defined as follows. Two blocks that share the same passing lane are defined as a big-block. Two sub-blocks from different big-blocks are not next to each other and there is no traffic congestion between them. Consequently, the vicinity matrix can be decomposed into several smaller matrices, each of which is only for one big-block. For a big-block, there are a limited number of feasible high-low patterns. Each feasible pattern, which satisfies Constraint (4.9), defines the high-low status of each sub-block in a big-block. By considering Constraint (4.9), a lot of infeasible patterns have been excluded. Therefore, a better solution is expected to be found since there is much less search space if the problem is formulated according to the big-block structure.

Besides the notations defined in Section 4.1.1, the following extra notations are introduced for this big-block formulation.

B	the number of big-blocks under consideration, $B = K/2$.
P	the number of patterns for a big-block.
S	the number of sub-blocks in one big-block, $S = 10$ in this model.
P_{ps}	whether the Sub-block s has high workload for pattern p, $1 \le p \le P$, $1 \le s \le S$.

There are also a new set of decision variables for this big-block formulation.

z_{btp} whether big-block Pattern p is selected for Big-block b in Shift t, $1 \le b \le B$, $1 \le t \le T$, $1 \le p \le P$.

The objective function is as same as the model for the yard allocation problem, which is to minimize the total number of yard crane shifts used to handle all the workload in the whole planning horizon. Two sets of new constraints are introduced into the formulation.

$$\sum_{p=1}^{P} z_{btp} = 1 \qquad \forall\ 1 \le b \le B,\ 1 \le t \le T \qquad (4.44)$$

$$h_{it} = \sum_{p=1}^{P} z_{btp} P_{ps} \qquad \forall\ 1 \le i \le I,\ 1 \le t \le T,\ b=[i/10],\ s=i\%10 \qquad (4.45)$$

$$z_{btp} = 0\ or\ 1 \qquad \forall\ 1 \le b \le B,\ 1 \le t \le T,\ 1 \le p \le P \qquad (4.46)$$

Constraint (4.44) ensures that only one pattern is selected for each big-block in each shift. Constraint (4.45) is used to represent h_{it} by the newly introduced variables. [*i*/10] and *i*%10 are used to convert the sub-block index *i* in the whole yard into the big-block index *b* and sub-block index *s* in one big-block.

To verify the effectiveness of the big-block formulation (denoted as BBP), the same input data are used to conduct the numerical experiments for the large-scale problem. To simplify the problem, some changes on the yard layout were made (as shown in Figure 4.3) so that the number of big-blocks is exactly half the total number of blocks. The computational results are summarized in Table 4.15.

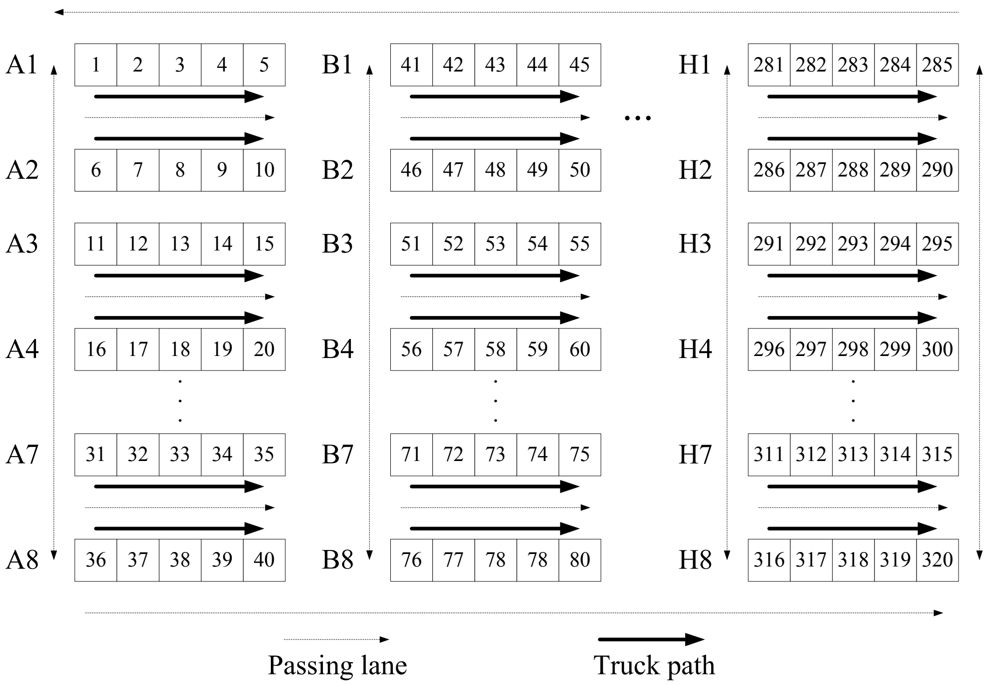

Figure 4.3 Simplified yard layout for the big-block model

Table 4.15 Results of the BBP model for the large-scale problem

Utilization	Computation Time (s)	Solution Status	Result Details
0.05	54854.29	Out of memory	104, Gap=15.9% (LB=89.758)
0.10	31254.18	Out of memory	152, Gap=56.2% (LB=97.318)
0.15	13523.62	Out of memory	185, Gap=58.9% (LB=116.43)
0.20	35521.54	Out of memory	216, Gap=55.8% (LB=138.68)
0.25	22423.16	Out of memory	220, Gap=51.5% (LB=145.17)
0.30	1	Infeasible	-
0.35	1	Infeasible	-
0.40	9324.11	Out of memory	283, Gap=27.8% (LB=221.41)
0.45	15111.33	Out of memory	316, Gap=24.0% (LB=254.75)
0.50	22411.16	Out of memory	352, Gap=23.3% (LB=285.53)
0.55	33421.17	Out of memory	364, Gap=16.3% (LB=313.04)
0.60	17443.94	Out of memory	397, Gap=15.4% (LB=344.14)
0.65	21233.03	Out of memory	415, Gap=12.1% (LB=370.17)
0.70	23235.33	Out of memory	452, Gap=12.4% (LB=402.15)
0.75	36443.35	Out of memory	485, Gap=13.0% (LB=429.09)
0.80	172342.23	Out of memory	526, Gap=15.7% (LB=454.76)
0.85	253322.03	Out of memory	556, Gap=14.9% (LB=483.71)
0.90	463323.92	Out of memory	609, Gap=17.5% (LB=518.48)
0.95	832212.45	Out of memory	622, Gap=14.9% (LB=541.46)

Utilization	Computation Time (s)	Solution Status	Result Details
1.00	702888.23	Out of memory	645, Gap=14.1% (LB=565.14)

As shown in Table 4.15, the large-scale problem by BBP formulation obtains similar results as the yard allocation model YAP. For most of the scenarios the problem cannot be solved to optimality within a reasonable time and always terminates as a result of insufficient memory. This result means that the problem with big-block formulation is still too complex to be solved by CPLEX directly.

In this chapter, the yard allocation problem is formulated assuming that the set of sub-blocks reserved for each vessel is given and input to the model. Although the yard allocation problem cannot be solved to optimality directly by CPLEX, three heuristic algorithms and a new big-block formulation are proposed to solve it. One of these heuristics, the sequential method, can obtain an optimal or near optimal solution for most of the cases tested. Based on the yard allocation problem and the solution method, the yard template problem is solved in the next chapter.

5 Solving the Yard Template Problem

Although the yard allocation problem cannot be solved to optimality by CPLEX, several heuristic algorithms are proposed, one of which, the sequential method, can solve the yard allocation problem effectively and efficiently for most scenarios tested. Hence a method is proposed, which aims at improving the solution quality iteratively based on the sequential method in the next section.

5.1 Solution Procedure

In the proposed solution method, a tabu-search based heuristic algorithm is proposed to find an initial yard template, and then the generated yard template is improved iteratively through an improvement algorithm based on the information from the yard allocation problem which is solved by the sequential method. The proposed method is shown in Figure 5.1 and described as follows.

Main algorithm:

Step 1: Calculate the lower bound based on the incoming workload in each shift and the capacity of yard cranes; generate an initial yard template by Algorithm IYT (see Section 5.3); then go to Step 2.

Step 2: Using the generated initial yard template in Step 1 or improved yard template in Step 3 as input, solve the yard allocation problem by the sequential method. If no feasible solution can be found for the yard allocation problem, gather the infeasibility information and repair the yard template by Algorithm INF (see Section 5.5); if the objective value of the solution obtained is equal to that of the lower bound, stop the procedure and the solution obtained is an optimal solution; if the objective value of the solution obtained is not equal to that of the lower bound and the maximum number of iterations has not been reached, go to Step 3; if the maximum number of iterations has been reached, go to Step 4.

Step 3: Improve the current yard template by Algorithm IMP (see Section 5.4) for a certain number of iterations, and then go back to Step 2.

Step 4: Stop the procedure. The best solution obtained so far is the final solution for the yard template problem.

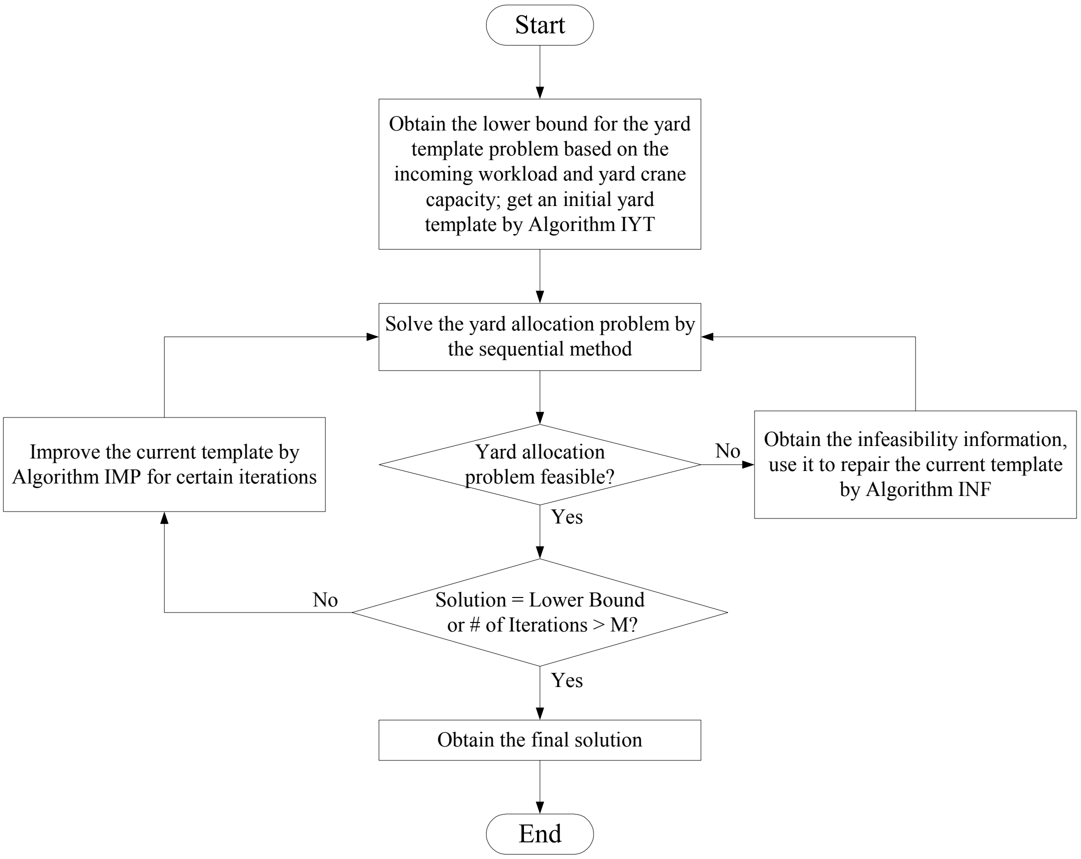

Figure 5.1 Flowchart for the main algorithm to solve the yard template problem

5.2 Finding a Lower Bound

To assess the quality of the solution to the yard template problem, an equitable lower bound is necessary. Since yard crane shifts cannot be shared by different shifts, and they can only be shared by different sub-blocks in the same shift, the smallest number of yard crane shifts needed to handle all the incoming containers can be determined shift by shift. In each shift, the lower bound is computed based on the yard crane capacity and the total incoming workload. The lower bound for the problem, which is the smallest number of yard crane shifts for the whole planning horizon, is calculated as follows.

$$LB = \sum_{t=1}^{T} \left\lceil \frac{W_t}{CC} \right\rceil \tag{5.1}$$

W_t is the total number of incoming containers in Shift t.

CC is the yard crane capacity per shift in terms of container moves.

$\lceil x \rceil$ returns the smallest integer number which is greater than or equal to x.

5.3 Generating an Initial Yard Template

As mentioned in Section 4.4.4, two sub-blocks from different big-blocks are not next to each other because they do not share the same truck path or passing lane and hence there is no traffic congestion between them. Consequently, the vicinity matrix can be decomposed into several smaller matrices, each of which is only for one big-block. Due to this big-block structure, the problem can be handled at the big-block level, i.e., the yard template is generated one big-block by one big-block sequentially, instead of generating the whole template for the whole terminal. Furthermore, the actual location of each big-block is not important. Therefore, the port operator can decide where to put the big-blocks by considering other factors.

Within a big-block, a vessel is assigned for each sub-block one sub-block by one sub-block. To generate a feasible initial yard template, the constraints dealing with the reservation should be satisfied. The constraints that need to be considered are Constraints (3.2), (3.3), (3.12) and (3.13). If any of these constraints is violated, the generated yard template will be infeasible. On the other hand, satisfying these constraints cannot guarantee the feasibility of the generated yard template because the constraints that directly deal with the allocation of the containers are not considered. Nevertheless, this is the most that can be done to avoid infeasibility at this stage. Therefore, there are a set of feasible vessels that can be assigned for each sub-block. A feasible vessel refers to a departing vessel that does not violate any of the following:

1. If the sub-block is reserved for this vessel, there will be two neighboring sub-blocks both in the loading process in a certain shift (Constraint (3.12)).
2. If the sub-block is reserved for this vessel, there will be more sub-blocks in the loading process in some shift than the limitation of the number of yard cranes that are allowed to reside in that block (Constraint (3.13)).

For each sub-block, the set of feasible vessels may be different. D_i is used to denote the set of feasible vessels that can be assigned for Sub-block i. The feasible vessel set D_i for Sub-block i depends on all the previous reservation of other sub-blocks in the same big-block up to now. For the first sub-block in one big-block, every vessel that has not been reserved sufficient sub-blocks will be in the feasible set; while for a sub-block that is not the first sub-block in one big-block, Constraints (3.12) and (3.13) are used to define the feasible vessels. If the feasible vessel set D_i is empty for certain Sub-block i, which means assigning any vessel to Sub-block i will result in an infeasible template, then backtrack one step and reserve Sub-block i-1 for a different vessel to replace the vessel that was originally assigned. A tabu list is defined for each sub-block to ensure a different vessel is selected when backtracking to the sub-block. The size of the tabu list is set to be 21, the same as the number of vessels.

It may not be prudent to randomly reserve sub-blocks for vessels. Instead, the potential traffic congestion can be avoided intelligently by the following method. If the incoming workload for two vessels is always heavy in the same shift, neighboring sub-blocks should be avoided being reserved for these two vessels. To evaluate the potential traffic congestion between two vessels, a conflicting factor is introduced, which is defined as the total number of conflicting shifts between two vessels. A conflicting shift for two vessels is a shift in which the numbers of incoming containers for these two vessels are both more than a certain amount (the threshold value). As shown in Figure 5.2, the numbers of incoming containers for Vessels j and j' in shifts 9, 10 and 11 are both greater than the threshold value. Therefore, the conflicting factor $C_{jj'}$ between Vessels j and j' is 3.

The conflicting factor $C_{jj'}$ is used as a guide when reserving sub-blocks for vessels. To find a good yard template, one sub-block can be reserved for the best vessel in terms of the conflicting factor. The best vessel has the least total number of conflicting factor with those vessels for which the neighbors of the sub-block are reserved. For example, when selecting a vessel for the Sub-block i, the best Vessel j^* is chosen as follows,

$$j^* = \arg\min_{j \in D_i} \left(\sum_{i' \in N_i} \sum_{j'=1}^{J} F_{i'j'} C_{jj'} \right) \tag{5.2}$$

where D_i is the set of feasible vessels for Sub-block i excluding those vessels on the tabu list of Sub-block i.

N_i is the set of sub-blocks that are neighbors of Sub-block i.

$F_{i'j'}$ = 1, if Sub-block i' has been reserved for Vessel j'.

= 0, otherwise.

$C_{jj'}$ is the conflicting factor between Vessels j and j'.

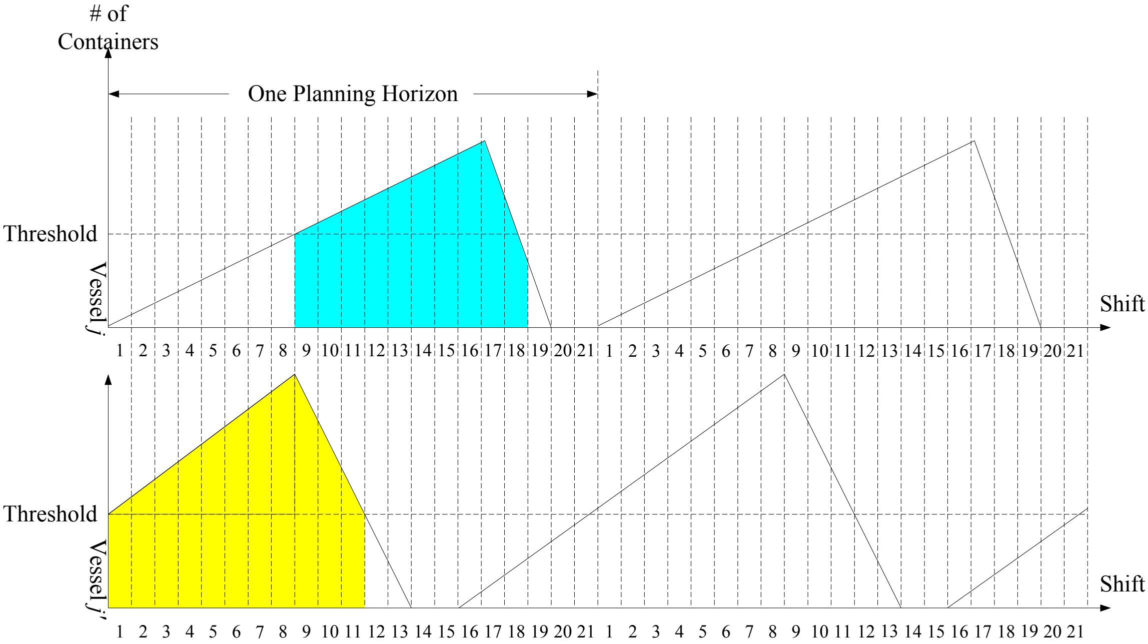

Figure 5.2 A schematic diagram for the conflicting factor

The following provides an overall procedure to obtain an initial yard template, which is denoted as Algorithm IYT. Sub-block index i is used to describe Algorithm IYT. We have the following relationship between the sub-block index i and the big-blocks index, in which Q_b is defined as the number of sub-blocks in Big-block b.

$i=\sum_{b'=1}^{b} Q_{b'}+s$, the sub-block is the s^{th} sub-block in the $(b+1)^{\text{th}}$ big-block.

For example, Sub-block 1 (i = 1) is the first sub-block in Big-block 1; Sub-block Q_1+2 ($i = Q_1+2$) is the second sub-block in Big-block 2.

Algorithm IYT:

Step 1: Start from the first sub-block in the first big-block, i.e., sub-block index $i = 1$.

Step 2: If Sub-block i is the first sub-block in one big-block, reserve Sub-block i for the best Vessel j^* (breaking tie by randomly selecting one vessel) according to Equation (5.2) (set $z_{ij^*} = 1$) and add Vessel j^* into the tabu list of Sub-block i; and then handle the next sub-block ($i = i+1$); otherwise go to Step 3.

Step 3: If the feasible vessel set D_i is empty, go to Step 4; otherwise, reserve the best Vessel j^* (breaking tie by randomly selecting one vessel) from the feasible vessel set D_i for Sub-block i according to Equation (5.2) and add Vessel j^* into the tabu list of Sub-block i; check whether all the vessels have been reserved sufficient sub-blocks. If yes, stop the procedure as one initial yard template has been obtained; if not, go to Step 2 to handle the next sub-block ($i = i+1$).

Step 4: Empty the tabu-list of Sub-block i. Backtrack one step to the previous Sub-block ($i = i-1$) and reserve it for the best Vessel ($\hat{j}$) based on the updated feasible set D_i ($\hat{j}$ is different from j^* because the previous best Vessel j^* is on the tabu list of Sub-block i and it is not included in the updated D_i), and then go back to Step 2.

The flowchart for Algorithm IYT to generate the initial yard template is shown in Figure 5.3.

The threshold value used to define the conflicting factor is important and should be chosen carefully. Generally it depends on the traffic volume, the bounds of high-low workload, and the capacity of yard cranes. If the threshold value is too big or too small, it cannot provide any useful information to generate the yard template. Based on some initial experiments, a value of 50 is used in the following numerical experiments. The computational results show that this value can generate a good yard template as the initial template.

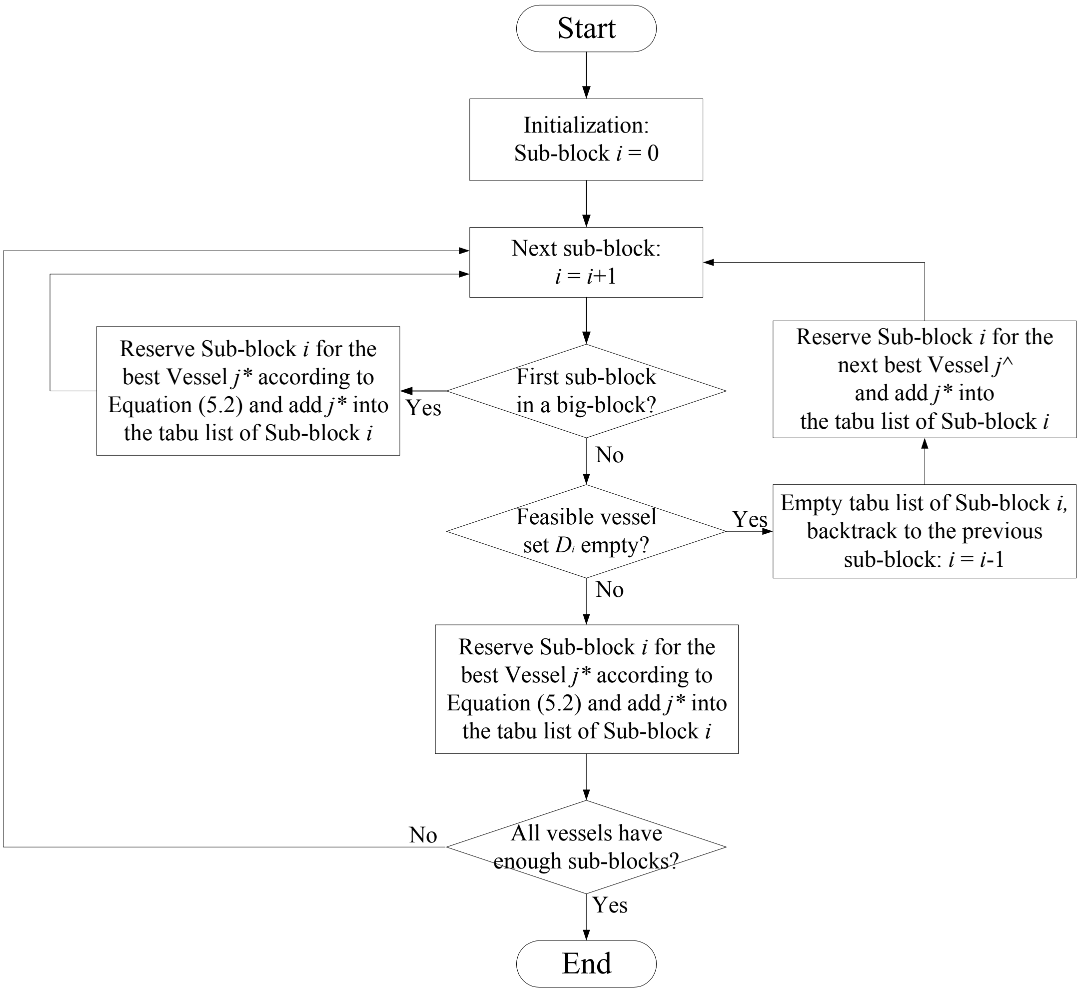

Figure 5.3 Flowchart for Algorithm IYT for generating initial yard template

5.4 The Improvement Algorithm

Through Algorithm IYT, the potential traffic congestion can be avoided by reserving a sub-block for the best vessel. However, if two vessels have the lowest conflicting factor, the sub-blocks reserved for these two vessels are likely neighbors of each other always.

Consequently, most of the sub-blocks reserved for one vessel cannot be used when the other vessel is in the loading process because the neighbors of a sub-block in the loading process cannot accept any workload in that shift (i.e., the neighbors of a sub-block in the loading process is "not available" in that shift). In this case, the incoming containers cannot be assigned to the storage locations or too few choices are available to generate a good solution. Generally sufficient sub-blocks should be available for each vessel to store the incoming containers in each shift. Based on this idea, an improvement algorithm is proposed to iteratively improve the generated yard template, which is denoted as Algorithm IMP. To describe Algorithm IMP, the concepts of bottleneck vessel and bottleneck shift are introduced. The bottleneck vessel for the yard template is defined as the vessel that has the smallest number of available sub-blocks in any shift. And the bottleneck shift is defined as the shift in which the bottleneck vessel has the smallest number of available sub-blocks, that is,

$$(J_b, T_b) = \underset{1 \le j \le J, 1 \le t \le T}{\arg\min} \left(AC_{jt}\right) \tag{5.3}$$

$$AC_{jt} = \sum_{i=1}^{I} F_{ij} V_{it} \tag{5.4}$$

$$V_{it} = \prod_{i' \in N_i} \left(1 - \sum_{j=1}^{J} F_{i'j} VL_{jt}\right) \tag{5.5}$$

where AC_{jt} is the number of the available sub-blocks for Vessel j in Shift t.

$F_{ij} = 1$ means Sub-block i is reserved for Vessel j, otherwise, $F_{ij} = 0$.

$V_{it} = 1$ means Sub-block i is available in Shift t, that is, Sub-block i is not a neighbor of a sub-block that is in the loading process in Shift t, otherwise, $V_{it} = 0$.

In Algorithm IMP, the bottleneck vessel and bottleneck shift are first determined for the yard template, and then one of the sub-blocks (say Sub-block i_1) that are reserved for the bottleneck vessel and are not available in the bottleneck shift ($F_{i1,Jb} = 1$ and $V_{i1,Tb} = 0$) is selected. A sub-block that is reserved for another vessel is randomly selected and the vessels reserved for these two sub-blocks are exchanged if the exchange will not cause the yard template to be infeasible. By repeating this procedure, $\underset{1 \le j \le J, 1 \le t \le T}{Min}\left(AC_{jt}\right)$ for the yard template may be increased to get more choices to allocate incoming containers and hence obtain a good yard template.

The following provides an overall procedure for Algorithm IMP.

Algorithm IMP:

Step 1: Determine the bottleneck vessel and bottleneck shift of the current yard template by Equation (5.3), breaking tie by selecting the vessel with the maximum total number of reserved sub-blocks.

Step2: Select one of the sub-blocks that are reserved for the bottleneck vessel and are not available in the bottleneck shift.

Step 3: Randomly select a sub-block that is reserved for another vessel so that Constraints (3.12) and (3.13) will not be violated if the vessels reserved for this sub-block and the sub-block selected in Step 2 are exchanged. If there is no feasible sub-block that can be used for exchange, terminate the procedure and use the generated template as the result.

Step 4: Exchange the vessels reserved for the two sub-blocks selected in Step 2 and Step 3, respectively.

Step 5: If $\underset{1\leq j\leq J,\,1\leq t\leq T}{Min}\left(AC_{jt}\right)$ increases after the exchanging or the maximum number of iterations has not been reached, go back to Step 1 for the next iteration; if $\underset{1\leq j\leq J,\,1\leq t\leq T}{Min}\left(AC_{jt}\right)$ does not increase and the maximum number of iterations has been reached, terminate the procedure and use the generated template as the result.

The corresponding flowchart is shown in Figure 5.4.

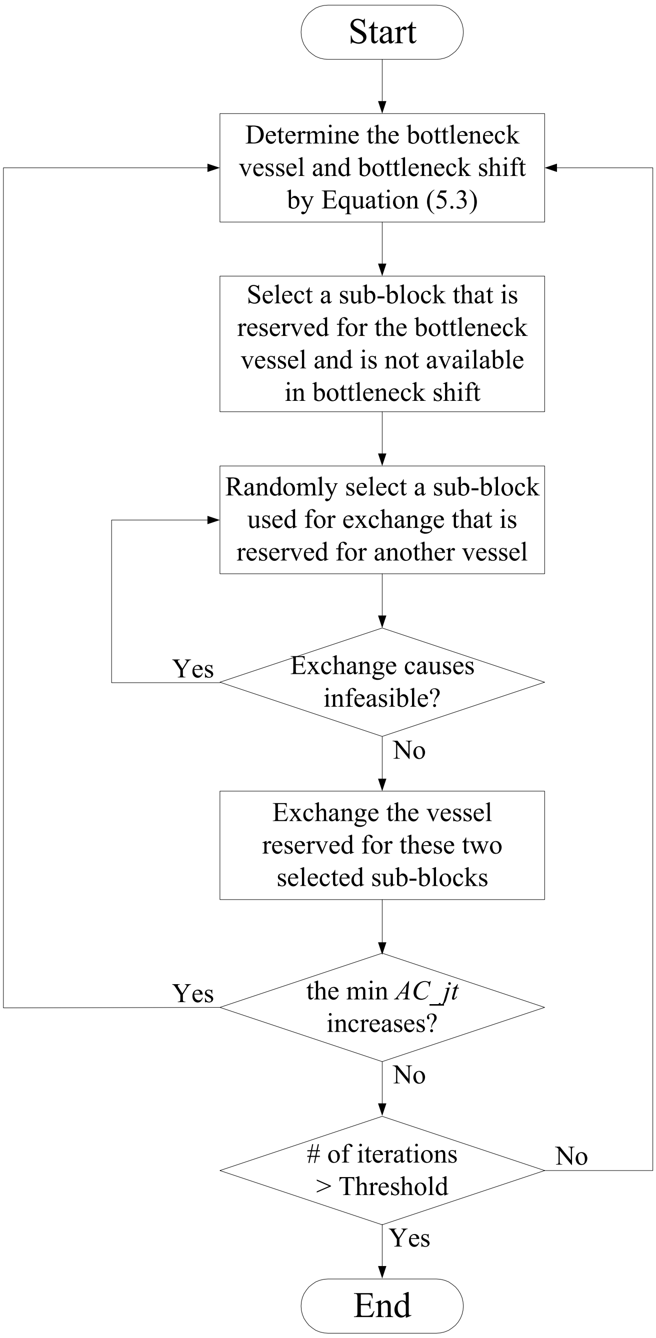

Figure 5.4 Flowchart for the improvement Algorithm IMP

5.5 The Repair Algorithm

If no feasible solution can be found for the yard allocation problem based on the current yard template, more useful information can be obtained from the result of the sequential method. For the sequential method, T different sequences are tested to solve the problem and for every sequence no feasible solution can be found. Intuitively, more attention should be paid to the first shift in which the single-shift model is infeasible because the solution procedure will be stopped once one single-shift model is infeasible. But for another sequence the first shift in which the single-shift model is infeasible may be a different shift. Therefore, the bottleneck shift is defined as follows:

$$t_b = \underset{1 \le t \le T}{\arg\max} \left(TM_t \right) \tag{5.6}$$

where TM_t is the number of times that Shift t is the first shift, in which the single-shift model is infeasible, for all the T sequences. After determining the bottleneck shift, the bottleneck vessel can be determined by the following equation.

$$J_b = \underset{1 \le j \le J, t = t_b}{\arg\min} \left(AC_{jt} \right) \tag{5.7}$$

This method of handling the infeasibility issue is denoted as Algorithm INF. The only difference between Algorithm INF and Algorithm IMP is on the determination of the bottleneck vessel and bottleneck shift. For Algorithm INF, the information obtained from the result of the sequential method is used to determine the bottleneck shift and bottleneck vessel; while for Algorithm IMP, the single-shift model for each shift is feasible, thus the indirect measure AC_{jt} is used to determine the bottleneck vessel and bottleneck shift by Equation (5.3). The flowchart for Algorithm INF is shown in Figure 5.5.

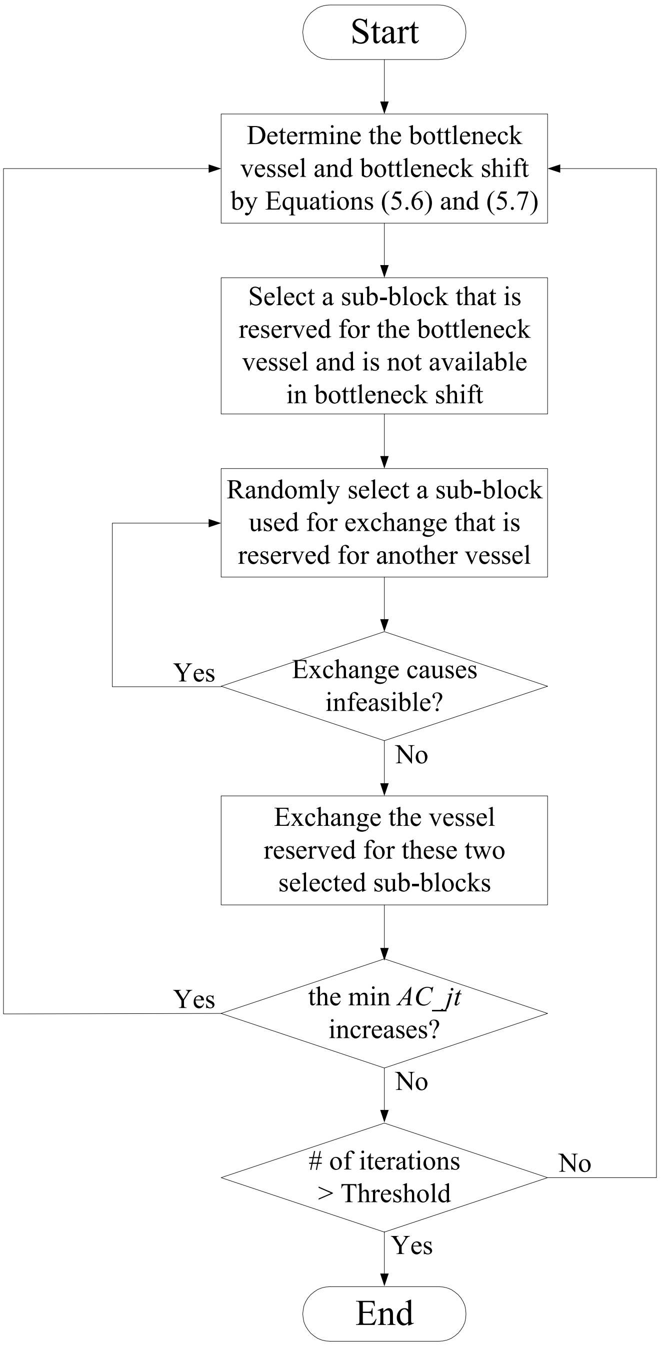

Figure 5.5 Flowchart for the repair Algorithm INF

If still no feasible solution can be found after running a certain number of iterations of Algorithm INF, one unreserved sub-block can be reserved for the bottleneck vessel. This is used only when no feasible template can be found because storage space is expensive and should be utilized highly.

5.6 Numerical Experiments

In this section, the solution procedure is implemented in C++ and run on the same computer as that for the yard allocation model YAP. The same input data as those for the large-scale yard allocation model YAP are used for the experiments first. Each single-shift model is solved using CPLEX 10.1. The utilization is defined as the ratio of the total storage space occupied by all the unloading containers to the total storage space in the storage yard. The number of iterations for the whole solution procedure is fixed at 5, which can generate satisfactory results; while the number of iterations for the improvement Algorithm IMP varies from 0 to 100. The computational results are summarized in Table 5.1.

Table 5.1 Results of the solution procedure for the large-scale problem (Case 1)

Utilization	Lower bound	Iteration = 0	Iteration = 10	Iteration =100	Results of YAP
0.50	290	Not Found	Not Found	290	296
0.55	322	Not Found	Not Found	322	323
0.60	350	Not Found	Not Found	350	353
0.65	377	Not Found	Not Found	377	381
0.70	409	Not Found	Not Found	409	409
0.75	436	Not Found	Not Found	436	438
0.80	465	Not Found	Not Found	465	472

As shown in Table 5.1, an optimal solution to the proposed model for every case is obtained through the proposed method. But no feasible solution can be found with the initial yard template because some of the constraints are not considered when generating the initial yard template. However, after running the repair and improvement procedure for a certain number of iterations, an optimal is obtained for each case. This means the repair algorithm can help to find a feasible solution based on the infeasibility information; while the improvement algorithm can improve the solution quality for most cases. Although the initial yard template might not be feasible, based on the initial template, the template can be improved iteratively until the optimal or near optimal solution is found. For comparison, the results for the yard allocation model YAP are listed in the sixth column. Better solution than the yard allocation model YAP is found for most of the cases as more yard template is under consideration for the yard template problem.

To verify whether it is because of coincidence in the input data so that an optimal solution can be found for each case tested, another set of input data (different from the data for YAP) is generated for the numerical experiments. The computational results are summarized in Table 5.2. For this set of input data, an optimal solution for each case can be found as well.

Table 5.2 Results of the solution procedure for the large-scale problem (Case 2)

Utilization	Lower bound	Iteration = 0	Iteration = 10	Iteration =100
0.50	288	289	289	288
0.55	320	320	320	320
0.60	347	349	348	347
0.65	374	375	375	374
0.70	405	406	405	405
0.75	434	435	435	434
0.80	461	462	461	461

In terms of the computational time, the solution procedure is quite efficient for this strategy problem. The initial yard template can be obtained within a few minutes and the final solution can be found within a few hours. It is quite reasonable for a large-scale planning problem.

5.7 Extreme Case Experiments

To evaluate the robustness of the proposed method for the yard template problem, some extreme case experiments are conducted. For Extreme Case 1, if a solution with the same objective value as the lower bound can be found, the average utilization of all the yard cranes is hundred percent in each shift. More specifically, the total workload for each shift is equal to $N \times CC$, where N is a positive integer number. The hypothetical data is generated to test the robustness of the solution procedure.

A set of input data for Extreme Case 1 is shown in Figure 5.6. The total number of arriving containers for all the vessels in each shift is plotted over the whole planning horizon. The computational results are summarized in Table 5.3.

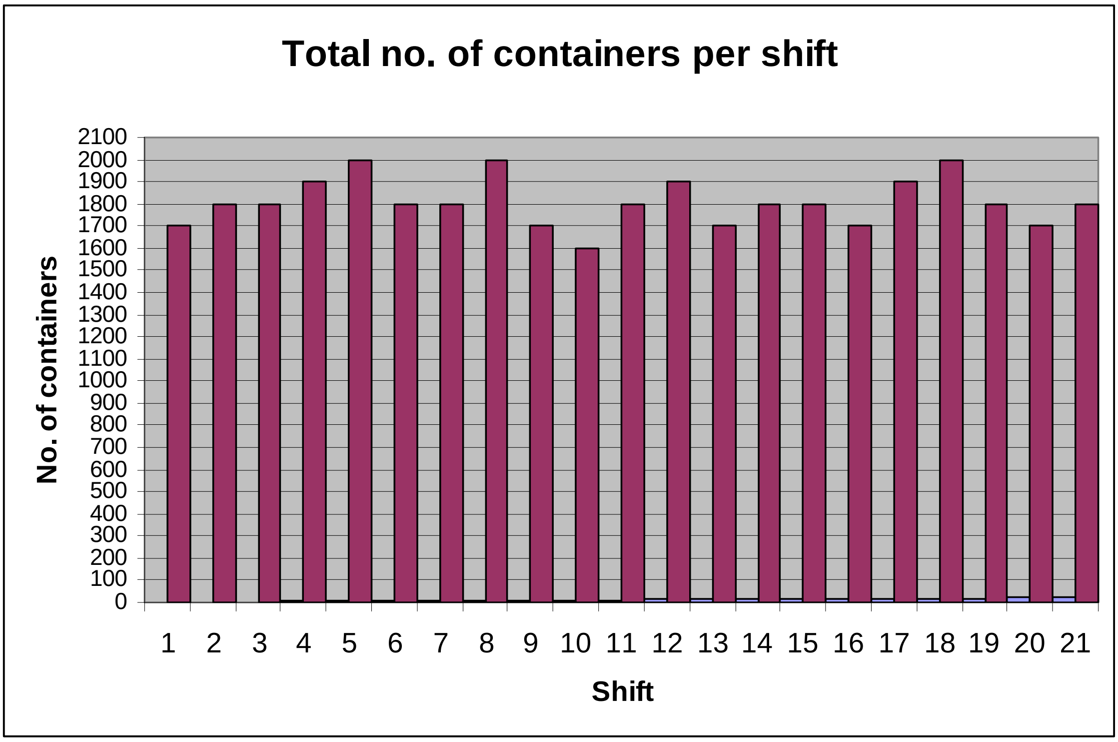

Figure 5.6 An example of input data (Extreme Case 1)

Table 5.3 Results of the solution procedure for the large-scale problem (Extreme Case 1)

Utilization	Lower bound for entire problem	Final solution	Gap
0.467643	267	267	0
0.522070	299	299	0
0.569766	326	326	0
0.617422	353	353	0
0.669948	384	384	0
0.721654	414	415	0.24%
0.794264	441	442	0.23%

As shown in Table 5.3, even for such extreme cases in which the utilization of the yard cranes is equal to or close to 100 percent, the gap between the lower bound and the solution obtained is less than 1%, which is acceptable to the port operator.

Although the port operator tries to spread the workload over the whole planning horizon, the arriving time of the vessels might not be exactly on the scheduled time. Hence in some shifts the workload is relatively heavy; while in other shifts the workload is relatively light. For Extreme Case 2, the input data are generated so that the total workload for all the vessels over the planning horizon has a wave pattern (as shown in Figure 5.7). In this case, the workload has concentrated heavy period and light period. The computational results are summarized in Table 5.4.

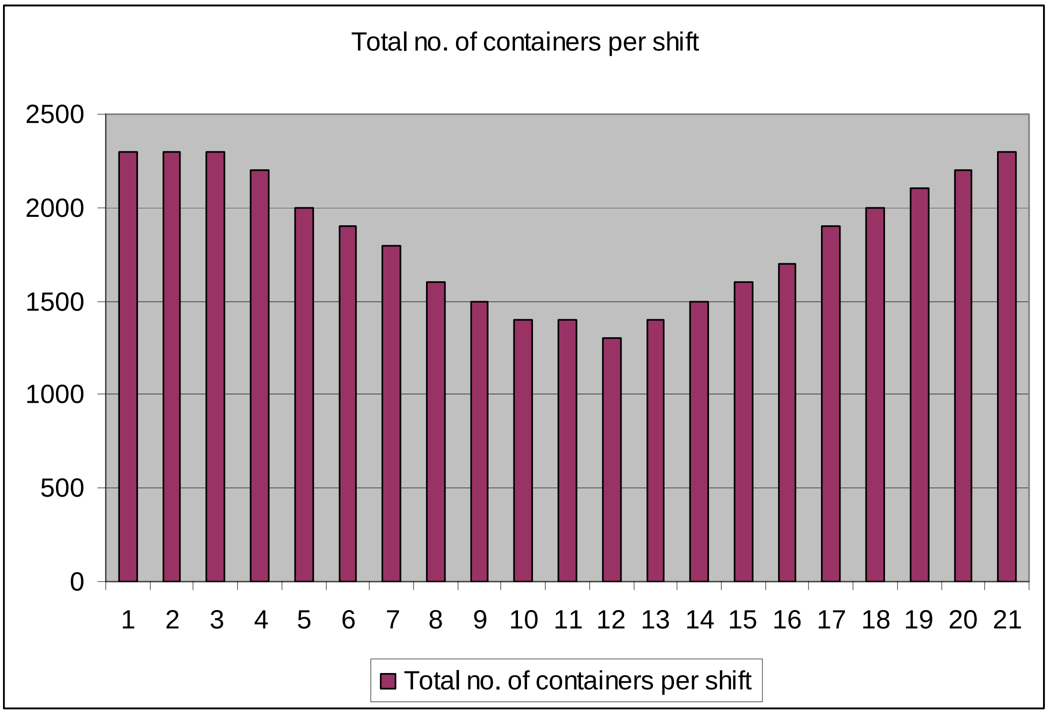

Figure 5.7 An example of input data (Extreme Case 2)

As shown in Table 5.4, the proposed method can handle the extreme case in which workload is concentrated in several periods. Although it cannot guarantee an optimal solution, the gap between the lower bound and the solution obtained is less than 1%.

Table 5.4 Results of the solution procedure for the large-scale problem (Extreme Case 2)

Utilization	Lower bound for the entire problem	Final solution	Gap
0.469712	268	268	0
0.517025	298	298	0
0.575078	329	330	0.30%
0.625234	358	359	0.28%
0.675132	387	389	0.52%
0.722561	415	416	0.24%
0.776546	441	441	0

The yard allocation problem YAP can be challenging, as the given yard template limits the solution. In addition, due to the tight constraints (Constraints (4.7) and (4.9)), it is not easy to find a good solution. However, by changing the template, there is more opportunity for the sub-blocks to pack activities while observing all the constraints, and hence make the problem easier to solve.

The solution procedure in this chapter tries to find a template which is able to pack the activities (i.e., make sure the crane shift is minimized). As there are some buffers in the space constraints, the template can be arranged in such a way that the activities in every shift can be packed. In general, if the intensity of the activities is not very high or the space constraint is not very tight, there will be a lot of possible templates which can result in efficient result (a complete pack). However, with the increase in the intensity of the activities (Extreme Case 2), it is harder to find a template which can pack well all the activities, and this is especially true when the space constraints become tighter.

In Chapters 3, 4, and 5, the storage yard management problem for export and transshipment containers is investigated. Import containers are usually stored in separate blocks from export and transshipment containers due to the different characteristics. The management problem for import containers is studied in the next chapter.

6 Simulation Study on the Docking Station Problem for Import Containers

Export and transshipment containers depart in large batches at designated time when the vessel comes. Hence, they can be stored in dedicated sub-blocks to reduce the number of reshuffles. In contrast, import containers arrive at the yard in large batches and in a predicted fashion, but depart one by one in an unpredictable order. Therefore, import containers are usually stored in separate blocks from export and transshipment containers so as to facilitate the ease of customer retrieval. These blocks are referred to as Full Container Load (FCL) blocks for import containers. Internal prime movers transport import containers to FCL blocks for storage and after that external trucks will come to collect import containers. Both internal prime movers and external trucks will queue up in the truck path beside the FCL blocks (see Figure 3.1) until a yard crane is available to serve them. In order to manage the competing demands for yard cranes in the FCL blocks, a docking station concept is proposed to change the current horizontal layout (i.e., yard blocks are parallel to the wharf) for FCL blocks to a vertical layout (i.e., yard blocks are perpendicular to the wharf). With the docking station concept, internal prime movers and external trucks are segregated which allows the port operator the flexibility of assigning yard crane service priority to internal prime movers and hence the ship turnaround time can be reduced when required. To verify the effectiveness of the docking station concept, two simulation models for the base layout and the proposed perpendicular layout are built respectively in this chapter.

6.1 Layout and Operations

6.1.1 Base Layout and Operations

As shown in Figure 6.1, all the blocks in the base layout are parallel to wharf. The blocks which are furthest from the quayside are the FCL blocks for import containers, whereas the blocks nearer to the quayside are for export and transshipment containers. Internal prime movers and external trucks queue up in the same truck path and wait to be served by a yard crane.

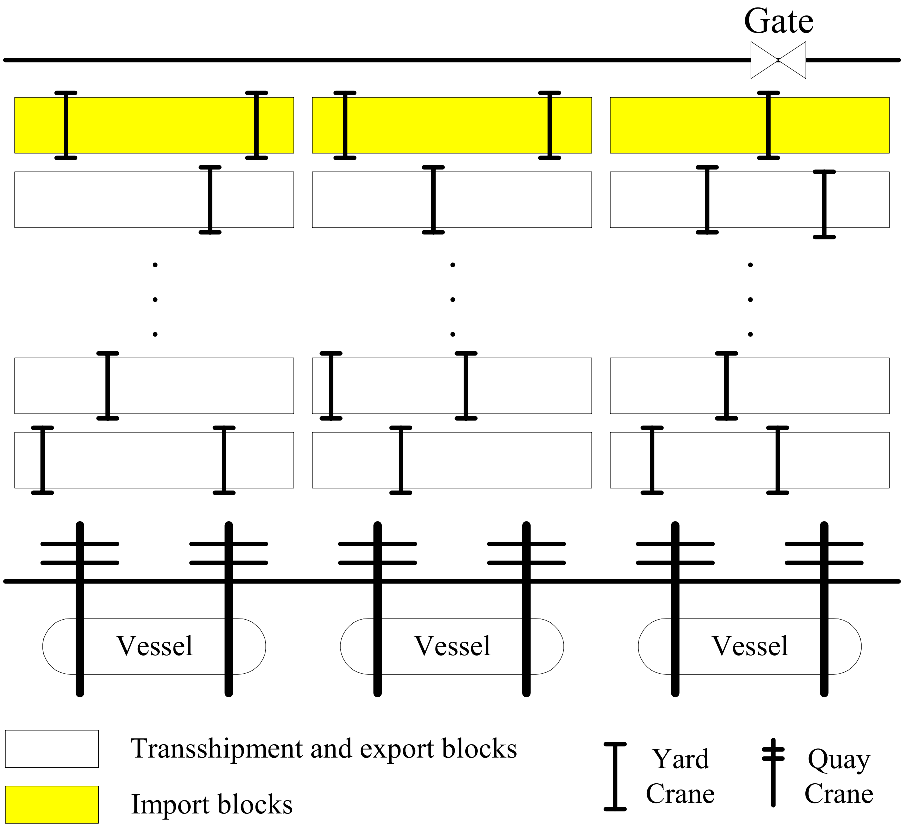

Figure 6.1 Current yard layout in the studied port

The problem in this layout is the competition for resources and the inability to prioritize the resource utilization. Internal prime movers should be served as soon as possible to reduce the vessel turnaround time, which is the overall objective of container terminals. External trucks should also be served within a time limit to ensure a service level. They compete for the yard cranes which are the scarce resources in this case. In addition, as the vehicles line up so near to one another, an FCFS principle has to be practiced for fairness sake. However, sometimes it is more desirable to give priority to internal prime movers so as to achieve a faster vessel turnaround time. But with some of external trucks arriving earlier or even blocking internal prime movers deliberately, practicing such a prioritization would be infeasible. Therefore, the docking station concept is proposed to tackle this problem.

6.1.2 Proposed Layout and Operations

A new layout is proposed under the docking station concept (see Figure 6.2). In the new layout, the blocks are now made perpendicular to the wharf. FCL blocks are assigned separately to import containers as before. External trucks now will wait at the docking bays at one end of the blocks (nearer to the gate); while internal prime movers will wait at the other end. By doing this, internal prime movers and external trucks are segregated so that service priority can be given to internal prime movers when necessary. In this concept, the currently used rubber tyred gantry (RTG) cranes need to be replaced with rail mounted gantry (RMG) cranes because the containers now need to be transported to or from the storage place by the yard cranes and RMG cranes are more stable than RTG cranes.

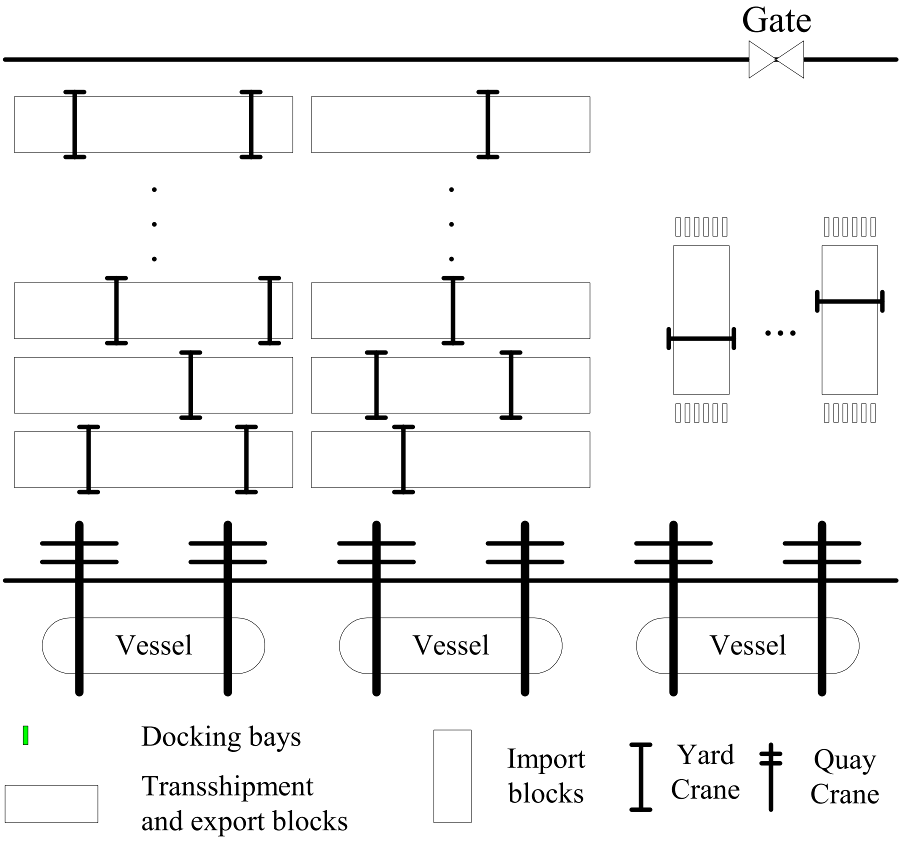

Figure 6.2 Proposed yard layout for the docking station concept

6.2 Model Assumptions

As this study mainly focuses on the FCL blocks for import containers, some assumptions are made in order to simplify the simulation model.

- The storage and retrieval of export and transshipment containers are not modeled in detail. The vehicles delivering or picking up containers will just go to any of the three sections, stay for a certain distribution of time which is derived from real life data, and then go back to the quayside or leave the yard.

- Each quay crane is assigned with a fixed number of 7 or 8 internal prime movers. This number is close to the average number of internal prime movers which are available to each quay crane in the studied port.

- Internal prime movers usually deliver import containers to the FCL blocks in bunch. To model the effect of bunching, it is assumed that all the import containers from one vessel would go to any of the two assigned FCL blocks until all import containers have been unloaded from the vessel.

6.3 Model Building

Simulation modeling is a powerful tool and often used to test out alternative designs and explore possibilities before making investment in new resources or implementing new policies. In this study, a simulation-based approach is adopted as well. Two simulation models by AutoMod 11.0 are developed, one simulating the current operations in the container storage yard (see Figure 6.3), and the other simulating the activities expected to take place after implementing the docking station concept (see Figure 6.4). As RMG cranes are used in Model 2 instead of RTG cranes in Model 1 and RMG cranes allow for greater width of the blocks, each block has 10 lanes (10 lanes×24 slots) as compared to the original 6 lanes (6 lanes×40 slots). The total area served by each yard crane remains unchanged, allowing the comparison to be made.

Model calibration is necessary so as to ensure that the first model is sufficiently representative of the current system. With the same input parameters, the two models are compared to verify whether the second model outperforms the first one. This comparison between the two models is justified for two reasons: firstly, the layouts for the two models have approximately the same total area; secondly, the number of critical resources (i.e. yard cranes) used in the two models are the same.

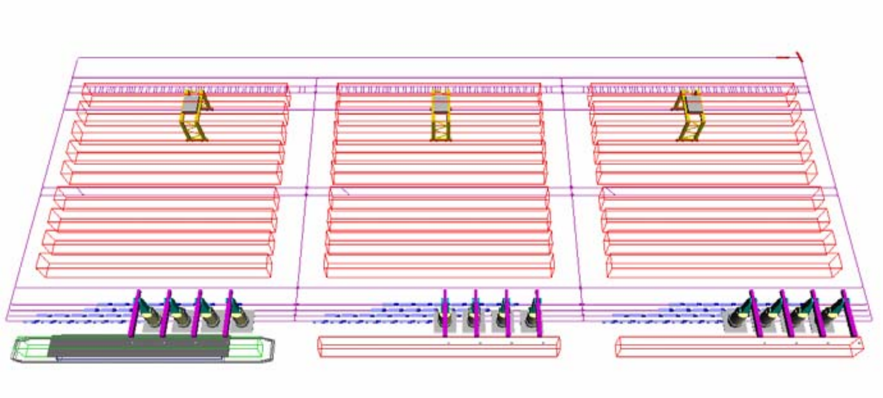

Figure 6.3 Simulation model for the base layout

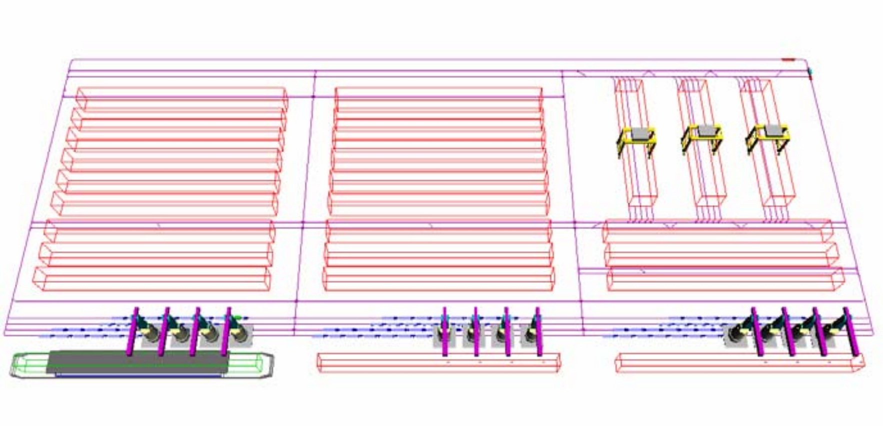

Figure 6.4 Simulation model for the proposed layout

6.4 Verification and Validation

The following actions are taken for model verification.

- Before writing the source code, flowcharts of the conceptual model are created; while after the model is complete, flowcharts of the model logic are made again. Translating the code back into a flowchart can uncover errors and weaknesses in the program. A flowchart of the model provides a graphical depiction of the model structure, which is easier to read and review than the actual code. Comparing flowcharts from the conceptual model and the computer model is part of the verification process.

- With the aid of simple animation features, the behavior of system entities such as vessels, internal prime movers, external trucks, and yard cranes, are closely observed to ensure that the required activity flow is being followed.

- Business graphs are used to track a variable, state, or other statistic in a graph and display it during a run, which provides a great way of identifying errant behavior in a model.

- To isolate the time and the section of code in which an entity is having problems the events of one or two particular entities instead of the whole model are traced, which reduces the tracing file size and makes the simulation run faster.

- Alarm and breakpoint are also used when debugging the model, which help us reduce the debugging effort and time.

- While building and verifying the model, a range of values are used to test the model in order to detect modeling problems early. AutoStat is used to make multiple runs that vary the range of input data to detect conditions under which the model fails or has unexpected behavior.

- Also, extreme values are used to test the model, even if those values might never be used during experimentation. The tests with extremely heavy workload and long run are performed and no deadlock or exception is detected, showing that the system is sustainable.

To make sure the model built matches the real world situation, the validation of the simulation model is done by comparing the simulation results with the actual data. Some simple sensitivity analysis and extreme-case tests are also conducted for the model validation. Model calibration is carried out, for example, by adding an unaccountable amount of service time ranging from 0 to 10 seconds to yard cranes. The purpose of validation and calibration is to confirm that the model has no flaws and is representative of the real system with fair precision.

Table 6.1 Results of model validation and calibration for the base layout

Measures	Actual data	Data from base model before calibration	Data from base model after calibration
Mean cycle time of internal prime movers t_i (min)	$t_i = 15 \pm 5$	14.16	15.63
Mean cycle time of external trucks t_o (min)	$t_o = 10 \pm 5$	9.52	10.45
Mean cycle time of yard cranes t_y (min)	$t_y = 2.5 \pm 0.5$	2.25	2.46

In addition, during the model development and validation phase, valuable feedback is also provided by experienced personnel from the studied port.

6.5 Simulation Results and Analysis

6.5.1 Measure of Performance

To observe the effect of the docking station in the FCL blocks, the following performance measures are proposed:

- Cycle time for both internal prime movers and external trucks which are going to the FCL blocks.
- Waiting time for both internal prime movers and external trucks which are going to the FCL blocks.

Cycle time for internal prime movers is defined as the total time taken by them to travel one loop from the quay to the FCL blocks and then back to the quay. Cycle time for external trucks is defined as the total time taken by them from entering the storage yard to leaving the yard. Statistics that are compared are: mean cycle time, 95th percentile of cycle time, and the average value of the top 5 percentile, and so on.

Waiting time is defined as the amount of time spent waiting, either due to other prime movers blocking the lane, or yard cranes not yet available. It is calculated as cycle time minus estimated traveling time. This performance measure is needed because it is necessary to assess the proportion of time that the vehicles spent waiting out of the total cycle time. As Model 2 has a slightly different layout from Model 1, if a significant change in cycle time is largely due to traveling time, any conclusion would be unfair. Moreover, the perception of vehicle drivers is greatly affected by the amount of waiting time instead of cycle time. Therefore, any significant change in waiting time is meaningful for the comparison.

In this study, the most important performance measure is the cycle time of internal prime movers. It is desirable to minimize the cycle time of internal prime movers because the port operator would like to achieve a shorter turnaround time for the vessels at berths; whereas for external trucks, there is not much emphasis on reducing the average cycle time, as long as the given service requirement could be met. This is also the reason why docking station concept is proposed because it allows service priority to be assigned to internal prime movers easily.

6.5.2 Warm-up Analysis

As the simulation model starts with an empty system, it is necessary to allow the system to reach a steady state before statistics are collected. The cumulative average cycle time of both internal prime movers and external trucks is smoothened after about 15 days (see Figures 6.5 and 6.6). Therefore, a 15-day warm-up period is chosen for this simulation. The run length is chosen to be 75 days to reduce the effect of initial bias. In addition, 5 independent simulation runs are carried out for each model, which is enough to reduce the variation of the simulation results.

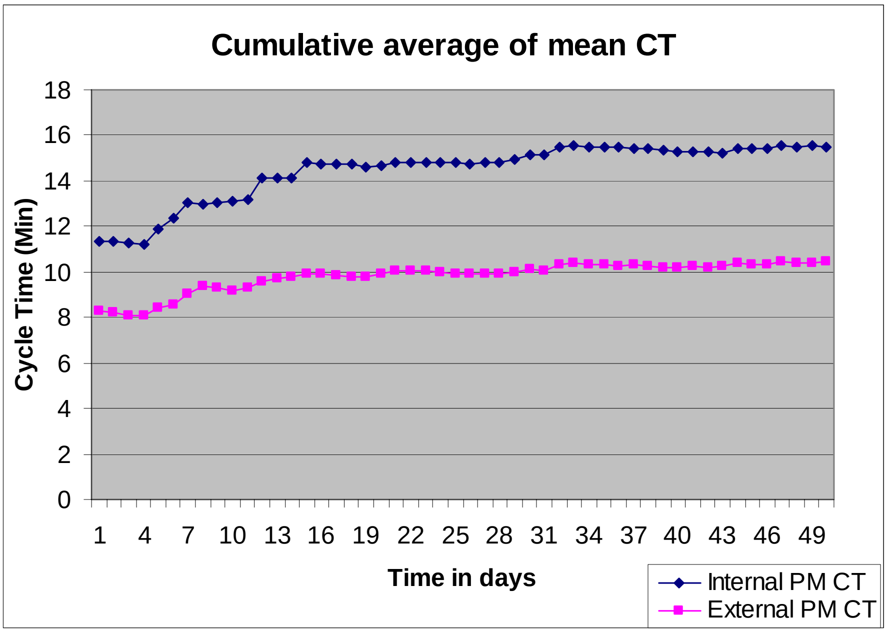

Figure 6.5 Cumulative average cycle time of vehicles with the first 15 days

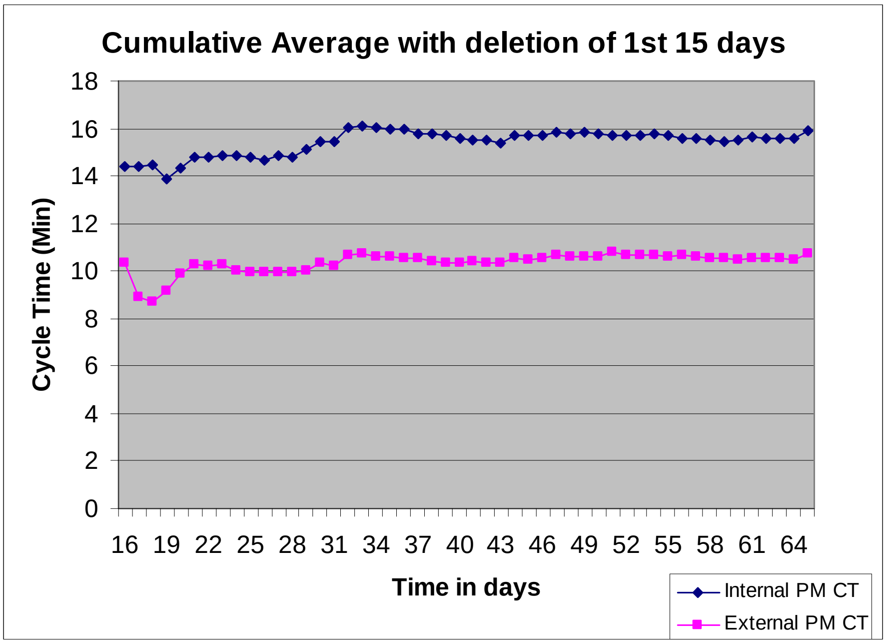

Figure 6.6 Cumulative average cycle time of vehicles w/o the first 15 days

6.5.3 Simulation Results for FCFS Case

The outputs from the model for the base layout and the one with docking stations are summarized in Tables 6.2 and 6.3. The service discipline for the two models is First Come First Served.

Table 6.2 Comparison for internal prime movers between the two models

Measures	Base Layout	Proposed Layout	Change
Mean cycle time (min)	15.63	18.95	21.24%
95% C.I.	(15.00, 16.27)	(17.89, 20.01)	-
Std. Dev. of cycle time	9.57	16.93	76.91%
95th percentile (min)	36	52	44.44%
Ave of top 5 percentile (min)	43.42	70.61	62.62%
% less than 30 min	90.27%	82.35%	-8.77%
Mean waiting time (min)	11.06	14.76	33.45%

Table 6.3 Comparison for external trucks between the two models

Measures	Base Layout	Proposed Layout	Change (%)
Mean cycle time (min)	10.45	10.82	3.54%
95% C.I.	(10.07, 10.83)	(10.37, 11.27)	-
Std. Dev. of cycle time	7.28	6.62	-9.07%
95th percentile (min)	26	27	3.85%
Ave of top 5 percentile (min)	35.67	31.85	-10.71%
% less than 30 min	96.34%	97.06%	7.47%
Mean waiting time (min)	7.45	7.12	-4.43%

It is observed that the performance of the model with docking station is slightly worse than the model for the base layout, especially for the internal prime movers. The mean cycle time of internal prime movers is increased by 21% in the model with docking station. As the traveling times of the two models are found to be close, the increase in cycle time is mainly resulted from the increase in waiting time, reflected by a 33% increase in waiting time for internal prime movers. The model with docking station also gives a higher variation in cycle time of internal prime movers, which is shown from the increased standard deviation and increased top 5 percentile average.

For the performance of external trucks, most of the statistics given by the two models are pretty close. For example, the 95% confidence interval for the mean cycle time from Model 1 is (10.07, 10.83), whereas Model 2 gives a confidence interval of (10.37, 11.27). As the two confidence intervals overlap, it is concluded that at a significance level of 5%, there is no significant difference between the mean cycle times of the two models.

From the simulation results, it is found that the docking station model is more sensitive to the bunching arrival of internal prime movers as compared to the current system. This is mainly due to the reduction in yard crane capacity in the docking station layout. In the base layout, whenever there is an unloading or loading job, the RTG crane only needs to travel once from its current location to the prime mover's location to unload/load the container. However, in the docking station layout, each unloading and loading job requires the RMG crane to make two trips. Although RMG cranes travel slightly faster than RTG cranes, the extra traveling distance in the docking station layout has a superseding effect and results in an overall reduction in yard crane capacity especially for the bunching arriving pattern.

6.5.4 Simulation Results for the Base Layout with Priority

By proposing a docking station layout, it is hoped that service priority can now be assigned to internal prime movers when necessary. Experiments with the two models are conducted to test the possibility of assigning priority to internal prime movers. The assignment of priority is done in this way: whenever there are more than one vehicle which have arrived at their destination slots and are waiting for the yard crane, the yard crane will choose to serve internal prime movers first.

By giving priority to internal prime movers in the current operation, a slight reduction in the mean cycle time of internal prime movers is observed, which is accompanied by an 11% increase in the mean cycle time of external trucks (see Table 6.4). This trade-off between the mean cycle times for internal and external vehicles is as expected. However, from the 95% confidence intervals, the reduction in cycle time is not significant enough to make a statistical impact.

Table 6.4 Comparison for the base layout with or without priority

		FCFS	With priority	Change (%)
Internal PMs	Mean cycle time	15.63	14.95	-4.35%
	95% C.I.	(15.00, 16.27)	(14.79, 15.11)	-
	Std. Dev. of cycle time	9.57	8.65	-9.61%
External Trucks	Mean cycle time	10.45	11.58	10.81%
	95% C.I.	(10.07, 10.83)	(11.43, 11.74)	-
	Std. Dev. of cycle time	7.28	9.79	34.48%

The reason for this insignificant reduction in the cycle time of internal prime movers is due to the failure to enforce a strict priority under the base layout. As internal prime movers and external trucks are mixed together in the FCL blocks, sometimes, even if the internal prime mover is served first, there are too many external trucks in the FCL blocks which have not been served, blocking the way of internal prime movers which need to go back to the quay side. Practically, it is also difficult to give priority to internal prime movers in real life because such an unfair practice will definitely arouse dissatisfaction among the external trucks' drivers. This is also one of the reasons to propose the docking station concept.

6.5.5 Simulation Results for the Proposed Layout with Priority

For the layout with docking stations, it is easy to assign priority to internal prime movers because the two types of vehicles are now segregated in two distant areas. In fact, an absolute priority is possible, by serving all the internal prime movers first before moving to the other end of the FCL blocks to serve external trucks. Table 6.5 shows the result of assigning absolute priority to internal prime movers. It is observed that docking station model with priority is able to achieve a 23% reduction from the one with FCFS in terms of the cycle time of internal prime movers. However, assigning absolute priority to internal prime movers is also impractical in real life as it incurs a great increase in the cycle time of external trucks by almost two times. A new operation method is needed to tackle this problem.

Table 6.5 Comparison for the proposed layout without or with absolute priority

		FCFS	With absolute priority	Change (%)
Internal PMs	Mean cycle time (min)	18.95	14.68	-22.53%
	95% C.I.	(17.89, 20.01)	(13.93, 15.42)	-
	Std. Dev. of cycle time	16.93	11.52	-31.96%
External Trucks	Mean cycle time (min)	10.82	29.95	176.80%
	95% C.I.	(10.37, 11.27)	(26.71, 33.19)	-
	Std. Dev. of cycle time	6.62	74.29	1022.21%

Another advantage of the docking station is to provide the port operator with the flexibility of manipulating the mean cycle time of internal prime movers by changing the service commitments to external trucks. Currently, it is required that 90% of the external trucks spend less than 30 minutes in the yard, i.e. within a target cycle time of 30 minutes, a 90% service level is required for external trucks. In statistical terms, this service requirement could be explained as: the 90th percentile for the cycle time of external trucks should not exceed 30 minutes.

The trade-off between the cycle times for internal prime movers and external trucks makes it impossible to reduce the cycle time of internal prime movers without lowering the service standard that is provided to external trucks. With the docking station layout, if the service requirement is set at a certain level, some kind of restricted priority can be given to internal prime movers by allowing yard cranes to serve any external truck which has waited for a certain amount of time (this threshold value can be adjusted according to the level of service to provide to external trucks). For example, with the current service requirement which states that 90% of external trucks need to be served within 30 minutes, the threshold value could be set at 20 minutes (i.e. yard crane will serve any external truck that has waited for more than 20 minutes first) in order to fulfill this service requirement. Table 6.6 shows how the cycle time of internal prime movers could be reduced by lowering the service requirement for external trucks. The 90th percentile for the cycle time of external trucks is equivalent to the minimum service target which could be promised to ensure a 90% service level. Figure 6.7 represents this variation in the cycle time of internal prime movers under different service targets.

Table 6.6 Variation in average cycle time of internal prime movers and the corresponding variation in the 90th percentiles for the cycle time of external trucks

Average internal PM cycle time (min)	90th percentile for external truck cycle time (min)
18.99	29
18.78	38
18.14	47
18.11	56
17.16	65
14.68 (absolute internal priority)	75

For example, from Table 6.6, it could be deduced that if the service requirement is revised, promising 90% of external trucks can be served within 47 minutes, instead of 30 minutes, a corresponding level of priority can be given to internal prime movers and achieve a cycle time of 18.14 minutes.

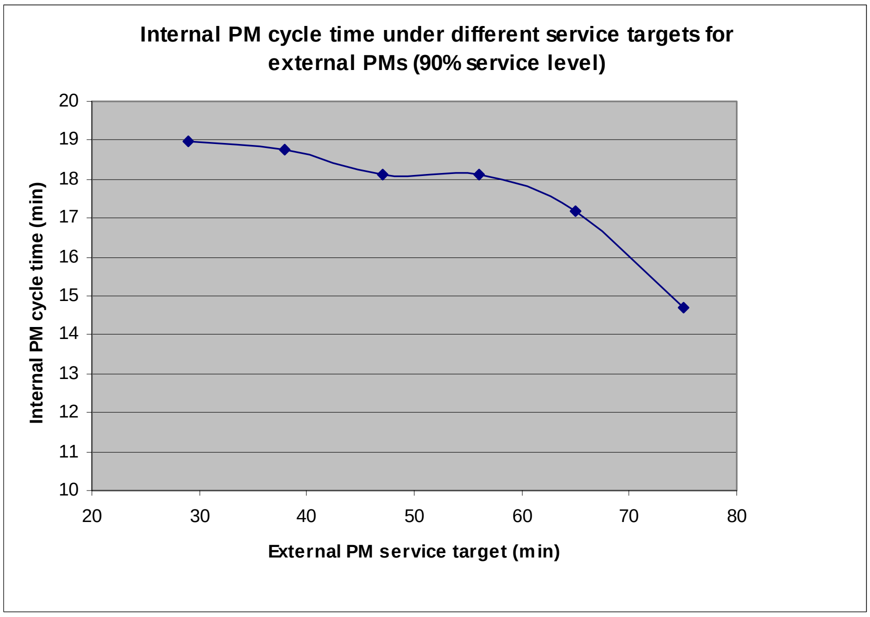

Figure 6.7 Variation in the cycle time of internal prime movers under different service requirements

The docking station model with priority is able to reduce the cycle time of internal prime movers while controlling the service provided to external trucks within some reasonable standard. However, as compared to the base layout with FCFS principle, the docking station model with priority is still not good, because to keep the cycle time of internal prime movers below the current level it requires too much sacrifice on the service provided to external trucks. The 90th percentile for the external truck cycle time even exceeds 70 minutes, which is absolutely too high.

Simulation results show that the cycle time of internal prime movers can be reduced when priority is given to them, but the required service level for external trucks needs to be slightly lowered due to the reduction in yard crane service capacity as a result of the extra movement of yard cranes. Therefore, a new method of operations in docking station layout is proposed in the next section to reduce the yard crane's effective traveling distance per handling, with which the internal prime movers' cycle time could be significantly reduced while the current service requirement for external trucks can also be met.

6.5.6 Recommended Improvements

The docking station layout, in terms of the design itself, is disadvantageous because it creates a bottleneck at the yard crane. As it is mentioned in Section 6.5.3, the main problem of docking stations is the increase in traveling distance of yard cranes and subsequently the reduction in yard crane capacity. However, docking station concept does provide the flexibility to give priority to internal prime movers, which is desirable for the port operator. Therefore, the yard crane's average traveling distance has to be reduced in order for the docking station to have an improved performance,

One possible suggestion is to reduce the length of the FCL blocks. But if no more yard cranes can be afforded, the total number of the FCL blocks will remain the same, with a reduced total area. Such a reduced storage area for import containers requires the rate at which these containers are picked up to be fast enough. Therefore, this suggestion should be exercised with great caution; otherwise the space capacity will be insufficient during extremely busy periods.

Alternatively, a more efficient method of operation can be adopted in the docking station. Therefore, a new model is proposed to solve the problem of increased traveling distance and at the same time take advantage of the flexibility to assign priority.

Whenever an internal prime mover arrives at the FCL blocks, the yard crane will unload the container and temporarily store it in a nearby location (e.g. the nearest four slots; see Figure 6.8). These import containers may be shifted to a location closer to the other end later. Although this involves double handling, there are always lull periods in the yard during which the yard cranes are idle. This method could reduce traffic congestion during the peak periods when a lot of prime movers come into the FCL blocks. As a result, when each unloading job comes, the yard crane only travels up to 4 slots to put it down; when each loading job comes, the yard crane only picks it up from the remaining 20 slots in the yard. This new method of operation effectively reduces the traveling distance especially when the yard crane continuously performs the same unloading/loading job on one side. When priority is given to internal prime movers, the yard crane usually needs to serve one sequence of jobs on the internal side before moving to the external side; therefore, the performance is expected to further improve under this modified model. The new model still tries to give priority to internal prime movers whenever possible but the method of operation is changed. The process of shifting the containers during lull periods is not modeled, as it is assumed that there is always sufficient idle time for the yard cranes to do the shifting work.

The recommended model is built with absolute priority to internal prime movers, and the simulation results are summarized in Table 6.7. There is a great reduction of 32% in the cycle time of internal prime movers as compared to the model for the base layout with FCFS principle. Moreover, almost 90% of external trucks could be served within 30 minutes. In other words, by the recommended model with absolute priority to internal prime movers, a significant reduction in the cycle time of internal prime movers is achieved while fulfilling the current service requirement for external trucks.

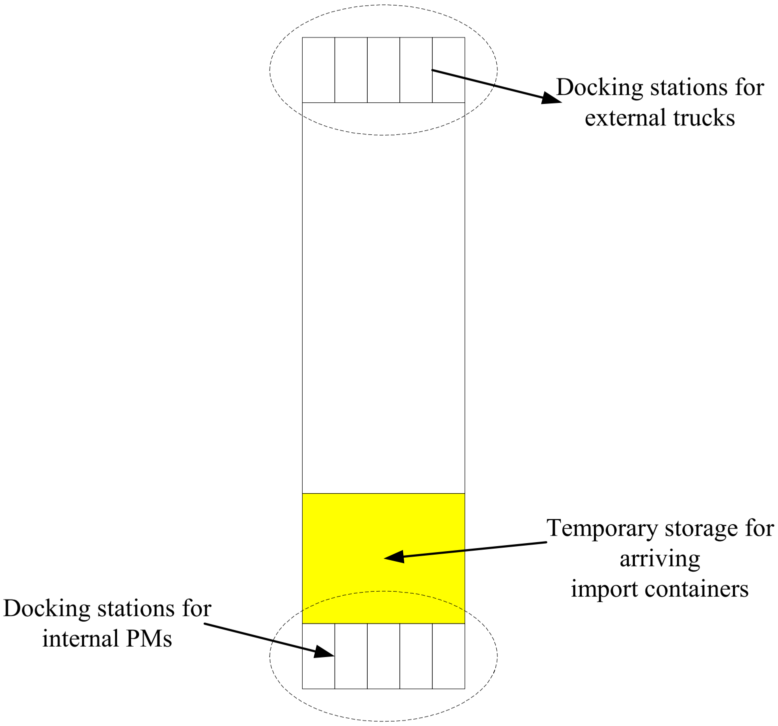

Figure 6.8 Illustration of recommended improvement

Table 6.7 Results of the recommended model with absolute priority to internal PMs

		Base layout model with FCFS principle	Recommended model with absolute priority to internal PMs	Change (%)
Internal PMs	Mean cycle time (min)	15.63	10.65	-31.86%
External trucks	Mean cycle time (min)	10.45	17.89	71.20%
	Std. Dev. of cycle time	7.28	37.44	414.29%
	% less than 30 min	96.34%	89.66%	-6.93%

Experiments with the recommended model are conducted with restricted priority to internal PMs also. A threshold value of 25 minutes is placed on the service for external trucks. Table 6.8 shows the results from the experiments.

The recommended model with restricted priority to internal PMs gives more satisfactory results. A significant reduction in the cycle time of internal prime movers can be achieved and the required level of service for external trucks is maintained, the standard deviation of the cycle time of external trucks is also well under control.

Table 6.8 Results of the recommended model with restricted priority to internal PMs

		Base layout model with FCFS principle	Recommended model with restricted priority to internal PMs	Change (%)
Internal PMs	Mean cycle time (min)	15.63	12.6	-19.39%
External trucks	Mean cycle time (min)	10.45	10.94	4.69%
	Std. Dev. of cycle time	7.28	8.28	13.74%
	% less than 30 min	96.34%	91.49%	-5.03%

7 Conclusions and Future Research

Container traffic has been growing steadily over the last 20 years. As a result, the carrying capacity of container vessels is becoming larger and larger. In order to handle larger container vessels and more volume of containers, port operators not only need to provide longer berth and deep water, they also need to improve the efficiency of the logistic activities in the container terminal. To improve the performance of container terminals, various methods have been proposed for all kinds of sub-processes or the whole terminal. However, these papers do not sufficiently address the particular needs of transshipment hubs, but more on general terminals which emphasize on import and export activities. The objective of this thesis is to study the storage yard management problem particularly for transshipment hubs, where the transshipment of containers is the major activity and the yard activity is heavy.

In this thesis two actual problems, the yard template problem and the docking station problem, are studied which are raised by a leading transshipment port operator. The yard template problem is to store export and transshipment containers in dedicated sub-blocks according to their destination vessel, which is used to reduce the number of reshuffles. A high-low workload balancing protocol is proposed to indirectly reduce the potential traffic congestion of prime movers. A mathematical model is developed to study the yard template problem and some heuristic algorithms and procedures are proposed to solve the model. The docking station problem is used to manage the competing demands for yard cranes in the container storage yard especially for import container blocks. According to the docking station concept, the current horizontal layout of the FCL blocks for import containers will be changed to a vertical layout. With the docking station concept, internal prime movers and external trucks are segregated which allows the port operator the flexibility of assigning yard crane service priority to internal prime movers and hence the ship turnaround time can be reduced when required. To verify the effectiveness of the docking station concept, two simulation models for the base layout and the proposed perpendicular layout are built respectively.

7.1 Conclusions

This thesis looks into an actual problem, the yard template problem, which is based on the consignment strategy and high-low workload balancing protocol. Currently, the port operator does not have any formal planning model to deal with the yard template problem and the decisions are based on intuition and past experiences. Hence a decision planning tool is developed to provide a holistic, scientific and systematic way of addressing the problem which takes into consideration of port operator's actual requirements. The model is particularly useful for transshipment hubs where transshipment of containers is the major activity and the yard activity is heavy. The solution to this model can be used directly by the port operator for planning purpose.

This yard template model under some scenarios cannot be solved to optimality by the commercial software package, but an iterative improving solution procedure is developed based on the MIP model to obtain satisfactory results. Although the proposed solution procedure could not guarantee an optimal solution, a bound which is useful in quantifying the quality of these solutions is developed. Furthermore, a feasible solution to the MIP model is also practical for the port operator to use. So far this is the first study to address the yard template problem with consignment strategy and high-low workload balancing protocol for a transshipment hub.

The simulation study verifies the effectiveness of the docking station concept for import container blocks. It turns out that the cycle time of internal prime movers can be reduced when priority is given to them, but the required service level for external trucks needs to be slightly lowered due to the reduction in yard crane service capacity as a result of the extra travelling distance of yard cranes. However, a new method of operations in docking station is proposed to reduce the yard crane's effective traveling distance per handling, with which the internal prime movers' cycle time could be significantly reduced while the current service requirement for external trucks can also be met.

Although the problems themselves, the requirements and some of the input data are from the studied port, the situation in transshipment hubs are similar and the problems might be practical for any transshipment hub. Therefore, the strategies, the formulation, and the solution method used to solve the yard template problem, the yard allocation problem, and the docking station problem are applicable to any transshipment port where transshipment of containers is the major activity and the yard activity is heavy.

7.2 Future Research Topics

There are several topics related to the scope of this thesis where future research can be conducted.

Firstly, for the yard template model, the uncertainty in the input data is not considered, especially for the loading time of each vessel and the arriving workload in each shift. Uncertainty in the input data will definitely make the problem much more complex, but it is the real situation and is a potential future research topic.

Secondly, in the yard template model, the division of sub-blocks is deterministic which means each sub-block is reserved for one vessel and the reservation cannot be changed. In reality, some slots in one sub-block may be reserved for one vessel in some shifts and are reserved for another vessel in other shifts. A future research direction is to deal with this dynamic situation.

Thirdly, for the docking station problem, the storage space and the number of yard cranes are kept the same for the base layout and the proposed layout in order to be fair for these two models. However, this configuration may not be optimal for the proposed layout. A future research topic is to determine the optimal amount of storage space and the optimal number of yard cranes by simulation optimization technique.

References

[1] Agerschou, H., Lundgren, H., Søensen, T., Ernst, T., Korsgaard, J., Schmidt, L.R., and Chi, W.K. Planning and Design of Ports and Marine Terminals. John Wiley and Sons, Chichester. 1983.

[2] Alicke, K. Modeling and Optimization of the Intermodal Terminal Mega Hub. OR Spectrum 24, pp.1-17. 2002.

[3] Asef-Vaziri, A., Cadavid, M., and Dougherty, E. A Combined Container Handling System in Maritime Terminals. Paper presented at Annual Meeting INFORMS Atlanta 2003, Oct. 2003, California State University, Northridge. 2003a.

[4] Asef-Vaziri, A., Khoshnevis, B., and Parsaei, H. Potentials for ASRS and AGVS in Maritime Container Terminals. Working paper, University of Houston. 2003b.

[5] Avriel, M. and Penn, M. Exact and Approximate Solutions of the Container Ship Stowage Problem. Computers and Industrial Engineering 25 (1-4), pp.271-274. 1993.

[6] Avriel, M., Penn, M., Shpirer, N., and Witteboon, S. Stowage Planning for Container Ships to Reduce the Number of Shifts. Annals of Operations Research 76, pp. 55-71. 1998.

[7] Avriel, M., Penn, M., and Shpirer, N. Container Ship Stowage Problem: Complexity and Connection to the Coloring of Circle Graphs. Discrete Applied Mathematics 103 (1-3), 271-279. 2000.

[8] Baker, C. High Time for Straddles. Cargo Systems, pp. 23-26. 1998.

[9] Ballis, A. and Abacoumkin, C. A Container Terminal Simulation Model with Animation Capabilities. Journal of Advanced Transportation 30 (1), pp.37-57. 1996.

[10] Bish, E.K., Leong, T., Li, C., Ng, J.W.C., and Simchi-Levi, D. Analysis of a New Vehicle Scheduling and Location Problem. Naval Research Logistics 48, pp.363-385. 2001.

[11] Bish, E.K. A Multiple-Crane-Constrained Scheduling Problem in a Container Terminal. European Journal of Operational Research 144, pp.83-107. 2003.

[12] Bish, E.K., Chen, F.Y., Leong, Y.T., Nelson, B.L., Ng, J.W.C., and Simchi-levi, D. Dispatching Vehicles in a Mega Container Terminal. OR Spectrum 27, pp.491-506. 2005.

[13] Böse, J., Reiners, T., Steenken, D., and Voβ, S. Vehicle Dispatching at Seaport Container Terminals Using Evolutionary Algorithms. In: Sprague R H (ed) Proceedings of the 33rd Annual Hawaii International Conference on System Sciences, DTM-IT, pp.1-10. IEEE, Piscataway. 2000.

[14] Bostel, N. and Dejax, P. Models and Algorithms for Container Allocation Problems on Trains in a Rapid Transshipment Shunting Yard. Transportation Science 32 (4), pp.370-379. 1998.

[15] Briskorn, D., Drexl, A., and Hartmann, S. Inventory-based Dispatching of Automated Guided Vehicles on Container Terminals. OR Spectrum 28, pp.611-630. 2006.

[16] Bruzzone, A. and Signorile, R. Simulation and Genetic Algorithms for Ship Planning and Shipyard Layout. Simulation 71, pp.74-83. 1998.

[17] Carrascosa, C., Rebollo, M., Julian, V., and Botti, V. A MAS Approach for Port Container Terminal Management: the Transtainer Agent. In: Actas de SCI'01, pp.1-5. International Institute of Informatics and Systemics, Orlando, FL. 2001.

[18] Chan, S.H. Dynamic AGV-container Job Deployment Strategy. Master of Science, National University of Singapore. 2001.

[19] Chen, T. Yard Operations in the Container Terminal: A Study in the "Unproductive Moves". Maritime Policy and Management 26 (1), pp.27-38. 1999.

[20] Chen, C.Y., Chao, S.L., and Hsieh, T.W. A Time-Space Network Model for the Space Resource Allocation Problem in Container Marine Transportation. In Proceedings of the 17th International Symposium on Mathematical Programming, 2000, Atlanta, USA. 2000.

[21] Chen, C.S., Lee, S.M., and Shen, Q.S. An Analytical Model for the Container Loading Problem. European Journal of Operational Research 80 (1), pp.68-76. 1995.

[22] Chen, Y., Leong, Y.T., Ng, J.W.C., Demir, E.K., Nelson, B.L., and Simchi-Levi, D. Dispatching Automated Guided Vehicles in a Mega Container Terminal. Paper presented at INFORMS Montreal, May 1998.

[23] Cheung, R.K. and Chen, C.Y. A Two-Stage Stochastic Network Model and Solution Methods for the Dynamic Empty Container Allocation Problem. Transportation Science 32 (2), pp.142–162. 1998.

[24] Chung, Y.G., Randhawa, S.U., and Mcdowell, E.D. A Simulation Analysis for a Transtainer-based Container Handling Facility. Computers and Industrial Engineering 14 (2), pp.113-115. 1988.

[25] Chung, R.K., Li, C.L., and Lin, W. Interblock Crane Deployment in Container Terminals. Transportation Science 36 (1), 79-93. 2002.

[26] Crainic, T.G., Gendreau, M., and Dejax, P. Dynamic and Stochastic Models for the Allocation of Empty Containers. Operations Research 41, 102-126. 1993.

[27] Daganzo, C.F. The Crane Scheduling Problem. Transportation Research B 23 (3), pp.159-175. 1989.

[28] Davies, A.P. and Bischoff, E.E. Weight Distribution Considerations in Container Loading. European Journal of Operational Research 114, pp.509-527. 1999.

[29] De Castilho, B. and Daganzo, C.F. Handling Strategies for Import Containers at Marine Terminals. Transportation Research B 27 (2), pp.151-166. 1993.

[30] Dekker, R., Voogd, P., and Asperen, E. Advanced Methods for Container Stacking. OR Spectrum 28 (4), pp.563-586. 2006.

[31] Dubrovsky, O., Levitin, G., and Penn, M. A Genetic Algorithm with a Compact Solution Encoding for the Container Ship Stowage Problem. Journal of Heuristics 8, pp.585-599. 2002.

[32] Duinkerken, M.B., Ottjes, J.A., Evers, J.J.M., Kurstjens, S., Dekker, R., and Dellaert, N. Simulation Studies on Inter Terminal Transport at the Maasvlakte. In the proceedings 2nd TRAIL PhD Congress 1996 “Defence or Attack”, May 1996, Rotterdam, Netherlands.

[33] Duinkerken, M.B., Evers, J.J.M., and Ottjes, J.A. TRACES: Traffic Control Engineering System: A Case Study on Container Terminal Automation. In the Proceedings of the 1999 Summer Computer Simulation Conference (SCSC 1999), July 1999, Chicago, USA. ISBN 1-56555-173-7.

[34] Duinkerken, M.B. and Ottjes, J.A. A Simulation Model for Automated Container Terminals. In the Proceedings of the Business and Industrial Simulation Symposium (ASTC 2000), April 2000, Washington D.C., USA. ISBN 1-56555-199-0.

[35] Duinkerken, M.B., Evers, J.J.M., and Ottjes, J.A. A Simulation Model for Integrating Quay Transport and Stacking Policies on Automated Container Terminals. In the Proceedings of the 15th European Simulation Multi-conference (ESM 2001), June 2001, Prague, Czechoslovakia. ISBN 1-56555-225-3.

[36] Duinkerken, M.B., Evers, J.J.M., and Ottjes, J.A. Improving Quay Transport on Automated Container Terminals. In the Proceedings of the IASTED International Conference Applied Simulation and Modeling (ASM 2002), June 2002, Crete. ISBN 0-88986-334-2.

[37] Duinkerken, M.B., Dekker, R., Kurstjens, S.T.G.L., Ottjes, J.A., and Dellaert, N.P. Comparing Transportation Systems for Inter-Terminal Transport at the Maasvlakte Container Terminals. OR Spectrum 28, pp.469-493. 2006.

[38] Edmond, E.D. and Maggs, R.P. How Useful are Queue Models in Port Investment Decisions for Container Berths. Journal of the Operational Research Society 29 (8), pp.741-750. 1978.

[39] Eisenberg, R., Stahlbock, R., Voβ, S., and Steenken, D. Sequencing and Scheduling of Movements in an Automated Container Yard Using Double Rail-Mounted Gantry Cranes. Working paper, University of Hamburg. 2003.

[40] Eley, M. A Bottleneck Assignment Approach to the Multiple Containers Loading Problem. OR Spectrum 25, 54-60. 2003.

[41] Evers, J.J.M. and Koppers, S.A.J. Automated Guided Vehicle Traffic Control at a Container Terminal. Transportation Research A 30 (1), pp.21-34. 1996.

[42] Gademann, A.J.R.M. and van de Velde, S.L. Positioning Automated Guided Vehicles in a Loop Layout. European Journal of Operational Research 127 (3), pp.565-573. 2000.

[43] Gambardella, L.M., Rizzoli, A.E., and Zaffalon, M. Simulation and Planning of an Intermodal Container Terminal. Simulation 71 (2), pp.107-116. 1998.

[44] Gambardella, L.M., Mastrolilli, M., Rizzoli, A.E., and Zaffalon, M. An Optimization Methodology for Intermodal Terminal Management. Journal of Intelligent Manufacturing 12, pp.521-534. 2001.

[45] Gibson, R., Carpenter, B., and Seeburger, S. A Flexible Port Traffic Planning Model. In the Proceedings of the 1992 Winter Simulation Conference, pp.1296-1306. 1992.

[46] Giemsch, P. and Jellinghaus, A. Organization Models for the Containership Stowage Problem. Paper presented at the annual international conference of the German Operations Research Society (OR 2003), Sep. 2003, Heidelberg, Germany.

[47] Grunow, M., Günther, H.O., and Lehmann, M. Dispatching Multi-Load AGVs in Highly Automated Seaport Container Terminals. OR Spectrum 26 (2), pp.211-235. 2004.

[48] Grunow, M., Günther, H.O., and Lehmann, M. Strategies for Dispatching AGVs at Automated Seaport Container Terminals. OR Spectrum 28, pp.587-610. 2006.

[49] Guan, Y., Xiao, W.Q., Cheung, R.K., and Li, C.L. A Multiprocessor Task Scheduling Model for Berth Allocation: Heuristic and Worst Case Analysis. Operations Research Letters 30, 343-350. 2002.

[50] Guan, Y. and Cheung, R.K. The Berth Allocation Problem: Models and Solution Methods. OR Spectrum 26, pp.75-92. 2004.

[51] Haghani, A., and Kaisar, E.I. A Model for Designing Container Loading Plans for Containerships. Working paper, University of Maryland, presented at Transportation Research Board 2001 Annual Meeting. 2001.

[52] Hanafi, S., Jesus, J., and Semet, F. The Container Assignment Problem: A Case Study on the Port of Lille. Working paper, University of Valenciennes, presented at ODYSSEUS 2003.

[53] Hartmann, S. A General Framework for Scheduling Equipment and Manpower on Container Terminals. OR Spectrum 26, pp.51-74. 2004.

[54] Hartmann, S. Generating Scenarios for Simulation and Optimization of Container Terminal Logistics. OR Spectrum 26 (2), pp.171-192. 2004.

[55] Hayuth, Y., Pollatschek, M.A., and Roll, Y. Building a Port Simulator. Simulation 63, pp.179-189. 1994.

[56] Henesey, L., Wernstedt, F., and Davidsson, P. A Market Based Approach to Container Port Terminal Management. In: Proceedings of the 15th European Conference on Artificial Intelligence, Workshop – Agent Technologies in Logistics, Lyon, France, 2002.

[57] Holguín-Veras, J. and Walton, C. On the Development of a Computer System to Simulate Port Operations Considering Priorities. In: Proceedings of the 28th Conference on Winter Simulation, pp. 1471-1478. ACM Press, New York. 1996.

[58] Holguín-Veras, J. and Jara-Díaz, S. Optimal Pricing for Priority Service and Space Allocation in Container Ports. Transportation Research B 33, pp.81-106. 1999.

[59] Hulten, L.A.R. Container Logistics and its Management. PhD thesis, Chalmers University of Technology: Department of Transportation and Logistics. 1997.

[60] Imai, A., Nagaiwa, K., and Tat, C.W. Efficient Planning of Berth Allocation for Container Terminals in Asia. Journal of Advanced Transportation 31 (1), pp.75-94. 1997.

[61] Imai, A., Nishimura, E., and Papadimitriou, S. The Dynamic Berth Allocation Problem for a Container Port. Transportation Research-B 35 (4), pp.401-417. 2001.

[62] Imai, A., Nishimura, E., and Papadimitriou, S. Berth Allocation with Service Priority. Transportation Research-B 37 (5), pp.437-457. 2003.

[63] Kang, J.G. and Kim, Y.D. Stowage Planning in Maritime Container Transportation. Journal of the Operational Research Society 53, pp.415-426. 2002.

[64] Kia, M., Shayan, E., and Ghotb, F. Investigation of Port Capacity under a New Approach by Computer Simulation. Computers & Industrial Engineering 42, pp.533-540. 2002.

[65] Kim, K.H. Evaluation of the Number of Reshuffles in Storage Yards. Computers and Industrial Engineering 32 (4), pp.701-711. 1997.

[66] Kim, K.H. and Bae, J.W. Re-marshaling Export Containers in Port Container Terminals. Computers & Industrial Engineering 35, pp.655-658. 1998.

[67] Kim, K.H. and Bae, J.W. A Dispatching Method for Automated Guided Vehicles to Minimize Delays of Containership Operations. International Journal of Management Science 5 (1), pp.1-25. 1999.

[68] Kim, K.Y. and Kim, K.H. A Routing Algorithm for a Single Transfer Crane to Load Export Containers onto a Containership. Computers and Industrial Engineering 33 (3-4), pp.673-676. 1997.

[69] Kim, K.H. and Kim, H.B. The Optimal Determination of the Space Requirement and the Number of Transfer Cranes for Import Containers. Computers and Industrial Engineering 35 (3-4), pp.427-430. 1998.

[70] Kim, K.H. and Kim, H.B. Segregating Space Allocation Models for Container Inventories in Port Container Terminals. International Journal of Production Economics 59, pp.415-423. 1999a.

[71] Kim, K.H. and Kim, K.Y. Routing Straddle Carriers for the Loading Operation of Containers Using a Beam Search Algorithm. Computers & Industrial Engineering 36 (1), pp.109-136. 1999b.

[72] Kim, K.Y. and Kim, K.H. A Routing Algorithm for a Single Straddle Carrier to Load Export Containers onto a Containership. International Journal of Production Economics 59 (1-3), pp.425-433. 1999c.

[73] Kim, K.Y. and Kim, K.H. An Optimal Routing Algorithm for a Transfer Crane in Port Container Terminals. Transportation Science 33 (1), pp.17-33. 1999d.

[74] Kim, K.H. and Kim, H.B. The Optimal Sizing of the Storage Space and Handling Facilities for Import Containers. Transportation Research-B 36, pp.821-835. 2002.

[75] Kim, K.Y. and Kim, K.H. Heuristic Algorithms for Routing Yard-Side Equipment for Minimizing Loading Times in Container Terminals. Naval Research Logistics 50, pp.498-514. 2003.

[76] Kim, K.H. and Moon, K.C. Berth Scheduling by Simulated Annealing. Transportation Research-B 37 (6), pp.541-560. 2003.

[77] Kim, K.H. and Park, K.T. A Note on a Dynamic Space-Allocation Method for Outbound Containers. European Journal of Operational Research 148 (1), 92-101. 2003.

[78] Kim, K.H., Park, Y.M., and Ryu, K.R. Deriving Decision Rules to Locate Export Containers in Storage Yards. European Journal of Operational Research 124, pp.89-101. 2000.

[79] Kim, K.H., Wang, S.J., Park, Y.M., Yang, C.H., and Bae, J.W. A Simulation Study on Operation Rules for Automated Container Yards. Presentation/Proc. of the 7th Annual International Conference on Industrial Engineering. 2002.

[80] Kim, K.H., Lee, K.M., and Hwang, H. Sequencing Delivery and Receiving Operations for Yard Cranes in Port Container Terminals. International Journal of Production Economics 84 (3), pp.283-292. 2003.

[81] Kim, K.H., Kang, J.S., and Ryu, K.R. A Beam Search Algorithm for the Load Sequencing of Outbound Containers in Port Container Terminals. OR Spectrum 26, pp.93-116. 2004.

[82] Koh, P.H., Goh, J.L.K., Ng, H.S., and Ng, H.C. Using Simulation to Preview Plans of Container Port Operations. In the Proceedings of the 26th Conference on Winter Simulation, pp.1109-1115. Society for Computer Simulation International, San Diego, CA. 1994.

[83] Konings, J.W. Integrated Centres for the Transshipment, Storage, Collection and Distribution of Goods: A Survey of the Possibilities for a High-Quality Intermodal Transport Concept. Transport Policy 3 (1-2), pp.3-11. 1996.

[84] Koo, P.H., Lee, W.S., and Jang, D.W. Fleet Sizing and Vehicle Routing for Container Transportation in a Static Environment. OR Spectrum 26 (2), pp.193-209. 2004.

[85] Kozan, E. Increasing the Operational Efficiency of Container Terminals in Australia. Journal of the Operational Research Society 48 (2), pp.151-161. 1997a.

[86] Kozan, E. Comparison of Analytical and Simulation Planning Models of Seaport Container Terminals. Transportation Planning and Technology 20 (3), pp.235-248. 1997b.

[87] Kozan, E. and Preston, P. Genetic Algorithms to Schedule Container Transfers at Multimodal Terminals. International Transactions in Operational Research 6 (3), pp.311-329. 1999.

[88] Kozan, E. Optimizing Container Transfers at Multimodal Terminals. Mathematical and Computer Modeling 31 (10-12), pp.235-243. 2000.

[89] Kozan, E. and Preston, P. Mathematical Modeling of Container Transfers and Storage Locations at Seaport Terminals. OR Spectrum 28 (4), pp.519-537. 2006.

[90] Kurstjens, S.T.G.L., Dekker, R., Dellaert, N.P., Duinkerken, M.B., Ottjes, J.A., and Evers, J.J.M. Planning of Inter Terminal Transport at the Maasvlakte, TRAIL Proceedings of the 2nd TRAIL Congress. 1996.

[91] Lai, K.K. and Shih, K. A Study of Container Berth Allocation. Journal of Advanced Transportation 26, pp.45-60. 1992.

[92] Lee, L.H., Chew, E.P., Tan, K.C., and Han, Y. An Optimization Model for Storage Yard Management in Transshipment Hubs. OR Spectrum 28 (4), pp.539-561. 2006.

[93] Leeper, J.H. Integrated Automated Terminal Operations. Transportation Research Circular 33 (2), pp.23-28. 1988.

[94] Legato, P. and Mazza, R.M. Berth Planning and Resources Optimization at a Container Terminal via Discrete Event Simulation. European Journal of Operational Research 133, pp.537-547. 2001.

[95] Lehmann, M., Grunow, M., and Günther, H.O. Deadlock Handling for Real-Time Control of AGVs at Automated Container Terminals. OR Spectrum 28, pp.631-657. 2006.

[96] Leong, C.Y. Simulation Study of Dynamic AGV-Container Job Deployment Scheme. Master of Science, National University of Singapore. 2001.

[97] Li, C.L. and Vairaktarakis, G.L. Loading and Unloading Operations in Container Terminals. IIE Transactions 36, pp.287-297. 2004.

[98] Lim, A. The Berth Planning Problem. Operations Research Letters 22, pp.105-110. 1998.

[99] Lim, J.K., Kim, K.H., Yoshimoto, K., and Lee, J.H. A Dispatching Method for Automated Guided Vehicles by Using a Bidding Concept. OR Spectrum 25, pp.25-44. 2003.

[100] Lin, W. On Dynamic Crane Deployment in Container Terminals. Master of philosophy in industrial engineering and engineering management, University of Science and Technology, 2000.

[101] Linn, R.J., Liu, J., Wan, Y.W., Zhang, C., and Murty, K.G. Rubber Tyred Gantry Crane Deployment for Container Yard Operation. Computers & Industrial Engineering 45, pp.429-442. 2003.

[102] Liu, C.I., Jula, H. and Ioannou, P.A. Design, Simulation, and Evaluation of Automated Container Terminals. IEEE Transactions on Intelligent Transportation Systems 3 (1), pp.12-26. 2002.

[103] Liu, C.I., Jula, H., Vukadinovic, K., and Ioannou, P. Automated Guided Vehicles System for Two Container Yard Layouts. Transportation Research Part C 12, pp.349-368. 2004.

[104] Mastrolilli, M., Fornara, N., Gambardella, L.M., Rizzoli, A.E., and Zaffalon, M. Simulation for Policy Evaluation, Planning and Decision Support in an Intermodal Container Terminal. In the proceedings of the International Workshop "Modeling and Simulation within a Maritime Environment", Sep. 1998, pp.33-38. Society for Computer Simulation International, Riga, Latvia.

[105] Mattfeld, D. The Management of Transshipment Terminals. Habilitation thesis, University at Bremen. 2003.

[106] Meersmans, P.J.M. Optimization of Container Handling Systems. Thela Thesis, Amsterdam. 2002.

[107] Meersmans, P.J.M. and Wagelmans, A.P.M. Effective Algorithms for Integrated Scheduling of Handling Equipment at Automated Container Terminals. Technical Report EI 2001-19, Erasmus University Rotterdam, Econometric Institute. 2001a.

[108] Meersmans, P.J.M. and Wagelmans, A.P.M. Dynamic Scheduling of Handling Equipment at Automated Container Terminals. Technical Report EI 2001-33, Erasmus University Rotterdam, Econometric Institute, 2001b.

[109] Merkuryev, Y., Tolujew, J., Blümel, E., Novitsky, L., Ginters, E., Viktorova, E., Merkuryeva, G., and Pronins, J. A Modeling and Simulation Methodology for Managing the Rigaharbour Container Terminal. Simulation 71, pp.84-95. 1998.

[110] Moon, K.C. A Mathematical Model and a Heuristic Algorithm for Berth Planning. PhD thesis, Pusan National University, 2000.

[111] Moorthy, R. and Teo, C.P. Berth Management in Container Terminal: the Template Design Problem. OR Spectrum 28, pp.495-518. 2006.

[112] Moorthy, R.L. and Wee, H.G. Deadlock Prediction and Avoidance in an AGV System. Master of science, Sri Ramakrishna Engineering College, National University of Singapore. 2000.

[113] Moorthy, R.L., Wee, H.G., Ng, W.C., and Teo, C.P. Cyclic Deadlock Prediction and Avoidance for Zone-Controlled AGV System. International Journal of Production Economics 83 (3), pp.309-324. 2003.

[114] Mosca, R., Giribone, P. and Bruzzone, A. Simulation and Automatic Parking in a Training System for Container Terminal Yard Management. In: Proceedings of ITEC, pp.65-72. 1994.

[115] Muller, G. Intermodal Freight Transportation, 3rd ed. Eno Foundation for Transportation, Westport, CN. 1995.

[116] Murty, K.G., Liu, J., Wan, Y., and Linn, R. A Decision Support System for Operations in a Container Terminal. Decision Support Systems 39 (3), pp.309-332. 2005.

[117] Murty, K.G., Wan, Y., Liu, J., Tseng, M.M., Leung, E., Lai, K.K. and Chiu, H.W.C. Hong Kong International Terminals Gains Elastic Capacity Using a Data-Intensive Decision-Support System. Interface 35 (1), pp.61-75. 2005.

[118] Nam, K.C. and Ha, W.I. Evaluation of Handling Systems for Container Terminals. Journal of Waterway, Port, Coastal and Ocean Engineering 127 (3), pp.171-175. 2001.

[119] Nam, K.C., Kwak, K.S., and Yu, M.S. Simulation Study of Container Terminal Performance. Journal of Waterway, Port, Coastal and Ocean Engineering 128 (3), pp.126-132. 2002.

[120] Narasimhan, A. and Palekar, U.S. Analysis and Algorithms for the Transtainer Routing Problem in Container Port Operations. Transportation Science 36 (1), pp.63-78. 2002.

[121] Nevins, M.R., Macal, C.M., Love, R.J., and Bragen, M.J. Simulation, Animation and Visualization of Seaport Operations. Simulation 71, pp.96-106. 1998.

[122] Nishimura, E., Imai, A., and Papadimitriou, S. Berth Allocation Planning in the Public Berth System by Genetic Algorithms. European Journal of Operational Research 131, pp.282-292. 2001.

[123] Nishimura, E., Imai, A., and Papadimitriou, S. Yard Trailer Routing at a Maritime Container Terminal. Transportation Research Part E 41, pp.53-76. 2005.

[124] Ottjes, J.A., Duinkerken, M.B., Evers, J.J.M., and Dekker, R. Robotised Inter Terminal Transport of Containers: A Simulation Study at the Rotterdam Port Area. In: Proceedings of the 8th European Simulation Symposium (ESS 1996), October 1996, Genua [SCS], 1996.

[125] Ottjes, J.A. and Veeke, H.P.M. Simulation of a New Port Ship Interface Concept for Inter Modal Transport. In: Proceedings of the 11th European Simulation Symposium (ESS 1999), October 1999, Erlangen. 1999.

[126] Ottjes, J.A., Veeke, H.P.M., and Duinkerken, M.B. Simulation Studies of Robotized Multi Terminal Systems. In: Proceedings International Congress on Freight Transport Automation and Multimodality (FTAM), May 2002, Delft [TRAIL]. 2002.

[127] Ottjes, J.A., Veeke, H.P.M., Duinkerken, M.B., Rijsenbrij, J.C., and Lodewijks, G. Simulation of a Multi-terminal System for Container Handling. OR Spectrum 28, pp.447-468, 2006.

[128] Park, K.T. and Kim, K.H. Berth Scheduling for Container Terminals by Using a Subgradient Optimization Technique. Journal of the Operational Research Society 53, pp.1054-1062. 2002.

[129] Park, Y.M. and Kim, K.H. A Scheduling Method for Berth and Quay Cranes. OR Spectrum 25, pp.1-23. 2003.

[130] Peterkofsky, R.I. and Daganzo, C.F. A Branch and Bound Solution Method for the Crane Scheduling Problem. Transportation Research B 24 (3), pp.159-172. 1990.

[131] Powell, W.B. and Carvalho, T.A. Real-Time Optimization of Containers and Flatcars for Intermodal Operations. Transportation Science 32 (2), pp.110-126. 1998.

[132] Preston, P. and Kozan, E. An Approach to Determine Storage Locations of Containers at Seaport Terminals. Computers & Operations Research 28, pp.983-995. 2001.

[133] Ramani, K.V. An Interactive Simulation Model for the Logistics Planning of Container Operations in Seaports. Simulation 66 (5), pp.291-300. 1996.

[134] Rebollo, M., Julian, V., Carrascosa, C., and Botti, V. A MAS Approach for Port Container Terminal Management. In: Proceedings of the 3rd Iberoamerican workshop on DAI and MAS, 2000, pp.83-94. Atibaia, Sao Paulo, Brasil. 2000.

[135] Reveliotis, S.A. Conflict Resolution in AGV Systems. IIE Transactions 32, pp.647-659. 2000.

[136] Rizzoli, A.E., Gambardella, L.M., Zaffalon, M., and Mastrolilli, M. Simulation for the Evaluation of Optimised Operations Policies in a Container Terminal. In: HMS99, Maritime & Industrial Logistics Modeling and Simulation, Sep. 1999, Genoa, Italy.

[137] Roach, P.A. and Wilson, I.D. A Genetic Algorithm Approach to Strategic Stowage Planning for Container-Ships. Working paper, University of Glamorgan, UK. 2002.

[138] Saanen, Y.A. Examining the Potential for Adapting Simulation Software to Enable Short-Term Tactical Decision Making for Operational Optimisation. Technical Report, TBA Nederland/Delft University of Technology. 2000.

[139] Scheithauer, G. LP-Based Bounds for the Container and Multi-Container Loading Problem. International Transactions in Operational Research 6 (2), pp.199-213. 1999.

[140] Schneidereit, G., Voβ, S., Parra, A., and Steenken, D. A General Pickup and Delivery Problem for Automated Guided Vehicles with Multi-loads: A Case Study. Working paper, University of Hamburg. 2003.

[141] Sculli, D. and Hui, C.F. Three Dimensional Stacking of Containers. Omega 16 (6): pp.585-594. 1988.

[142] Sgouridis, S.P., Makris, D. and Angelides, D.C. Simulation Analysis for Midterm Yard Planning in Container Terminal. Journal of Waterway, Port, Coastal and Ocean Engineering, pp.178-187. 2003.

[143] Shabayek, A.A. and Yeung, W.W. A Simulation for the Kwai Chung Container Terminal in Hong Kong. European Journal of Operational Research 40, pp.1-11. 2002.

[144] Shen, W.S. and Khoong, C.M. A DSS for Empty Container Distribution Planning. Decision Support Systems 15, pp.75-82. 1995.

[145] Shields, J.J. Container Stowage: A Computer Aided Preplanning System. Marine Technology 21 (4), pp. 370-383. 1984.

[146] Steenken, D. Fahrwegoptimierung am Containerterminal under Echtzeitbedingungen. OR Spectrum 14 (3), pp.161-168. 1992.

[147] Steenken, D., Henning, A., Freigang, S., and Voβs, S. Routing of Straddle Carriers at a Container Terminal with the Special Aspect of Internal Moves. OR Spectrum 15 (3), pp.167-172. 1993.

[148] Steenken, D., Winter, T., and Zimmermann, U.T. Stowage and Transport Optimization in Ship Planning. In: Grötschel M, Krumke S O, Rambau J (eds) Online optimization of large scale systems, pp 731–745. Springer, Berlin. 2001.

[149] Steenken, D., Voβ, S., and Stahlbock, R. Container Terminal Operation and Operations Research - a Classification and Literature Review. OR Spectrum 26, 3-49, 2004.

[150] Taleb-Ibrahimi, M., De Castilho, B., and Daganzo C.F. Storage Space vs Handling Work in Container Terminals. Transportation Research B 27, pp.13-32. 1993.

[151] Thiers, G.F. and Janssens, G.K. A Port Simulation Model as a Permanent Decision Instrument. Simulation 71, pp.117-125. 1998.

[152] Van der Heijden, M., Ebben, M., Gademann, N., and van Harten, A. Scheduling Vehicles in Automated Transportation Systems: Algorithms and Case Study. OR Spectrum 24, pp.31-58. 2002.

[153] Van der Meer, R. Operational Control of Internal Transport, ERIM Ph.D. series Research in Management 1. 2000.

[154] Van Hee, K.M., Huitink, B., and Leegwater, D.K. Portplan, Decision Support System for Port Terminals. European Journal of Operational Research 34 (3), pp.249-261. 1988.

[155] Van Hee, K.M. and Wijbrands, R.J. Decision Support Systems for Container Terminal Planning. European Journal of Operational Research 34 (3), pp.262-272. 1988.

[156] Veeke, H.P.M. and Ottjes, J.A. Detailed Simulation of the Container Flows for the IPSI Concept. In the proceedings of the 11th European Simulation Symposium (ESS 1999), October 1999, Erlangen.

[157] Veeke, H.P.M. and Ottjes, J.A. A Generic Simulation Model for Systems of Container Terminals. In the proceedings of the 16th European Simulation Multiconference (ESM 2002), June 2002, Darmstadt, Germany.

[158] Vis, I.F.A. and Harika, I. Comparison of Vehicle Types at an Automated Container Terminal. OR Spectrum 26, pp.117-143. 2004.

[159] Vis, I.F.A. and de Koster, R. Transshipment of Containers at a Container Terminal: An Overview. European Journal of Operational Research 147, pp.1-16. 2003.

[160] Vis, I.F.A., Koster, R., de Roodbergen, K.J., and Peeters, L.W.P. Determination of the Number of Automated Guided Vehicles Required at a Semi-Automated Container Terminal. Journal of the Operational Research Society 52, pp.409-417. 2001.

[161] Wilson, I.D. and Roach, P.A. Principles of Combinatorial Optimization Applied to Container-Ship Stowage Planning. Journal of Heuristics 5, pp.403-418. 1999.

[162] Wilson, I.D. and Roach, P.A. Container Stowage Planning: A Methodology for Generating Computerized Solutions. Journal of the Operational Research Society 51, pp.1248-1255. 2000.

[163] Wilson, I.D., Roach, P.A., and Ware, J.A. Container Stowage Pre-planning: Using Search to Generate Solutions, a Case Study. Knowledge-Based Systems 14 (3-4): pp.137-145. 2001.

[164] Yang, C.H., Choi, Y.S., and Ha, T.Y. Performance Evaluation of Transport Vehicle at Automated Container Terminal Using Simulation. OR Spectrum 26 (2), pp. 149-170. 2004.

[165] Yun, W.Y. and Choi, Y.S. A Simulation Model for Container-Terminal Operation Analysis Using an Object-Oriented Approach. International Journal of Production Economics 59 (1-3), pp.221-230. 1999.

[166] Yun, W.Y. and Choi, Y.S. Simulator for Port Container Terminal Using an Object Oriented Approach. Working paper, Pusan National University. 2003.

[167] Zaffalon, M., Rizzoli, A.E., Gambardella, L.M., and Mastrolilli, M. Resource Allocation and Scheduling of Operations in an Intermodal Terminal. In the Proceedings of 10th European Simulation Symposium and Exhibition, Simulation in Industry, pp.520-528, Oct. 1998, Nottingham, United Kingdom.

[168] Zhang, C., Wan, Y., Liu, J., and Linn, R.J. Dynamic Crane Deployment in Container Storage Yards. Transportation Research-B 36, pp.537-555. 2002.

[169] Zhang, C., Liu, J., Wan, Y.W., Murty, K.G., and Linn, R.J. Storage Space Allocation in Container Terminals. Transportation Research B 37, pp.883-903. 2003.