SPACE-SHARING STRATEGIES FOR STORAGE YARD MANAGEMENT IN A TRANSSHIPMENT HUB PORT

JIANG XINJIA

NATIONAL UNIVERSITY OF SINGAPORE

2012

SPACE-SHARING STRATEGIES FOR STORAGE YARD MANAGEMENT IN A TRANSSHIPMENT HUB PORT

JIANG XINJIA

(B.Eng., Nanjing University)

A THESIS SUBMITTED

FOR THE DEGREE OF DOCTOR OF PHILOSOPHY

DEPARTMENT OF INDUSTRIAL AND SYSTEMS ENGINEERING

NATIONAL UNIVERSITY OF SINGAPORE

2012

Declaration

I hereby declare that the thesis is my original work and it has been written by me in its entirety. I have duly acknowledged all the sources of information which have been used in the thesis.

This thesis has also not been submitted for any degree in any university previously.

Jiang Xinjia

23 Sep 2012

Acknowledgement

In memory of my mother who guided me with her wisdom, passion and determination.

This thesis would never have been written without the support from the people who enriched my knowledge and experience in many ways.

I would like to express my deepest appreciation to my two supervisors, A/Prof. LEE Loo Hay and A/Prof. CHEW Ek Peng. They offered me invaluable guidance and encouragement through the whole course of my research. I would also like to thank A/Prof. TAN Kok Choon, who generously provided me with his expertise on port operations.

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 my peers and fellow students in the Department of Industrial and Systems Engineering. Thank you for sharing your experience on research and life in Singapore during the four years.

Last but not least, I would like to thank my family for their continuous support and understanding, especially my father. It was your encouragement that helped me stand up and survive from the tough period.

Jiang Xinjia

Table of Contents

Declaration i

Acknowledgement ii

Table of Contents iii

Summary vi

List of Tables viii

List of Figures ix

List of Abbreviations xii

List of Notations xiv

Chapter 1. Introduction 1

1.1 Background of container terminals 2

1.2 Storage yard management 4

1.3 Space allocation planning 6

1.4 Contribution of the thesis 8

1.5 Organization of the thesis 10

Chapter 2. Literature Review 12

2.1 The design of storage yard 14

2.1.1 Equipment selection 14

2.1.2 Yard layout and configuration 14

2.1.3 Decision support and simulation systems 15

2.2 The transport vehicle management 16

2.2.1 Fleet sizing problem 16

2.2.2 Vehicle dispatching problem 16

2.3 The yard crane management 18

2.3.1 Yard crane dispatching 18

2.3.2 Yard crane deployment 21

2.4 The storage space management 23
2.4.1 Location assignment 23
2.4.2 Space allocation 27
2.5 Research gaps and motivations 30
Chapter 3. The Partial Space-sharing Strategy 32
3.1 Problem description 32
3.2 Solution approaches 37
3.2.1 Template generation 39
3.2.2 Two different methods for space allocation and workload assignment 44
3.3 Numerical experiments 54
3.3.1 Experiment descriptions 54
3.3.2 Experiment results 55
3.4 Conclusions 58
Chapter 4. The Flexible Space-sharing Strategy 60
4.1 Problem description 60
4.2 Model formulation 62
4.3 Solution approaches 69
4.3.1 Decomposed model for one shift 71
4.3.2 Shift-picking methods 74
4.4 Numerical experiments 77
4.4.1 Performance of flexible space-sharing strategy 79
4.4.2 Comparison of shift-picking methods 80
4.4.3 Impact of the yard crane limit 82
4.4.4 Effect of the sub-block size 83
4.5 Conclusions 85
Chapter 5. Short-term Space Allocation 86
5.1 Problem description 86
5.2 Model formulation 89
5.2.1 MIP model for the GSA method 90
5.2.2 MIP model for the SALP method 97

5.3 Numerical experiments 102
5.3.1 Experiment description 102
5.3.2 Comparison of short-term planning methods 104
5.3.3 Comparison between the storage strategies 109
5.4 Conclusions 112
Chapter 6. Conclusions and Future Research 114
Bibliography 117
Appendices 134
Appendix A: Candidate's publication list arising from the PhD work 134

Summary

In transshipment ports, the containers to the same destination vessel are usually stored together to facilitate the loading process, which is called the "consignment". The "yard template" is used to define the reservation of the storage locations for destination vessels. However, the consignment strategy is known to be inefficient in space utilization since each storage location must be dedicated to a particular vessel. To improve the space utilization while retaining the advantage of consignment, new storage strategies are proposed in this thesis, namely the "partial space-sharing strategy" and the "flexible space-sharing strategy".

In the "partial space-sharing strategy", part of the storage space is allowed to be shared between two adjacent storage locations. The space in each storage location is divided into non-sharing and sharing parts. When less space is needed by a storage location, the sharing space in this storage location can be lent to the adjacent locations. The sharing space will then be returned, before the major workload comes into this storage location. Since the major containers to each destination vessel arrive at different periods, the storage locations preserved for the vessels will also need the sharing space during different shifts. An integrated framework is developed to decide the yard template and the container assignment at the same time. Two approaches are proposed to decide the size of sharing and non-sharing space in each storage location. Experimental results show that the partial space-sharing strategy is able to improve the land utilization, while guaranteeing the least yard crane deployment.

In the "flexible space-sharing strategy", the same storage location is allowed to be reserved for two vessels. The amount of space will only be allocated to a specific vessel on the arrival of corresponding containers. By controlling where to stack the

containers in the storage locations, the containers to each vessel are not mixed and the consignment feature can be preserved. This strategy is first formulated as a mixed integer program. As the MIP model has a block diagonal structure, we develop a search algorithm which combines MIP and heuristics to find the solution. The results show that the "flexible space-sharing strategy" can handle much more containers within the same storage space compared with the "non-sharing strategy".

In the previous studies, the storage strategies are all studied for long-term planning. During the operation, the actual containers that will come in are only known for a short period in advance. Thus, short-term space allocation is needed to assign the incoming containers taking into account of transport vehicles, yard cranes and space capacity. Currently, the space is allocated based on the experience of port operators and the rule of thumb. To remedy this, we develop two systematic short-term planning methods, namely the "greedy space allocation (GSA)" and the "space allocation considering the long-term plan (SALP)". MIP models are formulated for the two methods respectively. The numerical experiments show that the SALP method is preferred over the GSA method, but the portion of long-term plan considered affects the performance of the SALP method.

List of Tables

Table 4.1 Comparison of the two storage strategies..80

Table 5.1 Parameter setting for different storage strategies 94

List of Figures

Figure 1.1 Growth of container traffic 1

Figure 1.2 Components and operations in a typical container terminal 2

Figure 1.3 Import, export and transshipment 3

Figure 1.4 General picture of the storage yard 5

Figure 1.5 Container stacking in a block 5

Figure 2.1 The structure of literature review 13

Figure 3.1 A storage yard configuration 33

Figure 3.2 The buildup pattern of the coming workload for one vessel 33

Figure 3.3 Schematic diagram of one block for the static yard template and the space-sharing yard template 34

Figure 3.4 A schematic diagram for space capacity of one sub-block 35

Figure 3.5 A schematic diagram for the space capacity of each sub-block in one block 36

Figure 3.6 The framework for solution 38

Figure 3.7 Block configurations with common size = 1,2,3 slots respectively 45

Figure 3.8 Relative locations occupied by two sub-blocks without sharing 51

Figure 3.9 Relative locations occupied by two sub-blocks with sharing 51

Figure 3.10 Input data with “even pattern” (left) and “wave pattern” (right) 55

Figure 3.11 Results from “even pattern” of input data 56

Figure 3.12 Results from “wave pattern” of input data 56

Figure 4.1 Yard template for non-sharing strategy (up) and flexible space-sharing strategy (down) 61

Figure 4.2 An example of space occupation under flexible space-sharing strategy...61

Figure 4.3 The staying period of a container 64

Figure 4.4 Main search algorithm..70

Figure 4.5 Batch-list method..75

Figure 4.6 Greedy-tabu method 77

Figure 4.7 The example workload patterns for one vessel 78

Figure 4.8 Comparison of shift-picking methods 81

Figure 4.9 Impact of the yard crane limit 83

Figure 4.10 Effect of the sub-block size 84

Figure 5.1 An example of the container flow for a destination vessel 103

Figure 5.2 Traffic violation under different planning methods (Sufficient space with vessel delay)...105

Figure 5.3 Traffic violation under different planning methods (Tight space with no vessel delay)...106

Figure 5.4 Loading violation under different planning methods (Tight space with no vessel delay)...107

Figure 5.5 Traffic violation under different planning methods (Tight space with vessel delay)...107

Figure 5.6 Loading violation under different planning methods (Tight space with vessel delay)..108

Figure 5.7 Traffic violation under different storage strategies (Sufficient space with vessel delay)..109

Figure 5.8 Crane violation under different storage strategies (Tight space with no vessel delay)..110

Figure 5.9 Space violation under different storage strategies (Tight space with no vessel delay)..111

Figure 5.10 Crane violation under different storage strategies (Tight space with vessel delay)..111

Figure 5.11 Space violation under different storage strategies (Tight space with vessel delay)..111

List of Abbreviations

AGV:	Automated Guided Vehicle
ALV:	Automated Lifting Vehicle
ASC:	Automated Stacking Crane
CAP:	Container Assignment Problem
DCAP:	Decomposed Container Assignment Problem
FEU:	Forty-foot Equivalent Unit
GA:	Genetic Algorithm
GSA:	Greedy Space Allocation
IYT:	Initial Yard Template
KPI:	Key Performance Index
LB:	Lower Bound
MBS:	Minimal Block Space
MIP:	Mixed Integer Programming
OBC:	Overhead Bridge Crane
PM:	Prime Mover
QC:	Quay Crane
RMG:	Rail Mounted Gantry
RP:	Random Pick
RTG:	Rubber Tyred Gantry
SA:	Simulated annealing

SALP: Space Allocation Considering the Long-term Plan

SC: Straddle Carrier

TEU: Twenty-foot Equivalent Unit

TS: Time Sequence

WAP: Workload Assignment Problem

YC: Yard Crane

YTG: Yard Template Generation

List of Notations

A_j The smallest number of sub-blocks that should be reserved for vessel 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 time, which may vary according to the condition of different blocks, $1 \le k \le K$.

C_t The "yard crane limit" is total number of yard cranes allowed for discharging jobs during shift t across the whole yard template, $1 \le t \le T$.

C_{kg}^{t} The remaining loading capacity of container group g in block k excluding space allocation in shift t, $1 \le t \le T$, $1 \le k \le K$, $1 \le g \le G$.

CC The capacity of each yard crane in terms of container moves per shift, which is 100 in this model.

CLS_i The left space capacity of non-sharing space for sub-block i, $1 \le i \le I$.

$CLS_{ii'}$ The left space capacity of sharing space between sub-block i and i', $1 \le i \le I$, $1 \le i' \le I$.

CLC_i The left yard crane capacity of sub-block i for loading process, $1 \le i \le I$.

CS The original space capacity of each sub-block in terms of TEUs, which is 240 (5 tiers×6 lanes×8 slots) in this thesis.

d_k The number of yard cranes allocated to block k for discharging containers in the current shift, $1 \le k \le K$.

d_{kt} The number of yard cranes allocated to block k in shift t for discharging containers, $1 \le k \le K$, $1 \le t \le T$.

D^d The total number of yard cranes allowed for discharging containers during the current shift.

D_k^l The number of yard cranes deployed for loading in block k during the current shift, $1 \le k \le K$.

E_j The maximum number of shifts allowed for loading the containers to vessel j, $1 \le j \le J$.

F_i The shift when sub-block i finishes its loading process, $1 \le i \le sub$.

$F_{jtt'}$ A container is staying in the storage yard or not during shift t'. The value is decided according to the discharging shift t of the container and the loading completion shift of its destination vessel j.

= 1, if the container is staying during shift t', $1 \le j \le J$, $1 \le t \le T$, $1 \le t' \le T$.

= 0, otherwise, $1 \le j \le J$, $1 \le t \le T$, $1 \le t' \le T$.

G Total number of container groups in one block, which is 6 in this thesis.

h_{it} = 1, if the total workload allocated to sub-block i for unloading in shift t is high, that is, $HL \le x_{it}+y_{it} \le HU$, $1 \le i \le I$, $1 \le t \le T$.

= 0, if the total workload allocated to sub-block i for unloading in shift t is low, that is, $LL \le x_{it}+y_{it} \le LU$, $1 \le i \le I$, $1 \le t \le T$.

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.

K The number of blocks under consideration.

LB_t The least number of yard cranes that can be deployed for unloading in shift t, $1 \le t \le T$.

L_j = 1, if vessel j is in loading procedure during the current shift, $1 \le j \le J$.

= 0, otherwise, $1 \le j \le J$.

L_{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, $1 \le j \le J$, $1 \le t \le T$.

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.

M_k The maximum number of yard cranes allowed to deploy in block k at during any shift, $1 \le k \le K$.

N_i The set of sub-blocks that are neighbors of sub-block i, $1 \le i \le I$.

NB_i The set of sub-blocks those are adjacent to sub-block i, where space sharing is possible between two sub-blocks, $1 \le i \le I$.

NL_{kt} Number of yard cranes deployed for loading jobs in block k during shift t, which is given by the yard template, $1 \le k \le K$, $1 \le t \le T$.

$R_{it'}^{t}$ The remaining space capacity of sub-block i in shift t' excluding space allocation in shift t, $1 \le i \le I$, $1 \le t \le T$, $1 \le t' \le T$.

R_{i0}^{T1} The amount of 20ft containers stored in the left corner of sub-block i at the beginning of the current shift, $1 \le i \le I$.

R_{i0}^{T2} The amount of 20ft containers stored in the right corner of sub-block i at the beginning of the current shift, $1 \le i \le I$.

R_{i0}^{F1} The amount of 40ft containers stored in the left corner of sub-block i at the beginning of the current shift, $1 \le i \le I$.

R_{i0}^{F2} The amount of 40ft containers stored in the right corner of sub-block i at the beginning of the current shift, $1 \le i \le I$.

r_i^{T1} The number of TEUs in the left corner of sub-block i at the end of the current shift. $1 \le i \le I$.

r_i^{T2} The number of TEUs in the right corner of sub-block i at the end of the current shift. $1 \le i \le I$.

r_i^{F1} The number of FEUs in the left corner of sub-block i at the end of the current shift. $1 \le i \le I$.

r_i^{F2} The number of FEUs in the right corner of sub-block i at the end of the current shift. $1 \le i \le I$.

r_{it}^{T1} The remaining TEUs in the left corner of sub-block i at the end of shift t. $1 \le i \le I$, $1 \le t \le T$.

r_{it}^{T2} The remaining TEUs in the right corner of sub-block i at the end of shift t. $1 \le i \le I$, $1 \le t \le T$.

r_{it}^{F1} The remaining FEUs in the left corner of sub-block i at the end of shift t. $1 \le i \le I$, $1 \le t \le T$.

r_{it}^{F2} The remaining FEUs in the right corner of sub-block i at the end of shift t. $1 \le i \le I$, $1 \le t \le T$.

s The common size of sharing space between each pair of adjacent sub-blocks.

s_i The space that belongs to sub-block i and cannot be shared with its neighbors, $1 \le i \le I$.

$s_{ii'}$ The space that can be shared between sub-blocks i and i', $1 \le i \le I$, $1 \le i' \le I$.

S_{ti} The set of shifts from the end of the loading time of sub-block i to the current shift t, $1 \le i \le I$, $1 \le t \le T$.

S_{it}^{L} The lower boundary of the slot range occupied by sub-block i in shift t, $1 \le i \le sub$, $1 \le t \le T$.

S_{it}^{U} The higher boundary of the slot range occupied by sub-block i in shift t, $1 \le i \le sub$, $1 \le t \le T$.

S_{it}^{40} The pair of adjacent slots of sub-block i needed for 40ft containers in shift t, $1 \le i \le sub$, $1 \le t \le T$.

$S_{jtt'}$ The parameter indicates whether a container, which is discharged in shift t and to be loaded onto vessel j, will still be present in the yard in shift t'. The value is decided according to the discharging shift t of the container and the loading completion shift of its destination vessel j.

= 1, if the container will stay during shift t', $1 \le j \le J$, $1 \le t \le T$, $1 \le t' \le T$.

= 0, otherwise, $1 \le j \le J$, $1 \le t \le T$, $1 \le t' \le T$.

sub The number of sub-blocks in each block.

T The number of shifts under consideration in the planning horizon.

U The workload limit for two neighboring sub-blocks, which is 100 in this model.

u_{ijt} A variable used to indicate the value of ($v_{it} \times z_{ij}$) in linear form, $1 \le i \le I$, $1 \le j \le J$, $1 \le t \le T$.

u_{it} The violation of space capacity for sub-block i in shift t, $1 \le i \le I$, $1 \le t \le T$.

V The lower bound for total number of available sub-blocks in all shifts.

v_{it} = 1, if sub-block i is available during shift t, $1 \le i \le I$, $1 \le t \le T$.

= 0, otherwise, $1 \le i \le I$, $1 \le t \le T$.

v_{kg} The violation of loading capacity for container group g in block k, $1 \le k \le K$, $1 \le g \le G$.

$v^1_{ii'}$ The traffic violation between two neighboring sub-blocks i and i' in the current shift. $1 \le i \le I$, $1 \le i' \le I$.

v^2 The crane violation in the current shift.

v^3_{kg} The loading violation for container group g in block k in the current shift. $1 \le k \le K$, $1 \le g \le G$.

v^4_i The space violation in sub-block i in the current shift. $1 \le i \le I$.

v^3_{kgt} The loading violation for container group g in block k during shift t. $1 \le k \le K$, $1 \le g \le G$, $1 \le t \le T$.

v^4_{it} The space violation in sub-block i during shift t. $1 \le i \le I$, $1 \le t \le T$.

$W_{ii't}$ = 1, if the sharing space between sub-blocks i and i' ($s_{ii'}$) belongs to part of capacity of sub-block i in shift t. $W_{ii't}$ can be obtained from the loading time of sub-blocks i and i', $1 \le i \le I$, $1 \le i' \le I$, $1 \le t \le T$.

= 0, if the sharing space between Sub-blocks i and i' ($s_{ii'}$) belongs to part of capacity of sub-block i' in Shift t, $1 \le i \le I$, $1 \le i' \le I$, $1 \le t \le T$.

W_t The parameter indicates whether shift t from the long-term plan is considered.

= 1, if shift t is considered, $2 \le t \le T$.

= 0, otherwise, $2 \le t \le T$.

WL_i The amount of loading containers in sub-block i during the current shift. $1 \le i \le I$.

WX_j The number of 20ft containers arriving at the terminal in the current shift and will be loaded onto vessel j. $1 \le j \le J$.

WY_j The number of 40ft containers arriving at the terminal in the current shift and will be loaded onto vessel j, $1 \le j \le J$.

WX_{jt} The number of 20ft 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 40ft containers arriving at the terminal in shift t and will be loaded onto vessel j finally. It is given and input to the model, $1 \le j \le J$, $1 \le t \le T$.

x_{it} The number of 20ft containers that are allocated to sub-block i for unloading in shift t, $1 \le i \le I$, $1 \le t \le T$.

x_i^1 The number of 20ft containers that are assigned to the left corner of sub-block i in the current shift, $1 \le i \le I$.

x_i^2 The number of 20ft containers that are assigned to the right corner of sub-block i in the current shift, $1 \le i \le I$.

x_{it}^1 The number of 20ft containers that are assigned to the left corner of sub-block i in shift t, $1 \le i \le I$, $1 \le t \le T$.

x_{it}^2 The number of 20ft containers that are assigned to the right corner of sub-block i in shift t, $1 \le i \le I$, $1 \le t \le T$.

X_{it}^1 The parameter indicates the given long-term space allocation plan. The amount of 20ft containers that come into the left corner of sub-block i in shift t. $1 \le i \le I$, $2 \le t \le T$.

X_{it}^2 The parameter indicates the given long-term space allocation plan. The amount of 20ft containers that come into the right corner of sub-block i in shift t. $1 \le i \le I$, $2 \le t \le T$.

y_{it} The number of 40ft containers that are allocated to sub-block i for unloading in shift t, $1 \le i \le I$, $1 \le t \le T$.

y_i^1 The number of 40ft containers that are assigned to the left corner of sub-block i in the current shift, $1 \le i \le I$.

y_i^2 The number of 40ft containers that are assigned to the right corner of sub-block i in the current shift, $1 \le i \le I$.

y_{it}^1 The number of 40ft containers that are assigned to the left corner of sub-block i in shift t, $1 \le i \le I$, $1 \le t \le T$.

y_{it}^2 The number of 40ft containers that are assigned to the right corner of sub-block i in shift t, $1 \le i \le I$, $1 \le t \le T$.

Y_{it}^1 The parameter indicates the given long-term space allocation plan. The amount of 40ft containers that come into the left corner of sub-block i in shift t. $1 \le i \le I$, $2 \le t \le T$.

Y_{it}^{2} The parameter indicates the given long-term space allocation plan. The amount of 40ft containers that come into the right corner of sub-block i in shift t. $1 \le i \le I$, $2 \le t \le T$.

z_{ij} = 1, if sub-block i is reserved for vessel j, $1 \le i \le I$, $1 \le j \le J$.

= 0, otherwise, $1 \le i \le I$, $1 \le j \le J$.

Z_{ij} = z_{ij}, indicator of sub-block reservation for different vessels, adopting the value solved in template generation step, $1 \le i \le I$, $1 \le j \le J$.

Z_{ij}^{1} The parameter indicates the reservation of the left corner of a sub-block.

= 1, if the containers to vessel j fill from the left corner of sub-block i, $1 \le i \le I$, $1 \le j \le J$.

= 0, otherwise, $1 \le i \le I$, $1 \le j \le J$.

Z_{ij}^{2} The parameter indicates the reservation of the right corner of a sub-block.

= 1, if the containers to vessel j fill from the right corner of sub-block i, $1 \le i \le I$, $1 \le j \le J$.

= 0, otherwise, $1 \le i \le I$, $1 \le j \le J$.

Chapter 1. Introduction

In 1955, former trucking company owner Malcom McLean and engineer Keith Tantlinger developed a simple yet brilliant idea which initiated the modern freight transportation. They proposed to put the cargoes in steel containers as transportation units, instead of packing and unpacking every time when the cargoes were transferred from one transportation mode to another. The steel containers not only provide protections against theft, weather and pilferage, but also greatly improve the efficiency of cargo transportation and reduce the logistic cost. The introduction of the standardized steel containers, transportation facilities and handling equipment, has further achieved a worldwide acceptance of container transportation (Levison, 2006).

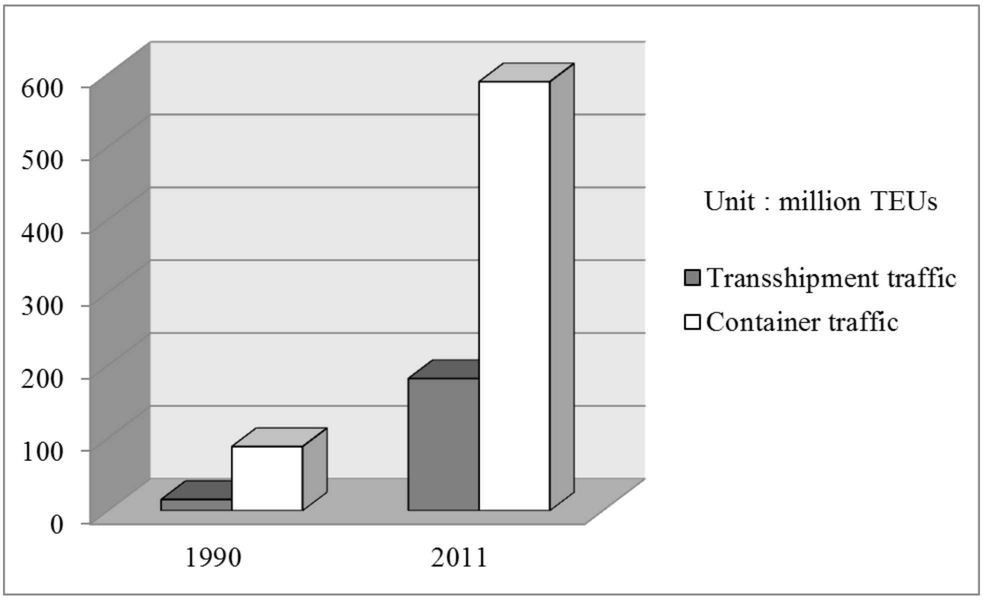

Figure 1.1 Growth of container traffic

In 2004, over 60% of the world's general sea cargos were transported in containers, while some routes between economically strong countries were 100% containerized (Steenken et al., 2004). The growth of container traffic during the last two decades can be shown in Figure 1.1. According to Drewry Shipping Consultants, the annual container traffic has increased around 6.7 times from 88.150 million TEUs (twenty-foot equivalent unit) in 1990 to 588.905 million TEUs in 2011. The growth of annual transshipment traffic (vessel to vessel container transportation) is even faster, at 11.7 times, from 15.504 million TEUs in 1990 to 181.596 million TEUs in 2011. The container terminals have played an important role in the growth of container traffic. However, the increasing container traffic and the use of mega container vessels like "Emma Maersk" also pose challenge for container terminals to provide efficient services.

1.1 Background of container terminals

To deliver the cargoes to the final destination, the containers usually need to be transported via different transportation modes, such as vessels, trains and trucks. The container terminals serve as the crucial interface between these transportation modes. The containers discharged from one transportation mode can also be temporarily stored in the port terminals, before being transported via the next one. A typical container terminal consists of three parts, namely the quayside, the landside and the storage yard, as shown in Figure 1.2.

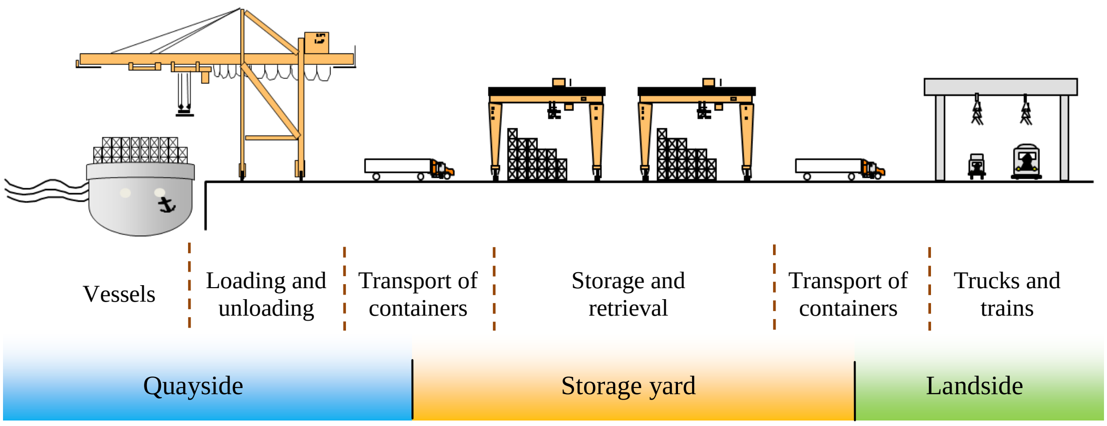

Figure 1.2 Components and operations in a typical container terminal

The quayside of a terminal offers berthing places for vessels, where the containers are loaded and unloaded by quay cranes. At the landside, the containers are received from or delivered to the local customers with trucks and trains. The main territory of a container terminal is used as the storage yard, where the containers can be temporarily stored. The storage and retrieval of containers can be performed by yard cranes, such as RTGs (rubber tyred gantry), RMGs (rail mounted gantry) and OBCs (overhead bridge crane). The containers are transferred between the quayside and the storage yard by transport vehicles, such as AGVs (automated guided vehicle), ALVs (automated lifting vehicle), SCs (straddle carrier) and PMs (prime mover). The same equipment can be used to transport containers between the storage yard and the landside, except AGVs which are preferred only to serve the quayside.

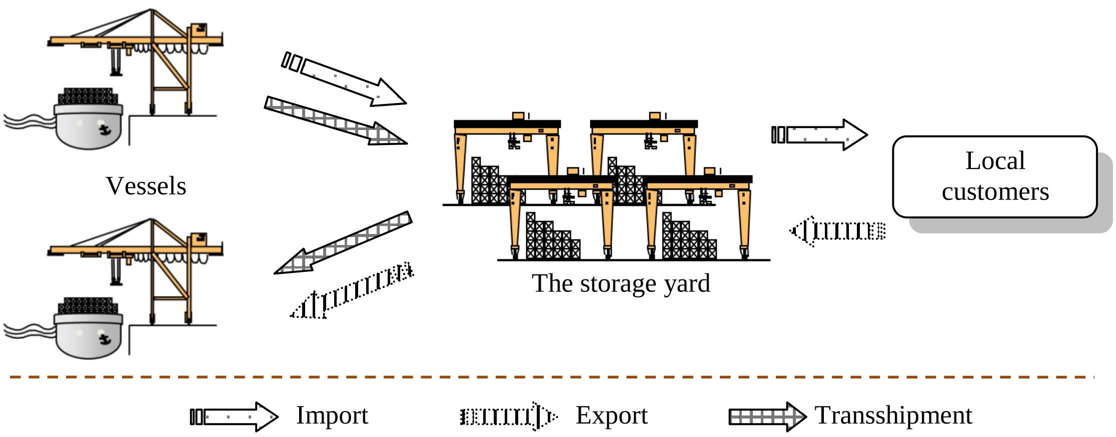

Figure 1.3 Import, export and transshipment

The containers handled in the port terminals can be categorized into three types, namely “import”, “export” and “transshipment”. For import, the containers arrive in large batches which are brought in by vessels. They will be temporarily stored in the yard until being retrieved by individual local customers. For export, the containers are brought in by the local customers and accumulated in the storage yard. When their destination vessel arrives, the export containers will be loaded together onto the vessel. For transshipment containers, the process is a little different. The containers will be temporarily stored in the yard after being brought in by a vessel. Instead of being retrieved by local customers, they will be eventually loaded onto other vessels and transported to their next destinations. The container terminal that initiated this study is a transshipment hub port, where around 85% of the container handling activities are transshipment.

There are many key performance indexes (KPI) which are currently used to measure the performance of a container terminal. A crucial KPI is the "vessel turnaround time", which is defined as the time spent by a vessel at a terminal for loading and discharging of containers. On the perspective of shipping liners, if a vessel spends less time at a terminal, it will have more time to deliver cargoes which in turn earns more profit. On the perspective of port operators, shorter vessel turnaround time can increase the number of vessels served per day and also attract more customers for the efficient service. Thus, this KPI is especially important for a transshipment hub port which earns its profit through providing transshipment service. To reduce the vessel turnaround time, both practitioners and researchers have focused much attention on quay-side operations to improve the loading and discharging of containers. However, the overall terminal performance will not benefit much from faster quay-side operations without the effective storage and retrieval of containers in the storage yard.

1.2 Storage yard management

The storage yard management mainly considers three kinds of yard resources, namely transport vehicles, yard cranes and the storage space. Due to the limited land, containers in the yard are usually stacked in multi-level blocks. The whole storage yard is managed as many blocks, which can be shown in Figure 1.4. In the port we study, the storage blocks are parallel to the quay side. The prime movers are used to transport the containers from and to the storage blocks. The access points for the prime movers are by the side of each block. The container stacking at each storage block is performed by RTGs.

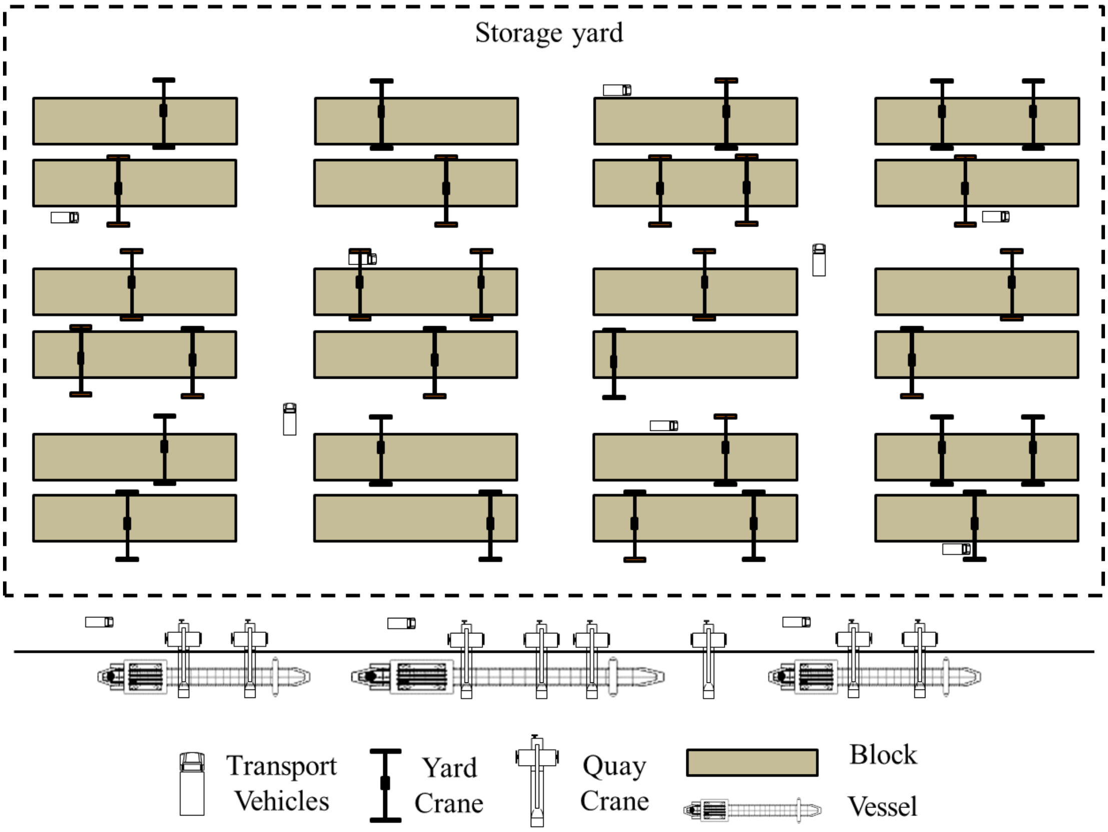

Figure 1.4 General picture of the storage yard

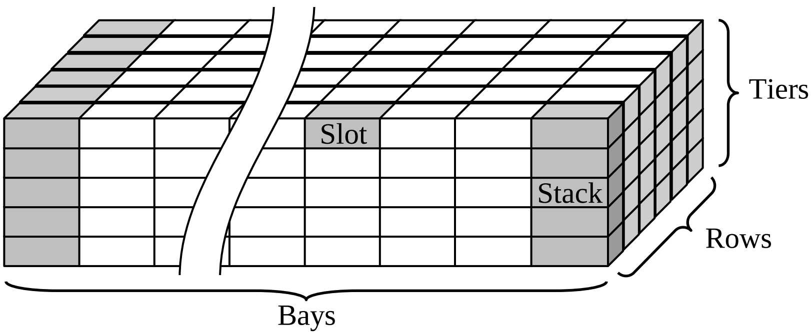

Figure 1.5 Container stacking in a block

Within each block, the containers are stacked on top of another by the yard cranes. As shown in Figure 1.5, a typical block can be described in three dimensions, namely “bay”, “row” and “tier”. The configuration of a block depends on the yard cranes used for container stacking. The basic unit of the storage space is “slot”, which can fit one TEU (20-foot equivalent unit). Several containers stacked on top of each other form a “stack”. When the container stacking is high, there are two major concerns in storage yard management, namely “reshuffles” and “congestions”. Firstly, “reshuffles” are the extra moves to reposition the containers on top of a requested one. As reshuffles increase the handling time, they need to be reduced during the operations. Secondly, the amount of transportation activities per acre increases when containers are stacked higher. Thus systematic planning is required for transportation activities to reduce the chance of congestions.

As a transshipment hub port, the storage yard management has its special characteristics compared with a gateway port. For a gateway port, the loading and discharging activities can be considered independently by storing the export and import containers in separate areas. For transshipment hubs, the loading and discharging activities are both in large batches and happen simultaneously within the same storage yard. This makes the storage yard management much more challenging. As the containers are usually stored in the same storage location until being loaded, the storage location of the discharged containers will determine the location for the loading containers. Thus, it is important to plan properly so that efficiency can be improved. In this thesis, we address research problems arising from a leading transshipment hub port. The proposed strategies and planning methods can be applied to other transshipment ports worldwide.

1.3 Space allocation planning

The main purpose of storage yard management is to decide where to store the incoming containers and how to deploy the yard cranes and prime movers to handle the containers. Once the space is allocated to the incoming containers, they will be transported to the corresponding storage locations by the prime movers and stacked by the yard cranes. Thus, the space allocation to incoming containers determines the workload for prime movers and yard cranes at each storage location. The efficiency of storage yard management depends greatly on the space allocation to incoming containers (Lee et al, 2006).

To avoid double handling, the incoming containers are usually stored at the same storage location until being retrieved. The loading activity at each storage location is just a result of the space allocation to incoming containers. Thus, the space allocation plan not only needs to consider the discharging of incoming containers, but also has to take into account the loading activities.

The planning of space allocation depends on how the containers are discharged and loaded at the quay side. The shipping liners usually operate several vessels to perform a service which calls at the port of rotation with fixed schedule. In this way, the service will call at the container terminal periodically, regardless of which vessel performs the service. In the port we study, each service usually calls at the terminal once a week. As a result, the space allocation is also planned on a weekly basis.

Generally, the "consignment strategy" is used in the yard for a transshipment port, where containers to the same destination vessel are stored together. This is to facilitate faster loading process as it reduces reshuffles as well as long distance movements of yard cranes. To achieve this strategy, a block is further managed as smaller storage locations, which usually consist of several consecutive bays. Each storage location is dedicated for incoming containers to a particular destination vessel. The "yard template" is used to define the reservation of storage locations for the destination vessels. However, the consignment strategy is known to be inefficient in space utilization. The main reason is that each storage location is dedicated to a particular vessel in advance. Since the majority of containers will only arrive near to their loading time (Han et al. 2008), much storage space is only occupied for a short period of time. With the rapid growth of the container traffic, more and more containers need to be handled and temporarily stored in the yard. New long-term storage strategies are required to improve the space utilization while retaining the advantage of consignment.

On the other hand, the current storage strategies are all studied for long-term planning. During the operation, loading and unloading times can change because of vessel delays. The amount of containers discharged from each vessel is also different from week to week. The information of incoming containers is only known for a short period in advance. Thus, short-term space allocation is needed to assign the incoming containers based on the latest information. Currently, the space is allocated based on the experience of port operators and the rule of thumb, which do not fully consider the general picture of storage yard management. To remedy this, systematic short-term planning methods are needed, which can take into account of transport vehicles, yard cranes and storage space.

1.4 Contribution of the thesis

The main contribution of this thesis can be listed as follows.

- An innovative "partial space-sharing strategy" is proposed to improve the space utilization while retaining the advantage of consignment. Instead of dedicating the storage location to a particular vessel, part of the storage space is allowed to be shared between two adjacent storage locations. The space in each storage location is divided into non-sharing and sharing parts. When less space is needed by a storage location, the sharing space in this storage location can be lent to the adjacent locations. The sharing space will then be returned from its neighbors, before the major workload comes into this storage location.

- A framework which integrates yard template and space allocation is proposed to implement the "partial space-sharing strategy". As the original MIP model is unable to be solved for a real scale problem, the decomposition method is used to separate the yard template and space allocation into two sub-problems. These two sub-problems are solved iteratively with the framework to decide the yard template and space allocation at the same time. Two different approaches are also incorporated in the framework to decide the size of sharing and non-sharing space.
- Based on the findings from the "partial space-sharing strategy", a more advanced storage strategy is proposed, namely the "flexible space-sharing strategy". In this strategy, there is no prefixed space boundary between the storage space reserved for two destination vessels. The space allocation can be self-balanced with the amount of incoming containers to each vessel. This strategy allows the same storage location to be reserved for two vessels. The amount of space will only be allocated to a specific vessel on the arrival of corresponding containers. By controlling where to stack the containers in the storage locations, the containers to each vessel are not mixed and the consignment feature can be preserved.
- Two systematic planning methods are developed to formalize the short-term space allocation. During the operation, the actual containers that will come in are only known for a short period in advance. Currently, the space is allocated based on the experience of port operators and the rule of thumb, which do not fully consider the overall picture of yard operations. In this thesis, the short-

term space allocation problem is formalized to take into account the yard cranes, prime movers and the storage space. Systematic planning methods are developed to improve the short-term operation efficiency as well as long-term impact through considering the long-term space allocation plan.

1.5 Organization of the thesis

This thesis consists of six chapters, which are organized as follows.

Chapter 2 reviews the studies dealing with the storage yard management. The related studies can be categorized into “the design of the storage yard” and “the management of yard resources”. There are mainly three kinds of yard resources under concern, including transport vehicles, yard cranes and the storage space. The management of yard resources will be reviewed in respective sections.

In Chapter 3, the “partial space-sharing strategy” is proposed. In this strategy, part of the storage space is allowed to be shared between two adjacent storage locations. An integrated framework is developed to decide the yard template and the container assignment at the same time. Two approaches are proposed to decide the size of sharing and non-sharing space in each storage location.

In Chapter 4, a more advanced concept is proposed, named the “flexible space-sharing strategy”. The idea is that the container space can be shared by two different vessels as long as their containers do not occupy the space at the same time. This strategy is first formulated as a mixed integer program. As the MIP model has a block diagonal structure, we develop a search algorithm which combines MIP and heuristics to find the solution.

Chapter 5 addresses the short-term space allocation problem. Two systematic short-term planning methods are developed, namely the “greedy space allocation (GSA)” and the “space allocation considering the long-term plan (SALP)”. MIP models are formulated for both short-term planning methods respectively.

In Chapter 6, the important findings in previous chapters are concluded. The limitations of the current studies and future research directions are also discussed.

Chapter 2. Literature Review

A container terminal is an integrated logistic system that offers the container handling and temporary storage between different transportation modes, such as vessels, trucks and trains. Various operations are performed in a container terminal, while all the operations are interrelated to some extent. Steenken et al. (2004) make a comprehensive review of the research works related to various port operations. According to their study, the terminal logistic and optimization studies can be categorized into the ship planning, the storage and stacking, and the transportation. The most discussed topic among the three is the ship planning, which includes major quayside operations, such as berth allocation, the stowage planning and the quay crane split. This is because the vessel turnaround time is a crucial performance measure for the container terminals worldwide. Both researchers and practitioners have focused much attention on improving quayside efficiency to shorten the vessel turnaround time. However, the overall terminal productivity will not benefit much from faster quay-side operations without the effective storage and retrieval of containers. The importance of storage yard management has also been highlighted in Vis et al. (2003), Günther and Kim (2006), Stahlbock and Voβ (2008), and Ku et al. (2010).

According to the types of operations and the detailed level of decisions, the studies related to the storage yard management can be further categorized as shown in Figure 2.1. The studies addressing the design stage include the equipment selection, the yard layout planning and the decision support systems. After the construction of the storage yard, the management problems can be classified according to the yard resources, which include the transport vehicles, the yard cranes and the storage space.

Due to the interactions among the yard resources, they are combined in some studies with different focus. In this chapter, a detailed literature review is presented according to this structure. The different focus and methods are shown for each category. Since our study is focused on the storage space management, the most related studies will be discussed in more details, while those related to other categories will be brief or only provided with the references.

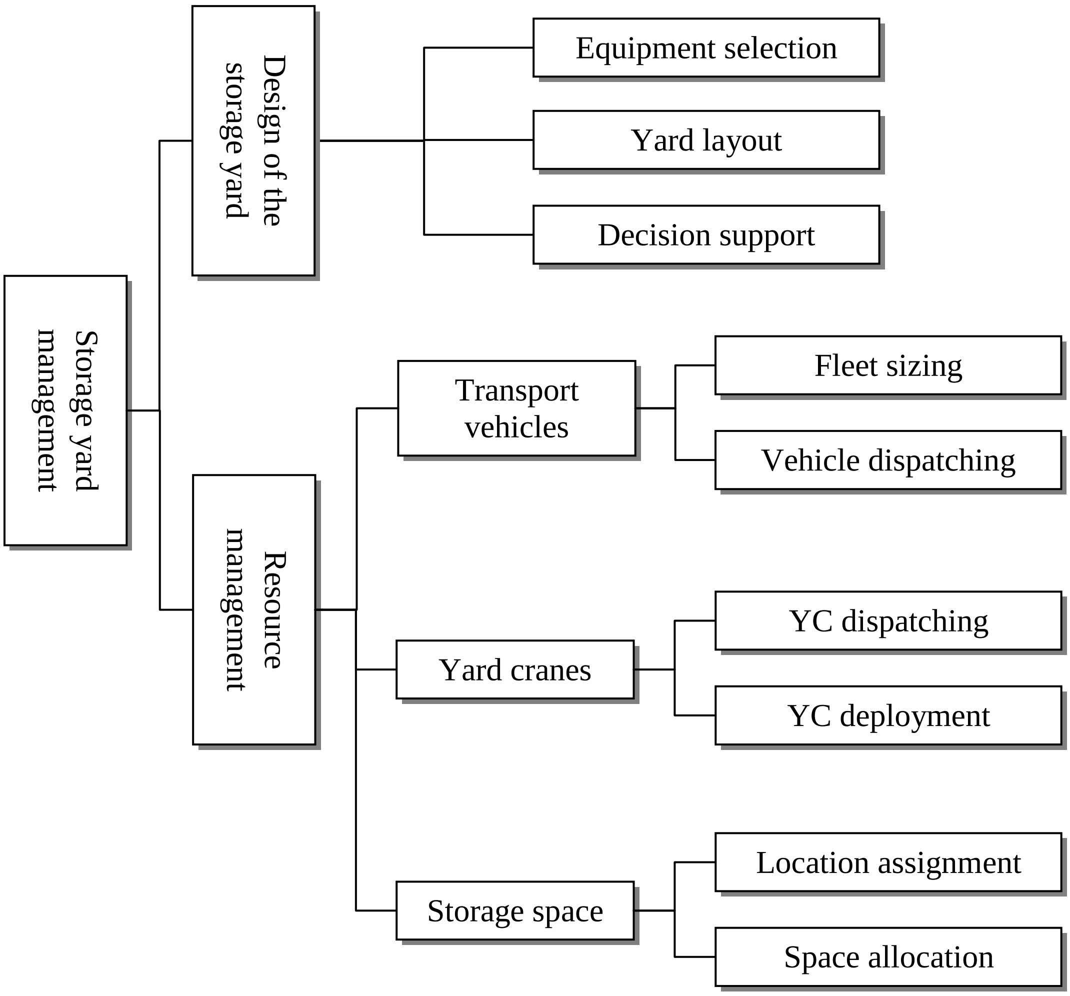

Figure 2.1 The structure of literature review

2.1 The design of storage yard

Since all the port operations are interrelated to some extent, the design of the storage yard is usually considered as a sub-system of the whole container terminal. The decisions related to the storage yard are shown as follows.

2.1.1 Equipment selection

The storage yard can be characterized by the combination of the equipment for container stacking and transportation. Liu et al. (2002) discuss four different design concepts for automated container terminals, where each concept has a unique combination of the yard equipment. Some studies are addressed especially for the comparison of stacking facilities or transport vehicles. The comparison of different SC, RTG and RMG systems are discussed in Chu et al. (2005), Saanen et al. (2005), and Vis (2006). The comparison and selection of transport vehicles are discussed in Baker (1998), Asef-Vaziri et al. (2003a, 2003b), Vis and Harika (2004), Yang et al. (2004) and Duinkerken et al. (2006).

2.1.2 Yard layout and configuration

The yard layout is another important aspect at the design stage of the storage yard. On one hand, the equipment selection will affect the yard layout. On the other hand, a better yard layout can further improve the performance of the same equipment.

Since the storage yard is managed as multi-level blocks, the major differences of the yard layouts are the direction of the blocks and the access points of each block. The direction of the blocks can be either parallel or perpendicular to the shore, which will affect the arrangement of travel lanes. The access points can be either at the end of a block or by the side of a block, which will affect the interaction between the yard cranes and transport vehicles. These two aspects have been discussed in Liu et al. (2004), Lee et al. (2007), Kim et al. (2008), Lee et al. (2008), and Park et al. (2010).

Moreover, a good yard layout design requires the optimal dimension of each block, including the number of bays, rows and layers. This is discussed in Murty (2007), Petering (2009), Petering and Murty (2009), and Lee and Kim (2010).

2.1.3 Decision support and simulation systems

Due to the rapid development of computer technology, various decision support and simulation systems have been developed. They can not only simulate and compare the alternative designs, but also provide references for the key design parameters. Many studies have been addressed in this category, including Kozan (1997), Gambardella et al. (1998), Nam et al. (2001), Murty et al. (2005 a, b), Parola and Sciomachen (2005), Bielli et al. (2006), Ottjes et al. (2006), Alessandri et al. (2007), Petering et al. (2009), Petering (2011), and Sun et al. (2012).

2.2 The transport vehicle management

The transport vehicles connect the storage yard with the quayside and landside of the container terminals. In order to guarantee the overall performance of the terminal, the vehicles shall transfer the containers efficiently and prevent the quay cranes and yard cranes from waiting. The management of the transport vehicles can be divided into two levels, namely the fleet sizing problem and the vehicle dispatching problem.

2.2.1 Fleet sizing problem

To guarantee the efficiency of container transportation within the terminal, enough transport vehicles shall be deployed to perform the jobs. On the other hand, the number of transport vehicles shall be minimized to control the operational cost. The fleet sizing problem is studied in Steenken (1992), Vis et al. (2001), Koo et al. (2004), Vis et al. (2005), and Kang et al (2008).

2.2.2 Vehicle dispatching problem

The vehicle dispatching problem is to determine the job sequence of each transport vehicle. This includes the assignment of transportation jobs to the vehicles and the travelling route of each vehicle. The vehicle dispatching problems can be studied as a general routing problem, without considering the difference between the kinds of vehicles. This is discussed in Bish et al. (2001), Narasimhan and Palekar (2002), Li and Vairaktarakis (2004), and Bish et al. (2005).

Due to the difference of transport vehicles, the container handling process can be different. The straddle carrier is one type of the alternative vehicles. It can not only transport the containers, but also pick up the containers without the help of yard cranes. This eliminates the handshakes in container handling, resulting in a different vehicle dispatching problem. The dispatching problem of straddle carriers is studied in Steenken et al. (1993), Kim and Kim (1999a, 1999b), Böse et al. (2000), and Vis et al. (2009).

Nowadays, the container terminal automation is a developing trend to improve the overall performance, while the AGVs are the most preferred choice in automated designs. Therefore, many studies can be found on the dispatching of AGVs, such as Evers and Koppers (1996), Chen et al. (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), Chan (2001), Leong (2001), van der Heijden et al. (2002), Lim et al. (2003), Moorthy et al. (2003), Schneidereit (2003), Grunow et al. (2004), Kim and Bae (2004), Nishimura et al. (2005), Briskorn et al. (2006), Lehmann et al. (2006), and Grunow et al. (2006).

The vehicle dispatching problem can also be combined with the location assignment problems to reduce the container transportation time. Kozan and Preston (1999) study the container transfer schedule with given container locations. The container location model is developed in Preston and Kozan (2001), where the container transfer schedule is given as a fixed input. Kozan and Preston (2006) present an iterative search algorithm which integrates the previous two models to optimize the storage locations and the transfer schedule simultaneously. More details will be discussed in the section on the storage space management.

2.3 The yard crane management

The yard cranes are the crucial equipment in the storage yard to perform the storage and retrieval of containers. The yard crane management can be divided into two levels, namely the yard crane dispatching and the yard crane deployment. For the yard crane dispatching problem, the main purpose is to decide the route of the yard cranes within a block. For the yard crane deployment problem, the main concern is the number of yard cranes to be deployed in each block and how to shift the yard cranes among the blocks.

2.3.1 Yard crane dispatching

When there is only one yard crane under consideration, the dispatching problem is to decide the number of containers to pick up in each bay as well as the sequence of the yard bays that the yard crane visits. Kim and Kim (1999c) consider the dispatching of a single yard crane, with a given load plan and a given bay plan for export containers. The number of containers to pick up is formulated as a transportation problem, while the visiting route is determined with a dynamic programming procedure. Narasimhan and Palekar (2002) analyze the generalized problem of scheduling yard cranes to pick up the containers. They do a theoretical investigation of the structural properties of the problem. Based on the results of the investigation, the problem is proved to be NP-Complete and formulated as an integer program. Both exact and heuristic algorithms are developed to solve the problem. Heuristics for the same problem is studied in Kim and Kim (2003), where a genetic algorithm and a beam search algorithm are developed for the dispatching of a single yard crane for loading jobs. The numerical experiments show that both algorithms can find near optimum solutions for small problems of ten yard bays. However, the neighborhood beam search algorithm outperforms the genetic algorithm for large scale problems. Kim et al. (2003) study the delivery and receiving operations for a single yard crane in order to reduce the total delay time of external trucks. Different sequencing rules are compared with the dynamic programming approach. In previous studies, the handling sequence of an individual container is not decided. Kim et al. (2004) take this into consideration and solve the whole problem in two sub-problems. The first sub-problem decides the travel route of a yard crane as well as the number of containers to pick up at each bay. The second sub-problem decides the load sequence for individual containers. A beam search algorithm is developed to combine these two sub-problems. Ng and Mark (2005 a, b) study the scheduling of a single yard crane for a given set of loading and unloading containers with different ready times. Efficient heuristics are provided to find the lower bounds and upper bounds, while a branch and bound algorithm is proposed to solve the NP-complete problem. The numerical experiments show that the optimal handling sequence for individual containers can be found for most instances.

When more than two yard cranes are studied at the same time, the cooperation of the yard cranes to serve the common quay crane and the interference among the yard cranes shall be considered. Zyngiridis (2005) uses integer linear programming to study the scheduling of one or two RMGs of equal size in a single block. The data from Rotterdam with different characteristics is used in the numerical experiment to evaluate the performance of the RMGs. It is found that the performance of single RMG is significantly affected by the block size and the fill level of each block, while the performance of two RGMs is only affected by the length of the block. Besides, the case of two RMGs is always better than the case of one RMG in terms of efficiency. Cao et al. (2006a) discuss the scheduling of yard cranes considering the loading sequence requirements. In their problem, two yard canes are deployed to serve loading containers to the same quay crane. The two yard cranes will not interfere with each other, as they pick up containers in separate blocks. A revised genetic algorithm is proposed to find the solutions. A variation of the problem is studied in Cao et al. (2006b) for a system with multiple yard cranes. An algorithm based on the simulated annealing and a greedy heuristic are developed to find the scheduling sequence of yard cranes to minimize the total handling time. The performance of the algorithms is tested with randomly generated experiments. It is shown that the greedy heuristic can find outperforming solutions compared with the simulated annealing algorithm in limited computation time. Ng (2005) studies the problem of scheduling multiple yard cranes which share the single bi-directional travel lane and cannot pass through each other. The scheduling problem is formulated as an integer programming model to minimize the total loading time or the sum of truck waiting times. A heuristic based on dynamic programming is developed to solve the problem, and an algorithm is provided to find the lower bound. Jung and Kim (2006) study the problem of scheduling multiple yard cranes which serve multiple quay cranes, where the adjacent yard cranes working in the same block have interference with each other. The algorithms based on GA and SA approaches are proposed to schedule the travelling route of the yard cranes and number of containers to pick up in each yard bay. Jung et al. (2006) extend the problem to schedule the loading sequence of the quay cranes considering the interference of multiple yard cranes. A greedy randomized adaptive search procedure is proposed for constructing a schedule to minimize the makespan of the quay cranes. The numerical experiments show that the improvement phase of the heuristic search algorithm is too time consuming for practice. Lee et al. (2007) study the problem of scheduling two yard cranes with the simulated annealing algorithm.

The two yard cranes travel in separate blocks and serve the same quay crane for loading operation. The main objective is to minimize the total loading time at the stack area. The numerical experiments show that the proposed algorithm can find solutions close to the lower bound. Chang et al. (2011) study the scheduling of yard cranes in a rolling horizon approach. The yard cranes are deployed in separate zones to avoid interference during operation. Due to the computational scale of the problem, a heuristic algorithm integrated with a simulation model is proposed to generate the initial solutions, which are further optimized with a genetic algorithm.

Some studies have also addressed especially for the crossover rail mounted RMG system. For this kind of system, two or more RMGs are deployed in the same block. Since the RMGs are of different size, they can pass through each other without blocking the way. This system is studied in Cao et al. (2008), Dorndorf and Schneider (2010), and Vis and Carlo (2010).

2.3.2 Yard crane deployment

Due to the vessel schedule and the storage strategies used for container assignment, the distribution of workload among the blocks changes over time. As the yard cranes are very expensive equipment, the port operators usually do not keep a fixed number of yard cranes in each block. Instead, the yard cranes are deployed dynamically among the blocks to fit the changing workload distribution and finish the workload within the planned time periods. Zhang et al. (2002) study the yard crane deployment to decide the deployment times and routes of the yard cranes, based on the given workload distribution in each period during the planning horizon. The problem is formulated as a mixed integer program and solved with Lagrangean relaxation techniques. Cheung et al. (2002) analyze the computational complexity of the problem. Besides a Lagrangian decomposition solution procedure, a new solution approach called the successive piecewise-linear approximation method is proposed to solve the problem. The numerical experiments show that their new solution approach is efficient and effective for large scale problems. The similar problem is also studied in Linn and Zhang (2003), and Linn et al. (2003). In previous studies, the workload is commonly estimated based on the number of containers to be handled in each period, which is often inaccurate in practice. Guo and Huang (2012) propose a hierarchical scheme to divide the workload and deploy the yard cranes. A time partition plan divides the planning horizon into smaller time windows; while a space partitioning algorithm flexibly divides the workload into non-overlapping zones for yard crane deployment in each time window. The yard crane dispatching problem can be combined for the job sequence in each zone.

However, these studies are all based on given or estimated workload distribution, which is determined by space allocation to incoming containers. Thus, many studies combine the yard crane deployment with the space allocation decisions. More details will be discussed in the next section.

2.4 The storage space management

To increase the stacking capacity of the limited land, containers are usually stacked into the multi-level blocks. The efficiency of stacking depends greatly on the space allocation to arriving containers (Lee et al. 2006), also known as the container assignment. According to the detailed level of decisions, the previous studies on container assignment can be categorized into two groups. The first group studies the location assignment, which determines the slots for individual containers within a bay. The second group studies the space allocation, which distributes the containers to blocks or bays in the whole yard.

2.4.1 Location assignment

When the containers are stacked in multi-levels, only those on top can be accessed directly. The extra moves to reposition the containers on top of a requested one are called “reshuffles”. As they increase the retrieval time, it is important to reduce the reshuffles through proper container stacking. Sculli et al. (1988) is one of the first to study the relationship between reshuffles and space utilization considering the stacking configuration. Watanabe (1991) and Kim (1997) then develop different methodologies to estimate the reshuffles in container retrieval, which enlighten the later studies. Chen (1999) discusses the main causes for reshuffles as well as the different methods that can be used to reduce the reshuffles.

One of the commonly used methods is to store the incoming containers at a proper location to minimize the expected reshuffles. Kim et al. (2000) propose a methodology to determine the storage location of arriving export containers considering the weight. It is assumed that heavy containers will be loaded before the light ones to keep the stability of the vessel. Thus, putting heavier ones on top will reduce the reshuffles. A dynamic programming model is formulated to find the optimal solution. As solving the model directly is time consuming, a decision tree is developed to support the real-time decisions. Kang et al. (2006) study the storage location under the similar set, but the weight information of each arriving container is only an estimate. A simulated annealing approach is proposed to derive a good stacking strategy with uncertain weight information. Simulation experiments show that the derived strategies are more efficient in reducing the reshuffles compared with the traditional same-weight-group-stacking strategy. The performance can be further improved if machine learning is applied to improve the accuracy of weight classification. Yang and Kim (2006) address a dynamic version as well as a static version of the location assignment problem. The arrival and retrieval times of the containers are given in the static version, which is described with a mathematical model and solved with a genetic algorithm. For the dynamic problem with uncertain arrival times, heuristic rules for determining the storage location are proposed. Park et al (2011) propose an online search algorithm which can dynamically adjust and optimize the stacking policy. Unlike the offline methods which compute the optimal solution before taking any action, the online algorithm continuously generates and evaluates variants of the stacking plan during operation. This is a good option to have a fast reaction to the dynamic setting. Chen and Lu (2012) address the similar problem combining both the space allocation stage and the location assignment stage. The first stage is solved with a mixed integer programming model considering the travelling distance and imbalance of workload. The detailed locations are solved in the second stage with a hybrid sequence stacking algorithm. Both stages are solved under a static case without considering the uncertainties.

However, even when the slots for each container are carefully planned, they can still be stacked in the wrong order upon their arrival, due to the lack of accurate information or for other reasons (Steenken et al. 2004). One solution is to pre-marshal the containers according to the retrieval requirements. Lee and Hsu (2007) study the pre-marshalling problem for a single bay to minimize the number of container movements during the pre-marshalling process. The problem is treated as a multi-commodity flow problem and formulated as an integer programming model. The similar topic is studied in Lee and Chao (2009) for much larger scale problems with a neighborhood search algorithm. Kim and Bae (1998) study the pre-marshalling of export containers among several bays. In order to reduce the reshuffles, the current bay layout needs to be converted into a desirable bay layout. They use the hierarchical approach to decompose the problem into three sub-problems. Firstly, choose the target bays from the current bays to match with the required layout. Secondly, decide the amount of containers to be moved from a specific bay to the target bays. Finally, determine the task sequencing of container moves to minimize the completion time of re-marshaling. The first and final sub-problems are solved with dynamic programming, while the second is solved as a transportation problem. Although this paper studies the pre-marshalling problem, the detailed relocation of individual containers is not considered. Choe et al. (2011) further study the pre-marshalling between bays considering the detailed locations of individual containers. A good target stacking configuration is generated with a simulated annealing algorithm, which is then evaluated with the crane working schedule for moving the containers.

Another solution is to reduce the future reshuffles which are caused by the reposition of containers on top of a requested one. Kim and Hong (2006) study the relocation of containers during the container pickup process. The relocations occur only at the moment of retrieving a target container, and the relocated containers may cause additional future relocations if the position is not well chosen. The objective is to reduce the expected additional relocations where both the stacking configuration of all containers in the bay and the retrieval precedence of containers are given. They propose a heuristic method to find solutions close to the optimum, while the computation time is much shorter than the branch and bound algorithm and suitable for real-time decisions. Wan et al. (2009) study the relocation of export containers within a stack to minimize the total amount of reshuffles. An integer programming model is formulated for a static problem to retrieve all the containers within a stack when there are no incoming containers. An IP-based heuristic is proposed to find the solution and the heuristic is then extended to the dynamic setting where there are continual incoming and outgoing containers. Lee and Lee (2010) propose a three-phase heuristic to solve for an optimal plan of container retrieval. The main objective is to minimize the weighted sum of the container movements and the crane working time. The numerical results show that the number of movements in the final plan is close to the lower bound, and the heuristic can solve instances of real scale problem. Caserta et al. (2011) consider a dynamic programming approach by transforming the container relocation problem into a shortest path problem. A corridor method is proposed to shorten the search tree and accelerate the search process. Caserta et al. (2012) further study the same problem and prove the NP-hardness of the container relocation problem. Two different binary integer formulations are proposed. The first maps the complete feasible region of the problem leading to large search space, while the second formulation decreases the feasible region with realistic assumptions and is more useful in application.

2.4.2 Space allocation

The space allocation problem studies the container assignments to blocks or bays, which decides the distribution of container handling activities in the yard as well as the deployment of yard resources. This is also the focus of our study. As the space allocation is greatly affected by the storage strategies, the research works can be further categorized.

For import containers, Castilho and Daganzo (1993) analyze the segregation and non-segregation strategies with a simulation study. The first strategy separates the new incoming containers from the old ones to facilitate the retrieval, while the second strategy mixes the containers. An idealized case is used to identify the conditions favoring each strategy. It is found that the segregation strategy can reduce reshuffles for import containers when the retrieval sequence is affected by the arrival time. Under the segregation strategy, Kim and Kim (1999d) study the space allocation and the yard crane requirement considering the arrival pattern of import containers. It is formulated as an assignment problem constrained by the space capacity, and solved using Lagrangian relaxation technique. Kim and Kim (2002) further explore the same strategy, where they study on how to optimize the space needed for the given container volume. Sauri and Martin (2011) follow the research line of Castilho and Daganzo (1993) to further develop the segregation and non-segregation concepts. Three storage strategies are proposed for import containers, regarding how to mix the containers arriving in different batches. A mathematical model based on probabilistic distribution functions is developed to evaluate the reshuffles. The study shows that the choosing of proper strategy depends on the stacking height, the inter arrival time of vessels, and the dwell time of containers.

For export containers, Dekker et al. (2006) use detailed simulation experiments to test the different stacking policies in automated container terminals. It is found that category stacking for export containers can lower the number of reshuffles. Usually, the containers to the same destination vessel are stored together to facilitate the loading process, which is known as the “consignment strategy”. The major concern is how to reserve the space for future incoming containers to the same destination vessel. Taleb-Ibrahimi et al. (1993) point out that almost 50% of the space reserved will be empty waiting for future arrivals, if the space is only allocated for once. Thus, they propose to store the early incoming containers in a temporary area, before the permanent area is allocated. The best time to allocate the permanent space is decided based on the arrival pattern of export containers. However, their strategy improves the space utilization at the cost of double handling. Lim and Xu (2006) consider the space allocation under consignment as a two-dimensional packing problem. The space to each cluster of containers is allocated according to the requirements of incoming containers at different time periods, instead of assigning all the space for once.

Despite the difference in storage strategies, these studies all address the long-term problems where the incoming containers are given for each time period during the whole planning horizon. This is mainly because the space allocation in one period has impact on future period. However, during the short-term operation, the incoming containers are only known for a short period in advance. An effective solution is the rolling horizon method, which has been used in both Kim and Park (2003) and Zhang et al. (2003) for space allocation. The essence of the rolling horizon method is to plan based on the realized information of the current period and the estimation of the near future. After implementing the plan for the current period, a new plan will be formulated based on the latest realized and estimated information. In Kim and Park

(2003), a space allocation method is developed for export containers within limited space. The main target is to minimize the travel distance between the apron and the storage location. To utilize the storage space more efficiently, the space allocation for one vessel is divided into small stages, to satisfy the space requirements at each stage during the planning horizon. The space is allocated in terms of bays, and the rolling horizon approach is used when implementing the model. Zhang et al. (2003) study a storage space allocation problem by a rolling horizon approach, where the problem of each period is divided into two levels. The first level balances the workload assigned to each block, while the second level minimizes the transfer distance of containers. Various resources have been considered in the study, such as quay cranes, yard cranes, storage space and internal trucks. Bazzazi et al. (2009) study an extension for the first level decisions of the space allocation problem in Zhang et al. (2003). The type of containers affects the assignment of containers to blocks. A genetic algorithm is used to solve the problem.

Although many studies can be found for the space allocation problem, most of them are done for the gateway ports, which do not sufficiently address the particular needs of transshipment hubs. In a gateway port, the loading and unloading activities can be considered independently by separating import and export containers. For transshipment hubs like the Singapore port, around 74% of the containers are transshipment, which are discharged from one vessel and loaded onto other vessels. To avoid double handling, a container will stay in the same location until being retrieved. The assignment of incoming containers determines not only the distribution of current unloading activities, but also the distribution of future loading activities. As the discharging and loading activities are both in large batches, an improper plan of space allocation will lead to heavy congestions of container handling activities. Lee et al. (2006) study the space allocation to determine the long term yard crane deployment. A consignment strategy is used to store containers going to the same destination vessel together. A high-low workload balancing protocol is proposed to control the congestions under consignment. The main idea is to avoid high workload appearing in neighboring storage locations. A MIP model and solution approaches are developed for space allocation based on this idea. Han et al. (2008) extend the work of Lee et al. (2006) to consider "yard template", which determines the reservation of storage locations for the destination vessels. Containers are assigned to the locations reserved for their own destination vessel. A good yard template helps to control the congestions and facilitate the yard crane deployment. An integrated algorithm is proposed to solve the template planning and space allocation iteratively. However, both Lee et al. (2006) and Han et al. (2008) use the consignment strategy. Although this strategy facilitates a faster loading process, it is known to be inefficient in space utilization.

2.5 Research gaps and motivations

From the previous literature review, it can be found that most of the studies on space allocation are done for the gateway ports, which do not sufficiently address the particular needs of transshipment ports. Only a few studies are addressing the storage yard management problems in transshipment ports, such as Lee et al. (2006) and Han et al. (2008). There are two major research gaps for the space allocation in the transshipment ports.

- Space utilization. Generally, the "consignment strategy" is used in the yard for a transshipment port, where containers to the same destination vessel are stored together. This is to facilitate faster loading process as it reduces reshuffles as well as long distance movements of yard cranes. However, the consignment strategy is known to be inefficient in space utilization since each storage location must be dedicated to a particular vessel. With the rapid growth of the container traffic, more and more containers will be handled and temporarily stored in the yard. The scarcity of land is posing serious challenges for the port operator to provide efficient services. To improve the space utilization while retaining the advantage of consignment, new storage strategies are proposed in the following sections of this thesis, namely the “partial space-sharing strategy” and the “flexible space-sharing strategy”.

- Short-term space allocation. In the previous studies, the storage strategies are all studied for long-term planning. During the operation, the actual containers that will come in are only known for a short period in advance. Thus, short-term space allocation is needed to assign the incoming containers taking into account of transport vehicles, yard cranes and space capacity. Currently, the space is allocated based on the experience of port operators and the rule of thumb. To our knowledge, this is the first work for short-term space allocation in transshipment ports.

Chapter 3. The Partial Space-sharing Strategy

3.1 Problem description

To provide more flexibility during operation, the terminal we study is divided into sections, and vessels are assigned to sections, each corresponding to several berths, rather than the exact berth locations. Therefore, when we conduct the yard storage allocation within a section, the specific planned berth of a vessel need not be considered. Furthermore, import containers are not considered in this study, since they have different characteristics and are usually stored in separate blocks from export and transshipment containers.

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. All different sections of the storage yard are composed of some common basic modules: “sub-block” and “block”. To reduce the level of reshuffles, a consignment strategy is used, where export and transshipment containers going to the same destination vessel are stored together in the yard. The smallest unit for the consignment strategy in yard storage allocation process is a “sub-block”. The depth of each sub-block is 6 rows of containers, and the length of each sub-block is 8 slots (each slot can accommodate one 20ft container length-wise). The stacking height is 5 containers high (which we call tier). A certain number of sub-blocks in a row form a bigger unit, called a “block”. There is a dedicated lane for the movement of prime movers (the “truck path”) and a separate “passing” lane strictly to allow trucks to pass each other when required. The passing lane is only wide enough for one prime mover and it is shared between two neighboring container blocks.

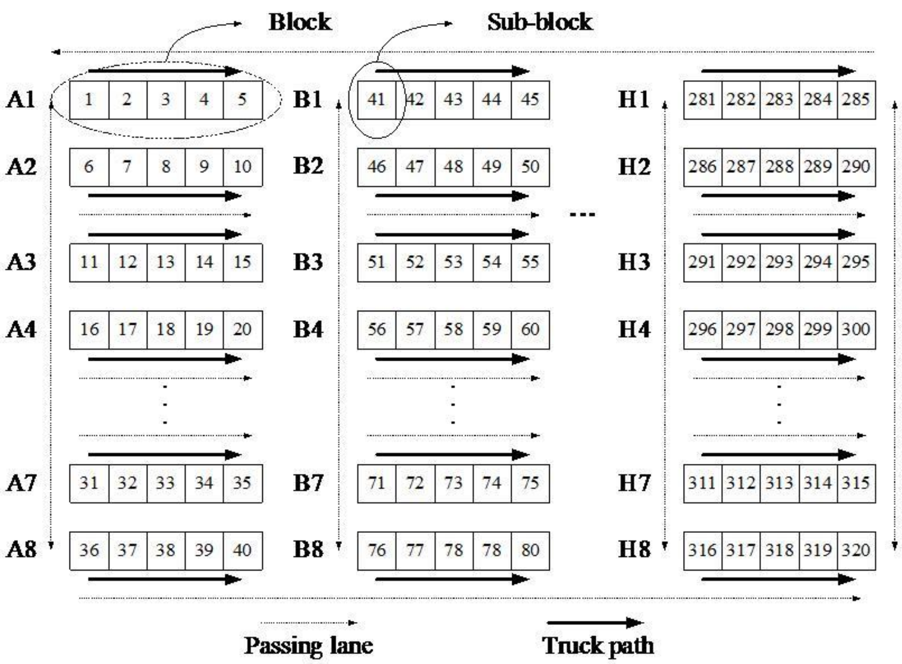

Figure 3.1 A storage yard configuration

According to the workload patterns provided by the port operator, there are two important characteristics of the incoming containers. The characteristics are that the higher incoming workload always happens near to their departure date, while the very low activity happens right after the vessel departs. Hence, it is a common practice for them to use triangular workload profile to do the planning.

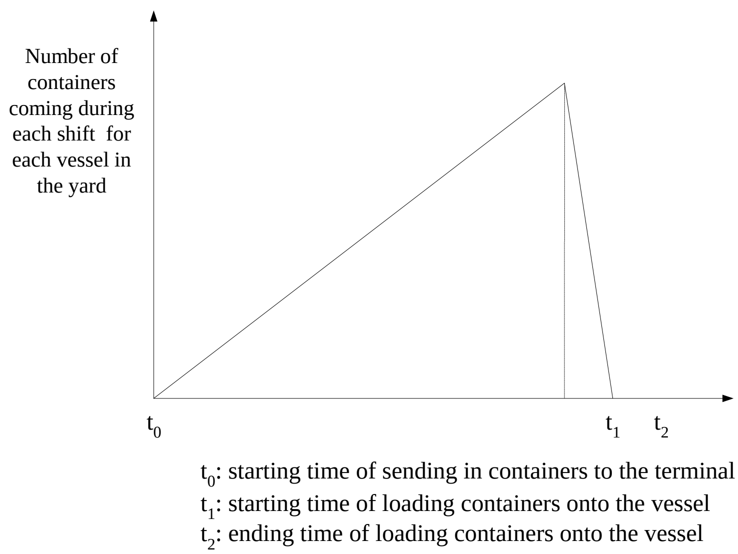

Figure 3.2 The buildup pattern of the coming workload for one vessel

For the static yard template (as in Lee et al. (2006) and Han et al. (2008)), all the sub-blocks in each block have a fixed space capacity, as shown in Figure 3.3. This means the maximum amount of space needed at the peak time will be exclusively assigned to each vessel during the whole planning horizon. As much space is only occupied for a short period, it clearly leads to under-utilization of the space. To enjoy the benefit of consignment while increasing the land utilization, we propose a space-sharing method which allows some space to be shared between adjacent neighbors. Essentially it will help to reduce the original space needed for a given workload. As shown in Figure 3.3, for the space-sharing yard template, each sub-block has certain amount of storage space for sharing. For example, s12 is the part that can be shared between sub-blocks 1 and 2.

Static Yard Template

Sub-block 1	Sub-block 2	Sub-block 3	Sub-block 4	Sub-block 5

Space-sharing Yard Template

s1	s12	s2	s23	s3	s34	s4	s45	s5

Figure 3.3 Schematic diagram of one block for the static yard template
and the space-sharing yard template

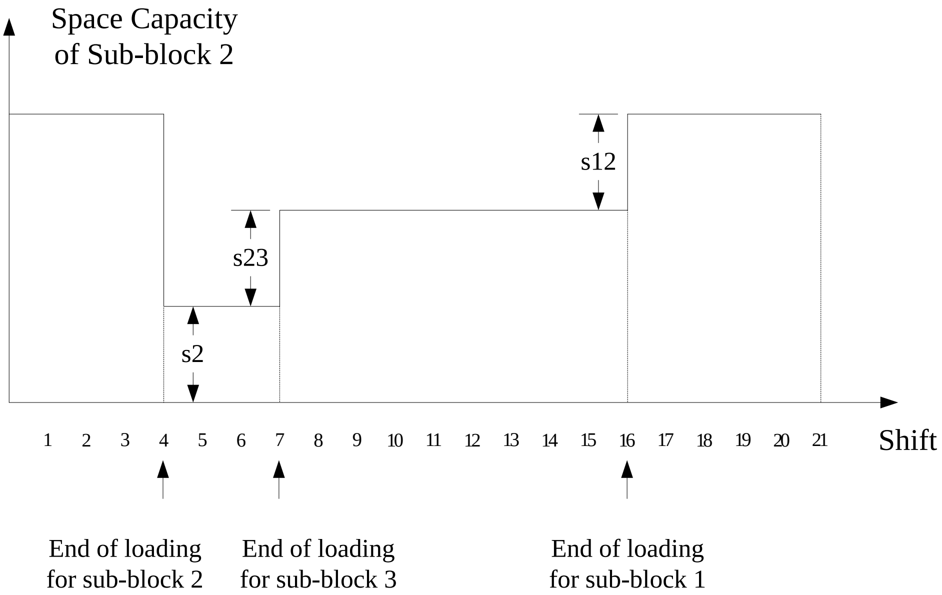

Figure 3.4 A schematic diagram for space capacity of one sub-block

As very few space is needed during the period right after the loading process, the sharing space of one sub-block can be lent to its neighbors. It will then be returned from its neighbors, before the major workload comes into this sub-block. Since the major workload arrives at different periods for different vessels, they will also need the sharing space during different shifts. We can take the sub-block 2 as an example to demonstrate how its space can change over time. Suppose that Sub-blocks 1, 2, and 3 have been assigned to different departing vessels, and the starting times of their loading operations are Shifts 14, 2 and 5 respectively. Then, the starting times of sharing the spaces to the neighbors for Sub-blocks 1, 2, and 3 are Shifts 16, 4, and 7 respectively, assuming that the loading operations last for 2 shifts. Since Sub-block 2 has Sub-blocks 1 and 3 as neighbors, the change of its space capacity over the 21 shifts can be plotted as in Figure 3.4. Similarly, the storage space of all the sub-blocks in one block changing over the 21 shifts is shown in Figure 3.5. In other words, the space capacity of one sub-block will decrease after the sub-block’s loading process, while it increases when its neighbors finish loading. However, the sum of a non-sharing space and its neighboring sharing spaces should be not more than the standard size of a sub-block given by the port operator.

To implement this space-sharing concept, three key issues should be resolved; namely, yard template, size of sharing space and workload assignment.

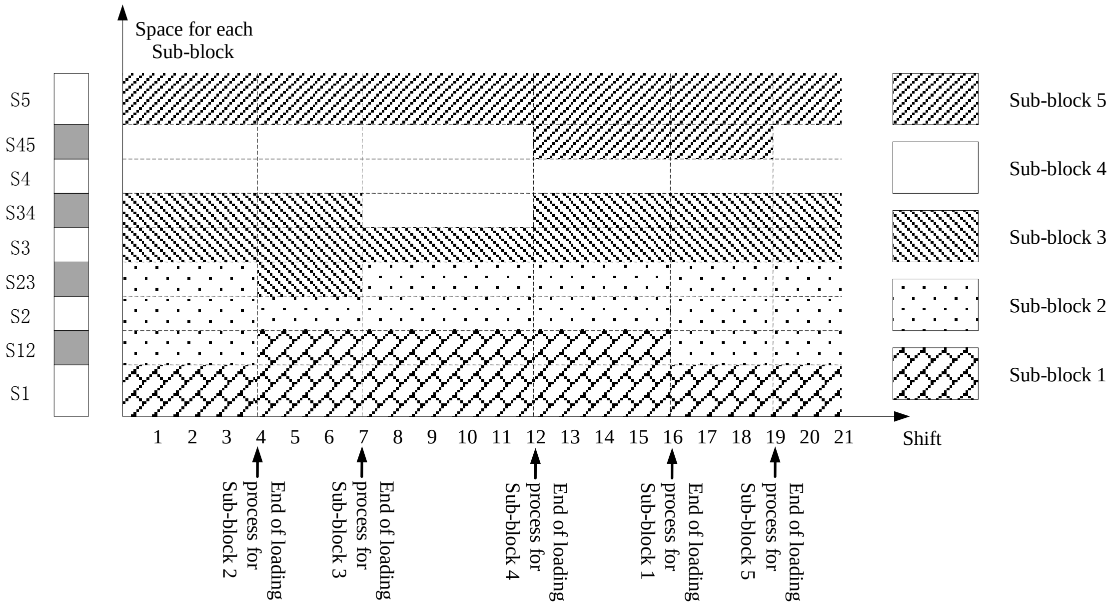

Figure 3.5 A schematic diagram for the space capacity of each sub-block in one block

Since the yard cranes and transporters handle one container at a time, the number of loading and unloading containers in each sub-block can be used to indicate the potential traffic. To ensure a smooth flow of traffic, we adopt the high-low workload balancing protocol and the vicinity matrix from Lee et al. (2006) and Han et al. (2008). The vicinity matrix is used to capture the neighborhood relationship among sub-blocks, while the high-low workload balancing protocol is implemented to avoid potential traffic congestion and to ensure high utilization of yard cranes.

In summary, for this problem, we need to determine a yard template for storage space first. Given the yard template, we will determine the amount of sharing space between each pair of adjacent sub-blocks to improve the land utilization. Meanwhile the number of containers assigned to each sub-block should also be decided for each shift. According to Han et al. (2008), the port operator does not have any formal planning model to determine the yard template, the sharing space, and container allocation. The decisions are based on yard planners' ingenuity and past experiences. As a means to remedy this, a framework that incorporates the concepts discussed above is developed in the next section.

3.2 Solution approaches

In this study, there are two main objectives of concern, namely operation efficiency and land productivity. The operation efficiency is measured by the number of yard cranes deployed in a shift. The land productivity measures the amount of space needed in order to handle a certain amount of workload, which is defined as "*total volume of incoming containers / land needed for container handling*". The land productivity can capture the land utilization. As the operation efficiency is more critical in the yard planning, we aim to find the minimum space needed to handle a given amount of workload while ensuring only using a least number of yard cranes in each shift. Since this problem is too complex to solve in an integrated model, a framework which combines space reservation and workload assignment is proposed in this section. The general picture of the framework can be shown in Figure 3.6.

The first step is "template generation", which decides the sub-block reservation for each vessel. Based on the information of incoming workload, we can get the requirements of the yard template, such as the minimum number of sub-blocks needed by each vessel. A yard template will then be generated satisfying the requirements.

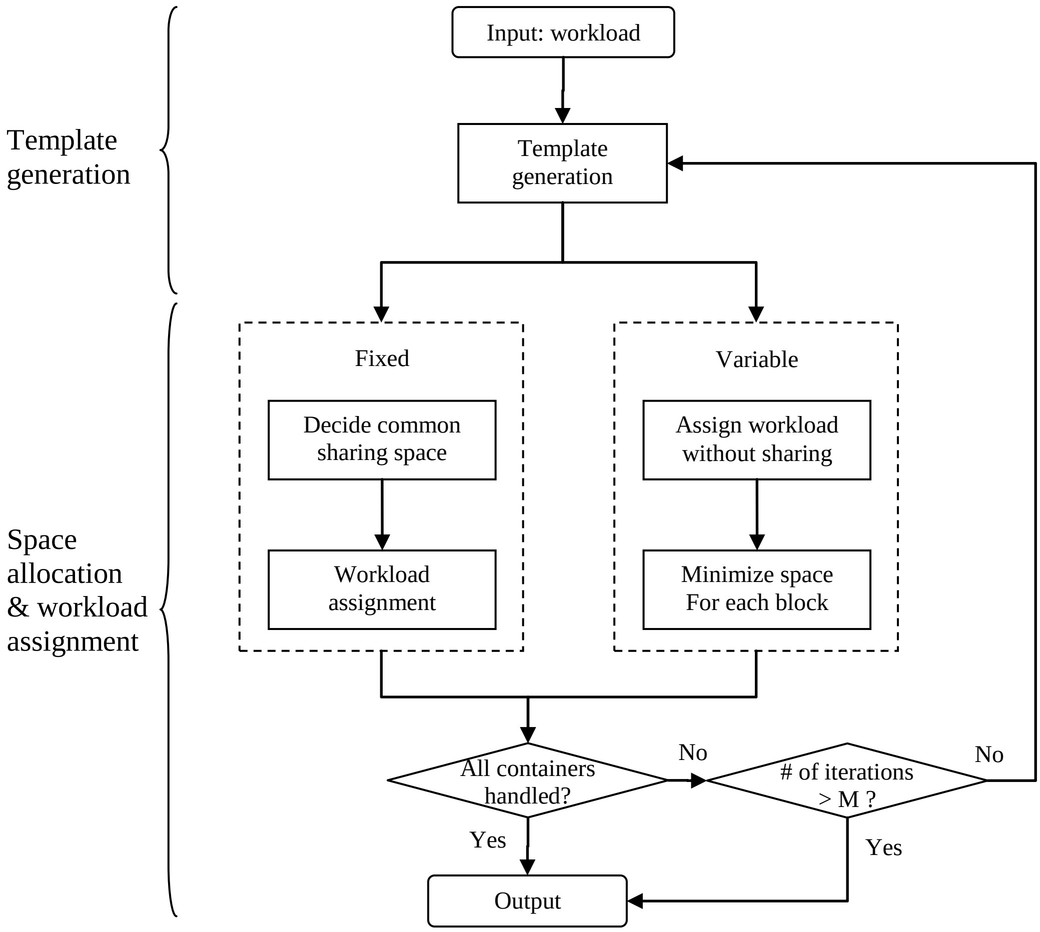

Figure 3.6 The framework for solution

The second step is “space allocation and workload assignment”, which decides both the size of sharing space and number of incoming containers for each sub-block in each shift to minimize the space needed. Two different methods can be used to achieve the tasks in this step. The first method is to use a common size of sharing space to simplify the problem. Incoming workload will then be assigned to each sub-block to maximize the common size of sharing space while guaranteeing the least number of yard cranes deployed in each shift. The details will be presented in the later part of this section. The second method is the variable sharing space method, in which the size of sharing space can be different across sub-blocks. In this method, we first assign the incoming workload without sharing space. Since the space needed in each shift for all sub-blocks is known after the workload assignment, the appropriate size of sharing space between adjacent sub-blocks can be decided through merging the space needs lasting for different periods to minimize the space needed.

To guarantee operation efficiency, the least number of yard cranes from Han et al. (2008) can be adopted to control yard cranes deployed in each shift. If all the containers can be handled within the space-sharing yard template while guaranteeing the operation efficiency, we get the final solution. Otherwise, the searching will go back to the first step and generate a new yard template as long as the iteration limitation allows.

3.2.1 Template generation

The yard template is a plan of the yard space reservation for different vessels. It will significantly affect the final result of workload assignment and yard crane deployment. In order to develop a better template, Han et al. (2008) provide an algorithm to generate, repair and improve the template iteratively. In their algorithm, an initial yard template is generated based on the relationship of vessels. Then the template is improved and repaired through swapping sub-blocks reserved for different vessels. The gist of their method is to look into the bottleneck to provide more available space for each vessel in each shift. Some important underlying criteria can be observed from their algorithm:

- In any shift, the number of loading sub-blocks in each block should be no more than one to prevent traffic congestion.
- No neighboring sub-blocks can be loading in the same shift.
- A sub-block is available to receive incoming containers in certain shift only when all its neighbors are not loading in the same shift.

- One sub-block can only be assigned to one vessel (i.e., no mixed stacking).
- The number of sub-blocks assigned to each vessel should be enough to satisfy the space needs of incoming containers.

Since the algorithm is mainly based on pair-wise swapping of sub-blocks, the final template obtained is dependent on the sequence of swapping and the way of searching. To overcome this, a new model is proposed in this section to generate the template directly.

Using the above criteria as constraints, a mathematical programming model can be developed to obtain an effective template. In the proposed model, the bottleneck vessel for the yard template is defined as the vessel that has the smallest number of available sub-blocks in any shift. Similarly, the bottleneck shift is defined as the shift in which the bottleneck vessel has the smallest number of available sub-blocks. In the template generation problem we are trying to get a yard template which provides more available sub-blocks across time. More importantly, the bottleneck vessel and bottleneck shift should be improved to provide more choices for the assignment of incoming workload, which ensures a better solution for the whole space-sharing yard planning problem.

The model parameters are as follows:

A_j The smallest number of sub-blocks that should be reserved for vessel 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 time, which may vary according to the condition of different blocks, $1 \le k \le K$.

I The number of sub-blocks under consideration.

J The number of vessels under consideration.

K The number of blocks under consideration.

L_{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, $1 \le j \le J$, $1 \le t \le T$.

N_i The set of sub-blocks that are neighbors of sub-block i, $1 \le i \le I$.

T The number of shifts under consideration in the planning horizon.

V The lower bound for total number of available sub-blocks in all shifts.

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

The decision variables are as follows:

v_{it} = 1, if sub-block i is available during shift t, $1 \le i \le I$, $1 \le t \le T$.

= 0, otherwise, $1 \le i \le I$, $1 \le t \le T$.

z_{ij} = 1, if sub-block i is reserved for vessel j, $1 \le i \le I$, $1 \le j \le J$.

= 0, otherwise, $1 \le i \le I$, $1 \le j \le J$.

u_{ijt} A variable used to indicate the value of ($v_{it} \times z_{ij}$) in linear form.

$1 \le i \le I$, $1 \le j \le J$, $1 \le t \le T$.

The mathematical programming model for the yard template generation problem (denoted as YTG) is as follows:

(YTG) $Max\ w$ (3.1)

Subject to:

$$w \leq \sum_{i}^{I} z_{ij} v_{it} \qquad \forall \ 1 \leq j \leq J, 1 \leq t \leq T \qquad (3.2)$$

Or equivalently (3.3) ~ (3.6):

$$w \leq \sum_{i}^{I} u_{ijt} \qquad \forall \ 1 \leq j \leq J, 1 \leq t \leq T \qquad (3.3)$$

$$u_{ijt} \leq z_{ij} \qquad \forall \ 1 \leq i \leq I, 1 \leq j \leq J, 1 \leq t \leq T \qquad (3.4)$$

$$u_{ijt} \leq v_{it} \qquad \forall \ 1 \leq i \leq I, 1 \leq j \leq J, 1 \leq t \leq T \qquad (3.5)$$

$$u_{ijt} \geq 0 \qquad \forall \ 1 \leq i \leq I, 1 \leq j \leq J, 1 \leq t \leq T \qquad (3.6)$$

$$\sum_{j=1}^{J} z_{ij} \leq 1 \qquad \forall \ 1 \leq i \leq I \qquad (3.7)$$

$$\sum_{i=1}^{I} z_{ij} \geq A_j \qquad \forall \ 1 \leq j \leq J \qquad (3.8)$$

$$\sum_{i \in B_k} \sum_{j=1}^{J} z_{ij} L_{jt} \leq C_k \qquad \forall \ 1 \leq k \leq K, 1 \leq t \leq T \qquad (3.9)$$

$$\sum_{i' \in N_i \cup \{i\}} \sum_{j=1}^{J} z_{i'j} L_{jt} \leq 1 \qquad \forall \ 1 \leq i \leq I, 1 \leq t \leq T \qquad (3.10)$$

$$1 - v_{it} \leq \sum_{i' \in N_i} \sum_{j=1}^{J} z_{i'j} L_{jt} \leq 3 - 3v_{it} \qquad \forall \ 1 \leq i \leq I, 1 \leq t \leq T \qquad (3.11)$$

$$\sum_{t}^{T} \sum_{i}^{I} v_{it} \geq V \qquad (3.12)$$

$$v_{it} \in \{0, 1\} \qquad \forall \ 1 \leq i \leq I, 1 \leq t \leq T \qquad (3.13)$$

$$z_{ij} \in \{0, 1\} \qquad \forall \ 1 \leq i \leq I, 1 \leq j \leq J \qquad (3.14)$$

The objective is to maximize the value of the bottleneck. We use w to represent the value, and therefore for any vessel in any shift, the number of available sub-blocks should be bigger than w as shown in Constraint (3.2). Since both v_{it} and z_{ij} are 0-1 integers, Constraint (3.2) is equal to the linear constraints (3.3) ~ (3.6). Constraint (3.7) ensures that a sub-block can be reserved for at most one vessel during the whole planning horizon and no change in the reservation can be made once the reservation is made. However, part of the sub-block can be shared with its neighbors. Constraint (3.8) ensures that sufficient sub-blocks are reserved for each vessel. Constraint (3.9) ensures that the number of sub-blocks in loading process should be less than the maximum number of yard cranes assigned to each block in each shift; otherwise the rest of the sub-blocks in that block will be unavailable during the shift. Constraint (3.10) ensures that no neighboring sub-blocks should be loading in the same shift to avoid potential traffic congestion. Constraint (3.11) defines the availability of a sub-block. According to the configuration of the template, a sub-block can have at most 3 neighbors. Therefore, when the number of loading neighbors is 1~3, the sub-block is unavailable; when the number of loading neighbors is 0, the sub-block is available. Constraint (3.12) ensures that the number of available sub-blocks should be bigger than the requirement. Constraints (3.13) and (3.14) are 0 or 1 value restrictions for decision variables.

The YTG model is difficult to solve to optimality directly using CPLEX. However, there is no necessity to run the model to optimality since the purpose of this model is to get a good initial template to be used for subsequent steps. In our implementation, we will set required values for the objective and V in constraint (3.12) to provide feasible solutions efficiently. The yard template will then be tested in "space allocation and workload assignment" to see if it is good enough. If the incoming containers cannot be handled in step 2, the required values in step 1 will increase accordingly to get a new yard template for the next iteration.

Later in our numerical runs, we will show that the YTG model performs better than the algorithm in Han et al. (2008) because it can generate feasible template for the higher workload level.

3.2.2 Two different methods for space allocation and workload assignment

After the template generation step, the yard template will be used for space allocation and workload assignment. In this step, both the size of the sharing space and workload assignment should be decided. Two different methods are presented in this section to achieve these two tasks.

3.2.2.1 Fixed sharing space

In the "fixed sharing space method", a common size of sharing space is used across the yard template to simplify the problem. When the common size is used, each sub-block will be divided into non-sharing space and sharing space with neighboring sub-blocks. As the size of sub-block is fixed and the number of sub-blocks in a block is constant, the larger scale of common size means more sharing space and so the overall space in a block is reduced, which can be shown in Figure 3.7. In order to improve the land productivity, we need to maximize the common size of sharing space to handle all containers.

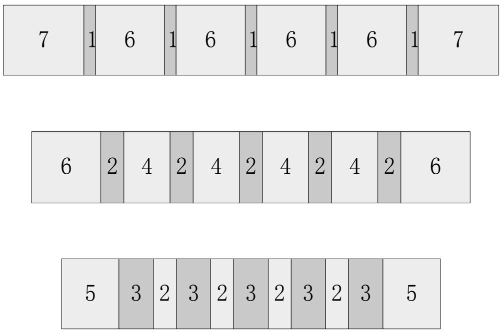

Figure 3.7 Block configurations with common size = 1,2,3 slots respectively

The model parameters are as follows:

CC The capacity of each yard crane in terms of container moves per shift, which is 100 in this model.

CLS_i The left space capacity of non-sharing space for sub-block i, $1 \le i \le I$.

$CLS_{ii'}$ The left space capacity of sharing space between sub-block i and i', $1 \le i \le I$, $1 \le i' \le I$.

CLC_i The left yard crane capacity of sub-block i for loading process, $1 \le i \le I$.

CS The original space capacity of each sub-block in terms of TEUs, which is 240 (5 tiers×6 lanes×8 slots) in this model.

HL The lowest value that a high workload can take.

HU The highest value that a high workload can take.

LB_t The least number of yard cranes that can be deployed for unloading in shift t, $1 \le t \le T$.

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.

NB_i The set of sub-blocks those are adjacent to sub-block i, where space sharing is possible between two sub-blocks, $1 \leq i \leq I$.

S_{ti} The set of shifts from the end of the loading time of sub-block i to the current shift t, $1 \leq i \leq I$, $1 \leq t \leq T$.

$W_{ii't}$ = 1, if the sharing space between sub-blocks i and i' ($s_{ii'}$) belongs to part of capacity of sub-block i in shift t. $W_{ii't}$ can be obtained from the loading time of sub-blocks i and i', $1 \leq i \leq I$, $1 \leq i' \leq I$, $1 \leq t \leq T$.

= 0, if the sharing space between Sub-blocks i and i' ($s_{ii'}$) belongs to part of capacity of sub-block i' in Shift t, $1 \leq i \leq I$, $1 \leq i' \leq I$, $1 \leq t \leq T$.

WX_{jt} The number of 20ft 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 40ft containers arriving at the terminal in shift t and will be loaded onto vessel j finally. It is given and input to the model, $1 \leq j \leq J$, $1 \leq t \leq T$.

Z_{ij} = z_{ij}, indicator of sub-block reservation for different vessels, adopting the value solved in template generation step, $1 \leq i \leq I$, $1 \leq j \leq J$.

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, $HL \le x_{it}+y_{it} \le HU$, $1 \le i \le I$, $1 \le t \le T$.

= 0, if the total workload allocated to sub-block i for unloading in shift t is low, that is, $LL \le x_{it}+y_{it} \le LU$, $1 \le i \le I$, $1 \le t \le T$.

s The common size of sharing space between each pair of adjacent sub-blocks.

s_i The space that belongs to sub-block i and cannot be shared with its neighbors, $1 \le i \le I$.

$s_{ii'}$ The space that can be shared between sub-blocks i and i', $1 \le i \le I$, $1 \le i' \le I$.

x_{it} The number of 20ft containers that are allocated to sub-block i for unloading in shift t, $1 \le i \le I$, $1 \le t \le T$.

y_{it} The number of 40ft containers that are allocated to sub-block i for unloading in shift t, $1 \le i \le I$, $1 \le t \le T$.

The workload assignment problem (WAP) can be formulated as follows to maximize the common size of sharing space.

(WAP) $Max\ s$ (3.15)

Subject to:

$$\sum_{i=1}^{I} Z_{ij} x_{it} = WX_{jt} \qquad \forall\ 1 \le j \le J, 1 \le t \le T \tag{3.16}$$

$$\sum_{i=1}^{I} Z_{ij} y_{it} = WY_{jt} \qquad \forall\ 1 \le j \le J, 1 \le t \le T \tag{3.17}$$

$$s_{ii'} = 30 \times s \qquad \forall\ 1 \le i \le I, 1 \le i' \le I \tag{3.18}$$

$$s_i + \sum_{i' \in NB_i} s_{ii'} = CS \qquad \forall\ 1 \le i \le I \tag{3.19}$$

$$\sum_{t'\in S_{it}} (x_{it} + 2y_{it}) \le s_i + \sum_{i'\in NB_i} W_{ii't} s_{ii'} \qquad \forall\, 1 \le i \le I,\ 1 \le t \le T \tag{3.20}$$

$$\sum_{t} (x_{it} + y_{it}) \le \sum_{j} \left(Z_{ij} \sum_{t} L_{jt} \times CC \right) \qquad \forall\ 1 \le i \le I \tag{3.21}$$

$$\sum_{i\in B_k} \left(x_{it} + y_{it} \right) \le d_{kt} CC \qquad \forall\, 1 \le k \le K \tag{3.22}$$

$$d_{kt} + \sum_{i\in B_k} \sum_{j=1}^{J} Z_{ij} L_{jt} \le C_k \qquad \forall\, 1 \le k \le K,\ 1 \le t \le T \tag{3.23}$$

$$\sum_{k} d_{kt} = LB_t \qquad \forall\, 1 \le t \le T \tag{3.24}$$

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

$$\forall\, 1 \le i \le I,\ 1 \le t \le T \tag{3.25}$$

$$\sum_{i'\in N_i} \left(x_{i't} + y_{i't} \right) \le M \left(1 - \sum_{j=1}^{J} Z_{ij} L_{jt} \right) \qquad \forall\, 1 \le i \le I,\ 1 \le t \le T \tag{3.26}$$

$$\sum_{i'\in N_i \cup \{i\}} h_{i't} \le 1 \qquad \forall\, 1 \le i \le I,\ 1 \le t \le T \tag{3.27}$$

$$s \in \{0, 1, 2, 3, 4\} \tag{3.28}$$

$$x_{it} \ge 0 \ \ y_{it} \ge 0 \qquad \forall\, 1 \le \mathrm{i} \le \mathrm{I},\ 1 \le t \le T \tag{3.29}$$

$$h_{it} \in \{\ 0, 1\ \} \qquad \forall\, 1 \le i \le I,\ 1 \le t \le T \tag{3.30}$$

$$d_{kt} \in \{Positive\ Integer\} \qquad \forall\, 1 \le k \le K,\ 1 \le t \le T \tag{3.31}$$

Constraints (3.16) and (3.17) ensure that all the workload arriving at the terminal in each shift for each vessel will be allocated to corresponding storage locations. Constraints (3.18) and (3.19) are used to calculate the size of sharing and non-sharing space for each sub-block. Constraint (3.20) ensures the space capacity restriction of each sub-block during each shift. The capacity includes the fixed part and the sharing part with its neighbors. Constraint (3.21) ensures that the containers in each sub-block should be loaded onto the destination vessel within a certain time span. Constraint (3.22) ensures that the yard cranes allocated to each block for unloading can handle all the unloading workload in each 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 (3.23) 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. Constraint (3.24) guarantees the least number of yard cranes deployed in each shift. To make full use of yard cranes, the workload allocated to each sub-block in each shift should be either high or low. In this model, constraint (3.25) is used to ensure this restriction. Constraint (3.26) ensures that all the neighbors of a sub-block in the loading process cannot accept any workload in that shift. Constraint (3.27) ensures that high unloading workload cannot be allocated to two sub-blocks that are neighbors of each other in the same shift. Constraints (3.28) to (3.31) are non-negativity and integrality restrictions.

As the maximum size of a sub-block is 8-slot, the scale of common size of sharing space can only take an integer value from 0 to 4 slots, as shown in constraint (3.28). In this case, we can set the objective value to different scales of common size and solve WAP for a feasible solution, while the solution for the largest scale of common size is kept as optimal. Besides, the only connection constraints from shift to shift are (3.20) and (3.21). If we replace these two constraints with (3.32) and (3.33) using the remaining space capacity and loading capacity in shift t, the workload assignment with fixed sharing space can be solved efficiently using the sequential method in Lee et al. (2006).

$$x_{it} + 2y_{it} \le CLS_i + \sum_{i' \in NB_i} W_{ii't} CLS_{ii'} \qquad \forall\, 1 \le i \le I \qquad (3.32)$$

$$x_{it} + y_{it} \le CLC_i \qquad \forall\, 1 \le i \le I \qquad (3.33)$$

3.2.2.2 Variable sharing space

In the previous method, the same size of sharing space is used for the yard template. Since the workload assignment does not match the same size, there will always be gaps of unused space. To fit variable size of sharing space to incoming containers, we can first assign the workloads as in static yard template planning. Then the size of sharing and non-sharing space can be decided based on the results of workload assignment.

The workload assignment based on the static yard template can be achieved using the WAP model by setting the common size of sharing space *s* as 0-slot. Based on the workload assignment solution of x_{it} and y_{it}, we now know the incoming and retrieval of containers for each sub-block. Then, the slots needed by each sub-block in each shift can be calculated with the remaining containers in the sub-block. If we number all the slots in a block as a sequence, the slots occupied by containers in two example sub-blocks can be shown in Figure 3.8. As the size of each sub-block is assumed to be 8 slots when we solve the workload assignment problem, sub-block 1 will use the slots chosen from slots 1 to 8, while sub-block 2 will choose from slots 9 to 16. To describe the problem more clearly, we use the range (S_{it}^L, S_{it}^U) to represent that the slot $S_{it}^L + 1$ to slot S_{it}^U are occupied by sub-block *i* in shift *t*. For instance, (8, 16) is used by sub-block 2 in shift 4, which means slot 9 to slot 16 are occupied.

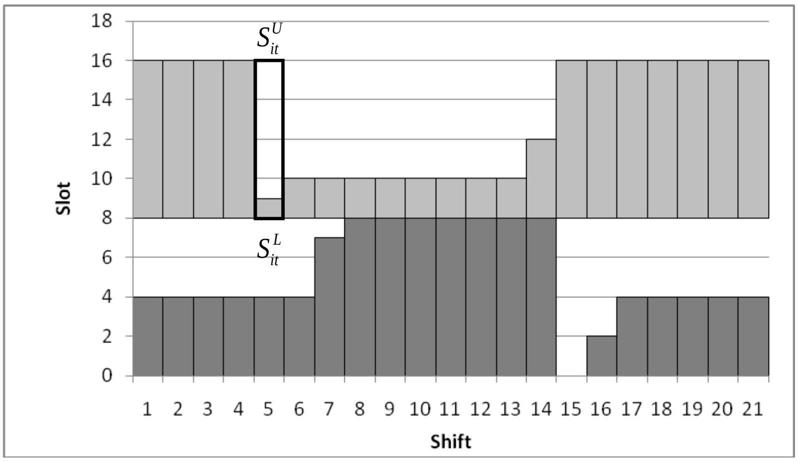

Figure 3.8 Relative locations occupied by two sub-blocks without sharing

Once a slot has incoming containers, it will be occupied until the end of the loading process of the sub-block. Hence if a slot is occupied by a sub-block in the current shift, the slot should also be assigned to the same sub-block in the subsequent shifts until the loading process ends. It can be found in Figure 3.8 that, slots 13 to 16 are not occupied by sub-block 2 from shifts 5 to 14, while slots 5 to 8 are occupied by sub-block 1 only from shifts 7 to 14. Therefore, 4 slots can actually be shared by the two sub-blocks, as they occupy different shifts. The size of sharing space can be decided between these two adjacent sub-blocks, as shown in Figure 3.9.

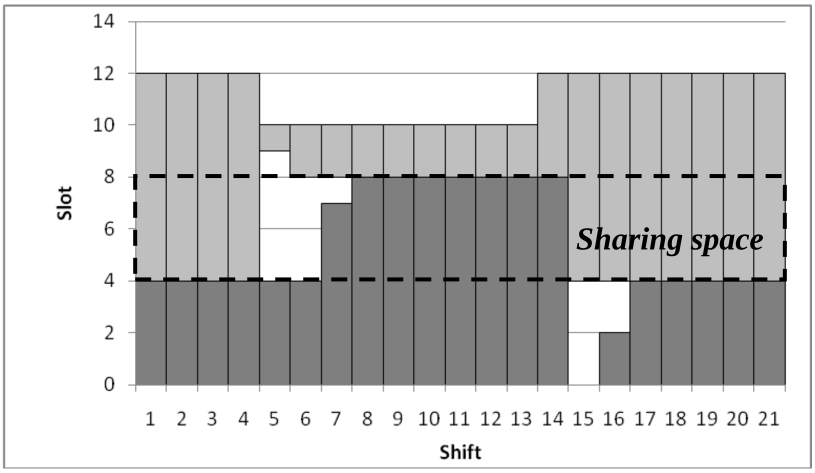

Figure 3.9 Relative locations occupied by two sub-blocks with sharing

Since space can only be shared between adjacent neighbors, to ensure no reshuffling, the size of each sharing space can be decided block by block. Based on the idea of variable sharing space method, we can develop the following model to solve the space-sharing problem. In this model, the objective is to minimize the total number of slots occupied by the whole block. The model parameters are as follows.

F_i The shift when sub-block *i* finishes its loading process, $1 \le i \le sub$.

sub The number of sub-blocks in each block.

The decision variables are as follows:

S_{it}^{L} The lower boundary of the slot range occupied by sub-block *i* in shift *t*, $1 \le i \le sub$, $1 \le t \le T$.

S_{it}^{U} The higher boundary of the slot range occupied by sub-block *i* in shift *t*, $1 \le i \le sub$, $1 \le t \le T$.

S_{it}^{40} The pair of adjacent slots of sub-block *i* needed for 40ft containers in shift *t*, $1 \le i \le sub$, $1 \le t \le T$.

The minimal block space problem (MBS) can be developed as follows:

$$\text{(MBS)}\ Min\ w \qquad (3.34)$$

Subject to:

$$w \ge S_{sub,t}^{U} \qquad \forall\ 1 \le t \le T \qquad (3.35)$$

$$(S_{it}^{U} - S_{it}^{L}) \times 30 \ge \sum_{t=F_i+1}^{t} (x_{it} + 2 \times y_{it}) \qquad \forall\ 1 \le i \le sub,\ 1 \le t \le T \qquad (3.36)$$

$$S_{it}^{40} \times 30 \ge \sum_{t=F_i+1}^{t} y_{it} \qquad \forall\ 1 \le i \le sub,\ 1 \le t \le T \qquad (3.37)$$

$$S_{it}^{U} - S_{it}^{L} \geq 2 \times S_{it}^{40} \qquad \forall\ 1 \leq i \leq sub,\ 1 \leq t \leq T \qquad (3.38)$$

$$S_{it}^{U} - S_{it-1}^{U} \geq 0 \qquad \forall\ 1 \leq i \leq sub,\ 1 \leq t \leq T,\ t \neq F_i+1 \qquad (3.39)$$

$$S_{it}^{L} - S_{it-1}^{L} \leq 0 \qquad \forall\ 1 \leq i \leq sub,\ 1 \leq t \leq T,\ t \neq F_i+1 \qquad (3.40)$$

$$S_{(i+1)t}^{L} - S_{it}^{U} \geq 0 \qquad \forall\ 1 \leq i < sub,\ 1 \leq t \leq T \qquad (3.41)$$

$$S_{it}^{U} - S_{it}^{L} \leq 8 \qquad \forall\ 1 \leq i \leq sub,\ 1 \leq t \leq T \qquad (3.42)$$

$$S_{it}^{U} \geq 0 \quad S_{it}^{L} \geq 0 \qquad \forall\ 1 \leq i \leq sub,\ 1 \leq t \leq T \qquad (3.43)$$

$$S_{it}^{L}, S_{it}^{U} \in \{Positive\ Integer\} \qquad \forall\ 1 \leq i \leq sub,\ 1 \leq t \leq T \qquad (3.44)$$

Constraint (3.35) captures the maximum slot occupation by the whole block. Constraint (3.36) ensures that the slots that occupied sub-block *i* during shift *t* are larger than the total space needed by containers stored in the sub-block. In our study, the 40ft containers need two consecutive slots for storage, and the space reserved for them can also store 20ft containers. Therefore we use constraints (3.37) to make sure that the incoming 40ft containers will always have space to store without reshuffles. Constraint (3.38) is then used to ensure that the pairs of slots needed by 40ft containers are included in the range occupied by the corresponding sub-block. Constraints (3.39) and (3.40) ensure that the space occupied by a sub-block at shift *t* will always be covered by the same sub-block at the following shift before the completion of the loading procedure. Constraint (3.41) ensures that a space unit cannot be occupied by different sub-blocks at the same time. The maximum number of slots allowed in a sub-block is guaranteed in Constraint (3.42). Constraints (3.43) and (3.44) are non-negativity and integrality restrictions.

The MBS model can be solved directly in a short time. The size of sharing and non-sharing space can be calculated with the following two equations.

The size of sharing space between sub-blocks *i* and *i+1* is

$$S_{i(i+1)} = \max_t S_{it}^U - \min_t S_{(i+1)t}^L \tag{3.45}$$

The size of non-sharing space for sub-block *i* is

$$S_i = \min_t S_{(i+1)t}^L - \max_t S_{(i-1)t}^U \tag{3.46}$$

3.3 Numerical experiments

In this section, the solution procedure is implemented in C++ and ran on the same computer as that for the static yard template problem (Pentium IV computer, CPU: 2.4 GHz, Memory: 512M). In all the numerical examples, the general scale of the problem is 64-block, 21-vessel and 21-shift.

3.3.1 Experiment descriptions

In order to test the robustness of our results and algorithms, we have generated different realizations of the workload based on the triangular pattern shown in Figure 3.2, and we have also varied the overall workload per week to test out different port throughput levels.

The “even pattern” and “wave pattern” are used to test two different situations of the workload. For there “even pattern”, there are no major variations in the workload among different shifts. For the “wave pattern”, the workload is highly uneven, i.e., in some shifts, the workload is relatively heavy; while in other shifts, the workload is relatively light. The “wave pattern” can be seen as the worst case scenario. Meanwhile, the workload for each vessel still follows the triangular form.

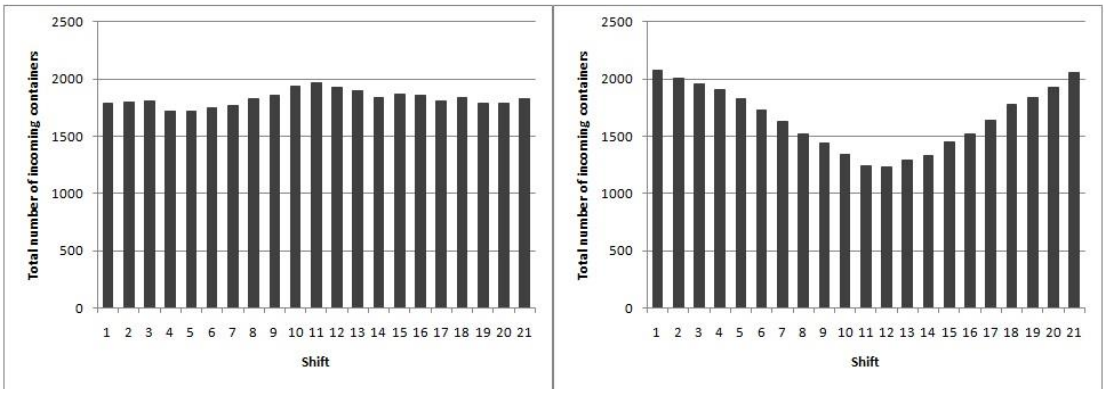

Figure 3.10 Input data with “even pattern” (left) and “wave pattern” (right)

For both “even pattern” and “wave pattern” scenarios, we first vary the total workload that is coming to the port. The workload level is the total volume of containers coming during the whole planning horizon (one week), which varies from 46,000 TEUs/week to 80,000 TEUs/week (50% to 85% of the original space given). Then for each workload level, we randomly generate ten different scenarios. Even through the general workload of a vessel still follows the triangular distribution, the parameters for each vessel may vary because of uncertainty. For example, the peak of container arrivals can vary from 3 to 10 shifts before the loading procedure. In addition, the volume of incoming containers in each shift may vary ±10% from the parameter set by the triangular distribution. In all, we have run more than 700 randomly generated scenarios.

3.3.2 Experiment results

The performance of “fixed sharing space” and “variable sharing space” for space-sharing yard template planning is presented in Figures 3.11 and 3.12 respectively. The land productivity achieved by the static yard template (which is without sharing) is also included in the figures and by definition, if we can find a feasible container allocation for a given workload level, the land productivity will be just equal to the total incoming workload divided by the original space given. For the space-sharing yard template planning, we will use the “fixed sharing space” and “variable sharing space” methods to find the sharing space, which eventually reduces the original space given, and so the land productivity increases in these two cases.

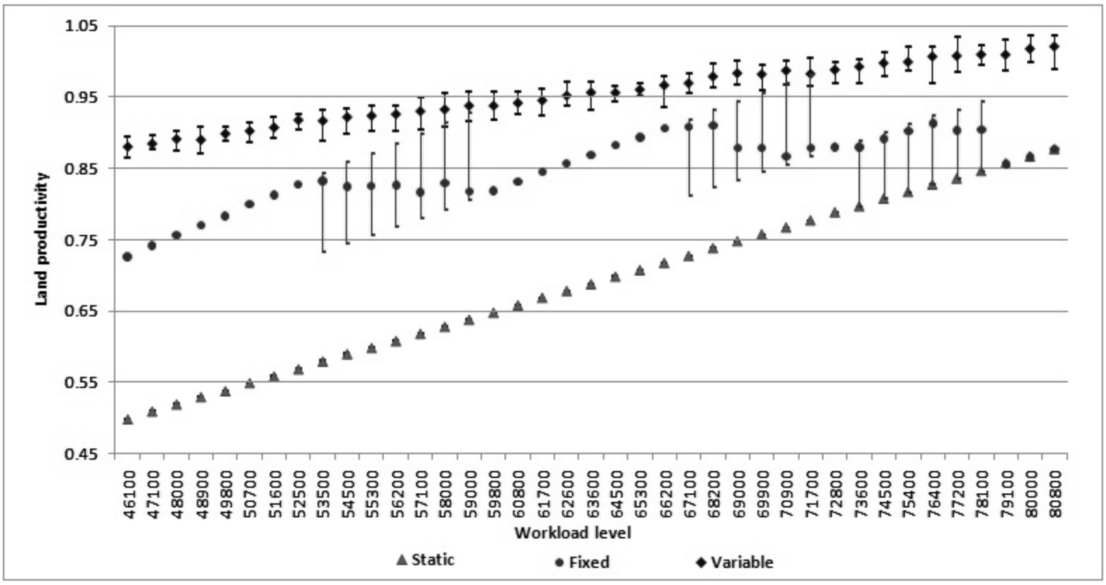

Figure 3.11 Results from “even pattern” of input data

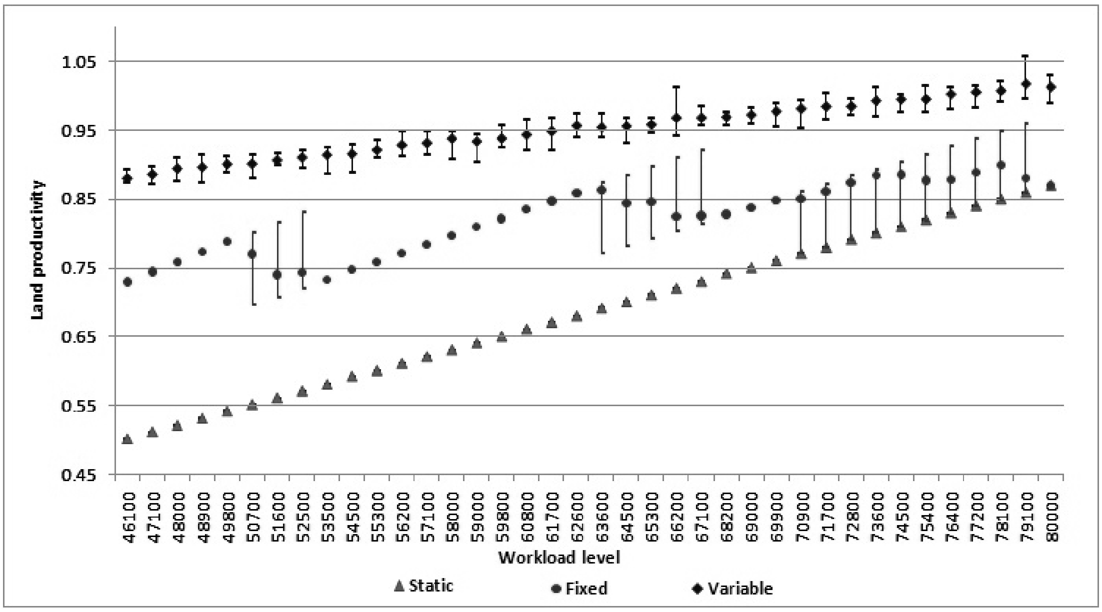

Figure 3.12 Results from “wave pattern” of input data

Figures 3.11 and 3.12 show the land productivity at different workload levels. For each workload level, we generate 10 sets of random input data. The highest, the middle and the lowest points for each workload level represent respectively the maximum, the average and the minimum values based on the 10 sets of runs. For most of the cases, the running time is within minutes, but for some worst cases, the running time can be hours and this is mainly caused by the high-low workload constraints.

Comparing the land productivity achieved by space-sharing yard template and static yard template under different workload levels, the following results can be observed.

Both the proposed methods can improve the land productivity when compared with the static model (Han et al. 2008). Note that for the consignment strategy, the static yard template planning can never achieve a land productivity exceeding a value of 1. However, the space-sharing yard template can achieve land productivity greater than 1 because each unit of storage space can accommodate up to 2 different types of containers while satisfying the constraints of consignment.

Among the two methods, the “variable sharing space method” performs better. Firstly, its land productivity dominates the other. Secondly, it has lower variability. The reason is that the “variable sharing space method” is able to have different sizes of sharing space for different sub-blocks, while the “fixed sharing space method” does not have that flexibility.

Moreover, the “variable sharing space method” is more robust. We conduct statistical tests to show whether land productivity is affected by the different patterns of workload. The results indicate that for the “fixed sharing space method”, we can achieve higher land productivity for the “even pattern” workload compared with the “wave pattern” workload. On the other hand, for the “variable sharing space method”, we cannot show that there is a significant difference in the land productivity between the two patterns, and this shows the robustness of the approach. The results for the fixed sharing space are not surprising. This is because for the wave pattern, the workload for the shifts is highly uneven, and therefore the space has to be catered for the worst case. Fixing a common size of the sharing space across the blocks will limit its flexibility in handling this uneven workload. On the other hand, for the variable sharing space, it has the flexibility to adjust, and for those with high workload, the sharing space is less, but for those with low workload, the sharing space can be more.

On the other hand, the YTG model proposed in this thesis performs better than the algorithm in Han et al. (2008). In terms of the computation time, it is comparable to Han's algorithm, but in terms of the solution quality, it is better. In our numerical example for the static yard template planning problem, Han's algorithm cannot provide feasible solutions at higher workload levels. For example at the workload level of 80,800 TEUs/week for the "even pattern" and 80,000 TEUs/week for the "wave pattern", Han's algorithm cannot provide a feasible template while the YTG model is able to provide a feasible solution.

3.4 Conclusions

In this chapter, an actual problem faced by a leading transshipment port operator is studied. Currently, the port operator uses a consignment strategy to reduce the reshuffling level for high equipment productivity. However, this strategy leads to under-utilization of storage space, due to pre-reservation of the maximum space needed during the whole planning horizon. With the increasing volume of transshipment container handling, the scarcity of storage space is urging new strategies to balance the equipment productivity and the land usage. A novel approach named partial space-sharing strategy is studied in this chapter to improve the land utilization while retaining the advantage of consignment strategy.

A framework which integrates space reservation and workload assignment is developed to solve the space-sharing yard template problem. In this framework, two different approaches are proposed to determine the sharing space namely, "fixed sharing space method" and "variable sharing space method". The numerical experiments show that, both the proposed methods can improve the land productivity under different port throughput levels compared to the static yard planning method which do not allow the sharing of space. The partial space-sharing strategy can even achieve a land productivity exceeding a value of 1, which is impossible for a yard template without sharing. Among the two methods, the variable sharing space method performs better both in land productivity and robustness.

Chapter 4. The Flexible Space-sharing Strategy

4.1 Problem description

In the previous chapter, a "partial space-sharing strategy" is proposed to improve the space utilization while retaining the advantage of consignment. Although the partial space-sharing strategy outperforms the non-sharing strategy, it has some limitations. Firstly, in the partial space-sharing strategy, there is always a clear boundary between the spaces reserved for different destination vessels. The performance of the partial space-sharing strategy depends on the size of sharing and non-sharing spaces. Secondly, once the size of sharing and non-sharing spaces is fixed at the planning stage, it will limit the flexibility of space allocation during operation.

To address this challenge while retaining the feature of the consignment, we propose a new approach named the "flexible space-sharing strategy". The idea is that the container space can be shared by two different vessels as long as their containers do not occupy the space at the same time. When less space is needed by one vessel, more space can be allocated to another vessel. In this way, the space allocated to each vessel can vary according to the amount of containers stored in the yard. The mechanism of this strategy is described as follows.

(1) Each sub-block is reserved for two different destination vessels, while two adjacent sub-blocks have one vessel in common. (The reservation of sub-blocks for vessels is known as the "yard template", as shown in Figure 4.1.)

(2) In each sub-block, one vessel fills from the left corner with incoming containers, while the other vessel fills from the right corner.

(3) When a vessel is common for two adjacent sub-blocks, the containers to this vessel fills one sub-block from the right corner and the other sub-block from the left corner as shown in Figure 4.1.

(V1)	(V2)	(V3)	(V4)	(V5)
S1	S2	S3	S4	S5

(V1, V2)	(V2, V3)	(V3, V4)	(V4, V5)	(V5, V6)
S1	S2	S3	S4	S5

Figure 4.1 Yard template for non-sharing strategy (up) and flexible space-sharing strategy (down)

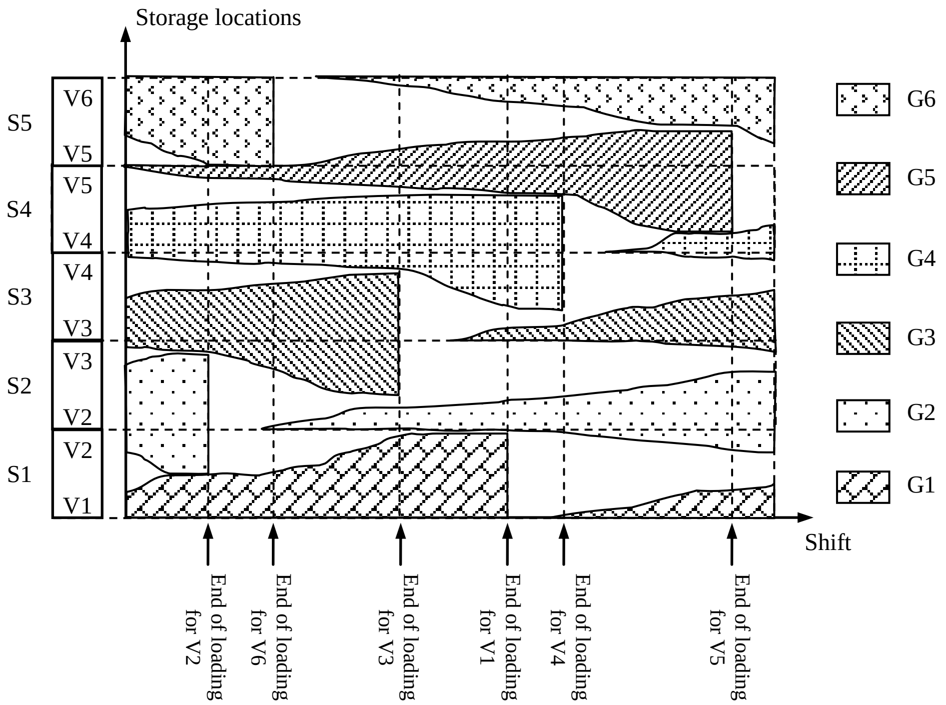

Figure 4.2 An example of space occupation under flexible space-sharing strategy

Under this strategy, the space occupation by each vessel in a block can be shown with the example in Figure 4.2. The containers to the same vessel in two adjacent sub-blocks form a cluster named the “container group”. As a block is managed in 5 sub-blocks, there will be 6 container groups (G1- G6) corresponding to the 6 vessels in each block. A yard crane will be dedicated to a container group during loading for a faster vessel turnaround time. To avoid interference between loading and discharging, if a container group is loading, the neighboring groups are not allowed to have any loading or discharging jobs.

When planning the space allocation, the yard template is given in advance. The main concern of space allocation is to decide where to assign the incoming containers, so that they can be handled in the limited storage space taking into account yard cranes and prime movers. A container assignment plan is called implementable, if it does not violate the following two conditions.

- The number of slots in a sub-block is called the “space capacity”. In any shift, the number of containers stored in a sub-block shall be no more than the “space capacity”, where a FEU (40-foot equivalent unit) is counted as two TEUs.
- The number of containers a yard crane can handle in the required loading period is called the “loading capacity”. The total number of containers assigned to a container group shall be less than the “loading capacity”.

For this study, we focus on how to find an implementable plan for deterministic incoming containers during the planning horizon. However, different realizations of incoming containers will be used in the numerical experiments to test the flexibility of the strategy. Besides, the number of yard cranes deployed will be limited to control the operational cost.

4.2 Model formulation

A mixed integer programming model will be formulated to find a container assignment plan based on the flexible space-sharing strategy. The number of incoming containers to each vessel per shift is called the “workload pattern”. The following assumptions are made when formulating the container assignment model.

- The workload pattern is deterministic and will repeat weekly.
- The planning horizon of the model is 21 shifts in a wraparound manner, as the workload pattern is repeated weekly.
- To avoid additional container handling effort, a container will stay in the same storage location until being loaded.
- As a sub-block consists of many stacks, it is possible to have a mixture of TEU (20-foot equivalent unit) and FEU (40-foot equivalent unit) in a sub-block. For simplicity, we assume the two types of containers can be stored in the same sub-block. However, the model can be easily modified to store TEU and FEU in separate sub-blocks.
- A yard crane cannot change to another block during a shift. This is to guarantee that the proper number of yard cranes is assigned to a specific block at the planning phase.

The model parameters are as follows:

B_k The set of sub-blocks that belong to block k, $1 \le k \le K$.

CC The yard crane handling capacity in terms of container moves per shift, which is 100 in this model.

CS The space capacity of each sub-block in terms of TEUs, which is 240 (5 tiers×6 rows×8 bays) in this model.

C_t The “yard crane limit” is the total number of yard cranes allowed for discharging jobs during shift t across the whole yard template, $1 \le t \le T$.

E_j The maximum number of shifts allowed for loading the containers to vessel *j*, $1 \le j \le J$.

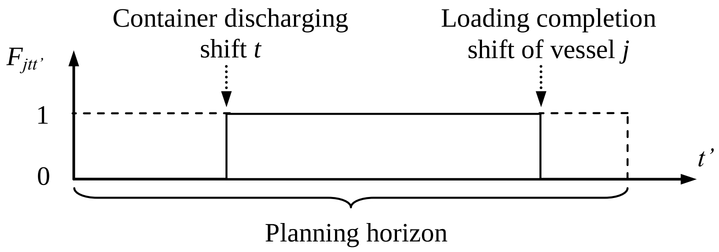

Figure 4.3 The staying period of a container

$F_{jtt'}$ The parameter indicates whether a container, which is discharged in shift *t* and to be loaded onto vessel *j*, will still be present in the yard in shift *t'*. The value is decided according to the discharging shift *t* of the container and the loading completion shift of its destination vessel *j*, as shown in Figure 4.3.

= 1, if the container will be present in shift *t'*, $1 \le j \le J$, $1 \le t \le T$, $1 \le t' \le T$.

= 0, otherwise, $1 \le j \le J$, $1 \le t \le T$, $1 \le t' \le T$.

G Total number of container groups in one block, which is 6 in this model.

HL The lowest value that a high workload can take, which is 50 in this model.

HU The highest value that a high workload can take, which is 100 in this model.

I Total number of sub-blocks in consideration.

J Total number of vessels in consideration.

K Total number of blocks in consideration.

L_{jt} = 1, if vessel *j* is in loading procedure during shift *t*, $1 \le j \le J$, $1 \le t \le T$.

= 0, otherwise, $1 \le j \le J$, $1 \le t \le T$.

LL The lowest value that a low workload can take, which is 0 in this model.

LU The highest value that a low workload can take, which is 20 in this model.

M A sufficiently large positive value.

M_k The maximum number of yard cranes allowed in block *k*, $1 \le k \le K$.

N_i The set of sub-blocks that are neighbors of sub-block *i*, $1 \le i \le I$.

NL_{kt} Number of yard cranes deployed for loading jobs in block *k* during shift *t*, which is given by the yard template, $1 \le k \le K$, $1 \le t \le T$.

T Total number of shifts in consideration.

WX_{jt} The number of 20ft containers arriving at the terminal in shift *t* and will be loaded onto vessel *j*, for a particular workload pattern. $1 \le j \le J$, $1 \le t \le T$.

WY_{jt} The number of 40ft containers arriving at the terminal in shift *t* and will be loaded onto vessel *j*, for a particular workload pattern. $1 \le j \le J$, $1 \le t \le T$.

Z_{ij}^1 The parameter indicates the reservation of the left corner of a sub-block.

= 1, if the containers to vessel *j* fill from the left corner of sub-block *i*, $1 \le i \le I$, $1 \le j \le J$.

= 0, otherwise, $1 \le i \le I$, $1 \le j \le J$.

Z_{ij}^2 The parameter indicates the reservation of the right corner of a sub-block.

= 1, if the containers to vessel *j* fill from the right corner of sub-block *i*, $1 \le i \le I$, $1 \le j \le J$.

= 0, otherwise, $1 \le i \le I$, $1 \le j \le J$.

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

The decision variables are as follows:

d_{kt} The number of yard cranes allocated to block k in shift t for discharging containers, $1 \le k \le K$, $1 \le t \le T$.

h_{it} = 1, if the amount of incoming containers allocated to sub-block i in shift t is high, that is, $HL \le x_{it}^1 + x_{it}^2 + y_{it}^1 + y_{it}^2 \le HU$, $1 \le i \le I$, $1 \le t \le T$.

= 0, if the amount of incoming containers allocated to sub-block i in shift t is low, that is, $LL \le x_{it}^1 + x_{it}^2 + y_{it}^1 + y_{it}^2 \le LU$, $1 \le i \le I$, $1 \le t \le T$.

u_{it} The violation of space capacity for sub-block i in shift t, $1 \le i \le I$, $1 \le t \le T$.

v_{kg} The violation of loading capacity for container group g in block k, $1 \le k \le K$, $1 \le g \le G$.

x_{it}^1 The number of 20ft containers that are assigned to the left corner of sub-block i in shift t, $1 \le i \le I$, $1 \le t \le T$.

x_{it}^2 The number of 20ft containers that are assigned to the right corner of sub-block i in shift t, $1 \le i \le I$, $1 \le t \le T$.

y_{it}^1 The number of 40ft containers that are assigned to the left corner of sub-block i in shift t, $1 \le i \le I$, $1 \le t \le T$.

y_{it}^2 The number of 40ft containers that are assigned to the right corner of sub-block i in shift t, $1 \le i \le I$, $1 \le t \le T$.

Using the "flexible space-sharing strategy", the container assignment problem (CAP) can be formulated as follows. To find an implementable plan, the objective is to minimize the violations of space capacity and loading capacity during the planning horizon.

$$\text{(CAP)}\ \ Min\ \ w = \sum_{t=1}^{T}\sum_{i=1}^{I} u_{it} + \sum_{k=1}^{K}\sum_{g=1}^{G} v_{kg} \tag{4.1}$$

Subject to:

$$\sum_{i=1}^{I} x_{it}^{1} \times Z_{ij}^{1} + \sum_{i=1}^{I} x_{it}^{2} \times Z_{ij}^{2} = WX_{jt} \qquad \forall\ 1 \le j \le J,\ 1 \le t \le T \tag{4.2}$$

$$\sum_{i=1}^{I} y_{it}^{1} \times Z_{ij}^{1} + \sum_{i=1}^{I} y_{it}^{2} \times Z_{ij}^{2} = WY_{jt} \qquad \forall\ 1 \le j \le J,\ 1 \le t \le T \tag{4.3}$$

$$\sum_{i \in B_k} (x_{it}^{1} + y_{it}^{1} + x_{it}^{2} + y_{it}^{2}) \le CC \times d_{kt} \qquad \forall\ 1 \le k \le K,\ 1 \le t \le T \tag{4.4}$$

$$\sum_{k=1}^{K} d_{kt} \le C_t \qquad \forall\ 1 \le t \le T \tag{4.5}$$

$$d_{kt} + NL_{kt} \le M_k \qquad \forall\ 1 \le k \le K,\ 1 \le t \le T \tag{4.6}$$

$$HL + (LL - HL)(1 - h_{it}) \le x_{it}^{1} + y_{it}^{1} + x_{it}^{2} + y_{it}^{2} \le LU + (HU - LU)h_{it}$$

$$\forall\ 1 \le i \le I,\ 1 \le t \le T \tag{4.7}$$

$$\sum_{i' \in N_i \cup \{i\}} h_{i't} \le 1 \qquad \forall\ 1 \le i \le I,\ 1 \le t \le T \tag{4.8}$$

$$\left(x_{it}^{2} + y_{it}^{2}\right) + \sum_{i' \in N_i} \left(x_{i't}^{1} + y_{i't}^{1}\right) \le M\left(1 - \sum_{j=1}^{J} Z_{ij}^{1} L_{jt}\right) \qquad \forall\ 1 \le i \le I,\ 1 \le t \le T \tag{4.9}$$

$$\left(x_{it}^{1} + y_{it}^{1}\right) + \sum_{i' \in N_i} \left(x_{i't}^{2} + y_{i't}^{2}\right) \le M\left(1 - \sum_{j=1}^{J} Z_{ij}^{2} L_{jt}\right) \qquad \forall\ 1 \le i \le I,\ 1 \le t \le T \tag{4.10}$$

$$\sum_{t=1}^{T}\left[(x_{it}^{1} + 2 \times y_{it}^{1}) \times \sum_{j=1}^{J} Z_{ij}^{1} F_{jtt'} + (x_{it}^{2} + 2 \times y_{it}^{2}) \times \sum_{j=1}^{J} Z_{ij}^{2} F_{jtt'}\right] - u_{it'} \le CS$$

$$\forall\ 1 \le i \le I,\ 1 \le t' \le T \qquad (4.11)$$

$$\sum_{t=1}^{T}(x_{it}^{1} + y_{it}^{1}) - v_{kg} \le \sum_{j=1}^{J} Z_{ij}^{1} E_j \times CC \qquad \forall\ 1 \le k \le K,\ g{=}1,\ i = 5\times(k-1)+g \qquad (4.12)$$

$$\sum_{t=1}^{T}\left[x_{(i-1)t}^{2} + y_{(i-1)t}^{2} + x_{it}^{1} + y_{it}^{1}\right] - v_{kg} \le \sum_{j=1}^{J} Z_{ij}^{1} E_j \times CC$$

$$\forall\ 1 \le k \le K,\ 2 \le g \le G\text{-}1,\ i = 5\times(k-1)+g \qquad (4.13)$$

$$\sum_{t=1}^{T}(x_{it}^{2} + y_{it}^{2}) - v_{kg} \le \sum_{j=1}^{J} Z_{ij}^{2} E_j \times CC \qquad \forall\ 1 \le k \le K,\ g{=}G,\ i = 5\times k \qquad (4.14)$$

$$x_{it}^{1}, y_{it}^{1}, x_{it}^{2}, y_{it}^{2}, u_{it} \ge 0 \qquad \forall\ 1 \le i \le I,\ 1 \le t \le T \qquad (4.15)$$

$$v_{kg} \ge 0 \qquad \forall\ 1 \le k \le K,\ 1 \le g \le G \qquad (4.16)$$

$$h_{it} \in \{\ 0, 1\ \} \qquad \forall\ 1 \le i \le I,\ 1 \le t \le T \qquad (4.17)$$

$$d_{kt} \in \{Positive\ Integer\} \qquad \forall\ 1 \le k \le K,\ 1 \le t \le T \qquad (4.18)$$

Constraints (4.2) and (4.3) ensure that all the containers arriving at the terminal in each shift for each vessel are assigned to sub-blocks in the storage yard. These containers can only be assigned to the sub-blocks which are reserved for their destination vessel. As the containers are handled one at a time, the number of containers can be used to indicate the workload for yard cranes and prime movers in the model. Constraint (4.4) gives the number of yard cranes deployed for discharging containers in block *k* during shift *t* based on the container assignment. Constraint (4.5) ensures that the total number of yard cranes deployed for discharging during shift *t* should be within the limits. The yard crane limit can be set according to the requirements of port operators. If the operators want to control the operational cost, the least number of yard cranes deployed in each shift from Han et al. (2008) can be used. Constraint (4.6) guarantees that the total number of yard cranes in each block during shift *t* should be no more than the maximum number of yard cranes allowed in a block. Constraint (4.7) ensures that the amount of incoming containers to sub-block *i* in shift *t* should be either high or low, to control the transportation activities. Constraint (4.8) guarantees that two neighboring sub-blocks cannot have high workload during the same shift. Both constraints (4.9) and (4.10) ensure that if a container group is loading, the neighboring groups will not have any incoming containers. Constraints (4.11) to (4.14) are used for space capacity and loading capacity under the "flexible space-sharing strategy". Constraint (4.11) captures the violations of space capacity. Due to the different discharging and loading times, the incoming containers will occupy the storage space for some periods. To ensure enough space for container storage at any period, the space occupation is checked in each shift of the planning horizon. The violations of loading capacity of each container group are captured by constraints (4.12) to (4.14). The container group at either end of a block is presented by constraints (4.12) and (4.14) respectively. For those not at the end of a block, each container group occupies space in two adjacent sub-blocks, which is presented by constraint (4.13). Constraints (4.15) to (4.18) are non-negativity and integrality restrictions.

4.3 Solution approaches

The CAP model presented in the previous section can be solved directly for small scale problems. But according to the experiments conducted in Lee et al. (2006), the container assignment problem under non-sharing strategy cannot be solved by CPLEX directly for a real scale problem of 64-block, 21-vessel and 21-shift. It is expected that the CAP model cannot be solved under the same scale, as it has much more decision variables and constraints. Since the CAP model is formulated for a planning horizon of 21 shifts, it has a “block diagonal structure”. Constraints (4.11) to (4.14) are the coupling constraints. The remaining constraints can be divided into blocks, each containing decision variables in only one shift. Based on this, the CAP model can be solved one block at a time, which corresponds to the container assignment in one shift. The coupling constraints can be used to update the solutions for each shift, in order to get an implementable plan.

As shown in Figure 4.4, a search algorithm can be developed with two stages, namely the “initial stage” and the “conflict resolution stage”. In both stages, the container assignment plan is solved shift by shift with the decomposed container assignment problem (DCAP). The idea of DCAP is to solve the container assignment in only one shift while keeping the decision variables in the other 20 shifts fixed.

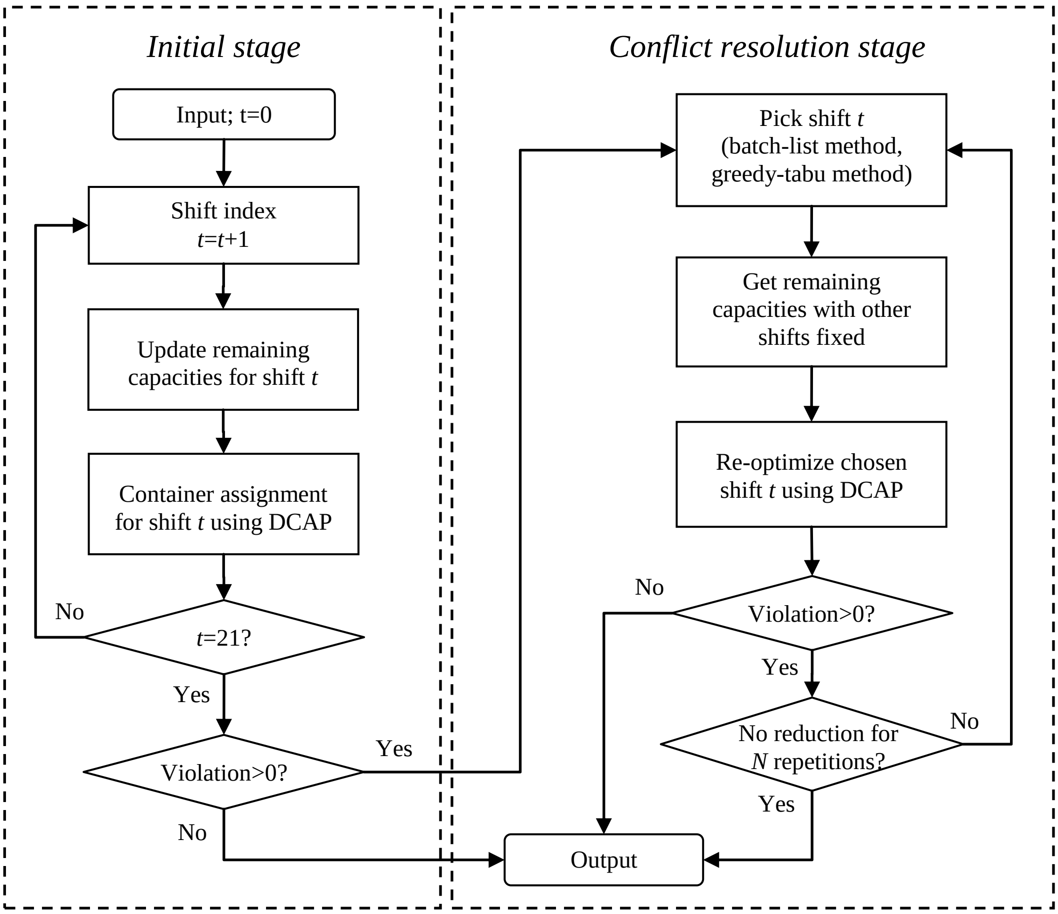

Figure 4.4 Main search algorithm

The "initial stage" is to get an initial solution during the planning horizon. The 21 shifts are solved in the increasing order of *t*. The remaining space capacity and loading capacity will be updated based on solutions of the solved shifts. After all 21 shifts are solved, the solution is inspected for the violation of space capacity and loading capacity. If there is no violation, an implementable plan is found; otherwise, the "conflict resolution stage" will be used to re-optimize the plan of container assignment.

In the "conflict resolution stage", the shifts are picked one at a time for re-optimization using DCAP while the remaining decision variables in the other 20 shifts are fixed. Several heuristics are introduced for shift-picking, namely the "batch-list method" and the "greedy-tabu method". The "conflict resolution stage" is repeated until an implementable plan is found or when there is no reduction of violations for *N* consecutive iterations.

4.3.1 Decomposed model for one shift

In both the "initial stage" and the "conflict resolution stage", the container assignment plan is solved shift by shift with the decomposed container assignment problem (DCAP). The idea is to fix the decision variables in 20 shifts, so that the block diagonal structure of the CAP model can be decomposed to one block. The coupling constraints can be modified accordingly for the decomposed model. For example, the coupling constraint (4.11) for space capacity can now be shown as equation (4.19), where the fixed decision variables have been moved to the right hand side.

$$
\begin{aligned}
&(x_{it}^1 + 2\times y_{it}^1)\times\sum_{j=1}^{J} Z_{ij}^1 F_{jtt'} + (x_{it}^2 + 2\times y_{it}^2)\times\sum_{j=1}^{J} Z_{ij}^2 F_{jtt'} - u_{it'} \\
&\le \mathrm{CS} - \sum_{t''\neq t}\left[(x_{it''}^1 + 2\times y_{it''}^1)\times\sum_{j=1}^{J} Z_{ij}^1 F_{jt''t'} + (x_{it''}^2 + 2\times y_{it''}^2)\times\sum_{j=1}^{J} Z_{ij}^2 F_{jt''t'}\right]
\end{aligned}
$$

$\forall\ 1 \le i \le I,\ 1 \le t' \le T$ (4.19)

The space occupied by incoming containers in sub-block *i* is measured in each shift *t'* during the planning horizon. The right hand side of equation (4.19) shows the remaining space capacity of sub-block *i* in shift *t'*, which can be replaced by $R_{it'}^{t}$ in the following equation. Since the value of remaining capacity in each shift *t'* depends on the un-fixed shift *t*, a superscript *t* is used in the notation.

$$R_{it'}^{t} = \mathrm{CS} - \sum_{t''\neq t}\left[(x_{it''}^{1} + 2\times y_{it''}^{1})\times\sum_{j=1}^{J} Z_{ij}^{1}F_{jt''t'} + (x_{it''}^{2} + 2\times y_{it''}^{2})\times\sum_{j=1}^{J} Z_{ij}^{2}F_{jt''t'}\right]$$

$\forall\ 1 \le i \le I,\ 1 \le t' \le T$ (4.20)

The coupling constraints (4.12) to (4.14) for the loading capacity can be modified similarly. The remaining loading capacity of container group *g* in block *k* can be noted as C_{kg}^{t}. The value of C_{kg}^{t} can be expressed with equations (4.21) to (4.23). Note that the remaining capacities can be negative, which implies that the violations of space and loading capacities already exist without the container assignment in shift *t*.

$$C_{kg}^{t} = \sum_{j=1}^{J} Z_{ij}^{1}E_{j}\times CC - \sum_{t'\neq t}(x_{it'}^{1} + y_{it'}^{1}) \qquad \forall\ 1 \le k \le K,\ g=1,\ i = 5\times(k-1)+g \quad (4.21)$$

$$C_{kg}^{t} = \sum_{j=1}^{J} Z_{ij}^{1}E_{j}\times CC - \sum_{t'\neq t}\left(x_{(i-1)t'}^{2} + y_{(i-1)t'}^{2} + x_{it'}^{1} + y_{it'}^{1}\right)$$

$\forall\ 1 \le k \le K,\ 2 \le g \le G\text{-}1,\ i = 5\times(k-1)+g$ (4.22)

$$C_{kg}^{t} = \sum_{j=1}^{J} Z_{ij}^{2}E_{j}\times CC - \sum_{t'\neq t}(x_{it'}^{2} + y_{it'}^{2}) \qquad \forall\ 1 \le k \le K,\ g=G,\ i = 5\times k \quad (4.23)$$

Based on these, the DCAP model can be formulated as follows. The objective of this model is to minimize the total violation of space capacity and loading capacity, which is consistent with the original problem. The DCAP model contains the set of constraints for one shift *t* and the modified coupling constraints (4.34) to (4.37).

$$\text{(DCAP)}\ \ Min\ \ w = \sum_{t=1}^{T}\sum_{i=1}^{I} u_{it} + \sum_{k=1}^{K}\sum_{g=1}^{G} v_{kg} \tag{4.24}$$

Subject to:

$$\sum_{i=1}^{I} x_{it}^{1} \times Z_{ij}^{1} + \sum_{i=1}^{I} x_{it}^{2} \times Z_{ij}^{2} = WX_{jt} \qquad \forall\ 1 \le j \le J \tag{4.25}$$

$$\sum_{i=1}^{I} y_{it}^{1} \times Z_{ij}^{1} + \sum_{i=1}^{I} y_{it}^{2} \times Z_{ij}^{2} = WY_{jt} \qquad \forall\ 1 \le j \le J \tag{4.26}$$

$$\sum_{i \in B_k} (x_{it}^{1} + y_{it}^{1} + x_{it}^{2} + y_{it}^{2}) \le CC \times d_{kt} \qquad \forall\ 1 \le k \le K \tag{4.27}$$

$$\sum_{k} d_{kt} \le C_t \tag{4.28}$$

$$d_{kt} + NL_{kt} \le M_k \qquad \forall\ 1 \le k \le K \tag{4.29}$$

$$HL + (LL - HL)(1 - h_{it}) \le x_{it}^{1} + y_{it}^{1} + x_{it}^{2} + y_{it}^{2} \le LU + (HU - LU)h_{it}$$

$$\forall\ 1 \le i \le I \tag{4.30}$$

$$\sum_{i' \in N_i \cup \{i\}} h_{i't} \le 1 \qquad \forall\ 1 \le i \le I \tag{4.31}$$

$$\left(x_{it}^{2} + y_{it}^{2}\right) + \sum_{i' \in N_i} \left(x_{i't}^{1} + y_{i't}^{1}\right) \le M\left(1 - \sum_{j=1}^{J} Z_{ij}^{1} L_{jt}\right) \qquad \forall\ 1 \le i \le I \tag{4.32}$$

$$\left(x_{it}^{1} + y_{it}^{1}\right) + \sum_{i' \in N_i} \left(x_{i't}^{2} + y_{i't}^{2}\right) \le M\left(1 - \sum_{j=1}^{J} Z_{ij}^{2} L_{jt}\right) \qquad \forall\ 1 \le i \le I \tag{4.33}$$

$$(x_{it}^{1} + 2 \times y_{it}^{1}) \times \sum_{j=1}^{J} Z_{ij}^{1} F_{jtt'} + (x_{it}^{2} + 2 \times y_{it}^{2}) \times \sum_{j=1}^{J} Z_{ij}^{2} F_{jtt'} - u_{it'} \le R_{it'}^{t}$$

$$\forall\ 1 \le i \le I,\ 1 \le t' \le T \tag{4.34}$$

$$x_{it}^{1} + y_{it}^{1} - v_{kg} \le C_{kg}^{t} \qquad \forall\, 1 \le k \le K,\ g=1,\ i = 5\times(k-1)+g \quad (4.35)$$

$$x_{(i-1)t}^{2} + y_{(i-1)t}^{2} + x_{it}^{1} + y_{it}^{1} - v_{kg} \le C_{kg}^{t} \qquad \forall\, 1 \le k \le K,\ 2 \le g \le G\text{-}1,\ i = 5\times(k-1)+g \quad (4.36)$$

$$x_{it}^{2} + y_{it}^{2} - v_{kg} \le C_{kg}^{t} \qquad \forall\, 1 \le k \le K,\ g=G,\ i = 5\times k \quad (4.37)$$

$$x_{it}^{1}, y_{it}^{1}, x_{it}^{2}, y_{it}^{2}, u_{it} \ge 0 \qquad \forall\, 1 \le i \le I \quad (4.38)$$

$$v_{kg} \ge 0 \qquad \forall\, 1 \le k \le K,\ 1 \le g \le G \quad (4.39)$$

$$h_{it} \in \{\,0, 1\,\} \qquad \forall\, 1 \le i \le I \quad (4.40)$$

$$d_{kt} \in \{Positive\ Integer\} \qquad \forall\, 1 \le k \le K \quad (4.41)$$

When using the DCAP model in the “initial stage”, the 21 shifts are solved one by one in the increasing order of *t*. For the “conflict resolution stage”, one shift shall be chosen from the 21 shifts to reduce the violations. Several shift-picking methods will be introduced in the next section.

4.3.2 Shift-picking methods

If the initial solution is not implementable, the “conflict resolution stage” will be used to re-optimize the container assignment. The shifts in the planning horizon will be re-optimized one at a time to reduce the violations. Two different methods can be used to generate the shift-picking sequence, namely the “batch-list method” and the “greedy-tabu method”. The first method keeps trying out the 21 shifts in batches until the termination of the main algorithm. The second method picks the shifts according to the performance of the solution. The shift causing most violations in each iteration will be chosen to be re-optimized.

4.3.2.1 Batch-list method

The main idea of the batch-list method is to keep trying the shifts one by one until the main search algorithm stops. To ensure that all the shifts in the planning horizon are picked, the shift-picking sequence is generated in batches of 21 shifts. The flow chart of this method can be shown in Figure 4.5. A batch-list will be initialized to contain all 21 shifts during the planning horizon. Once a shift is picked, it will be removed from the list. Note that when the batch-list becomes empty, all the shifts have been picked once. However, this does not mean the violations in the plan cannot be reduced further. According to equations (4.20) to (4.23), the remaining capacities for shift *t* are different if the decision variables in the other 20 shifts are changed. Therefore, we select the 21 shifts again to form a batch and repeat the solving procedure. Within a batch, the 21 shifts can be picked one by one in the increasing order of *t* (time sequence) or be picked in any random sequence (random pick).

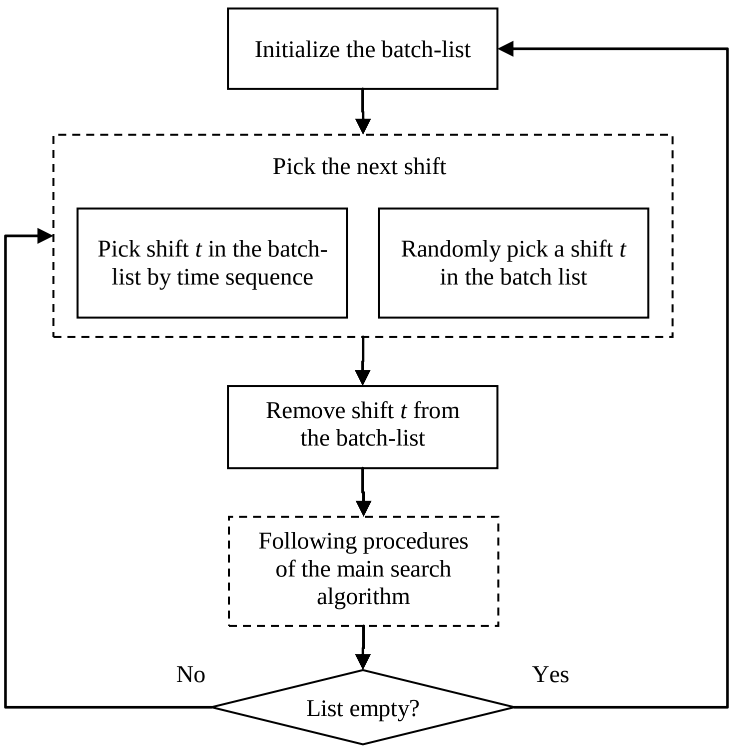

Figure 4.5 Batch-list method

4.3.2.2 Greedy-tabu method

In the previous method, the shift-picking sequence does not consider the performance of the container assignment plan. To improve the searching time, we propose to select the shift that causes the most violation. The violation caused by each shift can be measured by its marginal contribution, that is, taking the difference of total violations when all 21 shifts are considered and total violations (of 20 shifts) when the respective shift is removed. In each iteration, the total violations with 21 shifts are captured in the objective function (4.24) of the DCAP model, which can be denoted as w_0. The total violations without shift t can be calculated by adding the negative parts of the remaining capacities, as shown in equation (4.42).

$$w_t = \left| \sum_{i=1}^{I} \sum_{t'=1}^{T} \min\left\{R_{it'}^{t}, 0\right\} + \sum_{k=1}^{K} \sum_{g=1}^{G} \min\left\{C_{kg}^{t}, 0\right\} \right| \tag{4.42}$$

where $R_{it'}^{t}$ and C_{kg}^{t} are the remaining capacities in equations (4.20) to (4.23). The amount of violations caused by shift t can be denoted as P_t in equation (4.43).

$$P_t = w_0 - w_t \tag{4.43}$$

To avoid picking the same shift in consecutive iterations, the general steps of this shift-picking method can be as shown in Figure 4.6. After resolving the chosen shift, the container assignment plan will be updated. The shift that causes most violations is chosen to be re-optimized. When a shift is chosen, it will be recorded in a tabu-list to prevent from being picked. When the list is not full, the chosen shift will be recorded in the list directly; otherwise, the newly chosen shift will replace the shift which has been forbidden for the longest time in the list. In this way, the number of iterations to forbid a chosen shift equals to the length of the tabu-list. Later in the numerical experiments, we will find out the best length of the tabu-list.

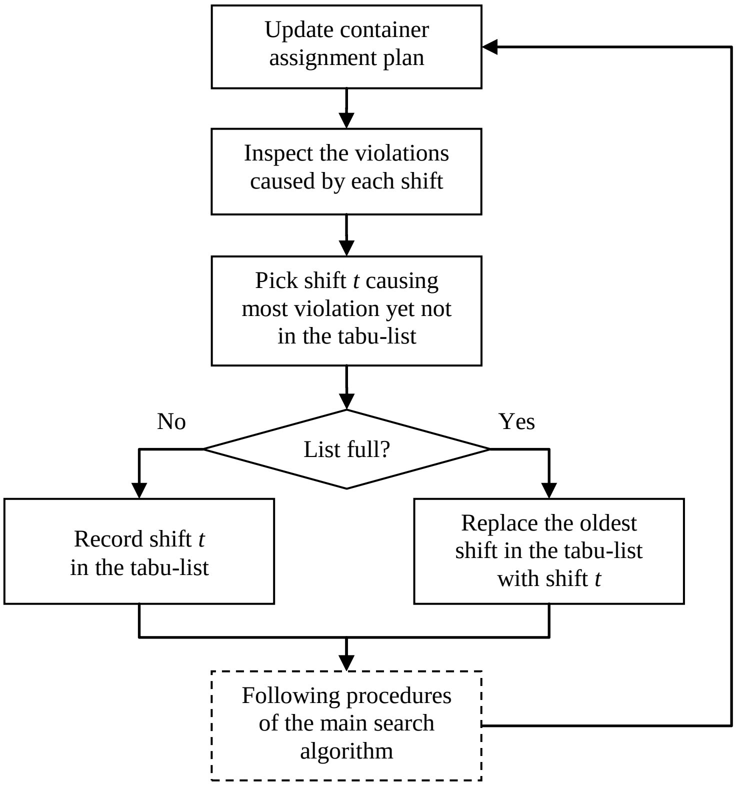

Figure 4.6 Greedy-tabu method

4.4 Numerical experiments

To test the performance of the solution approaches, the search algorithm with different shift-picking methods are implemented in C++ and each model was solved using CPLEX 11.2 with concert technology. The numerical experiments were run on Intel Core™2 computer (CPU: 2.66 GHz, 2.67 GHz; Memory: 4 GB). In all the numerical examples, the flexible space-sharing problem is solved under the scale of a real port problem (64-block, 21-vessel and 21-shift).

The performance of the storage strategies is affected by the scarcity of space for the incoming containers. In the experiments, the "land productivity" can be used to indicate the scarcity of space, which is defined as the ratio of "the total amount of TEUs during the planning horizon/the total number of slots in the storage yard". Here, each FEU among the incoming containers is counted as two TEUs, as it needs two slots in terms of space occupation. Higher land productivity indicates the higher scarcity of space. When land productivity is 1, the total volume of containers is equivalent to the storage space available. The experiment is carried out under different land productivity levels.

Under each land productivity level, the realization of incoming containers can be different. A realization is defined by the number of containers to each vessel per shift, which we call the “workload pattern”. According to the workload pattern provided by the port operator in Han et al. (2008), there are two important characteristics of the incoming containers. The characteristics are that the majority of containers arrive near to their departure time, while few containers come right after the vessel departs. Hence, it is a common practice for them to use a triangular workload profile to do the planning. However, the peak shifts of incoming containers and the number of containers in each shift can vary because of uncertainty. By varying these two factors, we can randomly generate different realizations of incoming containers under each land productivity level, as shown in Figure 4.7. This can be used to test the robustness of our algorithm.

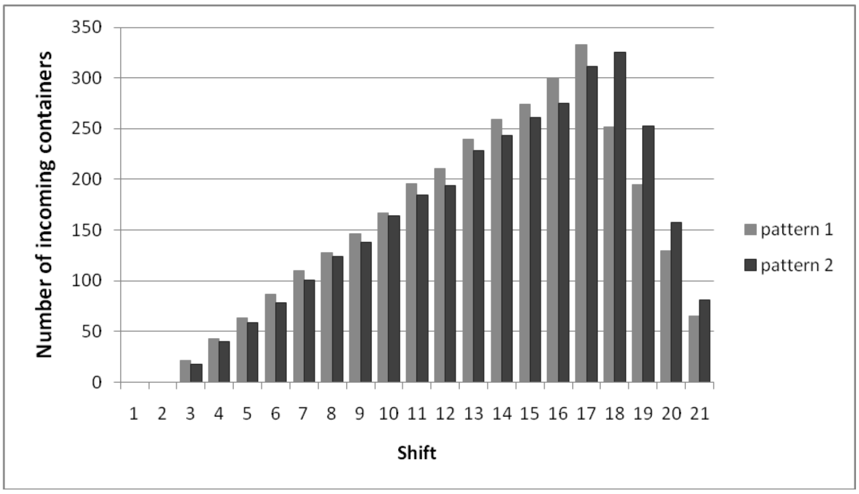

Figure 4.7 The example workload patterns for one vessel

4.4.1 Performance of flexible space-sharing strategy

In this section, the flexible space-sharing strategy is compared with the non-sharing strategy under common incoming containers during the planning horizon. For a fair comparison, the land productivity is varied from 1.00 to 1.19. Under each land productivity level, 10 different realizations are randomly generated to test the performance. The search algorithm presented in the previous section is used to find an initial plan and the final implementable plan for the “flexible space-sharing strategy”. Note that the algorithm can also be used for the “non-sharing strategy”, by reserving each sub-block for only one vessel in the yard template.

Table 4.1 shows the performance of both the “flexible space-sharing strategy” and the “non-sharing strategy”. The results are averaged from the 10 realizations under the same land productivity level. For the “non-sharing strategy”, the violations in the initial plan can be reduced to certain extent with the conflict resolution stage, but no implementable plan can be found. For the “flexible space-sharing strategy”, it is easier to plan the container assignment, as the initial plans under all land productivity levels have much less violations compared with the “non-sharing strategy”. When the land productivity level is below 1.16, our algorithm can find an implementable plan for all realizations. Even when the land productivity is above 1.17, the final plans using the “flexible space-sharing strategy” have much less violations than those using the “non-sharing strategy”. It can be concluded that the “flexible space-sharing strategy” can handle more containers than the “non-sharing strategy” within the same storage space. This can be attributed to the difference in space allocation between the two storage strategies. In the “non-sharing strategy”, the space is exclusively allocated to each vessel in advance, while much space is only occupied for a short period of time. In the "flexible space-sharing strategy", the space released by containers to one vessel after loading can be flexibly reused by containers to another vessel.

Table 4.1 Comparison of the two storage strategies

Land productivity	Non-sharing strategy		Flexible space-sharing strategy	
	Initial violations	Final violations	Initial violations	Final violations
1.00	6571	3433	228	0
1.01	8837	5593	294	0
1.02	10555	7420	686	0
1.03	13487	10204	844	0
1.04	16449	13572	1312	0
1.05	20106	17299	1626	0
1.06	22781	20120	1495	0
1.07	25518	22883	2252	0
1.08	29795	27102	2598	0
1.09	34320	31744	2789	0
1.10	37470	34957	3751	0
1.11	42018	39372	4669	0
1.12	45611	42925	4944	0
1.13	50709	47886	5718	0
1.14	54160	51397	6498	0
1.15	59666	56867	7604	0
1.16	64997	61969	8813	0
1.17	68950	65828	9879	6
1.18	73077	70000	10542	25
1.19	78143	74971	12791	45

4.4.2 Comparison of shift-picking methods

In the "conflict resolution stage", several shift-picking methods can be used, namely the "batch-list method (time sequence, TS)", the "batch-list method (random pick, RP)" and the "greedy-tabu method". After trying different tabu-list lengths for the "greedy-tabu method", the length around 12 turns out to be more robust and is used in this study.

To compare the performance of these methods, the land productivity is varied from 1.00 to 1.19. Ten different realizations are randomly generated under each land productivity level. The results show the three methods are about the same regarding the quality of container assignment plan, but their computation time is different. In Figure 4.8, the average computation time is calculated from the 10 realizations under the same land productivity. The "batch-list method (TS)" is used as the bench mark to normalize the results as the "percentage of computation time". Figure 4.8 shows the results for land productivity level below 1.16, where all shift-picking methods can find implementable solutions for each realization.

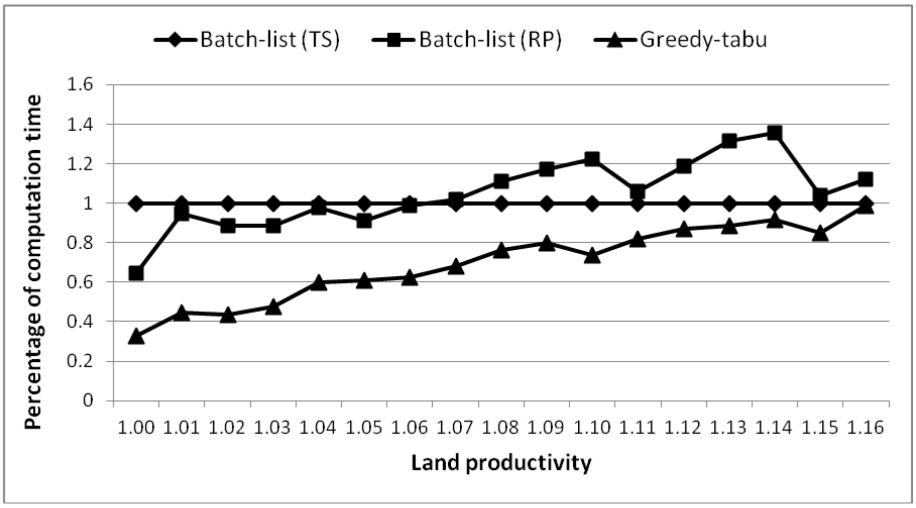

Figure 4.8 Comparison of shift-picking methods

It is observed that the "batch-list method (RP)" can sometimes use less time than the "batch-list method (TS)" under low land productivity levels, but it uses more time under higher land productivity levels. On the other hand, the "greedy-tabu method" always uses less time compared with the two batch-list methods. Thus, it is more efficient in finding an implementable plan. Note that the advantage of the "greedy-tabu method" tends to be reduced with the increase of land productivity level. An intuitive explanation is that the coupling constraints are more relaxed when land productivity is low, making it easier to re-optimize the container assignment plan.

Picking the shift that causes most violations will accelerate the search process. However, when land productivity is high, the coupling constraints become tight. It may not be easy for the shift that causes most violations to re-optimize its container assignment. The shifts that cause fewer violations need to be picked as well to reduce the violations, which reduces the advantage of the "greedy-tabu method".

4.4.3 Impact of the yard crane limit

In the CAP model, the container assignment is constrained by the "yard crane limit" (C_t, $1 \le t \le T$). This is to control the operational cost. In this section, we will show the impact of increasing yard crane limits in each shift on the total violations. The least number of yard cranes in each shift (Han et al. 2008) is used as the base value of the yard crane limit. We increase the base value by allowing 1, 2 and 3 additional yard cranes in each shift across the planning horizon. The experiments were carried out under the land productivity levels from 1.16 to 1.19, where 30 different realizations were randomly generated under each level. The performance is measured by the "percentage of zero violation", which is "the number of realizations where an implementable plan can be found/the total number of realizations tested". The higher percentage implies that it is easier to find an implementable plan. The results of the different yard crane limits can be combined in Figure 4.9, where the value 0 in the *x*-axis corresponds to the base value. The difficulty of container assignment increases when the land productivity level increases from 1.16 to 1.19.

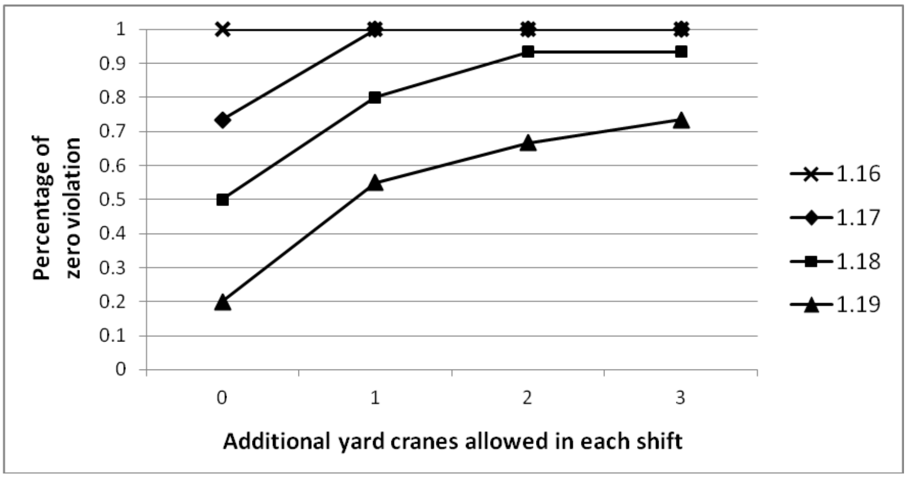

Figure 4.9 Impact of the yard crane limit

Comparing the results of different yard crane limits, the following conclusions can be made. With more yard cranes allowed in each shift, the percentage of zero violation is increasing. Therefore, more yard cranes will improve the container assignment within the limited space, which increases the land utilization. However, the marginal benefit of the additional yard crane is decreasing. This is because more yard cranes allow more choices for container assignment. But when the number of yard cranes keeps increasing, constraint (4.5) becomes less dominant compared with other constraints, which leads to the decreasing marginal benefit. For the overall benefit of the port, the operator should consider the trade-off between the land utilization and the operational cost.

4.4.4 Effect of the sub-block size

The configuration parameters of the storage yard section are provided by the port operator. Each sub-block consists of 8 consecutive bays. However, such sub-block size is chosen for the “non-sharing strategy” which is currently applied in the yard. For the “flexible space-sharing strategy”, it may not necessarily be the best choice. In this section, the sub-block size is increased from 4 bays to 8 bays to show the performance of the “flexible space-sharing strategy”. The land productivity level is varied from 1.0 to 1.26, while 30 different realizations are randomly generated under each level. Figure 4.10 shows the general trend with the high land productivity levels 1.22, 1.24 and 1.26.

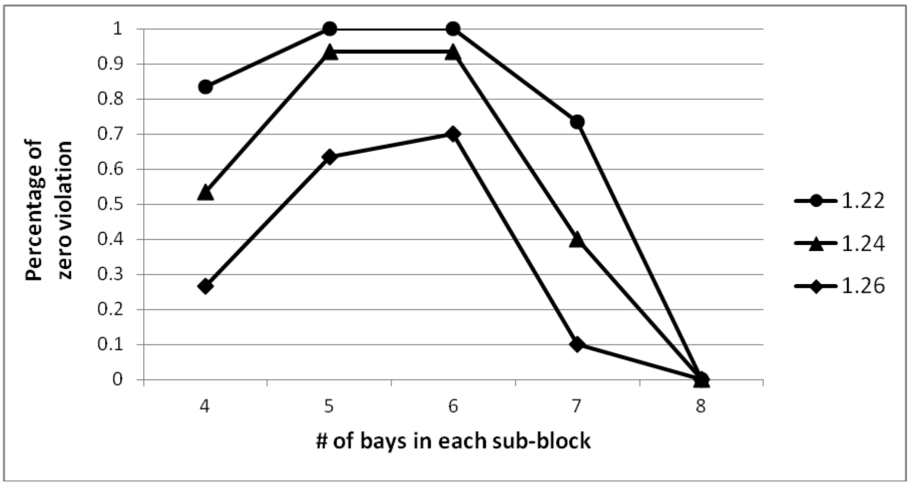

Figure 4.10 Effect of the sub-block size

It can be found that the sub-block size affects the percentage of zero violation under the same land productivity level. When the sub-block size increases from 4 to 6 bays, the percentage of zero violation is increasing. However, the percentage decreases, when the sub-block size increases from 6 to 8 bays. The main reason is that the sub-block size affects the flexibility of container assignment and the sharing space between two container groups. When the sub-block size increases from 4 to 6 bays, larger space capacity leads to more choices of container assignment. This improves the flexibility of container assignment, resulting in better performance. However, the size of each container group is limited by the “loading capacity”. When the sub-block size keeps increasing, the overlapping space between two container groups becomes less. This reduces the advantage of space-sharing, resulting in the decreasing trend when the sub-block size increases from 6 to 8 bays. Therefore, the port operator should choose the proper sub-block size to balance the space capacity and loading capacity.

4.5 Conclusions

To facilitate faster loading process, the "consignment strategy" is generally used in the transshipment ports, where containers going to the same destination vessel are stored together. Under this strategy, each storage location is dedicated to one destination vessel, resulting in much space only occupied for a short period of time. This leads to the under-utilization of the storage space. A "partial space-sharing strategy" is proposed in the previous chapter to improve the space utilization, but this strategy has some limitations during operation. To improve the space utilization while retaining the advantage of consignment, the "flexible space-sharing strategy" is proposed. The idea is that the container space can be shared by two different vessels as long as their containers do not occupy the space at the same time. This strategy allows each storage location to be shared by two vessels. By controlling where to stack the containers in the storage locations, the containers to each vessel are not mixed and the consignment feature can be preserved.

The container assignment problem (CAP) using this strategy is formulated as a mixed integer program. The model can be solved directly for small scale problems, while it cannot be solved for a real scale problem. Based on the "block diagonal structure" of the CAP model, a two-stage search algorithm incorporating MIP and heuristics is developed. The "initial stage" generates an initial solution shift by shift, while the "conflict resolution stage" reduces the violations in the solution to get an implementable plan. The numerical experiments show that the "flexible space-sharing strategy" can handle much more containers within the same storage space, compared with the "non-sharing strategy".

Chapter 5. Short-term Space Allocation

5.1 Problem description

In long-term planning, the space allocation can be decided with different methods proposed by Han et al. (2008), Jiang et al. (2012) and Jiang et al. (2013). The main purpose is to get a good yard template and the right number of resources to be deployed in each shift. However, the long-term space allocation is decided based on the expected container flow. During the operation, loading and unloading times can change because of vessel delays. The amount of containers discharged from each vessel is also different from week to week. The information of actual incoming containers is only known for a short period in advance. Thus, short-term planning is needed to decide the space allocation based on the latest information. Currently, the short-term space allocation is decided according to the experience of port operators and the rule of thumb, which do not fully capture the different aspects of yard operations. In this study, we develop systematic methods to decide the short-term space allocation, which considers prime movers, yard cranes and the storage space.

In short-term planning, the storage space is assigned to incoming containers with a given yard template. To improve the operation efficiency, the short-term planning shall take into account prime movers, yard cranes and the storage space. The considerations for these yard resources can be interpreted as the following four types of violations. Assuming the containers are handled one at a time, the number of containers assigned to each sub-block can be used to indicate the workload for prime movers and yard cranes.

- “Traffic violation”. As shown in Figure 5.1, there is a truck path for each block, while only one passing lane is shared between two blocks. When two sub-blocks are adjacent or using the same passing lane, the container handling activities in one sub-block will affect another. Such two sub-blocks are called “neighbors”. For example, sub-blocks 46, 48 and 52 are neighbors of sub-block 47. To prevent the potential congestion of prime movers, the total amount of workload in any two neighbors has a limit *U*. The “traffic violation” is defined as the amount of workload which exceeds the limit *U*.

- “Crane violation”. The short-term space allocation determines the amount of incoming containers assigned to each sub-block. This in turn determines the number of yard cranes deployed to handle the incoming containers across the whole yard. The “crane violation” is defined as the number of yard cranes which exceeds the least number of yard cranes to deploy according to the long-term plan. The crane violation leads to higher operational cost and shall be reduced in the short-term planning.
- “Loading violation”. The containers in the same container group are handled by the same yard crane during the loading process. A yard crane can only handle limited number of containers in one shift. To finish the loading jobs in required turnaround time of the destination vessel, the total number of containers assigned to a container group shall be no more than the “loading capacity” (number of containers per shift × number of loading shifts allowed by the vessel). The “loading violation” is defined as the amount of containers which exceeds the loading capacity in each container group.
- “Space violation”. A sub-block can only store a limited number of TEUs at the same time, which is called the “space capacity”. At any shift, the number of TEUs stored in each sub-block must be no more than the space capacity. The “space

violation" is defined as the number of TEUs which exceeds the space capacity. If the space violation occurs, buffer space is needed to store the excessive containers.

The effect of short-term space allocation on prime movers, yard cranes and the storage space can now be interpreted in terms of violations. In this study, we develop two systematic short-term methods to reduce the violations, namely the "greedy space allocation (GSA)" and the "space allocation considering the long-term plan (SALP)".

The GSA method plans the space allocation based on the short-term information, which includes the existing containers and the actual incoming containers. When the storage space is allocated to the incoming containers, there are containers already stored in the yard. The information of existing containers tells the amount of loading activities at each storage location across the yard. The space allocation to incoming containers determines the discharging activities, which must take into account of the loading activities to reduce the traffic violation. The assignment of incoming containers to each storage location with existing containers shall also reduce the loading violation and space violation. The incoming containers in future shifts are not considered.

On the other hand, the SALP method considers more information which includes the long-term space allocation plan. As the containers stay in the assigned sub-blocks until being loaded, the short-term space allocation impacts the future shifts in terms of loading violation and space violation. The long-term plan roughly tells how much space at each storage location needs to be allocated to incoming containers during the future shifts. If the short-term plan ignores the long-term plan, the decision of space allocation can be too greedy and cause more violations in the future. The loading violation and space violation can be measured based on the short-term information and the long-term plan. Thus, the SALP method aims to reduce not only the violations in the short-term, but also the loading violation and space violation in future shifts.

In the next section, MIP models will be formulated respectively for the GSA method and the SALP method. The short-term space allocation will also be evaluated under different storage strategies. Under each storage strategy, the incoming containers are assigned according to different rules, which will be considered in the MIP models.

5.2 Model formulation

Both the GSA method and the SALP method can be used to decide the short-term space allocation. In this section, mixed integer programming models are formulated for the two short-term planning methods respectively. The following assumptions are made when formulating the space allocation models.

- To avoid additional container handling effort, a container will stay in the same storage location until being loaded.
- Once a container is assigned to a storage location, the space will be occupied until all the containers to the same vessel are loaded.
- For simplicity, the loading activities of each vessel are equally distributed into each shift of the loading period.
- As a sub-block consists of many bays, it is possible to have a mixture of TEU (20-foot equivalent unit) and FEU (40-foot equivalent unit) in a sub-block. For simplicity, we assume the two types of containers can be stored in a sub-block. However, the model can be easily modified to store TEU and FEU in separate sub-blocks.
- A yard crane will not change block during a shift. This is to guarantee that the proper amount of yard cranes is assigned to a specific block at the planning phase.

Based on these assumptions, the MIP models are formulated for the GSA method and the SALP method respectively.

5.2.1 MIP model for the GSA method

The model parameters are as follows:

B_k The set of sub-blocks that belong to block k, $1 \le k \le K$.

CC The yard crane handling capacity in terms of container moves per shift, which is 100 in this model.

CS The space capacity of each sub-block in terms of TEUs, which is 240 (5 tiers×6 rows×8 bays) in this model.

D^d The total number of yard cranes allowed for discharging containers during the current shift.

D_k^l The number of yard cranes deployed for loading in block k during the current shift, $1 \le k \le K$.

E_j The number of shifts allowed for loading the containers to vessel j, $1 \le j \le J$.

G Total number of container groups in each block.

I Total number of sub-blocks in consideration.

J Total number of vessels in consideration.

K Total number of blocks in consideration.

L_j = 1, if vessel j is in loading procedure during the current shift, $1 \le j \le J$.

= 0, otherwise, $1 \le j \le J$.

M A sufficiently large positive value.

M_k The maximum number of yard cranes allowed in block k, $1 \le k \le K$.

N_i The set of sub-blocks that are neighbors of sub-block i, $1 \le i \le I$.

NL_k Number of yard cranes deployed for loading jobs in block k during the current shift, which is given by the yard template and the loading vessels, $1 \le k \le K$.

R_{i0}^{T1} The amount of 20ft containers stored in the left corner of sub-block i at the beginning of the current shift, $1 \le i \le I$.

R_{i0}^{T2} The amount of 20ft containers stored in the right corner of sub-block i at the beginning of the current shift, $1 \le i \le I$.

R_{i0}^{F1} The amount of 40ft containers stored in the left corner of sub-block i at the beginning of the current shift, $1 \le i \le I$.

R_{i0}^{F2} The amount of 40ft containers stored in the right corner of sub-block i at the beginning of the current shift, $1 \le i \le I$.

U The workload limit for two neighboring sub-blocks, which is 100 in this model.

WL_i The amount of loading containers in sub-block i during the current shift. $1 \le i \le I$.

WX_j The number of 20ft containers arriving at the terminal in the current shift and will be loaded onto vessel j. $1 \le j \le J$.

WY_j The number of 40ft containers arriving at the terminal in the current shift and will be loaded onto vessel j, $1 \le j \le J$.

Z_{ij}^{1} The parameter indicates the reservation of the left corner of a sub-block.

= 1, if the containers to vessel j fill from the left corner of sub-block i, $1 \le i \le I$, $1 \le j \le J$.

= 0, otherwise, $1 \le i \le I$, $1 \le j \le J$.

Z_{ij}^2 The parameter indicates the reservation of the right corner of a sub-block.

= 1, if the containers to vessel j fill from the right corner of sub-block i, $1 \le i \le I$, $1 \le j \le J$.

= 0, otherwise, $1 \le i \le I$, $1 \le j \le J$.

Note: Subscript i is for sub-block, j for vessel, k for block, g for container group.

The decision variables are as follows:

d_k The number of yard cranes allocated to block k for discharging containers in the current shift, $1 \le k \le K$.

r_i^{T1} The number of TEUs in the left corner of sub-block i at the end of the current shift. $1 \le i \le I$.

r_i^{T2} The number of TEUs in the right corner of sub-block i at the end of the current shift. $1 \le i \le I$.

r_i^{F1} The number of FEUs in the left corner of sub-block i at the end of the current shift. $1 \le i \le I$.

r_i^{F2} The number of FEUs in the left corner of sub-block i at the end of the current shift. $1 \le i \le I$.

$v_{ii'}^1$ The traffic violation between two neighboring sub-blocks i and i' in the current shift. $1 \le i \le I$, $1 \le i' \le I$.

v^2 The crane violation in the current shift.

v_{kg}^3 The loading violation for container group g in block k in the current shift. $1 \le k \le K$, $1 \le g \le G$.

v_i^4 The space violation in sub-block i in the current shift. $1 \le i \le I$.

x_i^1 The number of 20ft containers that are assigned to the left corner of sub-block i in the current shift, $1 \le i \le I$.

x_i^2 The number of 20ft containers that are assigned to the right corner of sub-block i in the current shift, $1 \le i \le I$.

y_i^1 The number of 40ft containers that are assigned to the left corner of sub-block i in the current shift, $1 \le i \le I$.

y_i^2 The number of 40ft containers that are assigned to the right corner of sub-block i in the current shift, $1 \le i \le I$.

The objective of the model is to minimize the four types of violations. All the violations in the objective function have been interpreted in the units of container moves. The first term is the traffic violation, where the weight is to remove the double counting of each sub-block. The second term is crane violation, where the container moves per shift for one yard crane is used to translate the crane violation into container moves. The third term is the loading violation, while the last term is the space violation. Since the space violation is considered as the last resort, a large enough number is used as the penalty weight.

$$Min\ w = \frac{1}{2}\sum_{i}\sum_{i' \in N_i} v_{ii'}^1 + CC \times v^2 + \sum_{k}\sum_{g} v_{kg}^3 + M \times \sum_{i} v_i^4 \qquad (5.1)$$

According to the yard template, the incoming containers can only be assigned to the sub-blocks reserved for their own destination vessel. In this model, two sizes of containers (TEU and FEU) are considered, which are described in constraints (5.2) and (5.3) respectively.

$$\sum_{i=1}^{I}(x_i^1 \times Z_{ij}^1) + \sum_{i=1}^{I}(x_i^2 \times Z_{ij}^2) = WX_j \qquad \forall\ 1 \le j \le J \qquad (5.2)$$

$$\sum_{i=1}^{I}(y_i^1 \times Z_{ij}^1) + \sum_{i=1}^{I}(y_i^2 \times Z_{ij}^2) = WY_j \qquad \forall\ 1 \le j \le J \qquad (5.3)$$

The two storage strategies have different forms of yard templates, which affect the rules of space allocation. For the "non-sharing strategy", each sub-block is dedicated to one destination vessel. For the "flexible space-sharing strategy", each sub-block is reserved for a pair of vessels. In the MIP model, the given yard template is described with the parameters Z_{ij}^1 and Z_{ij}^2. When Z_{ij}^1 equals to 1, the left corner of sub-block *i* is reserved by vessel *j*. When Z_{ij}^2 equals to 1, the right corner of sub-block *i* is reserved by vessel *j*. For non-sharing strategy, all Z_{ij}^2 is set to 0 so that each sub-block is dedicated to one destination vessel. The difference of the parameter setting for the two storage strategies can be shown in the following table.

Table 5.1 Parameter setting for different storage strategies

Storage strategies	Parameter setting		
Non-sharing	$\sum_{j=1}^{J} Z_{ij}^1 = 1$	$\forall\ 1 \le i \le I$	(5.4)
	$\sum_{j=1}^{J} Z_{ij}^2 = 0$	$\forall\ 1 \le i \le I$	(5.5)
Flexible space-sharing	$\sum_{j=1}^{J} Z_{ij}^1 = 1$	$\forall\ 1 \le i \le I$	(5.6)
	$\sum_{j=1}^{J} Z_{ij}^2 = 1$	$\forall\ 1 \le i \le I$	(5.7)

The potential congestion of prime movers is controlled in constraint (5.8). This constraint measures the total amount of loading and discharging activities in any two neighboring sub-blocks. The loading activities in sub-block *i* are the result of past decisions, which are presented with WL_i. When the amount of activities is larger than the workload limit, the overflow part is captured as the "traffic violation".

$$(x_i^1 + y_i^1 + x_i^2 + y_i^2 + WL_i) + (x_{i'}^1 + y_{i'}^1 + x_{i'}^2 + y_{i'}^2 + WL_{i'}) \le U + v_{ii'}^1$$

$$\forall\ 1 \le i \le I,\ i' \in N_i \qquad (5.8)$$

Constraints (5.9) to (5.11) control the yard crane deployment for discharging containers. Constraint (5.9) gives the number of yard cranes deployed in each block for discharging. Due to safety issues, the total number of yard cranes deployed in a block usually has a limit, which is described by constraint (5.10). The extra yard cranes are captured as the "crane violation", which is presented in constraint (5.11).

$$\sum_{i \in B_k} (x_i^1 + y_i^1 + x_i^2 + y_i^2) \le CC \times d_k \qquad \forall\ 1 \le k \le K \qquad (5.9)$$

$$d_k + D_k^l \le M_k \qquad \forall\ 1 \le k \le K \qquad (5.10)$$

$$\sum_{k=1}^{K} d_k \le D^d + v^2 \qquad (5.11)$$

The space occupation and loading activities are described in constraints (5.12) to (5.19). To make the description clearer, constraints (5.12) to (5.15) are used to calculate the containers in each sub-block at the end of the current period, which considers the existing containers and the short-term incoming containers.

$$R_{i0}^{T1} + x_i^1 = r_i^{T1} \qquad \forall\ 1 \le i \le I \qquad (5.12)$$

$$R_{i0}^{T2} + x_i^2 = r_i^{T2} \qquad \forall\ 1 \le i \le I \qquad (5.13)$$

$$R_{i0}^{F1} + y_i^1 = r_i^{F1} \qquad \forall\ 1 \le i \le I \qquad (5.14)$$

$$R_{i0}^{F2} + y_i^2 = r_i^{F2} \qquad \forall\ 1 \le i \le I \qquad (5.15)$$

Based on the containers in each sub-block at the end of the current shift, the loading violation for each container group is described in constraints (5.16) to (5.18). This is measured by comparing the amount of containers in a container group with the loading capacity.

$$r_i^{T1}+r_i^{F1}\le (CC+v_{kg}^3)\times\sum_j Z_{ij}^1E_j \qquad \forall\ 1\le k\le K,\ g{=}1,\ i=\left(G-1\right)\times k+g \qquad (5.16)$$

$$(r_{(i-1)}^{T2}+r_{(i-1)}^{F2})+(r_i^{T1}+r_i^{F1})\le (CC+v_{kg}^3)\times\sum_j Z_{ij}^1E_j$$

$$\forall\ 1\le k\le K,\ 2\le g\le \text{(G-1)},\ i=\left(G-1\right)\times k+g \qquad (5.17)$$

$$r_i^{T2}+r_i^{F2}\le (CC+v_{kg}^3)\times\sum_j Z_{ij}^2E_j \qquad \forall\ 1\le k\le K,\ g{=}\text{G},\ i=\left(G-1\right)\times(k+1) \qquad (5.18)$$

The space violation is described in constraint (5.19), where a FEU is counted as two TEUs in terms of space occupation. The total space occupation in a sub-block is compared with the space capacity to give the space violation.

$$r_i^{T1}+2\times r_i^{F1}+r_i^{T2}+2\times r_i^{F2}\le CS+v_i^4 \qquad \forall\ 1\le i\le I \qquad (5.19)$$

To further facilitate the loading process, constraints (5.20) and (5.21) are used to reduce the interference between loading and discharging activities. If a container group is in loading process, no discharging activities are allowed in its neighboring container groups.

$$\left(x_i^2+y_i^2\right)+\sum_{i'\in N_i}\left(x_{i'}^1+y_{i'}^1\right)\le M\left(1-\sum_{j=1}^{J}Z_{ij}^1L_j\right) \qquad \forall\ 1\le i\le I \qquad (5.20)$$

$$\left(x_i^1+y_i^1\right)+\sum_{i'\in N_i}\left(x_{i'}^2+y_{i'}^2\right)\le M\left(1-\sum_{j=1}^{J}Z_{ij}^2L_j\right) \qquad \forall\ 1\le i\le I \qquad (5.21)$$

Constraints (5.22) to (5.28) are integrality and non-negativity constraints for decision variables.

$$x_i^1,y_i^1,x_i^2,y_i^2\ge 0 \qquad \forall\ 1\le i\le I \qquad (5.22)$$

$$r_i^{T1},r_i^{F1},r_i^{T2},r_i^{F2}\ge 0 \qquad \forall\ 1\le i\le I \qquad (5.23)$$

$$d_k\in\left\{Positive\ Integer\right\} \qquad \forall\ 1\le k\le K \qquad (5.24)$$

$$v_{ii'}^1\ge 0 \qquad \forall\ 1\le i\le I,\ 1\le i'\le I \qquad (5.25)$$

$$v^2 \geq 0 \tag{5.26}$$

$$v_{kg}^3 \geq 0 \qquad \forall\, 1 \leq k \leq K,\ 1 \leq g \leq G \tag{5.27}$$

$$v_i^4 \geq 0 \qquad \forall\, 1 \leq i \leq I \tag{5.28}$$

Besides, the amount of loading activities in each sub-block during the current shift can be updated with equation (5.29). As described in the assumptions of the model, the total loading activities are equally distributed to each shift of the loading period.

$$WL_i = (R_{i0}^{T1} + R_{i0}^{F1}) \times \sum_j (Z_{ij}^1 L_j / E_j) + (R_{i0}^{T2} + R_{it}^{F2}) \times \sum_j (Z_{ij}^2 L_j / E_j) \qquad \forall\, 1 \leq i \leq I \tag{5.29}$$

5.2.2 MIP model for the SALP method

The SALP method considers the future impact from the short-term space allocation. To measure the loading violation and space violation in future shifts, we need to measure the containers not only at the end of the current shift, but also at the end of future shifts. Some variables are modified to describe the value in different shifts, which can be shown as follows. When shift index *t* is 1, the variable is for the current shift.

L_{jt} = 1, if vessel *j* is in loading procedure during shift *t*, $1 \leq j \leq J$, $1 \leq t \leq T$.

= 0, otherwise, $1 \leq j \leq J$, $1 \leq t \leq T$.

NL_{kt} Number of yard cranes deployed for loading jobs in block *k* during shift *t*, which is given by the yard template, $1 \leq k \leq K$, $1 \leq t \leq T$.

$S_{jtt'}$ The parameter indicates whether a container, which is discharged in shift *t* and to be loaded onto vessel *j*, will still be present in the yard in shift *t'*. The value is decided according to the discharging shift *t* of the container and the loading completion shift of its destination vessel *j*.

= 1, if the container will stay during shift *t'*, $1 \leq j \leq J$, $1 \leq t \leq T$, $1 \leq t' \leq T$.

= 0, otherwise, $1 \leq j \leq J$, $1 \leq t \leq T$, $1 \leq t' \leq T$.

T The number of shifts described in the model.

W_t The parameter indicates whether shift t from the long-term plan is considered.

= 1, if shift t is considered, $2 \leq t \leq T$.

= 0, otherwise, $2 \leq t \leq T$.

As the vessels arrive at the port once a week and each day consists of three 8-hour shifts, the long term space allocation plan is formulated for a 21-shift cycle. This plan roughly shows the amount of incoming containers which are assigned to each storage location in each shift of the cycle. In the SALP method, the long-term space allocation plan is considered to indicate the incoming containers to each storage location in a future shift. The parameters for the given long-term plan can be shown as follows. Since t = 1 is the index of the current shift, the parameters for all future shifts start with the index t = 2.

X_{it}^1 The amount of 20ft containers that come into the left corner of sub-block i in shift t. $1 \leq i \leq I$, $2 \leq t \leq T$.

X_{it}^2 The amount of 20ft containers that come into the right corner of sub-block i in shift t. $1 \leq i \leq I$, $2 \leq t \leq T$.

Y_{it}^1 The amount of 40ft containers that come into the left corner of sub-block i in shift t. $1 \leq i \leq I$, $2 \leq t \leq T$.

Y_{it}^2 The amount of 40ft containers that come into the right corner of sub-block i in shift t. $1 \leq i \leq I$, $2 \leq t \leq T$.

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

The decision variables are modified to measure the value in different shifts, which are shown as follows.

r_{it}^{T1} The remaining TEUs in the left corner of sub-block *i* at the end of shift *t*. $1 \le i \le I$, $1 \le t \le T$.

r_{it}^{T2} The remaining TEUs in the right corner of sub-block *i* at the end of shift *t*. $1 \le i \le I$, $1 \le t \le T$.

r_{it}^{F1} The remaining FEUs in the left corner of sub-block *i* at the end of shift *t*. $1 \le i \le I$, $1 \le t \le T$.

r_{it}^{F2} The remaining FEUs in the right corner of sub-block *i* at the end of shift *t*. $1 \le i \le I$, $1 \le t \le T$.

v_{kgt}^{3} The loading violation for container group *g* in block *k* during shift *t*. $1 \le k \le K$, $1 \le g \le G$, $1 \le t \le T$.

v_{it}^{4} The space violation in sub-block *i* during shift *t*. $1 \le i \le I$, $1 \le t \le T$.

The objective of the SALP method is to minimize the violations in the current and future shifts. The first four terms capture the corresponding violations in the current shift. The last term is added to measure the space violation and the loading violation in future shifts. The number of future shifts considered can be controlled by the weight W_t. As the space violation in future shifts is not necessarily going to happen, the penalty weights are not set to large numbers, which is different from the weight for the current shift.

$$\begin{aligned} Min\ w = & \frac{1}{2}\sum_{i}\sum_{i'\in N_i} v_{ii'}^{1} + CC \times v^{2} + \sum_{k}\sum_{g} v_{kg1}^{3} + M \times \sum_{i} v_{i1}^{4} \\ & + \sum_{t=2}^{T} W_t \times \left(\sum_{k}\sum_{g} v_{kgt}^{3} + \sum_{i} v_{it}^{4} \right) \end{aligned} \tag{5.30}$$

The SALP method and the GSA method have the same requirements on the yard template, the traffic control and the yard crane deployment. Thus, the constraints (5.31) to (5.36) are the same with the corresponding ones in the previous model.

$$\sum_{i=1}^{I}(x_i^1 \times Z_{ij}^1) + \sum_{i=1}^{I}(x_i^2 \times Z_{ij}^2) = WX_j \qquad \forall\ 1 \le j \le J \qquad (5.31)$$

$$\sum_{i=1}^{I}(y_i^1 \times Z_{ij}^1) + \sum_{i=1}^{I}(y_i^2 \times Z_{ij}^2) = WY_j \qquad \forall\ 1 \le j \le J \qquad (5.32)$$

$$(x_i^1 + y_i^1 + x_i^2 + y_i^2 + WL_i) + (x_{i'}^1 + y_{i'}^1 + x_{i'}^2 + y_{i'}^2 + WL_{i'}) \le U + v_{ii'}^1$$

$$\forall\ 1 \le i \le I,\ i' \in N_i \qquad (5.33)$$

$$\sum_{i \in B_k}(x_i^1 + y_i^1 + x_i^2 + y_i^2) \le CC \times d_k \qquad \forall\ 1 \le k \le K \qquad (5.34)$$

$$d_k + D_k^l \le M_k \qquad \forall\ 1 \le k \le K \qquad (5.35)$$

$$\sum_{k=1}^{K} d_k \le D^d + v^2 \qquad (5.36)$$

Similar constraints have been used to reduce the interference between loading and discharging activities, which are shown as follows.

$$\left(x_i^2 + y_i^2\right) + \sum_{i' \in N_i}\left(x_{i'}^1 + y_{i'}^1\right) \le M\left(1 - \sum_{j=1}^{J} Z_{ij}^1 L_j\right) \qquad \forall\ 1 \le i \le I \qquad (5.37)$$

$$\left(x_i^1 + y_i^1\right) + \sum_{i' \in N_i}\left(x_{i'}^2 + y_{i'}^2\right) \le M\left(1 - \sum_{j=1}^{J} Z_{ij}^2 L_j\right) \qquad \forall\ 1 \le i \le I \qquad (5.38)$$

Unlike the previous method, the SALP method considers the future impact of the current decisions. In order to achieve this, the container in each sub-block at the end of each shift is measured combining the existing containers, the current space allocation and the long-term space allocation plan. The corresponding constraints in the previous model are modified into the following constraints (5.39) to (5.42).

$$\left(R_{i0}^{T1}+x_i^1\right)\times\sum_{j=1}^{J}Z_{ij}^1S_{j1t}+\sum_{t'=2}^{t}(X_{it'}^1\times\sum_{j=1}^{J}Z_{ij}^1S_{jt't})=r_{it}^{T1} \qquad \forall\ 1\le i\le I,\ 1\le t\le T \qquad (5.39)$$

$$\left(R_{i0}^{T2}+x_i^2\right)\times\sum_{j=1}^{J}Z_{ij}^2S_{j1t}+\sum_{t'=2}^{t}(X_{it'}^2\times\sum_{j=1}^{J}Z_{ij}^2S_{jt't})=r_{it}^{T2} \qquad \forall\ 1\le i\le I,\ 1\le t\le T \qquad (5.40)$$

$$\left(R_{i0}^{F1}+y_i^1\right)\times\sum_{j=1}^{J}Z_{ij}^1S_{j1t}+\sum_{t'=2}^{t}(Y_{it'}^1\times\sum_{j=1}^{J}Z_{ij}^1S_{jt't})=r_{it}^{F1} \qquad \forall\ 1\le i\le I,\ 1\le t\le T \qquad (5.41)$$

$$\left(R_{i0}^{F2}+y_i^2\right)\times\sum_{j=1}^{J}Z_{ij}^2S_{j1t}+\sum_{t'=2}^{t}(Y_{it'}^2\times\sum_{j=1}^{J}Z_{ij}^2S_{jt't})=r_{it}^{F2} \qquad \forall\ 1\le i\le I,\ 1\le t\le T \qquad (5.42)$$

Based on the containers in each sub-block at the end of each shift, the loading violation for each container group is described in constraints (5.43) to (5.45). The space violation in each sub-block is described in constraint (5.46).

$$(r_{it}^{T1}+r_{it}^{F1})\times\sum_j Z_{ij}^1L_{jt}\le(CC+v_{kgt}^3)\times\sum_j Z_{ij}^1E_j$$

$$\forall\ 1\le k\le K,\ g{=}1, i=\left(G-1\right)\times k+g\ ,\ 1\le t\le T \qquad (5.43)$$

$$(r_{(i-1)t}^{T2}+r_{(i-1)t}^{F2})\times\sum_j Z_{ij}^2L_{jt}+(r_{it}^{T1}+r_{it}^{F1})\times\sum_j Z_{ij}^1L_{jt}\le(CC+v_{kgt}^3)\times\sum_j Z_{ij}^1E_j$$

$$\forall\ 1\le k\le K,\ 2\le g\le \text{(G-1)}, i=\left(G-1\right)\times k+g\ ,\ 1\le t\le T \qquad (5.44)$$

$$(r_{it}^{T2}+r_{it}^{F2})\times\sum_j Z_{ij}^2L_{jt}\le(CC+v_{kgt}^3)\times\sum_j Z_{ij}^2E_j$$

$$\forall\ 1\le k\le K,\ g{=}1, i=\left(G-1\right)\times(k+1)\ ,\ 1\le t\le T \qquad (5.45)$$

$$r_{it}^{T1}+2\times r_{it}^{F1}+r_{it}^{T2}+2\times r_{it}^{F2}\le CS+v_{it}^4 \qquad \forall\ 1\le i\le I,\ 1\le t\le T \qquad (5.46)$$

Constraints (5.47) to (5.53) are integrality and non-negativity constraints for decision variables.

$$x_i^1, y_i^1, x_i^2, y_i^2\ge 0 \qquad \forall\ 1\le i\le I \qquad (5.47)$$

$$r_{it}^{T1}, r_{it}^{F1}, r_{it}^{T2}, r_{it}^{F2}\ge 0 \qquad \forall\ 1\le i\le I,\ 1\le t\le T \qquad (5.48)$$

$$d_k\in\left\{Positive\ Integer\right\} \qquad \forall\ 1\le k\le K \qquad (5.49)$$

$$v^1_{ii'} \geq 0 \qquad \forall\ 1 \leq i \leq I,\ 1 \leq i' \leq I \tag{5.50}$$

$$v^2 \geq 0 \tag{5.51}$$

$$v^3_{kgt} \geq 0 \qquad \forall\ 1 \leq k \leq K,\ 1 \leq g \leq G,\ 1 \leq t \leq T \tag{5.52}$$

$$v^4_{it} \geq 0 \qquad \forall\ 1 \leq i \leq I,\ 1 \leq t \leq T \tag{5.53}$$

5.3 Numerical experiments

In this section, numerical experiments are carried out to compare the performance of the short-term space allocation using the GSA method and the SALP method. The short-term space allocation is also evaluated under the “non-sharing strategy” and the “flexible space-sharing strategy”. All procedures were implemented in C++ and the MIP models were solved using CPLEX 11.2 with concert technology. The numerical experiments were run on Intel Core i5-2000 computer (CPU: 3.30 GHz, 3.30 GHz; Memory: 8 GB). All the numerical experiments were solved under the scale of a real port problem with 64 blocks and 21 vessels.

5.3.1 Experiment description

5.3.1.1 The incoming containers

Due to confidentiality reasons, the port operator only provides the long-term pattern of incoming containers, instead of the actual data collected in port operation. The long-term pattern gives the amount of incoming containers to each destination vessel during each shift of the 21-shift cycle. Around 74% of these incoming containers are transshipment. Import containers are not considered, since they are usually stored in separate storage area from the export and transshipment containers. The transshipment containers are discharged from one vessel, and are temporarily stored in the yard until being loaded onto other destination vessels. According to the port operator, there are two major aspects which affect the fluctuation of actual incoming containers. Firstly, the amount of incoming containers to each destination vessel in a particular shift changes from week to week. Secondly, if a vessel is delayed, the incoming containers discharged from this vessel will also be delayed.

Based on this, we generate the transshipment and export containers separately. The transshipment containers are decomposed to the batches of containers from vessel *j* to vessel *j'*. The amount of containers in a particular batch of containers fluctuates around the mean value with ±30% randomness. When vessel *j* is delayed, all the batches of containers discharged from vessel *j* are delayed accordingly. In this section, we assume each vessel has 90% chances of arriving on time, 9% chance of one shift delay and 1% chance of 2 shifts delay. The chance can also be easily modified to test under other assumptions. On the other hand, the incoming export containers are not affected by the vessel delays, since they are not brought in by vessels. The amount of incoming containers to each vessel in a particular shift also fluctuates around the mean value with ±30% randomness. An example of the container flow with randomness and vessel delay for one destination vessel can be shown in Figure 5.1.

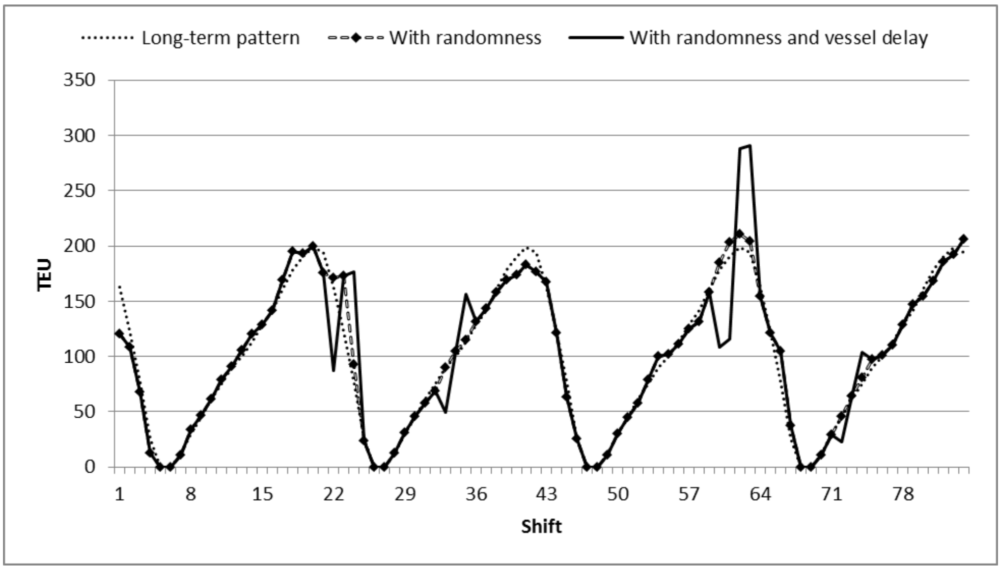

Figure 5.1 An example of the container flow for a destination vessel

5.3.1.2 Performance measure

In the numerical experiment, the input container flow is generated one shift at a time. After the short-term space allocation in one shift, the experiment performance is measured by the violations actually happening in the current shift. The violations that may happen in the future shifts are not counted.

The 21 vessels in the experiments have different loading shifts, while each vessel arrives at the terminal once a week. Despite the vessel delays, the consecutive 21 shifts usually include one complete cycle of loading and discharging for each vessel. Therefore, the statistical data of one week will be more representative than that of a single shift. All the shifts generated in the experiment are divided into cycles of consecutive 21 shifts. The violations happening in each shift within one cycle were summed together to measure the performance.

The performance of the experiments might be affected by two major factors, namely the "space tightness" and the "vessel delay". Four different scenarios are considered in the experiment by combining "sufficient space or tight space" and "with or without vessel delay". Common random numbers are used in all scenarios for a fair comparison of the short-term space allocation.

5.3.2 Comparison of short-term planning methods

In the experiment, the GSA method and the SALP method are compared under both storage strategies in each scenario. Since the comparison under each storage strategy leads to the same conclusions, the performance under the "flexible space-sharing strategy" is used for demonstration in this section. In the SALP method, the long-term space allocation plan includes all the shifts in a 21-shift cycle, but the number of shifts considered may affect the performance. In the experiments, we test three cases for the number of future shifts, including 3 shifts (1 day), 6 shifts (2 days) and 21 shifts (1 week).

5.3.2.1 Scenarios with sufficient space

When there is sufficient space and no vessel delay, both the GSA method and the SALP method can decide the space allocation with no violation.

However, when there is vessel delay, the container flow becomes more fluctuating, which increases the difficulty of space allocation. The workload violation is increased for both the GSA method and the SALP method, but the other kinds of violations remain zero. Figure 5.2 shows the 10-cycle moving average of traffic violation for both short-term methods. Although the violation is increased, there is no obvious difference when we vary the number of shifts considered. There is also no obvious difference between the GSA method and the SALP method.

Therefore, it can be concluded that when there is sufficient space, both the GSA method and the SALP method can be used for short-term space allocation.

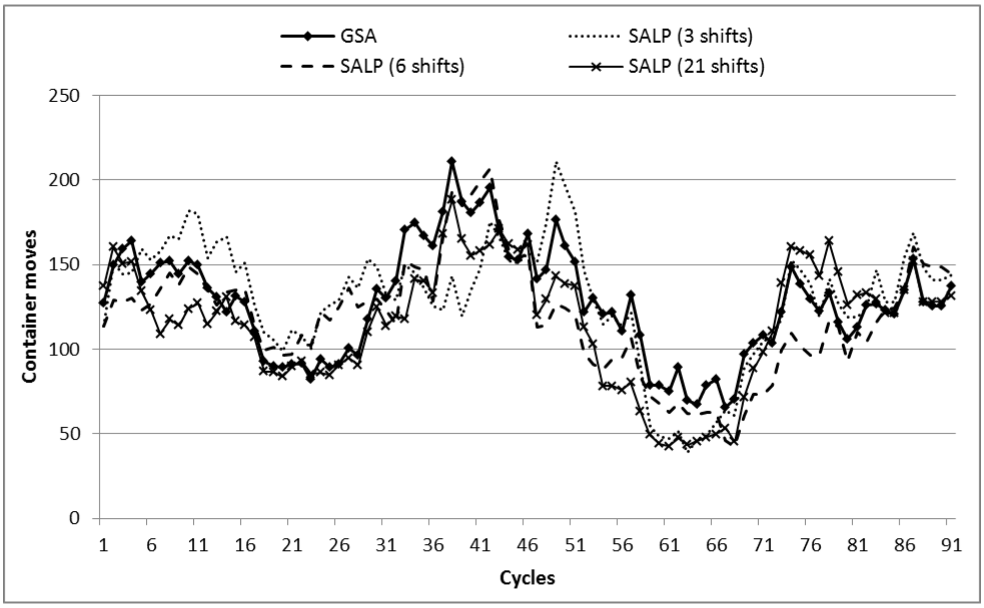

Figure 5.2 Traffic violation under different planning methods
(Sufficient space with vessel delay)

5.3.2.2 Scenarios with tight space

The first scenario we test in this section is tight space with no vessel delay. As the tight space increases the difficulty in space allocation, the traffic violation and the loading violation exist under both planning methods. The 10-cycle moving average of the violations can be shown in Figures 5.3 and 5.4 respectively. According to the results, the SALP method clearly outperforms the GSA method. Besides, the more future shifts are considered, the less violation occurs. The main reason is that the SALP method takes into account of the possible space allocation requirements in future shifts through the long-term space allocation plan. The decision of short-term space allocation tends to reserve the storage space that may be needed in the future. This helps to improve the future impact of the short-term decisions.

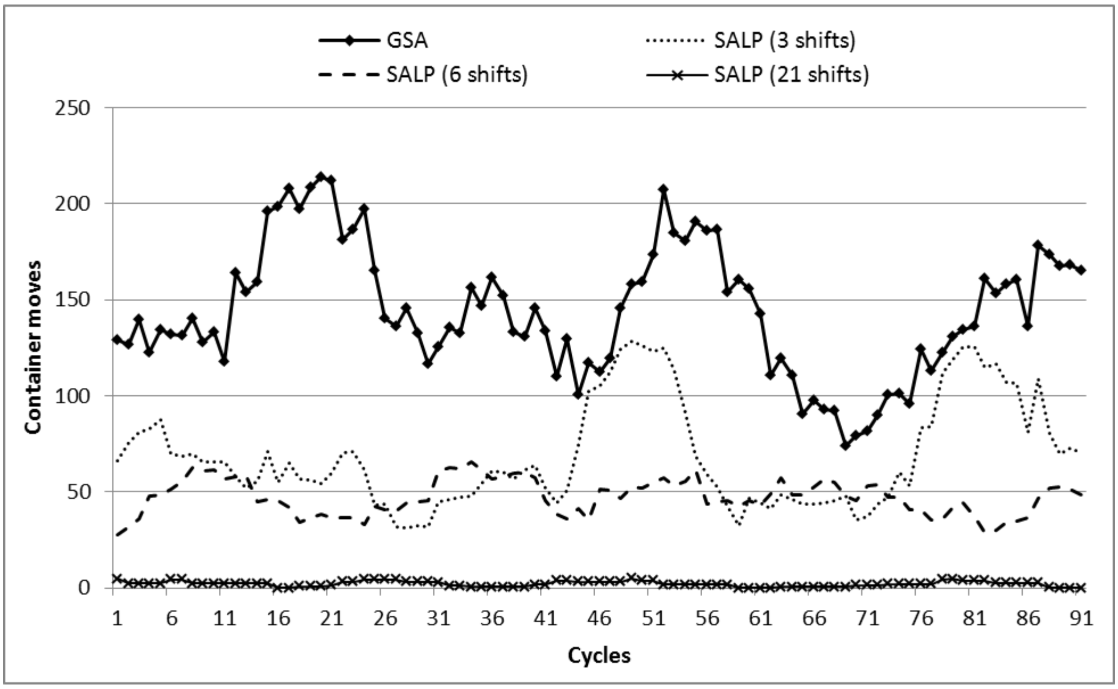

Figure 5.3 Traffic violation under different planning methods
(Tight space with no vessel delay)

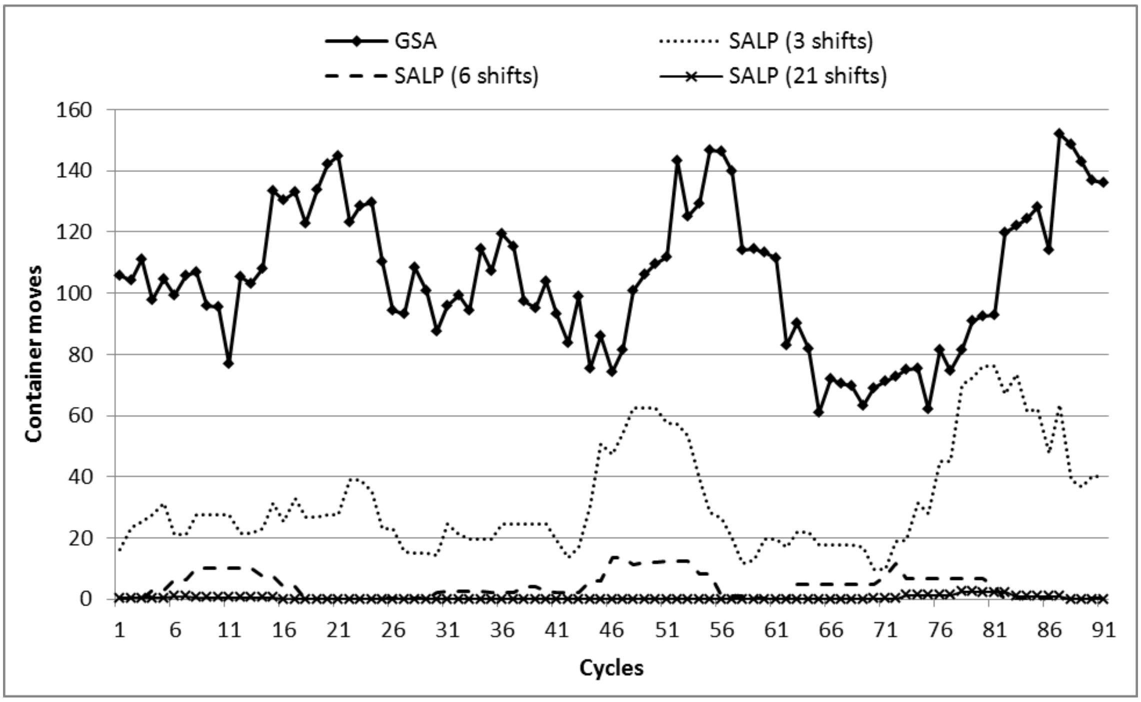

Figure 5.4 Loading violation under different planning methods
(Tight space with no vessel delay)

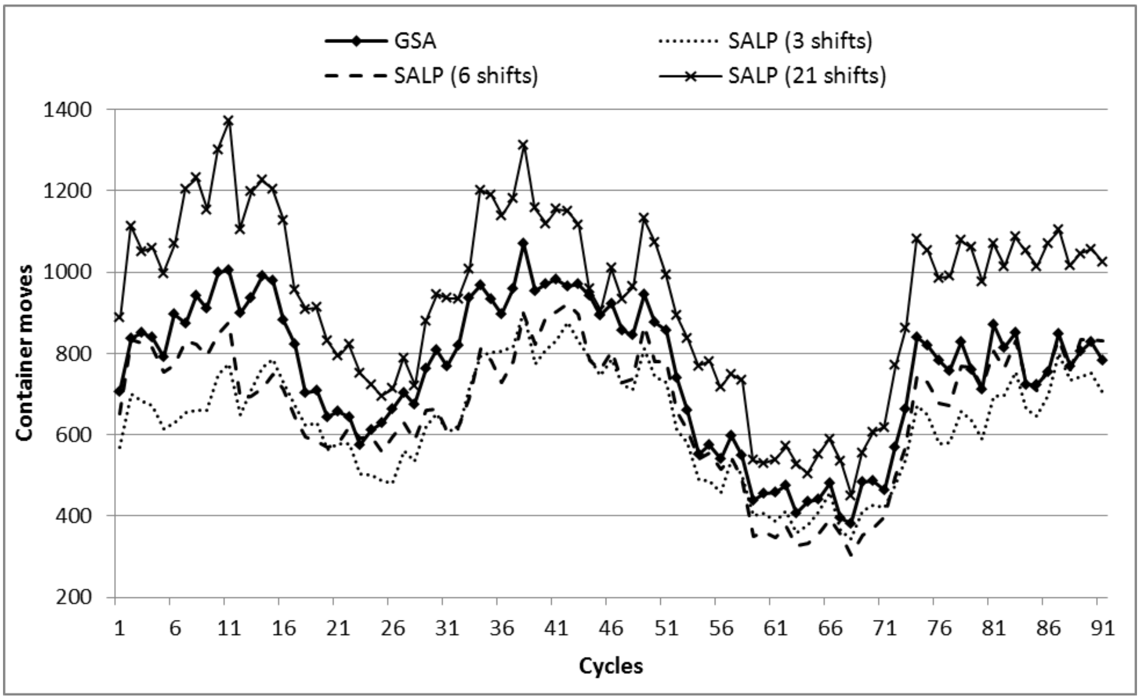

Figure 5.5 Traffic violation under different planning methods
(Tight space with vessel delay)

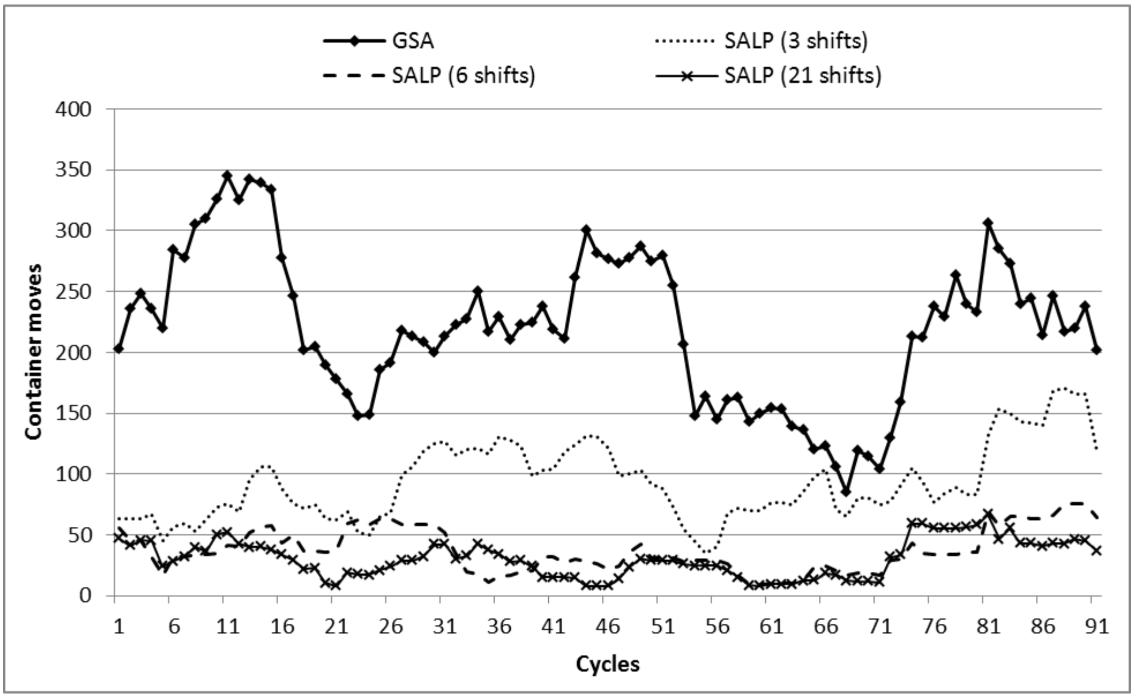

Figure 5.6 Loading violation under different planning methods
(Tight space with vessel delay)

When there is vessel delay, the performance shows some difference. The vessel delay causes more fluctuations in the container flow. The traffic violation and the loading violations are greatly increased under both planning methods. Although the loading violation still decreases with the number of shifts being considered, the traffic violation shows a different trend. According to the scales shown in Figures 5.5 and 5.6, the traffic violation dominates the total violation under each planning method. The SALP method considering 21 shifts leads to more violation compared with the GSA method. However, the SALP method is better than the GSA method when 3 or 6 shifts are considered. The underlying reason is that the estimation of the future through the long-term plan is less accurate when there is vessel delay. Considering more shifts during short-term space allocation will sacrifice the short-term efficiency so as to improve the future impact. When the long-term plan is inaccurate, considering more future shifts can mislead the short-term space allocation, but considering a portion of the long-term plan can still help to certain extent.

5.3.3 Comparison between the storage strategies

In this section, short-term space allocation is decided with the SALP method. As the 6-shift looking ahead turns out to be more robust in all scenarios, it is used to evaluate the short-term space allocation under the “non-sharing strategy” and the “flexible space-sharing strategy”.

5.3.3.1 Scenarios with sufficient space

When there is sufficient space and no vessel delay, the short-term space allocation under both the “flexible space-sharing strategy” and the “non-sharing strategy” can handle the incoming containers without any violations. When there is vessel delay, the container flow is more fluctuating, which causes traffic violation under both storage strategies. The 10-cycle moving average of the traffic violation can be shown in Figure 5.7. It can be seen from the graphic that the “flexible space-sharing strategy” has less violation compared with the “non-sharing strategy”. The main reason is that space-sharing provides more flexibility for short-term space allocation, which makes it more adaptable to the vessel delay.

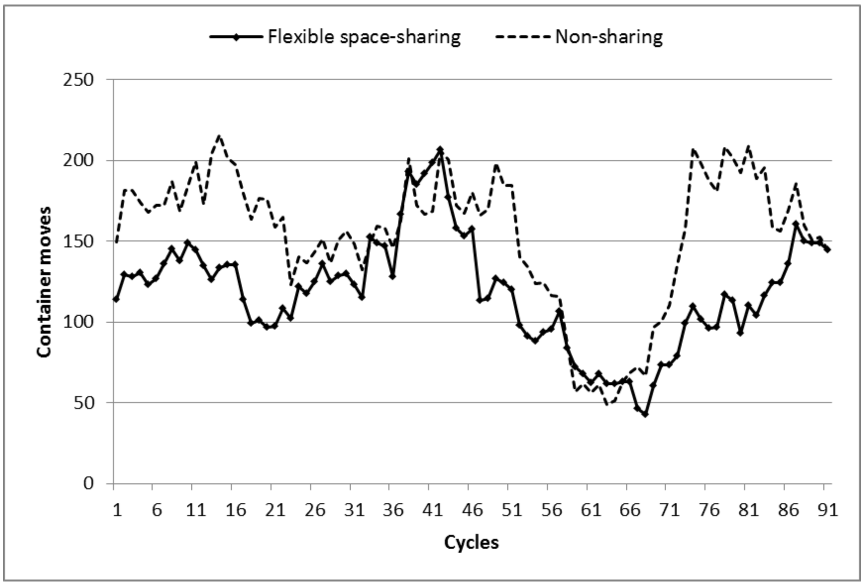

Figure 5.7 Traffic violation under different storage strategies
(Sufficient space with vessel delay)

5.3.3.2 Scenarios with tight space

When there is tight space and no vessel delay, the traffic violation and the loading violation under different storage strategies are about the same level, but the crane violation and the space violation are quite different. The 10-cycle moving average of the crane violation and the space violation can be shown respectively in Figure 5.8 and Figure 5.9. Under the non-sharing strategy, although extra yard cranes are deployed, much buffer space is still required to handle the incoming the containers. In contrast, the short-term space allocation under the flexible space-sharing strategy needs no buffer space, while the incoming containers can be handled with the least number of yard cranes. When there is vessel delay, the similar results can be found in Figure 5.10 and Figure 5.11 respectively.

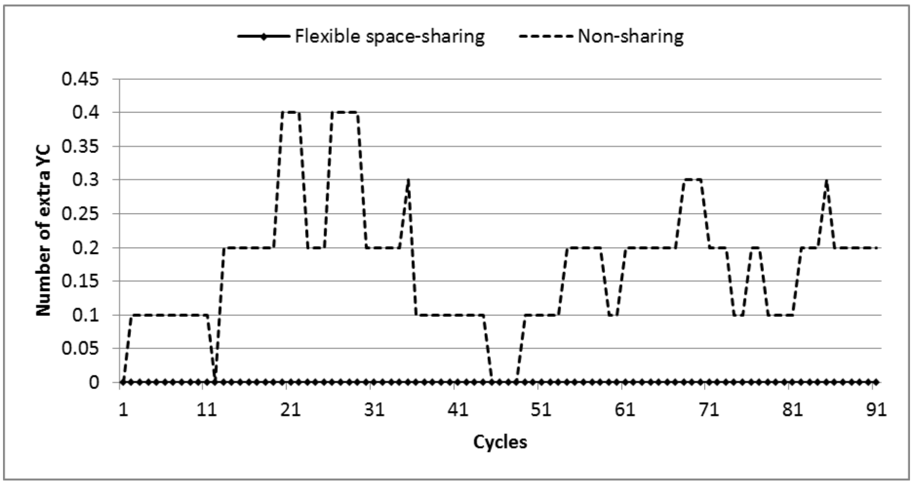

Figure 5.8 Crane violation under different storage strategies
(Tight space with no vessel delay)

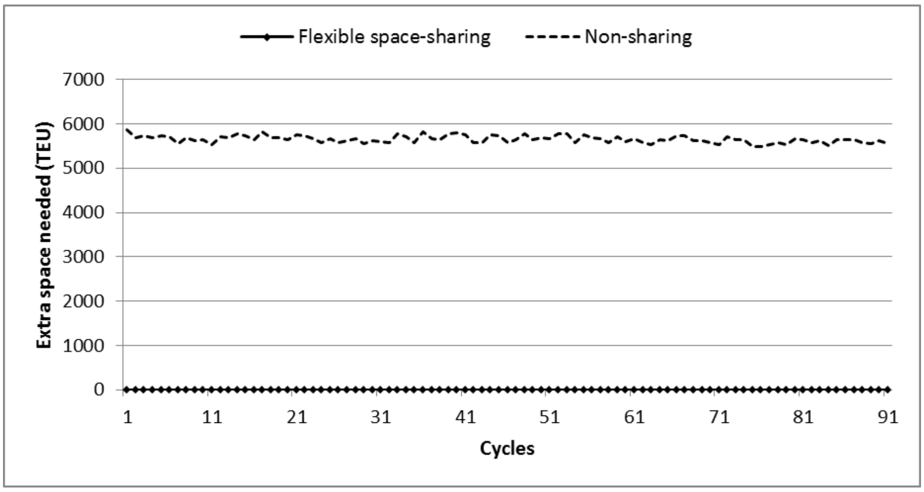

Figure 5.9 Space violation under different storage strategies
(Tight space with no vessel delay)

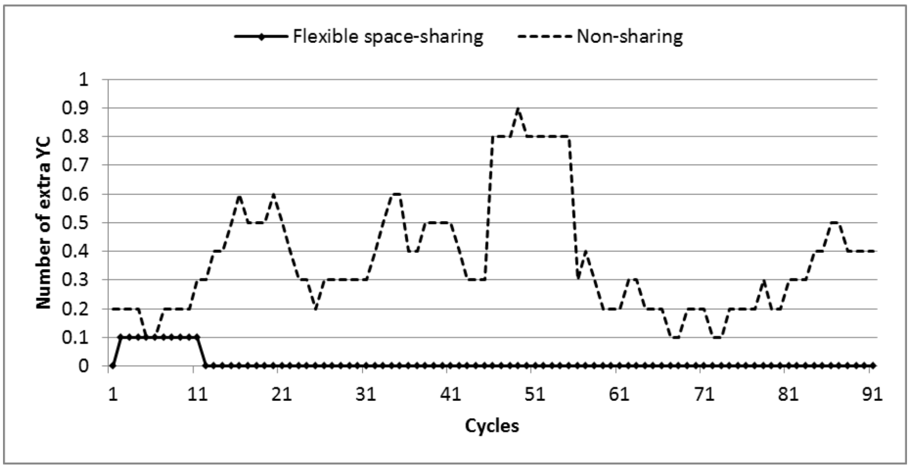

Figure 5.10 Crane violation under different storage strategies
(Tight space with vessel delay)

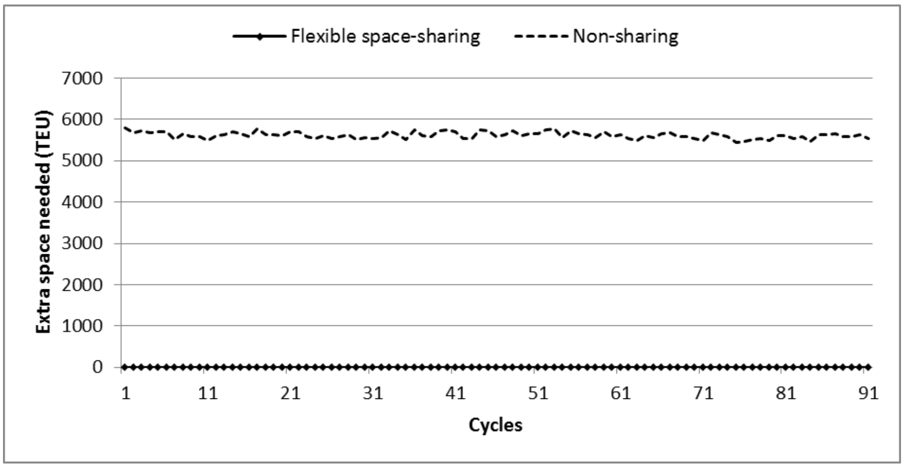

Figure 5.11 Space violation under different storage strategies
(Tight space with vessel delay)

It can be concluded that the short-term space allocation under the flexible space-sharing strategy outperforms that of the non-sharing strategy when space is tight. The flexible space-sharing strategy also makes the short-term space allocation more adaptable to the fluctuations caused by the vessel delay. Therefore, when the throughput is high and the land is scarce, it is recommended to use the flexible space-sharing strategy to increase the stacking capacity of the storage yard.

5.4 Conclusions

In transshipment ports, the containers to the same destination vessel are usually stored together to facilitate the loading process. This is known as the "consignment", while the storage locations reserved for each vessel are defined by the given "yard template". In the short-term operation, loading and unloading times can change because of vessel delays. The amount of containers discharged from each vessel is also different from week to week. The actual containers that will come in are only known for a short period in advance. Thus, short-term space allocation is needed to assign the incoming containers taking into account of transport vehicles, yard cranes and space capacity. Currently, the space is allocated based on the experience of port operators and the rule of thumb. In this chapter, we develop two systematic short-term planning methods, namely the "greedy space allocation (GSA)" and the "space allocation considering the long-term plan (SALP)". The GSA method only considers the short-term information, which includes the existing containers and the actual incoming containers. On the other hand, the SALP method considers more information which includes the long-term space allocation plan. MIP models are formulated for the two methods respectively. The short-term space allocation is evaluated under two different storage strategies, including the "non-sharing strategy" and the "flexible space-sharing strategy". The numerical experiments show that the SALP method is preferred over the GSA method, but the portion of long-term plan considered affects the performance of the SALP method. The short-term planning methods perform well under both storage strategies when space is sufficient. However, the performance under the “flexible space-sharing strategy” is better than the “non-sharing strategy” when space is tight.

Chapter 6. Conclusions and Future Research

In this thesis we study an actual problem faced by a leading transshipment hub port. Generally, the “consignment strategy” is used in the yard for a transshipment port, where containers to the same destination vessel are stored together. This is to facilitate faster loading process as it reduces reshuffles as well as long distance movements of yard cranes. However, the consignment strategy is known to be inefficient in space utilization since each storage location must be dedicated to a particular vessel. With the increasing volume of transshipment container handling, the scarcity of storage space is urging new studies to balance the equipment productivity and the land usage.

In chapter 3, a new approach named the “partial space-sharing strategy” is proposed to improve the space utilization while retaining the advantage of consignment. The “yard template” is used to define the reservation of storage locations for each destination vessel. We develop a framework which integrates yard template planning and workload assignment to solve the space allocation problem. In this framework, two different approaches are proposed to determine the sharing space, namely the “fixed sharing space method” and the “variable sharing space method”. The numerical experiments show that, both the proposed methods can improve the land productivity under different port throughput levels compared to the current consignment strategy which does not allow the sharing of space. The partial space-sharing strategy can even achieve a land productivity exceeding a value of 1, which is impossible for a yard template without sharing. Among the two methods, the variable sharing space method performs better both in land productivity and robustness.

In chapter 4, we propose a more advanced concept named the “flexible space-sharing strategy”. The idea is that the container space can be shared by two different vessels as long as their containers do not occupy the space at the same time. This strategy allows each storage location to be shared by two vessels. By controlling where to stack the containers in the storage locations, the containers to each vessel are not mixed and the consignment feature can be preserved. The container assignment problem (CAP) using this strategy is formulated as a mixed integer program. The model can be solved directly for small scale problems, while it cannot be solved for a real scale problem. Based on the "block diagonal structure" of the CAP model, a two-stage search algorithm incorporating MIP and heuristics is developed. The "initial stage" generates an initial solution shift by shift, while the "conflict resolution stage" reduces the violations in the solution to get an implementable plan. The numerical experiments show that the "flexible space-sharing strategy" can handle much more containers within the same storage space, compared with the "non-sharing strategy".

In chapter 5, the short-term space allocation problem is studied. During the operation, loading and unloading times can change because of vessel delays. The amount of containers discharged from each vessel is also different from week to week. The actual containers that will come in are only known for a short period in advance. Thus, short-term planning is needed to decide the space allocation taking into account prime movers, yard cranes and the storage space. Currently, the space is allocated based on the experience of port operators and the rule of thumb. In this thesis, we develop two systematic short-term planning methods, namely the "greedy space allocation (GSA)" and the "space allocation considering the long-term plan (SALP)". The GSA method only considers the short-term information, which includes the existing containers and the actual incoming containers. On the other hand, the SALP method considers more information which includes the long-term space allocation plan. MIP models are formulated for the two methods respectively. The short-term space allocation is evaluated under two different storage strategies, including the “non-sharing strategy” and the “flexible space-sharing strategy”. The numerical experiments show that the SALP method is preferred over the GSA method, but the portion of long-term plan considered affects the performance of the SALP method. The short-term planning methods perform well under both storage strategies when space is sufficient. However, the performance under the “flexible space-sharing strategy” is better than the “non-sharing strategy” when space is tight.

There are several topics related to this thesis that can be conducted for further research. Firstly, the space-sharing strategies are all considered at the space allocation phase, which does not consider the locations for individual containers. Future research is needed to implement the storage strategies at the real-time phase, which considers the locations of individual containers. Secondly, when the storage strategies are implemented for the short-term operations, the future shifts are estimated with the long-term space allocation plan. In future research, different estimation methods can be developed and combined with the short-term space allocation. This will further improve the performance of the space-sharing strategies.

Bibliography

[1] Alessandri, A., Sacone, S., and Siri, S. Modelling and optimal receding-horizon control of maritime container terminals. Journal of Mathematical Modelling and Algorithms, 6, pp. 109-133. 2007.

[2] 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.

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

[4] Baker, C. High time for straddles. Cargo Systems, 23-26. 1998.

[5] Bazzazi, M., Safaei, N., and Javadian, N. A genetic algorithm to solve the storage space allocation problem in a container terminal. Computers and Industrial Engineering, 56, pp. 44-52. 2009.

[6] Bielli, M., Boulmakoul, A., and Rida, M. Object oriented model for container terminal distributed simulation. European Journal of Operational Research, 175, pp. 1731-1751. 2006.

[7] 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.

[8] 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.

[9] 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.

[10] Briskorn, D., Drexl, A., and Hartmann, S. Inventory-based dispatching of automated guided vehicles on container terminals. OR Spectrum, 28, pp. 611-630. 2006.

[11] Cao, Z., Lee, D-H., Meng, Q. Scheduling of multiple yard crane system with container loading sequence consideration. Paper presented at TRB 2006 annual meeting, Washington, DC, 22-26, January, 2006a.

[12] Cao, Z., Lee, D-H., Meng, Q. Scheduling two-transtainer systems for the loading operation of containers using a revised genetic algorithm. Paper presented at TRB 2006 annual meeting, Washington, DC, 22-26, January, 2006b.

[13] Cao, Z., Lee, D-H., Meng, Q. Deployment strategies of double-rail-mounted gantry crane systems for loading outbound containers in container terminals. International Journal of Production Economics, 115, pp. 221-228. 2008.

[14] Caserta, M., Voß, S., and Sniedovich, M. Applying the corridor method to a blocks relocation problem. OR Spectrum 33, pp. 915-929. 2011.

[15] Caserta, M., Schwarze, S., and Voß, S. A mathematical formulation and complexity considerations for the blocks relocation problem. European Journal of Operational Research 219, pp. 96-104. 2012.

[16] Chang, D., Jiang, Z., Yan, W., and He, J. Developing a dynamic rolling-horizon decision strategy for yard crane scheduling. Advanced Engineering Informatics, 25, pp. 485-494. 2011.

[17] Chen, T. Yard operations in the container terminal - a study in the 'unproductive moves'. Maritime Policy and Management, 26, pp. 27-38. 1999.

[18] 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.

[19] Chen, L., and Lu, Z. The storage location assignment problem for outbound containers in a maritime terminal. International Journal of Production Economics, 135, pp. 73-80. 2012.

[20] Cheung, R.K., Li, C-L., and Lin, W. Interblock crane deployment in container terminals. Transportation Science, 36, pp. 79-93. 2002.

[21] Choe, R., Park, T., Oh, M.-S., Kang, J., and Ryu, K.R. Generating a rehandling-free intra-block remarshaling plan for an automated container yard. Journal of Intelligent Manufacturing, 22, pp. 201-217. 2011.

[22] Chu, C.Y., and Huang, W.C. Determining container terminal capacity on the basis of an adopted yard handling system. Transport Reviews, 25, pp. 181-199. 2005.

[23] De Castilho, B., and Daganzo, C.F. Handling Strategies for import containers at Marine Terminals. Transportation Research Part B, 27, pp. 151-166. 1993.

[24] Dekker, R., Voogd, P., and Asperen, E. Advanced methods for container stacking. OR Spectrum, 28, pp. 563-586. 2006.

[25] Dorndorf, U., and Schneider, F. Scheduling automated triple cross-over stacking cranes in a container yard. OR Spectrum, 32, pp. 617-632. 2010.

[26] 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.

[27] 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, Chicago, USA. 1999.

[28] Evers, J.J.M. and Koppers, S.A.J. Automated guided vehicle traffic control at a container terminal. Transportation Research Part A, 30, pp. 21-34. 1996.

[29] 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, pp. 565-573. 2000.

[30] Gambardella L.M., Rizzoli A.E., and Zaffalon M. Simulation and planning of an intermodal container terminal. Simulation, 71, pp. 107-116. 1998.

[31] Grunow, M., Günther H.O., and Lehmann, M. Dispatching multi-load AGVs in highly automated seaport container terminals. OR Spectrum, 26, pp. 211-235. 2004.

[32] Grunow, M., Günther H.O., and Lehmann, M. Strategies for dispatching AGVs at automated seaport container terminals. OR Spectrum, 28, pp. 587-610. 2006.

[33] Günther, H.O., and Kim, K.H. Container terminals and terminal operations. OR Spectrum, 28, pp. 437-445. 2006.

[34] Guo, X., and Huang, S.Y. Dynamic Space and Time Partitioning for Yard Crane Workload Management in Container Terminals. Transportation Science, 46, pp. 134-148. 2012.

[35] Han, Y., Lee, L.H., Chew, E.P., and Tan, K.C. A yard storage strategy for minimizing traffic congestion in a marine container transshipment hub. OR Spectrum, 30, pp. 697-720. 2008.

[36] Jiang, X., Lee, L.H., Chew, E.P., Tan, K.C., and Han, Y. A container yard storage strategy for improving land utilization and operation efficiency in a transshipment hub port. European Journal of Operational Research, 211, pp. 64-73. 2012.

[37] Jiang, X., Chew, E.P., Lee, L.H., and Tan, K.C. Flexible space-sharing strategy for storage yard management in a transshipment hub port. OR Spectrum, 35, pp. 417-439. 2013.

[38] Jung, S.H., and Kim, K.H. Load scheduling for multiple quay cranes in port container terminals. Journal of Intelligent Manufacturing, 17, pp. 479-492. 2006.

[39] Jung, D.H., Park, Y-M., Lee, B.K., Kim, K.H., Ryu, K.R. A quay crane scheduling method considering interference of yard cranes in container terminals. In: Gelbukh A, Reyes-Garcia CA (eds) MICAI 2006: Advances in Artificial Intelligence, Proceedings of the fifth Mexican International Conference on Artificial Intelligence, Apizaco, November 13-17. Springer, Berlin, pp. 461-471. 2006.

[40] Kang, S., Medina, J.C., and Ouyang, Y. Optimal operations of transportation fleet for unloading activities at container ports. Transportation Research Part B, 42, pp. 970-984. 2008.

[41] Kang, J., Ryu, K.R., and Kim, K.H. Deriving stacking strategies for export containers with uncertain weight information, Journal of Intelligent Manufacturing, 17, pp. 399-410. 2006.

[42] Kim, K.H. Evaluation of the number of reshuffles in storage yards. Computers and Industrial Engineering, 32, pp. 701-711. 1997.

[43] Kim, K.H., and Bae, J.W. Re-marshaling export containers in port container terminals. Computers and Industrial Engineering 35, pp. 655-658. 1998.

[44] 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, pp. 1-25. 1999.

[45] Kim, K.H. and Bae, J.W. A Look-Ahead Dispatching Method for Automated Guided Vehicles in Automated Port Container Terminals. Transportation Science, 38, pp. 224-234. 2004.

[46] 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.

[47] Kim, K.H. and Kim, K.Y. Routing straddle carriers for the loading operation of containers using a beam search algorithm. Computers and Industrial Engineering, 36, pp. 109-136. 1999a.

[48] 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, pp. 425-433. 1999b.

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

[50] 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. 1999d.

[51] Kim, K.H., and Kim, H.B. The optimal sizing of the storage space and handling facilities for import containers. Transportation Research Part B, 36, pp. 821-835. 2002.

[52] 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.

[53] Kim, K.H., and Hong, G.P. A heuristic rule for relocating blocks, Computers and Operations Research, 33, pp. 940-954. 2006.

[54] 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, pp. 283-292. 2003.

[55] Kim, K.H., and Park, K.T. A note on a dynamic space-allocation method for outbound containers. European Journal of Operational Research, 148, pp. 92-101. 2003.

[56] Kim, K.H., Park, Y.M., and Jin, M.J. An optimal layout of container yards. OR Spectrum, 30, pp. 675-695. 2008.

[57] Kim, K.H., Park, Y.M., and Ryu, K.R. Deriving decision rules to locate export containers in container yards. European Journal of Operational Research, 124, pp. 89-101. 2000.

[58] 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, pp. 193-209. 2004.

[59] Kozan, E. Comparison of analytical and simulation planning models of seaport container terminals. Transportation Planning and Technology, 20, pp. 235-248. 1997.

[60] Kozan, E., and Preston, P. Genetic algorithms to schedule container transfers at multimodal terminals. International Transactions in Operational Research, 6, pp. 311-329. 1999.

[61] Kozan, E., and Preston, P. Mathematical modeling of container transfers and storage locations at seaport terminals. OR Spectrum, 28, pp. 519-537. 2006.

[62] Ku, L.P., Lee, L.H., Chew, E.P., and Tan, K.C. An optimization framework for yard planning in a container terminal: case with automated rail-mounted gantry cranes. OR Spectrum, 32, pp. 519-541. 2010.

[63] Lee B.K., and Kim K.H. Optimizing the block size in container yards. Transportation Research Part E, 46, pp. 120-135. 2010.

[64] Lee, D-H., Cao, Z., and Meng, Q. Scheduling of two-transtainer systems for loading outbound containers in port container terminals with simulated annealing algorithm. International Journal of Production Economics, 107, pp. 115-124. 2007.

[65] Lee, Y., and Chao, S.L. A neighborhood search heuristic for pre-marshalling export containers. European Journal of Operational Research, 196, pp. 468-475. 2009.

[66] 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, pp. 539-561. 2006.

[67] Lee, L.H., Chew, E.P., Tan, K.C., Huang, H.C., Lin, W., and Han, Y. A simulation study on the uses of shuttle carriers in the container yard, in: Proceedings of the 39th Conference on Winter Simulation, pp. 1994-2002. 2007.

[68] Lee, L.H., Chew, E.P., Cheng, H.X., Han, Y. A study on port design automation concept, in: Proceedings of the 40th Conference on Winter Simulation, pp. 2726-2731. 2008.

[69] Lee, Y., and Hsu, N.Y. An optimization model for the container pre-marshalling problem. Computers and Operations Research, 34, pp. 3295-3313. 2007.

[70] Lee, Y., and Lee, Y.-J. A heuristic for retrieving containers from a yard. Computers and Operations Research, 37, pp. 1139-1147. 2010.

[71] 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.

[72] Levinson, M. The Box - How the Shipping Container Made the World Smaller and the World Economy Bigger. Princeton University Press. 2006.

[73] Li, C.L. and Vairaktarakis, G.L. Loading and unloading operations in container terminals. IIE Transactions, 36, pp. 287-297. 2004.

[74] 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.

[75] Lim, A., and Xu, Z. A critical-shaking neighborhood search for the yard allocation problem. European Journal of Operational Research, 174, pp. 1247-1259. 2006.

[76] Linn, R., Liu, J-Y., Wan, Y-W., Zhang, C., and Murty, K.G. Rubber tired gantry crane deployment for container yard operation. Computers and Industrial Engineering, 45, pp. 429-442. 2003.

[77] Linn, R.J., and Zhang, C. A heuristic for dynamic yard crane deployment in a container terminal. IIE Transactions, 35, pp. 161-174. 2003.

[78] Liu, C.I., Jula, H., and Ioannou, P.A. Design, simulation, and evaluation of automated container terminals. IEEE Transactions on Intelligent Transportation Systems 3, pp. 12-26. 2002.

[79] Liu, C.I., Jula, H., Vukadinovic, K., and Ioannou, P. Automated guided vehicle system for two container yard layouts. Transportation Research Part C: Emerging Technologies, 12, pp. 349-368. 2004.

[80] 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, pp. 309-324. 2003.

[81] Murty, K.G. Yard crane pools and optimum layouts for storage yards of container terminals. Journal of Industrial and Systems Engineering, 1, pp. 190-199. 2007.

[82] Murty, K.G., Liu, J., Wan, Y.W., and Linn, R. A decision support system for operations in a container terminal. Decision Support Systems, 39, pp. 309-332. 2005a.

[83] Murty, K.G., Wan, Y.W., Liu, J., Tseng, M.M., Leung, E., Lai, K.-K., and Chiu, H.W.C. Hongkong International Terminals gains elastic capacity using a data intensive decision-support system. Interfaces, 35, pp. 61-75. 2005b.

[84] Nam, K.C., and Ha, W.I. Evaluation of handling systems for container terminals. Journal of Waterway, Port, Coastal, and Ocean Engineering, 127, pp. 171-175. 2001.

[85] Narasimhan, A. and Palekar, U.S. Analysis and algorithms for the transtainer routing problem in container port operations. Transportation Science, 36, pp. 63-78. 2002.

[86] Ng, W.C. Crane scheduling in container yards with inter-crane interference. European Journal of Operational Research, 164, pp. 64-78. 2005.

[87] Ng, W.C. and Mak, K.L. An effective heuristic for scheduling a yard crane to handle jobs with different ready times, Engineering Optimization, 37, pp. 867-877. 2005a.

[88] Ng, W.C., and Mak, K.L. Yard crane scheduling in port container terminals. Applied Mathematical Modeling, 29, pp. 263-276. 2005b.

[89] Nishimura, E., Imai, A., and Papadimitriou, S. Yard trailer routing at a maritime container terminal. Transportation Research Part E, 41, pp. 53-76. 2005.

[90] 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.

[91] Park, T. Choe, R., Ok, S.M., and Ryu, K.R. Real-time scheduling for twin RMGs in an automated container yard. OR Spectrum, 32, pp. 593-615. 2010.

[92] Park, T., Choe, R., Kim, Y.H and Ryu, K.R. Dynamic adjustment of container stacking policy in an automated container terminal. International Journal of Production Economics, 133, pp. 385-392. 2011.

[93] Parola, F., and Sciomachen, A. Intermodal container flows in a port system network: analysis of possible growths via simulation models. International Journal of Production Economics, 97, pp. 75-88. 2005.

[94] Petering, M.E.H. Effect of block width and storage yard layout on marine container terminal performance. Transportation Research E, 45, pp. 591-610. 2009.

[95] Petering, M.E.H. Decision support for yard capacity, fleet composition, truck substitutability, and scalability issues at seaport container terminals. Transportation Research Part E, 47, pp. 85-103. 2011.

[96] Petering, M.E.H., and Murty, K.G. Effect of block length and yard crane deployment systems on overall performance at a seaport container transshipment terminal. Computers and Operations Research, 36, pp. 1711-1725. 2009.

[97] Petering, M.E.H., Wu, Y., Li, W., Goh, M., and de Souza, R. Development and simulation analysis of real-time yard crane control systems for seaport container transshipment terminals. OR Spectrum 31, pp. 801–835. 2009.

[98] Preston, P., and Kozan, E. An approach to determine storage locations of containers at seaport terminals. Computers and Operations Research, 28, pp. 983-995. 2001.

[99] Reveliotis, S.A. Conflict resolution in AGV systems. IIE Transactions, 32, pp. 647-659. 2000.

[100] Saanen, Y., and van Valkengoed, M. Comparison of three automated stacking alternatives by means of simulation. Proceedings of the 37th Conference on Winter Simulation, Orlando, December 4-7, ACM, New York, pp. 1567-1576. 2005.

[101] Sauri, S., and Martin, E. Space allocating strategies for improving import yard performance at marine terminals. Transportation Research Part E, 47, pp. 1038-1057. 2011.

[102] Schneidereit, G., Voβ, S., Parra, A., and Steenken, D. A general pickup and delivery problem for automated guided vehicles with multiloads: a case study. Working paper, University of Hamburg. 2003.

[103] Sculli, D., and Hui, C.F. Three dimensional stacking of containers. Omega International Journal of Management Science, 16, pp. 585–594. 1988.

[104] Stahlbock, R., and Voβ, S. Operations research at container terminals: a literature update. OR Spectrum, 20, pp. 1-52. 2008.

[105] Steenken, D. Fahrwegoptimierung am Containerterminal under Echtzeitbedingungen. OR Spektrum, 14, pp. 161-168. 1992.

[106] Steenken, D., Henning, A., Freigang, S., and Voβ, S. Routing of straddle carriers at a container terminal with the special aspect of internal moves. OR Spektrum 15, pp. 167-172. 1993.

[107] Steenken, D., Voβ, S., and Stahlbock, R. Container terminal operation and operations research - a classification and literature review. OR Spectrum, 26, pp. 3-49. 2004.

[108] Sun, Z., Lee, L.H., Chew, E.P., and Tan, K.C. MicroPort, A general simulation platform for seaport container terminals. Advanced Engineering Informatics, 26, pp. 80-89. 2012.

[109] Taleb-Ibrahimi, M., de Castilho, B., and Daganzo, C.F. Storage space vs handling work in container terminals. Transportation Research Part B, 27, pp. 13-32. 1993.

[110] Van der Meer, R. Operational control of internal transport, ERIM Ph.D. series Research in Management 1. 2000.

[111] Vis, I.F.A. A comparative analysis of storage and retrieval equipment at a container terminal. International Journal of Production Economics, 103, pp. 680-693. 2006.

[112] Vis, I.F.A., and Carlo H.J. Sequencing Two Cooperating Automated Stacking Cranes in a Container Terminal. Transportation Science, 44, pp. 169-182. 2010.

[113] 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.

[114] Vis, I.F.A., de Koster, R. and Savelsbergh, M.W.P. Minimum vehicle fleet size under time-window constraints at a container terminal. Transportation Science, 39, pp. 249-260. 2005.

[115] Vis, I.F.A., de 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. 2011.

[116] Vis, I.F.A., and Harika, I. Comparison of vehicle types at an automated container terminal. OR Spectrum, 26, pp. 117-143. 2004.

[117] Vis, I.F.A., and Roodbergen, K.J. Scheduling of Container Storage and Retrieval. Operations Research, 57, pp. 456-467. 2009.

[118] Wan, Y., Liu J., and Tsai P.C. The Assignment of Storage Locations to Containers for a Container Stack. Naval Research Logistics, 56, pp. 699-713. 2009.

[119] Watanabe, I. Characteristics and analysis method of efficiencies of container terminal: an approach to the optimal loading/unloading method. Container Age, 3, pp. 36-47. 1991.

[120] Yang, C., Choi, Y., and Ha, T. Simulation-based performance evaluation of transport vehicles at automated container terminals. OR Spectrum, 26, pp. 149-170. 2004.

[121] Yang, J.H., and Kim, K.H. A grouped storage method for minimizing relocations in block stacking systems. Journal of Intelligent Manufacturing, 17, pp. 453-463. 2006.

[122] Zaffalon, M., Rizzoli, A.E., Gambardella, L.M., and Mastrolilli, M. Resource allocation and scheduling of operations in an intermodal terminal. In the proceeding of 10th European Simulation Symposium and Exhibition, Simulation in Industry, Nottingham, United Kingdom, pp.520-528, 1998.

[123] Zhang, C., Liu, J., Wan, Y.W., Murty, K.G., and Linn, R.J. Storage space allocation in container terminals. Transportation Research Part B, 37, pp. 883-903. 2003.

[124] Zhang, C., Wan, Y., Liu, J. and Linn, R.J. Dynamic crane deployment in container storage yards. Transportation Research Part B, 36, pp. 537-555. 2002.

[125] Zyngiridis, I. Optimizing container movements using one and two automated stacking cranes. Master's thesis, Naval Postgraduate School, Monterey. 2005.

Appendices

Appendix A: Candidate's publication list arising from the PhD work

Jiang, X., Lee, L.H., Chew, E.P., Tan, K.C., and Han, Y. A container yard storage strategy for improving land utilization and operation efficiency in a transshipment hub port. European Journal of Operational Research, 211, pp. 64-73. 2012.

Jiang, X., Chew, E.P., Lee, L.H., and Tan, K.C. Flexible space-sharing strategy for storage yard management in a transshipment hub port. OR Spectrum, 35, pp. 417-439. 2013.

Jiang, X., Lee, L.H., Chew, E.P., and Tan, K.C. Short-term space allocation for storage yard management in a transshipment hub port. Submitted to European Journal of Operational Research, currently under review. 2012.

Lee, L.H., Chew, E.P., Tan, K.C., Jiang, X.J., and Han, Y. A dynamic yard template approach to container storage management in a transshipment hub. The 3rd International Conference on Transportation and Logistics (T-LOG 2010), May 2010, Fukuoka, Japan.

Jiang, X., Chew, E.P., Lee, L.H., and Tan, K.C. A space-sharing strategy for storage yard management in a transshipment hub port. INFORMS Annual Meeting, December 2011, Charlotte, North Carolina, USA.