# HIERARCHICAL MODELING AND ANALYSIS OF CONTAINER TERMINAL OPERATIONS

by

HACI MURAT ÖZDEMİR

Submitted to the Graduate School of Engineering and Natural Sciences
in partial fulfillment of
the requirements for the degree of
Master of Science

Sabancı University
July 2003

# HIERARCHICAL MODELING AND ANALYSIS OF CONTAINER TERMINAL OPERATIONS

APPROVED BY:

Assis. Prof. Dr. Tonguç Ünlüyurt …………………………. 

(Thesis Advisor)

Associate Prof. Dr. Meltem Denizel ………………………….

Assis. Prof. Dr. Bülent Çatay ………………………….

DATE OF APPROVAL: ………………………….

## ACKNOWLEDGEMENTS

I would like to express my gratitude to all those who gave me the possibility to complete this thesis.

First, I am deeply indebted to my thesis advisor, Assis. Prof. Dr. Tonguç Ünlüyurt for his continuous support, stimulating suggestions and patience. It would not have been possible to complete this thesis without his encouragement and trust. I have learned a lot from his broad and profound knowledge during the long hours that we have spent together.

I would generously like to thank committee members of my thesis, Assoc. Prof. Dr. Meltem Denizel and Assis. Prof. Dr. Bülent Çatay, whose critical suggestions and excellent remarks have contributed to my thesis.

I also wish to acknowledge my teaching supervisor Assis. Prof. Dr. Kemal Kılıç for his endless support and motivation throughout the year.

I would like to appreciate the founders, board of trustees, and administrators of Sabanci University who gave me the chance to do research in an academic environment. I would like to express sincere thanks to the faculty members, graduate students and other staff of the Faculty of Engineering and Natural Sciences, especially to my officemates for their encouragement. I am greatly grateful to Mr. Zafer Gürel for his precious help during the programming work.

I would like to acknowledge Mrs. Nancy Karabeyoğlu whose instructions, corrections, and editorial remarks improved my thesis.

Last but certainly not least, my special thanks go to my family especially to my parents for their financial and spiritual support. Their patient love and trust enabled me to complete this thesis.

**ABSTRACT**

After the breakdown of trade barriers among countries, the volume of international trade has grown significantly in the last decade. This explosive growth in international trade has increased the importance of marine transportation which constitutes the major part of the global logistics network. The utilization of containers and container ships in marine transportation has also increased after the eighties due to various advantages such as packaging, flexibility, and reliability.

Parallel to the container throughput, the capacities of ships and sizes of fleets as well as the number of terminals have been increased considerably. Substantial pressure of competition on ship operators and terminal managers has forced them to consider the issues regarding operational efficiency more deeply. Thus, the operational efficiency at port container terminals has become the major concern of terminal managers to satisfy the rapid transshipment of goods.

In this thesis, we focus on a set of decision problems regarding container terminal operations. Since these problems are interrelated hierarchically, we attempt to model and analyze them consecutively.

First, we consider the storage space allocation problem over a rolling horizon as an aggregate planning model. Since the model has the minimum cost flow network structure there exist polynomial time solution procedures via linear programming models. Although ship turnaround time is the principal performance criteria for whole container terminal operations, the total distances traveled by containers in the terminal throughout the planning horizon is determined as the surrogate objective function for the allocation model.

The output of the storage space allocation problem is used as the input for the next step of our methodology, namely the location matching model. With the location matching model, the routes of vehicles for each time period have been identified while minimizing the total distance traveled by the vehicles, which reveals the ship turnaround times. The routes that are found subject to the output of storage space allocation models are better than those of random allocation in terms of total distances traveled. Next, the vehicle scheduling problem is discussed for different levels of complexity. The solution procedures proposed for similar problems in the machine scheduling literature are provided.

Finally, we discuss the problem of simultaneous vehicle dispatching with precedence constraints. We have modeled the problem as a nonlinear mixed integer programming model and proposed an iterative solution procedure to obtain reasonable solutions in considerable times. Moreover, we have presented the worst-case performance analysis for this heuristic.

# ÖZET

Ülkeler arası ticari engellerin ortadan kalkmasından sonra geçtiğimiz on yılda uluslararası ticaretin hacmi önemli ölçüde büyüdü. Uluslararası ticaretteki bu ciddi büyüme küresel lojistik ağının önemli bir kısmını teşkil eden deniz taşımacılığının önemini arttırdı. Deniz taşımacılığında konteynır ve konteynır gemilerinin kullanımı da paketleme, esneklik ve güvenilirlik gibi sağladığı birçok avantaj nedeniyle seksenli yıllarda arttı.

Konteynırla üretilen işlere paralel olarak gemi kapasiteleri, filo büyüklükleri ve terminal sayısı da ciddi biçimde arttı. Rekabet nedeniyle gemi operatörleri ve terminal yönetenleri üzerinde oluşan büyük baskı onları operasyonel verimlilikle ilgili konulara odaklanmaya zorladı. Böylece liman konteynır terminallerindeki operasyonel verimlilik malların hızlı aktarımını sağlamak için terminal yönetenlerinin temel endişeleri halini aldı.

Bu tezde konteynır terminal operasyonlarıyla ilgili bir dizi karar problemine odaklandık. Bu problemler birbirleriyle hiyerarşisel olarak ilişkili olduklarından ardışık olarak modelledik ve analiz ettik.

İlk olarak, devreden zaman ufkunda depolama alanı ataması problemini bir toplu planlama problemi olarak ele aldık. Bu model minimum maliyet akış ağı yapısı taşıdığından doğrusal programlama yoluyla polinom zamanda çözüm yöntemleri mevcuttur. Tüm konteynır terminal operasyonları için ana performans kriteri gemi dolaşım süreleri olmasına rağmen depolama alanı ataması problemi için vekil amaç fonksiyonu konteynırlar tarafından planlama zamanı boyunca terminalde gezilen mesafeler toplamı olarak belirlendi.

Depolama alanı ataması modellerinin çıktısı yöntemimizin yer eşleştirme olarak adlandırılan bir sonraki adımı için girdi olarak kullandı. Yer eşleştirme problemiyle gemi dolaşım sürelerini açığa çıkaran araçlar tarafından gezilen toplam mesafe en küçüklenirken araçların rotaları belirlendi. En iyilenmiş depo alanı atama çıktılarıyla bulunan rotalar rassal atama çıktılarıyla bulunanlardan daha iyi sonuçlar verdi. Daha sonra farklı karmaşıklık düzeyleri için araç çizelgeleme problemi tartışıldı. Makine çizelgeleme literatüründeki benzer problemlere önerilen çözüm yöntemleri sunuldu.

Son olarak, öncelik kısıtlarıyla eşzamanlı araç sevk etme problemi incelendi. Problemi doğrusal olmayan karışık tamsayılı programlama modeli halinde modelledik ve makul zamanlarda iyi sonuçlar veren bir tekrarlanan sezgisel yöntem önerdik. Ayrıca bu sezgisel için en kötü durum performans analizini de sunduk.

# TABLE OF CONTENTS

## LIST OF FIGURES

# LIST OF TABLES

# 1. INTRODUCTION

## 1.1. Motivation

In the last decade, the globalization of trade has increased the volume and significance of international logistics issues dramatically. Although the global logistics network is an integrated system that comprises various modes of transportation, overseas or marine transportation has become the leading mechanism to handle intercontinental bulk cargo. In marine transportation, a great number of hubs, ports, and terminals serve both shippers and manufacturers to accomplish the rapid delivery of goods.

The greatest portion of international bulk transported overseas is carried in containers. Containers are large, standardized, metal-frame packages for bulk cargo, utilized to transport goods via various modes of transportation such as ships, trucks and rail. Containerization is defined as "*the utilization, grouping or consolidating of multiple units into a larger container for more efficient movement*", according to The Containerization Institute. Containers, which were first introduced in the mid-fifties, had several advantages compared to the former bulk in terms of productivity, packaging costs, and reliability. The introduction of containers speeds up the logistics cycle substantially because they ensure the reduction of time consumed in handling operations at ports, transfer points, and remaining modes of transportation system.

Since their introduction in the fifties, containers are frequently preferred for intercontinental transport. After the eighties, globalization began with the breakdown of trade barriers among countries, so the volume of international overseas transportation has significantly grown. As a result of this explosion in international trade, the size of fleets and the capacity of ships, terminals, and ports have been enlarged tremendously. Container throughput of a particular container terminal is the total number of containers handled by the terminal in a given time period. World container throughput growth and future projections made for the next 15 years in 1995 by Ocean Shipping Consultants are given in the following table.

Table 1.1 World container throughput growth projection

| | Optimistic view | | | Pessimistic view | | |
|---|---|---|---|---|---|---|
| Year | Index | Million TEU* | % Growth | Index | Million TEU | % Growth |
| 1995 | 100 | 142 | | 100 | 142 | |
| 2000 | 156 | 222 | 56 | 156 | 222 | 56 |
| 2005 | 236 | 235 | 51 | 215 | 306 | 38 |
| 2010 | 327 | 465 | 39 | 275 | 391 | 28 |

*Twenty feet equivalent unit

The realized container throughput values for the 1996-2000 period are listed in the following table to indicate the Turkey's portion in the world market.

Table 1.2 Container throughput growth, Turkey vs. the World

| | Container Throughput (1000 TEU) | | |
|---|---|---|---|
| Year | Turkey | World | % |
| 1996 | 555 | 150,753 | 0.37 |
| 1997 | 369 | 160,721 | 0.23 |
| 1998 | 1,262 | 174,880 | 0.72 |
| 1999 | 1,325 | 203,207 | 0.65 |
| 2000 | 1,577 | 225,294 | 0.70 |

The busiest container terminals are located in the Asia-Pacific region, where the volume of international trade is largest. The biggest port container terminals are listed in Table 1.3.

Table 1.3 Port Container Traffic League (year 2000)

| Rank | Port | Country | Throughput % |
|---|---|---|---|
| 1 | Hong Kong | China | 8.03 |
| 2 | Singapore | Singapore | 7.56 |
| 3 | Kaohsiung | Taiwan | 3.34 |
| 4 | Busan | South Korea | 3.29 |
| 5 | Rotterdam | Netherlands | 2.78 |
| 6 | Long Beach | USA | 2.49 |
| 7 | Shanghai | China | 2.16 |
| 8 | Los Angles | USA | 2.04 |
| 9 | Hamburg | Germany | 1.88 |
| 10 | Antwerp | Belgium | 1.81 |

The container ship operators handle the overseas transportation of containers. The container shipping business is also growing in terms of the ship sizes and fleet capacities parallel to the growth in container throughput. The largest container ship liners worldwide are listed with respect to their origin and the capacity.

Table 1.4 World's Leading Container Ship Liners

| Rank | Ship Liner | Origin | Capacity (1000 TEU) | Market Share (%) |
|---|---|---|---|---|
| 1 | MAERSK-SEALAND | DENMARK | 775 | 12.1 |
| 2 | P&O NEDLLOYD / BLUE STAR | ENGLAND | 402 | 6.3 |
| 3 | EVERGREEN / UNIGLORY | TAIWAN | 395 | 6.2 |
| 4 | MSC | SWITZERLAND | 309 | 4.8 |
| 5 | HANJIN / DSR SENATOR | KOREA | 303 | 4.7 |
| 6 | COSCO | CHINA | 251 | 3.9 |
| 8 | NYK | JAPAN | 229 | 3.6 |
| 9 | OOCL | CHINA | 162 | 2.5 |
| 10 | MOL | JAPAN | 148 | 2.3 |

If we observe the Turkish container shipping industry, the indicators regarding the traffic and volume of terminal operations are quite below those of other medians of transportation. The volume of container traffic in Turkish State Terminals covering almost all of the port terminal operations nationwide is illustrated between 1998 and 2002 as follows.

Table 1.5 Volume of Turkish Container Terminal Operations

| Type | Loading | | | | Unloading | | | | Total | |
|---|---|---|---|---|---|---|---|---|---|---|
| | TEU | | 2-TEU | | TEU | | 2-TEU | | | |
| Years | Full | Empty | Full | Empty | Full | Empty | Full | Empty | Quantity | TEU |
| 1998 | 138043 | 54003 | 104040 | 50499 | 119999 | 58506 | 120912 | 25427 | 671429 | 972,307 |
| 1999 | 158545 | 28122 | 125558 | 27460 | 100480 | 77194 | 108216 | 42679 | 668254 | 972,167 |
| 2000 | 154554 | 35482 | 128935 | 43867 | 113896 | 72955 | 138788 | 36632 | 725109 | 1,073,331 |
| 2001 | 165288 | 16363 | 145366 | 16603 | 81731 | 97341 | 96629 | 65793 | 685114 | 1,009,505 |
| 2002 | 185373 | 22642 | 163809 | 25469 | 103642 | 98620 | 123983 | 63014 | 786552 | 1,162,827 |

Since a container terminal is the interface of transshipment of containers from ship to ship or to other modes of transportation such as rail and trucks vice versa, the substantial growth in international trade and overseas transportation reveals the importance of operational efficiency at intermodal terminals. The speed of operations is the most vital criteria in container terminals for both shippers and manufacturers, as it is in other medians of the transportation industry. The hubs of the global logistics network, such as container terminals, should be operated efficiently so as to respond the customers' demand rapidly and to procure the right product at the right time at a compatible cost, the ultimate objective of all logistics activities. Thus, the major goal of the container terminal operators is minimizing the time between the arrival and departure of ships, called turnaround times, while maximizing the utilization of terminal facilities. Since the berthing and terminal operating time of a container ship accounts for the considerable proportion of its overall service time or cycle time for a given route, the main concerns of shipping lines address the operational swiftness at container terminals. Although container terminal managers charge for the duration of stay both for moored ships and stored containers, they try to sustain rapid operations so as to handle more ships and containers per day. As a result, due to the increasing pressure of competition among terminals and emerging capacity limitations globally, container terminal managers now focus on methodologies to increase the terminal throughput and decrease the ship turnaround times.

In order to perform fast and productive operations in a container terminal, an efficient coordination among activities should be maintained. Container terminals are actually complex systems that consist of several subsystems covering interrelated and sequential operations such as berthing, storage, transshipment, and so on. The size and complexity of container terminals necessitate computerized decision support systems to handle a great number of operations in considerable time frames. Depending on the configurations and requirements, various automation tools are employed in today's terminals where most of the processes are performed by state of the art computer applications.

Decisions regarding container terminal operations vary based on the level and consequence of the decision. For instance, macro level issues such as terminal location selection, determination of terminal configuration, and material handling system are strategic. These long-term decisions have been taken by top management. On the other hand, operational decisions at a container terminal are taken for each day, shift, or even more frequently. Storage space allocation, vehicle dispatching, routing vehicles and traffic control are some operational level issues, which should be considered for tight time frames.

### 1.2. Operations at a Container Terminal

Numerous tasks such as unloading/loading ships, transshipment of containers, container storage and retrieval are performed regularly in a typical container terminal. Thus, managing and controlling the components of such a system are complicated due to the large number of operations. Some of the operational level decisions are made after an analysis of alternatives, whereas others are made by the operator responsible for a particular task. For instance, a crane operator can determine the unloading sequence of containers from an arriving ship based on his/her experiences or intuition. However, the loading sequence of containers to a departing ship should be determined after a considerable evaluation since such a decision significantly influences further operational tasks.

A typical port container terminal can be divided into two distinct areas: one at the quayside, and the other at the landside. The quayside of the container terminal, also called the ship area or berth area, where ships are moored and stay during unloading (discharging) and loading (uploading) services. The landside of the terminal area is the storage space of containers, called the yard area, stack, or storage area. Basically, two types of cranes exist in a container terminal: Quay Cranes and Yard Cranes in order to unload and load containers at quayside and yard area, respectively. Dedicated vehicles perform the transshipment of containers between the berth and yard area. Daily processes at a container terminal can be clustered into hierarchical steps in order to analyze this complicated system. Figure 1.1 illustrates the processes at a container terminal.

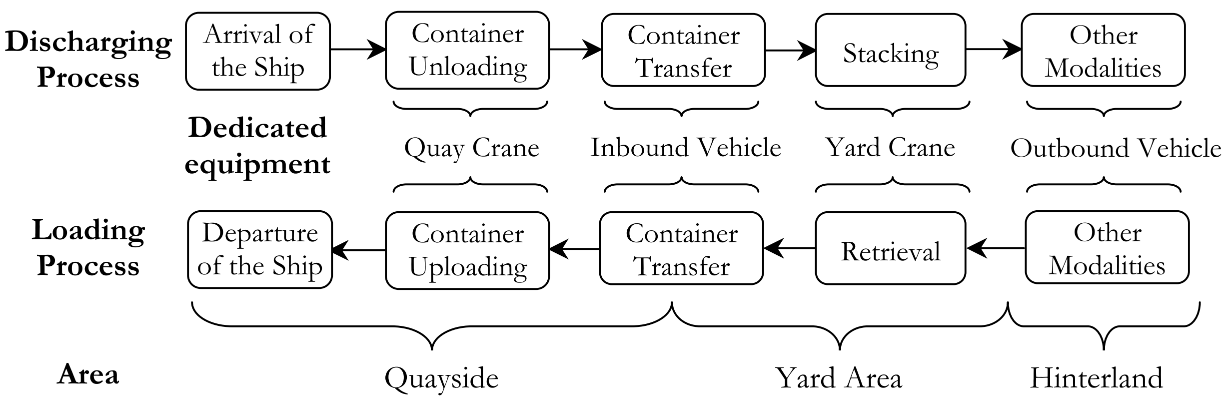



Figure 1.1 Process flow at a container terminal

The order of operations in a container terminal regarding an import/export container can be demonstrated as the reciprocal unloading/loading processes. First, a ship arrives at the port and moors to the berth; this process is called berthing and detailed information about ship content is received a few hours before the arrival of the ship. Thus, the list of containers to be unloaded from the ship and the list of containers to be loaded on the ship as well as their locations at the yard is known. Of course, ship loading begins after all containers in the unload list are already discharged. According to a given unload plan or crane job sequence, containers on the ship are discharged consecutively by manned Quay Cranes (QC). The container taken off by QC is loaded on a vehicle nearby the ship. Dedicated vehicles can be trucks, forklift trucks, straddle carriers, automated guided vehicles (AGVs), or a combination of them depending on the terminal configuration. After the receipt of container by the dispatched vehicle in the berth area (quayside), the container is transported to the yard area to be stacked and stored until departure. If trucks are used for inter-terminal transportation, the container is taken off from the truck by a Yard Crane (YC), and then loaded on a predetermined location in the yard area. Forklifts and straddle carriers are able to stack containers in the yard area, systems with AGVs are obviously served by automated stacking cranes (ASCs), and some other transporter-stacker pairs are also valid in practice. Time consumed during the container positioning operation at the yard area varies due to the existence of some re-handling moves on the stack. After a certain storage period, the container is retrieved from the stack and transported to the other modalities or to another departing ship. Thus, the unloading and transshipment process of a container is completed. Performing this process backwards demonstrates the process of loading a ship. We assume in our models that identical quay cranes, yard cranes, and trucks are used as container handling equipments for the terminal operations.

## 1.3. General Approach

Unfortunately, modeling the whole range of operations in a container terminal and solving the model to optimality are beyond today's computational capabilities. Most studies investigating terminal systems in the literature focus on a single problem or a small subset of problems such as container loading/unloading, vehicle dispatching or crane scheduling. Due to the interrelation among decision problems throughout the process flow, the output of a primary level problem presents the input to a succeeding decision. Hence, decisions regarding operational efficiency and corresponding optimization models for facilities should be ordered hierarchically from general to specific level problems. Zhang et al. (2001) propose a hierarchical structure for interrelated decisions for terminal operations as illustrated in Figure 1.2.

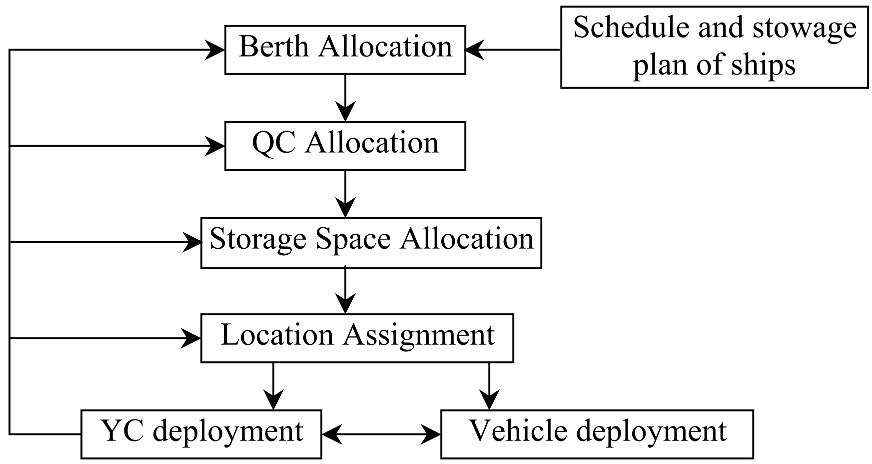



Figure 1.2 Hierarchical structure of operational decisions in a container terminal

Our approach in this study is also hierarchical, where storage space allocation, location matching and vehicle scheduling decisions are made consecutively. Although modeling of an integrated space allocation – location matching problem is possible; the complexity of solving such a huge model is beyond today's computational capacity in terms of considerable completion times of relevant experiments. Therefore, problems are decomposed into subproblems and a hierarchical modeling approach is used in this study.

**i. Storage Space Allocation**

An aggregate space allocation model over a rolling horizon is constructed. Since unloading and loading job sequences for each ship is known prior to the berthing, arrival and departure period for each and every container is known as well. Any container stored in the yard seizes the storage space between stacking and retrieval. Thus, storage space is reserved for the time periods within arrival and departure. A great number of locations are available at the yard area, and each location's distance from the berth is also known. The aggregate space allocation model returns the locations for each type of container, where the type of a container indicates the arrival and departure periods, so as to minimize the total distance traveled throughout the planning horizon. It is assumed that parameters such as number of containers arriving and departing at a particular time period are not certain over the rolling horizon. Therefore, the parameter update for each time period is inevitable. In our models, numerical experiments are done without any incoming information regarding parameters since the inconsistency in dynamic data is not predictable.

### ii. Location Matching

After completing the storage space allocation models, we extract the possible storage locations for each type of container. Although the results of space allocation are aggregate, they reveal reasonable inputs for the location matching problem so as to minimize total distance traveled by containers for each period. Of course, this objective also fulfills the overall objective of a container terminal, which is minimizing ship turnaround times. To minimize the total transshipment time of vehicles in a container terminal, the most efficient routes should be determined. The efficiency of a particular route can be defined as throughput, the number of containers traveled per unit distance. Thus the inefficient or empty travels should be minimized to increase the throughput of a vehicle. Our location matching model is based on the philosophy of finding routes, i.e. pairs of locations between which the distances traveled without the container will be minimum. The input of such a model is the unloading and loading locations over the planning horizon, and the output are the pairs of locations, which correspond to the routes or jobs to be handled by vehicles.

### iii. Vehicle Scheduling

Eventually, the routes found in the preceding model should be scheduled on vehicles and other equipment such as cranes. Vehicle scheduling can be carried out via three different scenarios. Because the first scenario ignores the equipment other than vehicles, the problem becomes scheduling identical vehicles such as parallel machine scheduling. In the second scenario, there is a quay crane constraint and each container should be scheduled on quay cranes. Thus, the problem is similar to another NP-hard problem; parallel machine scheduling with common servers. The third and the more complicated case is when parallel servers exist at each side of vehicle scheduling: yard cranes are also considered as constraints so the problem becomes the integrated scheduling of terminal equipment.

### iv. Simultaneous Vehicle Dispatching with Precedence Constraints

Models defined so far disregard the precedence relationship among containers or pairs of containers handled consecutively. In practice, precedence relationships are present both for unloading and loading job sequences. Since the loading job sequence is hardly flexible and stricter than the unloading job sequence, precedence relationship among containers that will be uploaded should be considered. Modeling simultaneous unloading and loading operations with precedence constraints requires scheduling constraints that reveal the precedence relationships and integer programming constraints that satisfy the assignment of each job to a particular position in the operations sequence. However, different heuristic methods are required to solve such models in reasonable times since they are nonlinear integer programming models. Solving assignment models iteratively and updating the model parameters after completing each iteration is the main motivation to handle such a model.

## 1.4. Thesis Outline

The remainder of this study is organized as follows. Chapter 2 provides a detailed literature review. Chapter 3 describes the details of the container terminal framework considered in this study. Storage space allocation models for both simple and extended cases are analyzed in Chapter 4. In Chapter 5, location matching models are proposed to illustrate considerable routes for vehicles, and the comparison of results associated with random allocation and optimal allocation scenarios is also given at the end of Chapter 5. The scheduling of vehicles with respect to general parallel machine fashion is performed via a simple heuristic (LTT) in Chapter 6; non-optimality and worst-case analysis of LTT are also illustrated referring to the scheduling literature. Quay crane and yard crane constraint scheduling issues are also proposed in Chapter 6. In Chapter 7, simultaneous vehicle dispatching for unloading and loading operations with precedence constraints is proposed via an iterative solution procedure. The thesis concludes with the remarks regarding research directions in future studies.

# 2. LITERATURE REVIEW

In this chapter, a review of published material on container terminal operations will be presented from the operations research / management science perspective. Although there are a great number of studies both in industry and academia regarding container terminals, this review focus on the research covering the issues supported by the operations research techniques.

As a comprehensive decision making methodology, operations research / management science contributes numerous solution approaches to the decision problems for container terminal operations from the strategic to the operational level. Although such studies seem to be a niche area for operations researchers, a great number of refereed journal and conference papers, industry projects as well as many of M.Sc. and Ph.D. theses are found in the literature.

This literature review section is structured according to the sequence of operations at container terminals.  Initially, the arrival of ships, berth and crane allocation problems are discussed, then the literature regarding container loading and unloading from/to ships is summarized. Next, the literature considering space allocation problems and studies associated with material handling systems are proposed consecutively. More minor operations at the yard side, such as stacking and re-handling, are illustrated in a separate section. This review concludes with the studies comprising whole container terminal operations or a large set of decision problems together.

## 2.1. Process Flow at a Typical Container Terminal

As mentioned earlier, the order of operations in a container terminal regarding an import/export container can be demonstrated as reciprocal unloading/loading sequences. First, a ship arrives to the port; this process is called berthing and various studies can be found mostly focusing on queuing theory for the berthing of ships since the process is analogically identical to the server-customer interaction since the berths serve the ships and the time between arrivals is random in most cases. Then, according to a given unload plan, containers on the ship are discharged consecutively by manned Quay Cranes (QC). The berth and QC allocations and load/unload plans are determined well before the arrival/departure of the ships. The container, taken off by the pre-assigned QC, is loaded on a vehicle nearby the ship; numerous studies discussing this type of container unloading and vehicle dispatching can be found in the literature. These allocated vehicles can be trucks, forklift trucks, straddle carriers, automated guided vehicles(AGVs), or a combination of them, depending on the terminal configuration. After the receipt of container by the dispatched vehicle in the berth area (quayside), the container is transported to the yard area where they will be stacked onto each other and stored until departure. Container transshipment between quayside and yard area is also a deeply investigated area of study. If trucks are used for inter-terminal transportation, the container is taken off from the truck by a Yard Crane (YC), and then loaded on the predetermined location in the yard area. In some configurations mostly used in North American terminals, forklifts and straddle carriers are able to stack containers in the yard area, systems with AGVs are obviously served by automated stacking cranes (ASCs), and some other transporter-stacker combinations are also used in actual cases. A great number of researchers focus on the container storage and retrieval operations at the yard area. In practice, transfer systems composed of two separate equipments for transportation and stacking, i.e. truck-yard crane, are called Indirect Transfer systems whereas the systems using single multipurpose equipment such as straddle carriers are called Direct Transfer systems. Time consumed during the container positioning operation at the yard area varies due to some re-handling moves on the stack. After a certain storage period, the container is retrieved from the stack and transported to other modalities or to another departing ship. Thus, the unloading and transshipment process of a container is completed. Performing this process flow backwards reveals the process of loading a ship.

## 2.2. Arrival Process, Berth and Crane Allocation

Process flow of a container at a terminal commences with the arrival of the ship. After having arrived, the ships stay for an almost uncertain time period to complete unloading and loading services at the terminal. Since a fixed number of berths are available at the quayside of the container terminal, an arriving ship has to wait until a berth becomes free to moor. Determination of the number of berths at a container terminal is a strategic decision that has to be made prior to the terminal construction. Edmond and Maggs' (1978) evaluation of queuing models could be useful in deciding on the number of berths that should be available at the quayside. They mention that some of the proposed models could be used when the model and parameters are chosen carefully and the results are evaluated precisely. Decisions regarding the type and number of material handling equipment at the quayside are also strategic level. The allocation of berths to the ships, exact number of quay cranes that work simultaneously on a ship and the assignment of quay cranes to the holds of ships are other decisions for the quayside operational problems.

Imai et al. (1997) study the problem of allocating berths to ships while optimizing the berth utilization. There may be two different scenarios regarding the berth allocation: first scenario allocates berths based on the order of arrivals according to the first come first served principle; second strategy ignores the order of arrivals and assigns ships to berths based on the closeness of staking area that most of the containers will be stored. Thus, ships' waiting times lengthen while the terminal utilization will be maximal. Conflicting objectives of terminal management and ship owners due to the trade off between the total staying time in the port and dissatisfaction of ship owners caused by the order in which the ships are berthed, could be considered as a multi-objective machine scheduling problem. Imai et al. (1997) formulate a biobjective nonlinear integer program to identify the set of non-inferior berth allocation, which minimizes the dual objectives of overall staying time and dissatisfaction on order of berthing. Overall staying time is the sum of staying, waiting, and berthing times where dissatisfaction equals the sum of the number of cases in which a ship arrives later and mooring earlier than a particular ship. The biobjective problem is reduced to a single objective problem where the single objective problem is similar to the assignment problem. The objective function is the sum of staying times plus the sum of dissatisfactions. Varying the weights identifies the set of non-inferior trade offs between the first and the second terms of the objective function. Numerical experiments show that the trade off increases if the size of the port increases.

Daganzo (1989) examines the crane scheduling problem at ports. He considers a berth of fixed length with a fixed number of cranes serving a number of ships. Ships are divided into holds with only one crane working one a hold at a time. Cranes can be moved from hold to hold quickly compared to the time that it takes to handle one hold. In most cases, ships arrive at different times and must queue for berthing space if the berths are full. It is mentioned that crane scheduling problem seems related to the queuing theory and machine scheduling problems for dynamic and static cases respectively. As the first stage of his study, static crane allocation problem is discussed and exact solution approach for problems with few ships via mixed integer programs is proposed. Optimal or near optimal results were obtained for several numerical examples. Although the performance was not tested, the proposed method is also expected to be effective for large size problems. In the second part, he provides a principle-based technique for the dynamic crane allocation case when the berths can hold only a fixed number of ships and queuing ships join the berths in the order of arrival. The technique is based on the methods proposed in the static case; the objective is still minimizing the ship delays. The results show that crane idle time is minimized and berth throughput is maximized, both of which reduce queuing delay as well.

A branch and bound solution method for a class of static crane allocation problems considered by Daganzo (1989) is studied further in Peterkofsky and Daganzo (1990). The static allocation model is formulated so as to minimize the weighted amount of time that ships spend at the port. The branch and bound method determines the best possible ship departure schedule. Dominance of infeasible solutions, boundary points, and construction of the branch and bound tree are illustrated. Branching procedures such as node selection, pruning and termination are explained with an example according to the proposed methodology. In order to determine the feasibility, the set of constraints are analyzed one by one. The feasibility of a set of constraints that have a capacitated transportation problem structure is checked via solving a derivative maximum flow problem by labeling algorithm. Computational performance of the method is evaluated by ten problems of different sized ships. Computational results for realistically sized problems with up to six ships using a microcomputer (6 Mhz) are satisfactory. Although the method is efficient for some cases, it does not address all crane scheduling problems, especially for the large and complex cases, due to time and memory resources.

Legato and Mazza (2001) focus on the berth planning subsystem of Gioia Tauro (Italy) container terminal for designing a specialized quantitative model for bottleneck analysis, operations management, and resources capacity optimization. A closed queuing network model is proposed to estimate congestion effects on the dwelling time of ships belonging to a given shipping company, out of a fixed number visiting the terminal. Visual SLAM language is used to simulate the queuing network approach in a modular implementation of system's processes description and interaction. The discrete event simulation model represents berthing policy with priorities, multiple crane allocation and non-exponentially distributed time between arrivals of major ships. After validation against actual data, the model is used for scenario analysis for berth planning and resources optimization via the "what-if" approach. Also, simulation tool has been shown to be effective in estimating how resource capacity upgrades and modifying resource allocation policies affect performance levels.

### 2.3. Container Unloading/Loading, Ship Stowage

In practice, the number and the specifications of containers that have to be unloaded are known shortly before the arrival of the ship. The unload plan, which is determined first, identifies which containers to be unloaded and in which hold they are positioned in the ship. Since the quay crane assignment to the holds of the ship is already done before unloading a ship, the quay crane operator successively unloads the containers for each hold. Within a hold, the operator is almost free to determine the order in which the containers can be unloaded. Thus, the containers are picked up according to their accessibility at the ship while maintaining the ship balance and some specific restrictions depending on the contents of the container. Since the unloading time of a container is directly proportional to its location on the ship, a large variance may occur in container unloading times. On the other hand, container loading process is hardly flexible and a good distribution of containers on the ship should be determined to ensure fast and efficient transshipment of containers. The stowage planning decision influences the operational times both at the current terminal and incoming terminals. Stacking later departing containers on top of the earlier ones may cause inefficient quay crane moves at the subsequent terminals on the route of the ship. As an operational level decision, container stowage planning should be studied so as to minimize overall loading / unloading times. Shields (1984) presents a computer-aided stowage planning system where the physical limitations of the ships and containers as well as the visiting sequence of ship are considered. This assistant system uses the Monte Carlo method; the most efficient ship loading sequence is selected among different possible ones. For every container, the exact place on the ship is indicated and the most efficient loading plan is displayed with the precise loading order of containers. This system has been used worldwide since 1981.

Wilson and Roach (2000) decouples the ship stowage problem into strategic and tactical planning process. Since finding an optimal solution for the overall stowage problem in reasonable times is not realistic, the following approach is proposed. In the first step, the containers are assigned to the blocks at the ship. Secondly, containers are assigned to the exact locations within the predetermined blocks. The branch and bound method and tabu search are used to find good solutions within reasonable time for the strategic and tactical steps respectively.

Penn et al. (2000) discuss the container ship stowage problem, its complexity, and connection to the coloring of circle graphs. The shifting of containers on board is defined as the temporary removal from and placement back of containers onto a stack of containers. For instance, if a container is placed on a vertical stack has a destination of $j$, while the containers stacked on it have destinations further than $j$, the latter containers should be shifted. Although shifting cost could be considerable for large ships, container stowage placement decisions are based on port operations efficiency and ship stability, but not enough attention has been given to minimize the number of shifts for a particular route. The computational complexity of this optimization problem is addressed. Penn et al. (2000) show that the problem is NP-complete, where a polynomial time algorithm for single column case exists. Also, they derive upper and lower bounds on the number of columns for which a plan can be found in polynomial time that will result in zero shifts. Further, they show that finding the minimum number of columns for which there is a zero shifts stowage plan is equivalent to finding the coloring number of circle graphs.

Chen et al. (1999) consider discharging and uploading containers to and from ships in a working paper. The problem is to dispatch vehicles to the containers so as to minimize the total turnaround time of a ship, which is the total time it takes to discharge all containers from the ship and upload new containers to the ship. They propose easily implementable heuristic algorithms and identify the absolute and asymptotic worst-case performance ratios of these heuristics. Dispatching first available vehicle to a discharging job is proposed as greedy algorithm, which is optimal for discharging job sequences whereas the reversed greedy algorithm is optimal for uploading job sequences for single crane case. For the combined job sequences, asymptotic optimality of the combined algorithm as well as the optimality of combined greedy algorithm is proven. Although it is simple, the greedy algorithm finds near optimal solutions for the multiple crane case. To get rid of the myopic nature of the greedy algorithm, a refined greedy algorithm is proposed and more satisfactory results are found with an average deviation of 1.55% from the optimality.

Li and Vairaktarakis (2001) analyze the same problem as of Chen et al. (1999), and improve the algorithms in terms of computational times and lower bounds. They propose FAT (First Available Truck) and LBT (Last Busy Truck) rules corresponding to the Greedy and Reversed Greedy algorithms of Chen et al. (1999) for the discharging and uploading job sequences respectively. In order to propose an optimal algorithm for the single crane case, they apply FAT and LBT rules to discharging and uploading job sequences respectively, and then concatenate the possible pairs of terminal discharging jobs set with leading uploading jobs set by solving a bottleneck assignment problem illustrated in Ahuja et al. (1993). The overall complexity of this solution method is $O(n^2m-1(n+m^{2.5}\log m)$, and reveals a significant improvement of one provided by Chen et al. (1999), which is $O(n^3m+1)$ where $n$ and $m$ denote the number of jobs and number of trucks respectively. The computational time of the proposed algorithm becomes $O(n^2m-1)$, which is polynomial in *n*. Three different heuristic algorithms are developed to find efficient solution procedures. The first algorithm matches the terminal discharging jobs with leading uploading jobs arbitrarily after having conducted the FAT and LBT rules respectively. They prove that $Z^H / Z^* \leq 2$ where $Z^H$ and $Z^*$ represents the objective function values of heuristic and optimal solutions respectively. Also, they indicate that this bound is tight for the first heuristic, whereas Chen et al. (1999) show that this heuristic (combined algorithm) has worst-case error bound of 200% (i.e. $Z^H / Z^* \leq 3$) with a running time of $O(n)$. The second heuristic uses the optimal matching with respect to the bottleneck assignment problem after applying the FAT and LBT rules. The lower bound of the first heuristic remains equal for the second heuristic, while the computational complexity becomes $O(\max\{n, n^{2.5}\log m\})$. The last heuristic uses a lower bound algorithm to generate more suitable terminal discharging and leading uploading sequences and then matches them optimally. Complexity of this heuristic is $O(n^4 \log n \log m)$. Computational results indicate that their optimal algorithm is efficient when the number of trucks is small such as 2 or 3. All proposed heuristics are effective and the last heuristic dominates the first and second heuristics in terms of performance. Also, they prove that the problem of minimizing the time to unload and load a ship is NP-hard for two crane case.

Bish et al. (2001) consider discharging containers from a ship and locating them in the terminal, and they propose a new vehicle scheduling and location problem. A crane job sequence given prior to the unloading operations determines the order of containers to be unloaded from the ship and each container has a number of potential storage locations on the yard area depending on its content and final destination. Since the number of vehicles is also limited, assigning containers to yard locations and dispatching vehicles to the containers so as to minimize the ship turnaround time is a combined problem called vehicle scheduling location problem. They show that the problem is NP-hard and develop an assignment problem based (APB) heuristic. The APB heuristic composed of solving an assignment problem that assigns containers to locations regardless of vehicle dispatching, and applying a greedy algorithm that assigns first *k* jobs, *k* being the number of vehicles, to the vehicles and remaining jobs to the first available vehicles. The zero unloading time case is investigated firstly, where it is proven that the heuristic solution error for this case can not exceed 100% and that error value goes to 0 when the number of jobs goes to infinity. For general case, unloading time is assumed to be a positive integer; the absolute and asymptotic worst-case ratios are found as 3 and 2, respectively. Computational experiments for different number of jobs and different sizes of fleet of vehicles are conducted. For a given number of jobs, the relative error generally decreases as the number of container increases which is consistent with the asymptotic worst-case performance results. Indeed, for small number of container, the error is at most 26.7% on average and no more than 48.0%, which is consistent with the absolute worst-case analysis results.

### 2.4. Storage Space Allocation

In most of the today's container terminals, containers are stacked and stored at the yard area for a particular time period.  An export container arrives with an external vehicle and is stored in the container terminal yard until departure. An import container comes with an arriving ship and waits to be retrieved by the dedicated external vehicle. A transshipment container, which arrives and departs with different ships at different time periods, also stays at the yard between arrival and departure. There are two different stacking types in practice: on chassis (undercarriage) and on ground stacking. If the containers are stacked on the chassis, which is mostly used in North American container terminals, they can be reached directly via the chassis. Otherwise, containers are stacked on the ground on top of each other to a particular height, depending on the container content and height of the yard crane bridge. On ground stacking is most common storage policy since storage spaces are limited in most cases.

Container storage area comprises blocks; each block is made up of rows and lanes. Containers are stored next to each other at each row and lane. Since the width of a block is shorter than the length, the number of containers stored next to each other in a row is smaller than that in a lane. The height of a particular block varies between two to eight containers depending on the configuration. If the yard cranes are utilized for container stacking at the yard area, container transfers from the transshipment vehicle to the yard can be carried out via either transfer points located at front and back end of the block or a lane dedicated for vehicle traffic along a block.

Determining the material handling equipment for storage and retrieval of containers at yard area is a strategic level decision. Forklift trucks, reach stackers, yard cranes, and straddle cranes are the most common alternatives. Yard cranes can be rubber tired or rail mounted. ASCs are common for automated container terminals, such as the Port of Rotterdam.

Since most operations take place at the yard area, sustaining efficient storage operations to ensure the efficiency of remaining operations becomes inevitable. Storage policies highly depend on terminal configuration such as handling equipment, stacking height, container grouping etc. For instance, higher stacking requires reshuffles or rehandles to reach a specific container since it may not be accessible. On the other hand, the higher the stacking, the less ground space is needed for the same number of containers. Although rehandling could be done in advance to eliminate possible delays during the storage/retrieval operations, such operations are unproductive moves and should be reduced.

The most recent paper by Kim and Park (2003) discusses storage space allocation of outbound (export) containers that will arrive at a storage yard. Although containers can be grouped into three distinct categories, outbound, inbound, and transshipment, their study focuses on the allocation of outbound (export) containers, which arrive at the yard several days before the arrival of corresponding ships. Yard equipment is also classified into two groups; direct and indirect transfer, respectively. Direct transfer compromises the dedicated equipment, which can handle both transfer and stacking operations whereas indirect transfer system consists of a delivery truck and dedicated stacking equipment such as yard crane or straddle carriers. In the direct transfer system, objective is to minimize the total distance traveled by trucks. In the indirect transfer system, the travel distance of transferring equipment as well as that of yard trucks should be minimized. The main focus of their study is to suggest a method for pre-allocating storage spaces for arriving outbound containers so that maximum efficiency in the loading operation is achieved. Objective functions and constraints regarding both the direct and indirect transfer systems are described and formulated. A basic model is formulated as a mixed integer linear program, and then two heuristic algorithms are suggested based on the duration of stay of containers and the sub-gradient optimization technique, respectively. A numerical experiment has been conducted to compare two heuristic algorithms.

Kim and Kim (1999a) analyze the segregating space allocation models for import containers at terminal storage yards. Arrival times of ships, the number of containers unloaded from the ships, and a measure reflecting duration of stay are considered to use segregation policy so as to minimize the expected total number of re-handles. Due to the segregation policy, containers unloaded during different periods are not mixed with each other at the same bay. Another relevant assumption states that the re-handled containers are moved to another slot in the same yard bay. They use a formula, which is presented in Kim (1997) to represent the relationship between the stack height and number of re-handles. Constant, cyclic, and dynamic arrival rates for import containers are investigated separately and optimal solutions are derived for each case by the Lagrangian relaxation technique. Numerical experiments and solutions are also provided to show the results for different problem instances.

Determining storage locations for export containers is also investigated in the latter study of Kim et al. (2000). In order to locate an arriving export container, they propose a methodology considering the weight of a container. The objective of their dynamic programming model is to minimize the number of relocation movements that could occur during the loading operations of ship. Containers are divided into three groups based on their weights. Since the heavy containers are picked up initially to load at the bottom levels of the ship, a relocation movement occurs when a lighter container is stacked on top of a heavier one. Assumptions state that each container could be relocated at most once and the arriving trucks are served on a first in first out bases, which means that re-sequencing the trucks is not allowed. Due to the lengthy computational time of dynamic programming approach, a decision tree induction based classification is applied to determine the storage locations. Information gain is used to determine the branching procedure; pruning and simplification are conducted to get more accurate decision trees. The performance of the decision tree approach is evaluated by the number of wrong decisions compared with the results found by dynamic programming. Numerical experiments reveal that the number of wrong decisions ranges between 1% and 5.5%, depending on the pruning parameters.

Kim and Kim (1998) discuss a method of determining both the optimal amount of storage space and optimal number of transfer cranes (yard cranes) for handling import containers. Two important decisions related to the investment cost of an import container yard are to determine the required space and the number of transfer cranes. Greater space for import containers for a given number of transfer cranes results in lower stack height, fewer number of re-handles during retrieval operations, longer travel distance by transfer cranes to pickup containers, and higher investment cost for the construction of the yard. More transfer cranes for a given amount of space results in shorter response time for a pickup call and higher investment cost for facilities. In summary, there is an economic trade-off among storage density, accessibility, investment cost, and level of service to outside trucks. They analyze this trade-off by minimizing the sum of the relevant cost components associated with the number of transfer cranes and the amount of space. Also, an analytic model is developed to estimate the various cost components related to handling of the import containers. Further, an attempt is made to simultaneously determine the optimal amount of storage space and the optimal number of transfer cranes for import containers.

Kozan and Preston (2001) model the seaport system with the objective of determining the optimal storage strategy for various container handling schedules. They examine the method employed in the storage of containers awaiting export at a seaport terminal. A container location model is developed with an objective function designed to minimize the turnaround time of container ships. Since the MIP formulation is NP-hard, the genetic algorithm (GA) is employed as one of the best known heuristic algorithms. Changes in the seaport infrastructure are considered and compared to the benchmark. The results section presents an analysis of different resource levels and a comparison with the current practice at the Port of Brisbane. The seaport system considered in the Kozan's studies is mostly specific for Australian terminals in terms of some operational issues and differs from the worldwide practice by temporary storage spaces.

In the most recent study of Zhang et al. (2001), storage space allocation problem in the storage yards of container terminals is considered using a rolling horizon approach. They considered the real practice in Hong Kong, where the inbound an outbound containers are mixed at the storage yard. Their decision problem is decomposed into two levels, and each level is formulated separately as a mathematical model. At the first level, the total number of containers to be placed in each storage block in each time period of planning horizon is set to balance two types of workloads among blocks. The nonlinear objective function for the workload balancing model is transformed to linear by several manipulations and the problem becomes an easily solvable network flow. The second level determines the number of containers associated with each ship that constitutes the total number of containers at each block in each period in order to minimize the total distance to transport the containers between their storage blocks and the ship berthing locations. With the numerical experiments run, they showed that the proposed method is efficient to get rid of work imbalance among storage block while avoiding possible bottlenecks in terminal operations.

Due to storage restrictions and several constrains, storage space allocation problems should be extended to satisfy practical requirements. Since a great number of allocation problems are based on network flow models, additional constraints to the models representing such problems become more complex. Cao and Uebe (1995) discuss the transportation problem with nonlinear side constraints, where the nonlinear side constraints avoid the assignment of a set of containers to a location at the same time. They propose a Tabu Search (TS) approach to improve efficiency of convenient branch and bound procedure. Applying TS to such a generalized problem results effective performance in terms of computational time compared to exact solution procedure. Their study suggests that similar problems consisting of a network flow structure with nonlinear side constraints could be examined with TS approaches since the results are promising for large and complex instances.

Veras and Diaz (1999) focus on the determination of optimal space allocation and optimal pricing for priority systems in container ports. They discuss how to allocate containers optimally and what is the storage pricing policy consistent with the optimal allocation. Hence, a joint problem is solved subject to the constraints regarding facility size. Demand has been taken into account through arrival rates, price elasticity and logistic opportunity costs, while supply has been introduced through marginal operating costs and land requirements. Results were obtained for welfare and profit maximizing rules, both for priority pricing and neutral price schemes. Conclusions for the analysis of optimal prices derived are illustrated for unrestricted welfare maximization, Ramsey pricing and profit maximization cases.

## 2.5. Transshipment of Containers

As mentioned earlier, containers have to be moved from the ship at the quayside to the yard area and vice versa. Determining the material handling system or system components is a strategic level decision that has to be made during the design of the container terminal. In practice, there is a great number of material handling equipments used in container terminals. They can be classified as automated vs. manned, direct vs. indirect, and so on. For container transfer from ship to yard, manned trucks, straddle carriers, forklift trucks could be used. Multiple trailer systems are used to transfer multiple containers together. As described in the container unloading/loading section previously, quay cranes are used to pick up and load containers from/to the ships to from/to the dedicated vehicles. Although some ships carry their cranes, the terminal equipments operate today's larger and frequently used ships. In practice, almost all of the quay cranes are manned.

At automated container terminals, AGV's are utilized for internal transport. In such a system AGVs are integrated with ASC's(automatic stacking cranes), which can pick up the container from the AGV at the transfer point of the yard block and move it to the final destination at the yard area. ASC's are also used to transfer containers from yard to the vehicles of other modalities. As a combination of AGV and ASC, automated lifting cranes (ALV), introduced recently, are capable of both lifting and transferring containers without using a crane.

After the material handling system is selected, the problem of determination of the necessary number of transfer vehicles should be solved. Steenken (1992) develops an optimization system to determine the number of straddle carriers and their routes. The problem is solved as a linear assignment problem. Vis (2001) presents a model and an algorithm to determine the number of AGVs at an automated container terminal. The problem can be modeled as a network flow problem and strongly polynomial time algorithm is developed.

Studies regarding material handling equipment used in container terminal operations have been carried out numerous times in the literature. Determining which vehicle transfers which container and the routes of vehicles are operational problems has been widely investigated in the literature. Steenken et al. (1993) focus on the problem of routing straddle carriers at the container terminal so as to minimize the empty travel distances by combining unloading and loading jobs. Steenken (1992) obtains savings of 13% in empty travels compared with the previously exiting situation by solving the problem as a linear assignment problem. Steenken et al. (1993) solve the problem by formulating it as a network problem with minimum costs. Numerical experiments for a real terminal system show 20-35% savings can be obtained in reasonable computational times.

Kim and Bae (1999) discuss dispatching containers to AGVs so as to minimize the delay of the ship and the total travel time of the AGVs. MIP formulations and a heuristic method for such a problem is given with numerical experiments. Kim and Kim (1999b) investigate the single transfer crane routing problem. They focus on the minimization of total handling time of transfer crane at the container storage yard by determining the optimal number of containers to be picked up at each yard bay as well as the optimal route of the transfer crane. Their modeling approach and optimal algorithm is applied without major changes to the straddle carrier routing problems for single and multiple carrier cases as illustrated below.

Kim and Kim (1999c) discuss the optimal routing of single straddle carrier, which is the frequently used transshipment equipment in port container terminals. They propose a MIP model with the objective of minimizing the total travel time of the straddle carrier and investigate the properties of optimal solutions to devise a solution procedure. The solution procedure is decomposed into two stages. In the first stage, the number of containers to be picked up during a sub tour is determined. In the second stage, the visiting sequence of yard bays by the straddle carrier is found. Their solution procedure could be summarized as follows. First, with respect to a set of transportation model constraints involved in MIP model all basic feasible solutions are generated.

Then the set of basic feasible solutions subject to the whole constraints is constructed by enumerating the solutions found in the first step. Next, the routing problem is solved via dynamic programming according to the solution set found in the second step and the least cost route is selected as optimal.

As further research, Kim and Kim (1999d) extend the previous study and discuss multiple straddle carriers (SC) routing problem during the loading operation of export containers in port container terminals. Since unloading time is proportional to the number of containers, the loading process is considered so as to minimize the total travel distance of straddle carriers in the yard. They mention that the loading time depends on loading sequence of containers so using efficient algorithms could reduce loading time significantly. The loading sequence is the order of containers that a quay crane loads onto a corresponding ship, assuming that export containers are handled by a combination of SCs and yard trucks. One SC and 3-4 yard trucks are assigned to a quay crane. The problem is comprised of the container allocation problem and carrier routing problem. The container allocation problem is illustrated as the transportation problem in the first step. An MIP is formulated and tested in LINDO. Since finding optimal values is extremely slow, a beam search algorithm, a specific type of Branch and Bound, is proposed with specific parameters such as width etc. A numerical experimentation is carried out in order to evaluate the performance of the algorithm. Computational results show that beam search is 114% greater than optimal values on average and it depends on some parameters of the problem and algorithm.

Since the workload distributions in the yard change over time, the deployment of yard cranes (rubber tired gantry cranes) among storage blocks is an important issue for terminal management. Liu et al. (2002) investigate this problem with the given forecasted workload of each block over time, where the objective is finding the crane routes among blocks and the time of deployment so that the total delayed workload is minimized. After having formulated the problem as a MIP model, they apply Lagrangian relaxation. In order to improve the performance of solution procedure and the quality of the solutions, they augment additional constraints to the original problem and modified the steps of Lagrangian approach accordingly. The efficiency of solutions generated by modified Lagrangian relaxation approach has been approved by computational experiments.

Cheung et al. (2002) consider the problem of scheduling the movements of cranes in container storage yard and propose a MIP problem to formulate the Interblock Crane Deployment problem in a rolling horizon manner so as to minimize the total unfinished workload at the end of each time period. Travel time is measured in number of time periods, where the cranes can leave a block only at the beginning of a period. The workload can arrive at the beginning of a time period and the amount of the workload is also measured in time periods. The amount of work done in a block per time period is proportional to the number of cranes in the block during that period. The maximum number of cranes working simultaneously in one block is limited with 2 to get rid of possible collision within the block. After having analyzed the complexity of the mixed integer linear problem, it is shown to be NP-hard. Lagrangian Decomposition and Successive Piecewise Linear Approximation methods are applied to solve the integer program. The Lagrangian decomposition method is used to decompose the problem into a network flow problem and a linear subproblem. Additionally, the Successive Piecewise Linear Approximation method is introduced to approximate the problem by linear network flow problem iteratively. Computational experiments are conducted to show the efficiency and effectiveness of proposed methods for large sized problems.

Kozan and Preston (1999) propose genetic algorithms to schedule the container transfers at multimodal terminals. Since optimizing container transfers the multimodal container terminal is known to be NP-Hard, genetic algorithm is used to reduce container handling and transfer times as well as the ship times at the port by speeding up transfers. The layout used as an input consists of two distinct storage spaces: berthing and marshalling area for temporal storage and the yard storage areas for remaining storage period. The investigated multimodal terminal is connected with a rail intermodal terminal. The main objectives of the proposed model are to determine optimal storage strategies and container handling schedules. Chromosome representation, algorithm, and crossover operations are described as the steps of solution technique. Computational experiments carried out for the Port of Brisbone are provided extensively for different layout alternatives, the number of yard machines and storage policies. Simulations reveal that the GA is relatively good since the near-optimal results are found even with the simplest GA implementations in a reasonable time. Results show that using the nearest rows to store containers is better than random allocation of the container. Also, both the height of stack and the number of yard machines have a dramatic influence on the handling times.

Since solving all the operational level problems at a container terminal in a single integrated model is well beyond today's computing capability, Bish (2003) decomposes the problems into two levels. In the first step, dispatching vehicles to containers, assigning unloaded containers to storage locations, and determining the schedule of loading and unloading operations on the quay cranes should be done. Locating yard cranes in the storage area and determining the sequence of locations served by each location are the decisions that can be made in real time control after solving the problems in the first step. Bish (2003) deals with the problem described in the first step so as to determine a storage location for each unloaded container, dispatch vehicles to containers, and schedule the unloading and loading operations on the quay cranes. There exist an unloading and a loading ship in the terminal area both of them are served simultaneously by an equal number of quay cranes and a pool of vehicles. The problem is called multiple-crane constraint vehicle scheduling and location (MVSL). After a review of literature regarding machine scheduling, material handling systems and resource constraint scheduling, it is approved that the MVSL problem is NP-hard since the more simple problems which are similar to the MVSL problem are NP-hard. Thus, a heuristic algorithm is presented, called the Transhipment Problem Based List Scheduling Heuristic. First, a transhipment problem is solved to assign each container to a storage location and match this location with a location of container waiting to be uploaded. Such an assignment and matching problem is easy to solve without integrality constraints due to the totally unimodular property of transhipment problem. By solving this model, loaded trips associated with container discharging and uploaded are combined. These combined trips are composed of a loaded trip from discharging crane to the storage location, an empty trip between storage locations, a loaded trip from the storage location to the uploading crane. Thus, minimizing allocation and matching related trips result in a minimum makespan. In the next step, combined trips found via assignment and matching model are scheduled on the vehicles with the list heuristic. The combined trips are ordered in non-decreasing order of processing times and the next combined trip in the list is scheduled on the next vehicle served by a crane at the discharging ship. The properties of the heuristic solution are also investigated. Asymptotic worst-case performance is given with a lower bound on the optimal makepan. A fixed bound on the heuristic's deviation from optimality for all large instances of the MSVL problem is derived. If the crane processing times and minimum travel time of a loaded trip are at least as great as the travel time between cranes at the berth, then the heuristic makespan is at most 133% away from the optimal makespan for a large number of containers. The solution procedure is extended for more specific cases where the number of discharging and uploading container are different. Computational analysis provides the average percent deviation of the heuristic makespan from the lower bound over all the replications. The computational results are consistent with the asymptotic worst-case analysis. Moreover, stochasticity of travel times is included in the computational study as an extension.

Narasimhan and Palekar (2002) define the problem of minimizing the time taken to load and unload the containers from the container stack yard onto the ship as transtrainer routing problem, where transtrainer is the dedicated equipment to load and unload containers from/to trucks to/from container stacking yard blocks respectively. They investigate the theoretical aspects of the problem and prove that the problem is NP-Complete. The problem is formulated as an integer program with the given load and bay plans. The overall objective is minimizing the total setup and inter-bay traveling times. A branch and bound based enumerative method is developed to obtain an exact solution to the problem. The properties of optimum solutions and related proofs of lemmas are given. The problem is decomposed into enumerating the degenerate solutions and then arranging the partial sequences in the degenerate solutions to obtain final route for transtrainer. Several lower bounds to prune the size of tree are also developed. They design a specific enumerative heuristic with a worst-case performance ratio of 1.5 since the absence of a polynomial time heuristic with a bounded worst case unless P=NP is proved. In addition, computational studies with randomly generated problems are conducted to evaluate the exact and heuristic algorithms.

Kozan (2000) evaluates the major factors influencing the transfer efficiency of seaport container terminals. A network model is proposed and solved so as to minimize the total throughput time, which is handling time for all containers from the ships at berth and transferring time of containers to the yard area. The overall objective of the proposed model is to minimize the handling time of the containers from the first arrival at the port until the ship carrying containers departs from the port. It is mentioned that the model presented could be seen as a decision support system in the context of investment appraisal of intermodal container terminals and an application is completed for the expansion strategies of the Fishburn Port in Australia.

Meersmans and Wagelmans (2001a) consider the problem of integrated scheduling of various types of container handling equipment at an automated container terminal, where the objective is to minimize the makespan of the schedule as usual. They investigate the case of the Port of Rotterdam, where several terminals use automated equipment such as automated guided vehicles (AGV's) and automated stacking cranes (ASC's). The typical layout of the automated container terminal is given, stacking lanes are positioned vertical to the berth, an AGV area exists between stacking lanes and ship where the AGVs are routed, as well as transfer points of containers from AGVs to ASCs and ASCs to other modalities at the front and at the end of stacking lanes, respectively. Former studies are reviewed as an introduction, modeling and complexity of the problem is proposed with a proof of NP-hardness of the integrated scheduling problem. A branch and bound solution algorithm is developed; branching rule, search strategy, and combinatorial lower bounds regarding the algorithm are also given. A beam search variant algorithm is proposed to speed up the computational time by filtering the solutions found at each level. Although beam search could not find optimal results, computational results show that it would be more efficient and practically applicable for large size problems without extensive fine-tuning.

In a later study, Meersmans and Wagelmans (2001b) consider the same problem in a dynamic environment, where the handling times are not known beforehand and that the order in which the different pieces of equipment handle containers need not be specified completely in advance. They mention, instead of static schedules, partial schedules must be updated when new information on realizations of handling times becomes available. The dynamic version of beam search algorithm takes a small number of containers into account within a rolling horizon. The performance of static and dynamic version is compared and the longer planning horizons used the better average performance for dynamic beam search algorithm found. This result holds for both deterministic and stochastic scenarios. In the second part, various well-known dispatching rules are considered. Although straightforward implementation of these rules generated deadlocks, modified versions yield feasible solutions. Moreover, they compare the performances of these rules with the performance of beam search algorithm. On average, beam search performs best, but some dispatching rules such as FCFS (first come first served) and MWR (most work remaining) come very close.

Van der Meer (2000) analyzes the control of guided vehicles in the vehicle based internal transport systems such as container terminals. Results show how different vehicle dispatching policies behave in different environments. Evers and Koppers (1996) develop a formal tool to describe traffic infrastructure and its control, and then the tool is evaluated with simulation. They conclude that the developed tool is powerful for modeling transportation infrastructure and its control.

### 2.6. Stacking/Re-handling Operations

Kim (1997) examines the effects of re-handling work to the performance of transfer cranes in a container terminal. Since the re-handling time occupies a large portion of container handling operations, accurate estimation of re-handles is an important factor to evaluate terminal throughput rate, a significant metric to determine the material handling specifications, number of equipments and to evaluate the alternative layouts. It is assumed that outbound trucks containers are picked up randomly from the initial stack and loaded on outbound trucks without additional containers being added until all containers are removed. The expected number of re-handles for a random target container depends on the total number of containers, number of rows and the distribution of the height of the stacks. An exact evaluation, regression analysis and approximation formula are proposed for the expected number of re-handles for the next pickup and expected number of total re-handles with several tables and equations. The approximation formula for the total number of re-handles is compared with a conventional method, IOS (index of selectivity). Computational results show that approximation formula outperforms the IOS method in both accuracy and lack of bias.

In order to speed up the loading operation of export containers onto a ship, replacing containers in proper positions, called re-marshaling operation, is a usual practice in port container terminals. It is assumed that current layout of containers is given and the desirable layout for efficient movements is provided. Kim and Bae (1998) analyze the conversion of current layout to the desirable by moving the least number of containers to the shortest possible travel distance. The re-marshaling problem is decomposed into three hierarchical sub problems: bay matching, move planning, and task sequencing. First, current to desirable layout conversion is formulated as the bay-matching problem by dynamic programming. In the next step, the number of containers to be moved from one bay to another is determined by the solution of a well-known transportation problem. Using the iterative procedure for the bay matching and move planning problems, the final solution is found and the corresponding tasks are sequenced by dynamic programming so as to minimize total completion time.

## 2.7. Overall Container Terminals

Gambardella et al. (1998) present a decision support system for the management of an intermodal container terminal. They propose two modules to focus on resource allocation problem at the La Spezia Container Terminal (LSCT). First, the optimization module is employed that uses integer programming approach by formulating the problem as a linear network flow. Although getting optimality in a complex MIP is a time consuming task, good solutions validated with experimental results can be found quickly with the LP SOLVE software. Next, a simulation tool covering the operational details of system resources is developed to support the terminal manager's decision for various scenarios. The simulator provides a test bed for checking the validity and robustness of the policy computed by optimization tool.

Gambardella et al. (2001) propose a hierarchical formulation and solution to the problems of resource allocation and scheduling of loading and unloading operations in a container terminal. The objective of resource allocation problem is minimizing the costs by properly allocating resources while respecting the ship's deadlines over a planning horizon without knowing in advance the exact storage positions of in yard. The terminal is modeled as a network, where nodes represent the resources as yard areas, cranes, ships and arcs represent the decision variables whose capacities and costs depend upon resources. The model is extended over shifts to cover the rolling horizon. An MIP is formulated and solved with branch and bound in a short time. The scheduling problem is a lower level problem at a different level of detail. The unloading list of containers is known a few hours before the arrival of the ship. Since resources are already allocated and container positions are known from the primary model, the objective of this stage becomes moving containers between ships and yard so as to minimize the total makespan while avoiding deadlocks among resources. Hence, the problem is modeled as an extended flexible job shop scheduling problem. Deep investigation of extended neighborhood function and self-tuning Tabu Search are presented. Using neighborhood function and tabu search in succession, it is possible to compute schedules quickly, which makes the procedure implementable for real terminals. As the second step, discrete event simulation is conducted via the validated and calibrated simulator proposed in Gambardella et al. (1998). Computational results show that optimizing resource allocation, which reduces costs by 1/3, can be adopted with optimized loading and unloading lists efficiently. Also, simulation results show that optimized lists reduce the number of crane conflicts and the average length of truck queues in the terminal.

The main contributions operations research has made in the area of container handling activities are illustrated as an overview study of Meersmans and Dekker (2001). They propose a literature survey of the use of operations research models and methods in the design and operations of container terminals. Decisions at the strategic, tactical as well as operational level are given after the brief descriptions of activities taking place at container terminals. They mention the specific nature of container related operations and container terminals. Decision problems are classified as container stowage, berth and crane allocation, container loading; quay transport in terms of scheduling of cranes, vehicles, carriers and traffic control, stacking as well as the design and analysis of the overall container terminal. They conclude that the studies regarding container terminal operations vary from integer programming formulations, queuing models, and simulations approaches.

Vis and Koster (2003) present an overview paper on transshipment of containers at a container terminal recently. They mention the dramatic improvement in container transportation and the requirement of efficient port terminal operations. After stating the processes at container terminals step by step, they classify the decision problems from the arrival of the ship to the transfer of containers to other modes of transportation.

First, the arrival process is described and the corresponding relevant literature summarized with the studies covering queuing models for the arrival process of ships and berth allocation. Container stowage, crane scheduling, crane allocation problems are illustrated in the unloading and loading operations section. AGV dispatching and control, vehicle scheduling and routing literature is given in the transshipment of containers section. Storage space allocation, container stacking and re-handling problems are considered as the stacking decisions. Eventually, the studies regarding inter-terminal transport and other modes of transportation are investigated. Complete container terminal cases conclude the overview. They mention that numerous researches have been done to solve decision problems in container terminals; however various problems are still open to investigation.

Shabayek and Yeung (2002) analyze the Kwai Chung container terminal, one of the busiest and leading terminals of Asia in terms of the total container throughput. An application of simulation model using Witness software is developed and described. Witness is superior to other simulation software since it can simulate the situation in which servers share berths with each other is flexible and has animation capabilities. The objective is to investigate to what extent a simulation model could predict the actual container terminal operations with a higher order of accuracy. The proposed model can be used for cost analysis, the planning of future additional berths and to estimate the performance improvement in case of handling equipment variations. Simulation runs show that the model provides good results in predicting the actual operating system of the container terminal.

# 3. CONTAINER TERMINAL FRAMEWORK

According to the recent literature on container logistics, various container terminal configurations can be found all over the world. Container terminal configuration depends on the system requirements, resource capacities, and specific characteristics of the port. The number of berths, total storage area in the yard, layout of the yard area, material handling system and processing pattern of containers are some critical inputs that identify the properties of a container terminal. Most of the decisions to determine the system specifications are concluded at the strategic level, before dealing with operational problems.

In order to configure a common framework for container terminals, we have scanned a number of terminal layouts and operational systems investigated and modeled in the literature. Since most of the research in this area are conducted for the consultancy of a particular container terminal, shipping line, or any industry partner, the frameworks indicate actual terminal cases for the projects, and thus there is no common framework representing container terminals worldwide. Nevertheless, process flow and major operational issues are almost similar in a variety of studies regarding container terminals. Due to the absence of an actual case and associated data in our study, models and methodologies illustrate a general container terminal prototype. Thus, we need to simplify the operational complexity of terminal framework and settle on an abstract, flexible and expandable schema with a set of relevant assumptions.

To express the insights of the system more properly, an extensive terminology regarding container terminal operations is given in the next section of this chapter. Terminology is followed by the description of terminal layouts and distance metrics associated with the layouts taken into account in numerical experiments. The last section explains how the parameter settings regarding the container flow are performed prior to computations.

## 3.1. Terminology

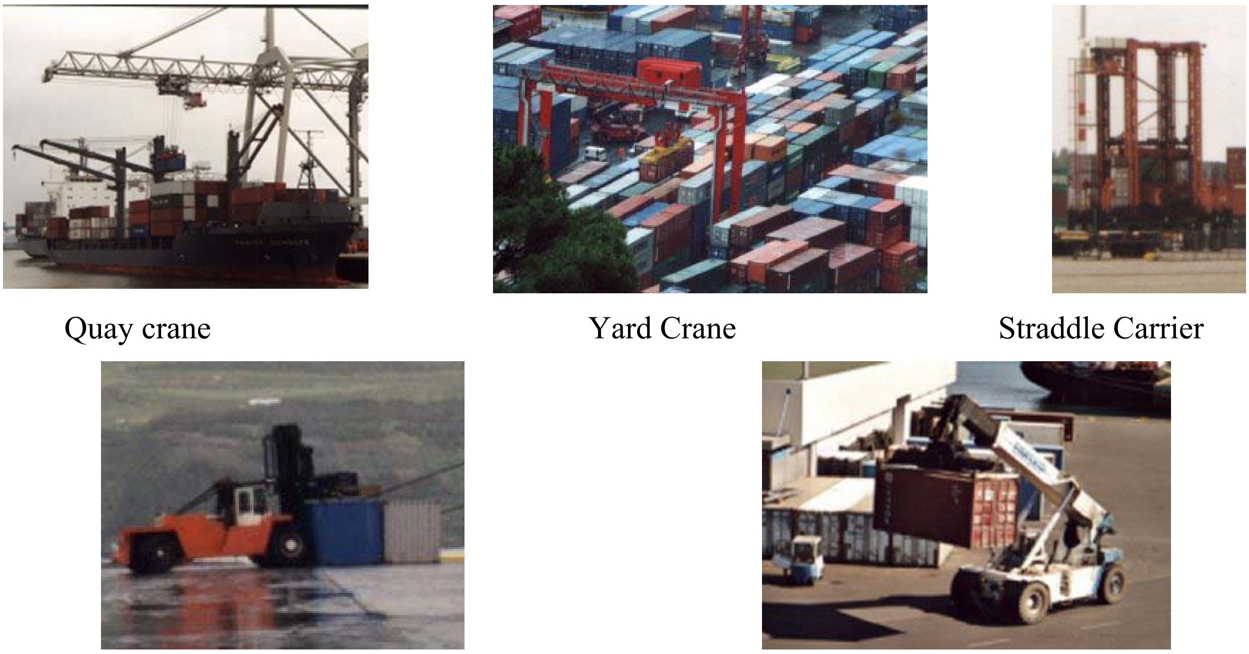
The container is a standardized metal frame package for bulk cargo. Containers can be transshipped via different modes of transportation such as rail, truck, and marine since the dimensions and specifications are appropriate. There exist various container types when the dimensions are considered as the criteria for classification. In practice, two types of containers are used most frequently all over the world: 1-TEU and 2-TEU containers, based on the length of the container. TEU, the acronym for “twenty feet equivalent unit", is defined to identify container types and related measures used in logistics systems. The containers are made up of iron and steel elements with the fixture on the corners. A door placed on the backend allows loading and unloading cargo.

As mentioned earlier, a container terminal is composed of two distinct areas called quayside and yard area, respectively. Quayside is the area where ships stay at berths between arrival and departure. Berths are the places at the quayside where the ships moor to be unloaded and loaded. Yard area is the stacking place of containers, where containers are stacked for a particular time period between storage and retrieval. The yard area is the main decoupling point between the import and export flows, either from sea to sea or from sea to land and vice versa. The yard area of a container terminal is divided into a number of blocks. Actually, two different stacking policies exist in today’s container terminals worldwide. In North American terminals, containers are stacked on the chassis, which allows them to be accessed directly. Due to the space restrictions, containers are stacked on top of each other in European and Asian terminals. In such configurations, containers are not directly reachable since some re-handling operations are required to access a particular container at a lower level in the stack. The number of containers to be stacked on top of each other varies from 3 to 6 depending on the height and accessibility of retrieval equipment as well as container characteristics. In practice, similar containers are placed on top of each other to ensure safety and to accomplish rapid transfers.

In almost all of the container terminals, container unloading and loading operations from/to ships are performed by manned quay cranes (QCs). QCs have trolleys that can move along the crane arm to transport the container from the ship to the transport vehicle and vice versa. A spreader, a pickup device attached to the trolley, picks the containers. Each ship, depending on its size, is divided into a number of holds to operate simultaneously via parallel QCs. The QCs can move on rails to the different holds to take/put containers off/on the deck and holds.

After unloading, containers are placed on a vehicle, which moves the container to the yard area. The types of this transfer vehicle vary so much; they can be a truck with a trailer or multiple trailers, an automated guided vehicle, a straddle carrier, a forklift truck, a reach stacker, or any other specific transfer equipment in practice. As mentioned earlier, a transporter-stacker combination is called as Indirect Transfer system whereas the system composed of single vehicle capable for both operations is called a Direct Transfer. For instance, the truck-yard crane (stacker crane or gantry crane) pair is the most frequently used Indirect Transfer system, whereas straddle carriers are the most common vehicles of Direct Transfer.

After the transfer of container from berth to yard, containers are placed on a predetermined position. Yard cranes or stacking cranes pick up containers from trucks and stack them on the storage location. They can provide high-density storage and can be automated in some instances. Yard cranes can be rail mounted, rubber tired, or put on a concrete or steel structure such as overhead bridge cranes. Rail mounted yard cranes, called rail mounted gantry cranes in some cases, are stable and fast, but inflexible. Rubber tired gantry cranes are more flexible in terms of deployment between blocks. Instead of a truck-yard crane combination, a bi-purpose equipment called a straddle carrier can be used. The straddle carrier combined the functionalities of a transport vehicle and stacking crane. It is able to drive a container over, lift it up to 3 or 4 containers high, and move it around. Additionally, forklift trucks are able to transport and stack containers. The reach stackers are rubber-tired equipment able to lift containers up more than forklift trucks.  The images below illustrate the container handling equipments.



Quay crane | Yard Crane | Straddle Carrier

Forklift Truck | Reach Stacker

Figure 3.1 Container Handling Equipment

### 3.2. Layout and Distances

Since the analytical investigation carried out in this study covers a generic container terminal, structural characteristics of terminal such as layout, distances, handling equipment etc. should be determined prior to the problem definition.

As mentioned earlier in this chapter, container terminals are constructed of two main areas: the berth area (quayside) and the yard area. The unloading and loading of ships are performed in the berth area by QCs, and then containers are transported to their final destinations in the yard area by vehicles. The terminal yard is divided into a set of blocks: each block consists of several lanes and rows; and in each row, four to seven containers can be stacked vertically. At any location, four to six containers can be stacked on top of each other. A brief top-view of the terminal yard and a particular block layout were depicted in Figure 3.2 and Figure 3.3, respectively.

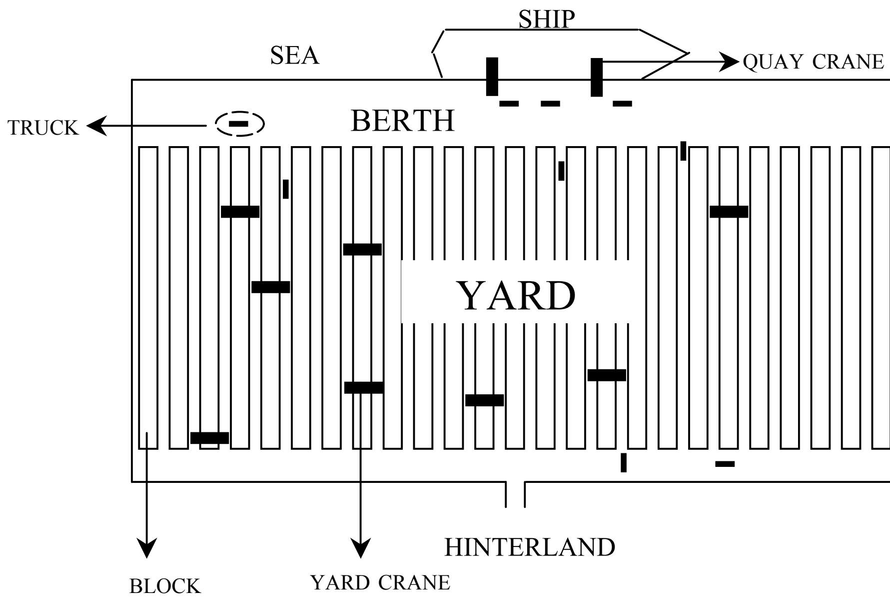



Figure 3.2 Top-view of the terminal yard

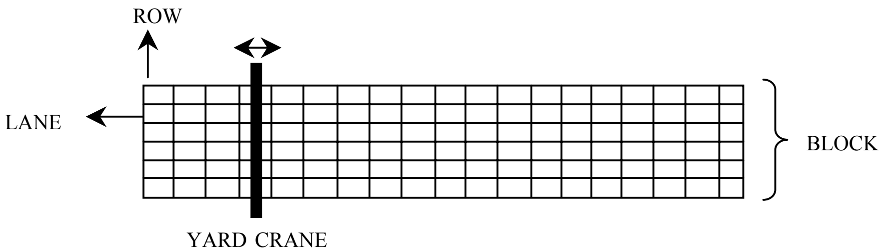



Figure 3.3 Block Layout

The most crucial criteria to be considered when deciding on the container terminal prototype investigated in this study are the comprehensiveness and extendibility. In almost all of the studies in the literature, the yard areas of container terminals are composed of a number of blocks, so we divided the yard area into a number of blocks. Since several berths are available for ships in practice, our prototype has multiple berths. We assume that the quay crane-truck-yard crane triple is used as the handling system, as in most of the container terminals. A sketch of the container terminal layout is depicted as follows. Each bar represents a block and the points are the origins of berths.

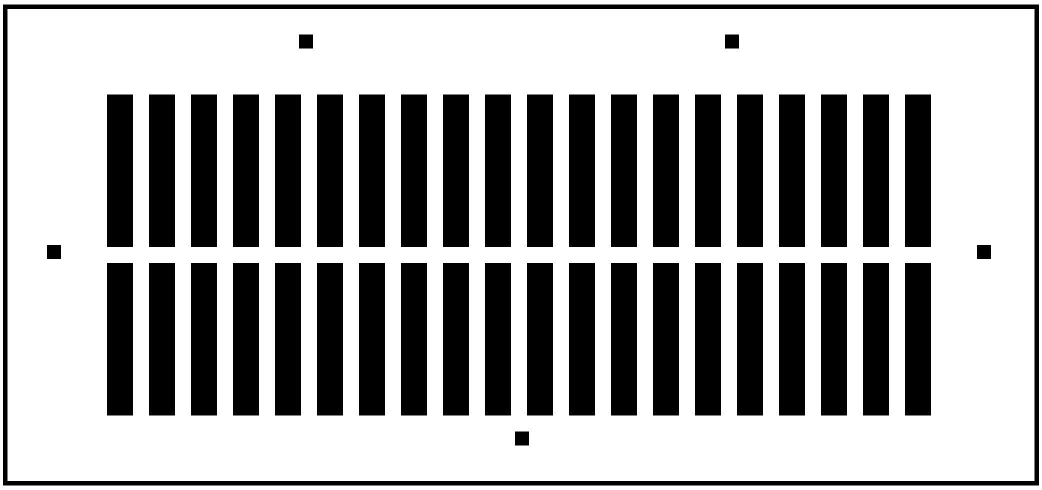


Figure 3.4 Container Terminal Layout (Block view)

We have four berthing locations: two of them are in the front side -- one for the right and one for the left sides, respectively. The point at the bottom side represents the entrance/exit gate of the container terminal from/to the hinterland. Although each hold of a containership can be served by different quay cranes, which means that a number of quay cranes may serve a ship at one time, we assume that the depicted berthing locations are the "origins" or the "center of gravity" of the berth positions of ships. Thus, we have five different distribution points to fulfill container inflow and outflow. The figure below demonstrates the associated distances within the terminal area.

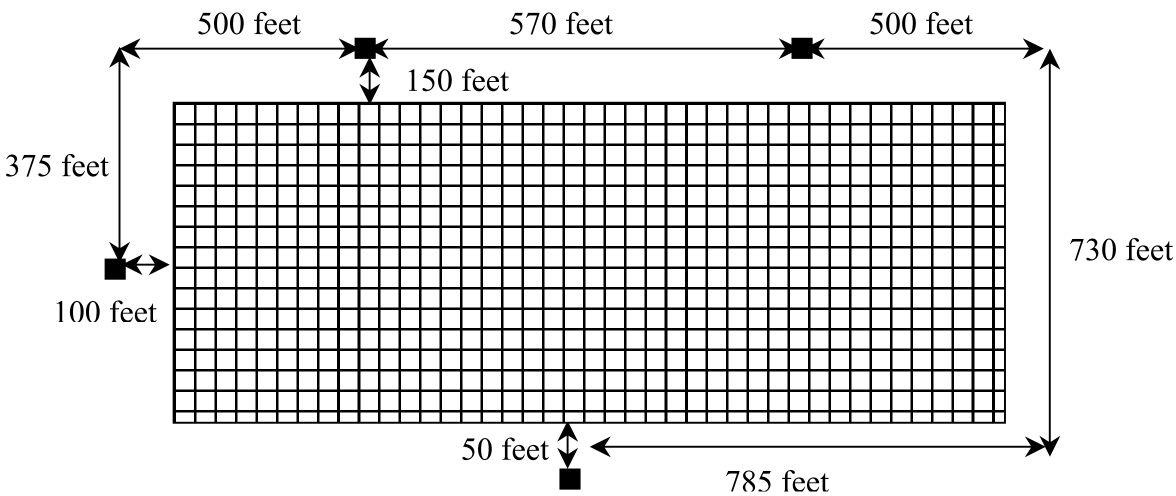



Figure 3.5 Distances of the terminal layout

The parameters regarding container terminal area are summarized below. There exist 800 locations in the storage yard if we assume that each location has a capacity of 10 containers. Since the number of locations increases as the location capacity decreases, the layout parameter settings allow us to extend such parameters throughout numerical experiments. The blocks can be aligned horizontally or vertically. According to the different orientations of blocks, we consider four different types of container layout alternatives. The different layout alternatives do not influence computational results regarding the space allocation case because the Manhattan (rectilinear) Distance metric is used when identifying the distance values, as can be seen in Section 4.4.1. Since trucks should turn around corners to access locations belonging different blocks, travel distances within blocks may vary due to orientation.

Table 3.1 Layout Parameters

| | |
|---|---|
| Number of blocks: 40 | Number of locations in a block: 20 |
| Total number of locations: 800 | Stack height for each location: 5 |
| Lanes at each location: 2 | Total container capacity: 8000 |
| Number of berths: 4 | Number of entrance/exit gates: 1 |
| Length of the yard area= 1370 feet | Width of the yard= 530 feet |
| Length of a block= 250 feet | Width of a block= 40 feet |
| Distance within blocks= 30 feet | Distance between berths and yard = 100 feet |
| Distance between gate and yard = 50 feet | |
| Length of whole terminal area= 1570 | Width of whole terminal area= 730 |

The northwest corner of the sketch is assumed to be the origin *O (0,0)*. According to the parameters listed above, coordinates of berths and gate will be as follows.

Table 3.2 Coordinates of berths and gate

| Location | $x$ | $y$ |
|---|---|---|
| Berth 1 | 0 | 365 |
| Berth 2 | 500 | 0 |
| Berth 3 | 1070 | 0 |
| Berth 4 | 1570 | 365 |
| Gate | 785 | 730 |

In order to determine the parameters associated with each location at the yard area, locations are numbered, starting from the first block to the last one. As mentioned earlier, we assume that each location reserves two containers stacked next to each other if there exist 800 locations in total. Thus, the capacity of each location will be 10 when the stack height is 5 per location. In what follows, the numbering of locations for such a configuration is given where $O_1$ is the origin for Block 1.

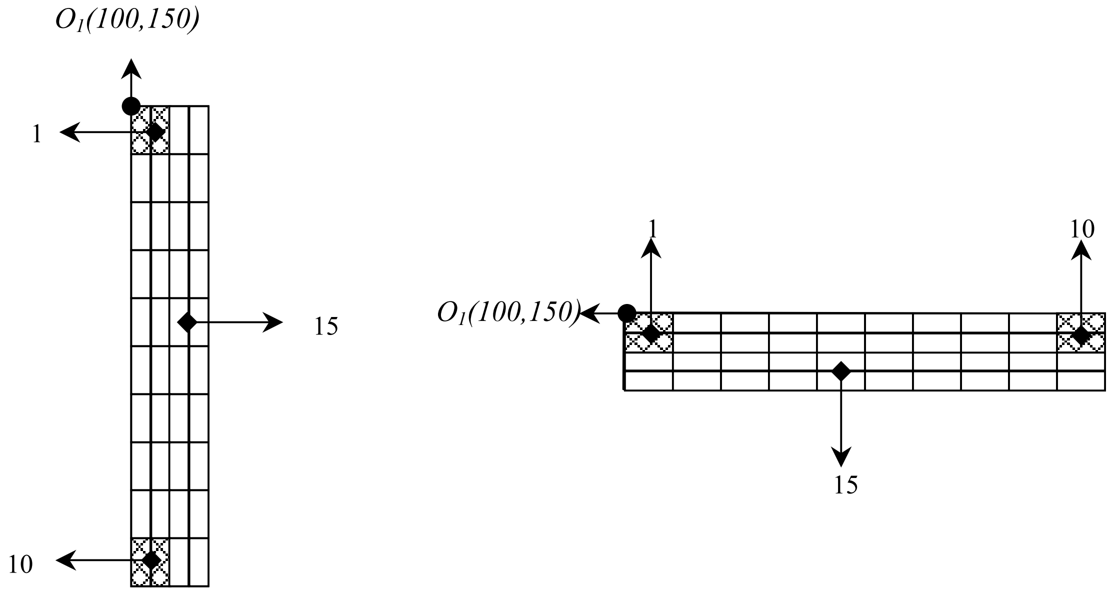



Figure 3.6 Numbering of locations for Block 1 for vertical and horizontal alignments

We can now calculate the coordinates of each location according to the parameters and numbering defined above. Since there exist four different types of layouts, there are different formulas associated with the coordinates of locations. After numbering, the location coordinate formulas based on layout types are stated as follows.

$b_i$ : block of location $i$ $\quad$ $b$ : total number of blocks

$s_i$ : section of location $i$ $\quad$ $a_i$ : group of location $i$

$l$ : number of lanes in a block $\quad$ $r$ : number of rows in a block

$$b_i = \left\lceil \frac{i}{rl} \right\rceil \qquad s_i = \left\lceil \frac{10i}{rlb} \right\rceil \qquad a_i = \left\lceil \frac{2i}{rlb} \right\rceil$$

$W$ : width of a block $\quad$ $L$ : lenght of a block $\quad$ $I$ : distance between blocks

$x_i$ : $x$ coordinate of location $i$ $\quad$ $x_o$ : $x$ coordinate of origin

$y_i$ : $y$ coordinate of location $i$ $\quad$ $y_o$ : $y$ coordinate of origin

Table 3.3 Coordinate calculation formulas

| | |
|---|---|
| Layout 1 | $x_i = x_o + \left[(b_i - 1) \bmod \frac{b}{2}\right](W+I) + \frac{W}{2l} + \left\{\left\lfloor \frac{i-1}{r} \right\rfloor \bmod l\right\}\frac{W}{l}$ <br> $y_i = y_o + \left\lfloor \frac{b_i - 1}{b/2} \right\rfloor (L+I) + \frac{L}{2r} + \left[(i-1) \bmod r\right]\frac{L}{r}$ |
| Layout 2 | $x_i = x_o + \left[(b_i - 1) \bmod \frac{b}{8}\right](L+I) + \frac{L}{2r} + \left[(i-1) \bmod r\right]\frac{L}{r}$ <br> $y_i = y_o + \left\lfloor \frac{b_i - 1}{b/8} \right\rfloor (W+I) + \frac{W}{2l} + \left\{\left\lfloor \frac{i-1}{r} \right\rfloor \bmod l\right\}\frac{W}{l}$ |
| Layout 3 | for $s_i = 1,2,3,6,7$ apply LAYOUT 2 <br> for $s_i = 4,5,8,9,10$ apply LAYOUT 1 |
| Layout 4 | for $s_i = 1,3,5,7,9$ apply LAYOUT 2 <br> for $s_i = 2,4,6,8,10$ apply LAYOUT 1 |

In order to enrich the numerical experiments as well as to approve the robustness of results for different instances four different layout alternatives depicted below are tested.

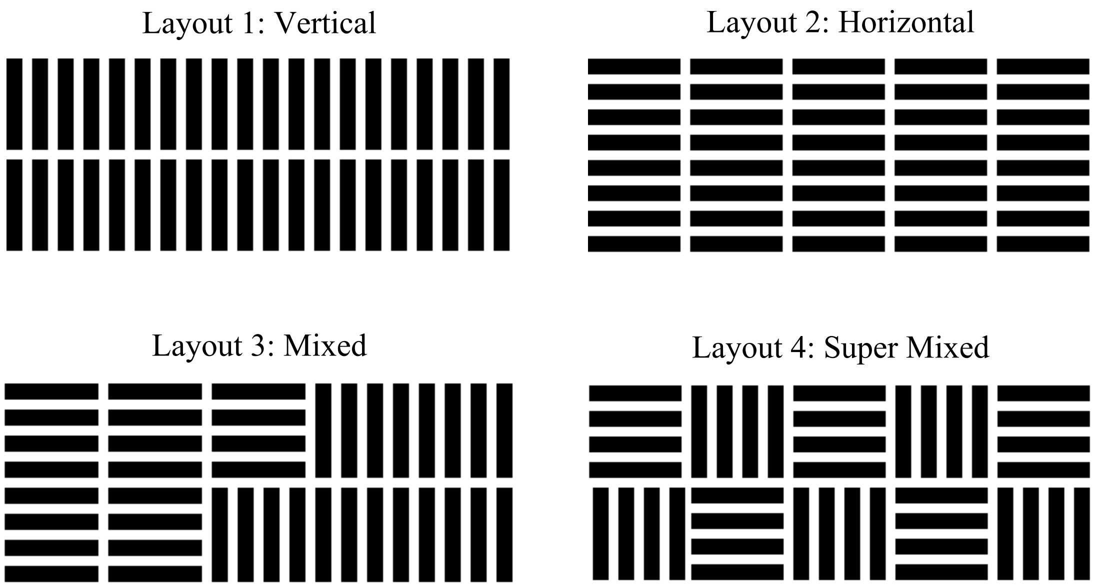



Figure 3.7 Layout Alternatives

### 3.3. Parameter Settings for the Numerical Experiments

A number of parameters are required to construct the mathematical models considered in this study. There exist different inputs for different types of models. For instance, berth allocations are the input for crane allocation problem. The unload and load plans of ships are the main input for space allocation, vehicle scheduling, and routing as well as the ship stowage problems. The content of each arriving ship to the container terminal is known a sufficient time period before the arrival. The position of an unloading container on the ship, the owner of the container, the possible storage areas regarding the owner or destination are already known prior to the discharging operations. The exact storage location, the vehicle to be dispatched, and the route for the dedicated vehicle should be determined for an arriving container. For any departing container with ships or uploading container, the exact storage location at the yard area is known. Thus, the stowage position and uploading sequence is determined so as to reduce ship turnaround times at the current and remaining ports on the route of the ship.

#### 3.3.1. Arrival-Departure Parameters

The major inputs considered in our storage space allocation model are the arrival-departure parameters. Since it is an aggregate planning model, the more detailed information regarding containers is assumed to be unknown for such a decision. With the arrival-departure parameters, arrival and departure periods of containers are known. In other words, we have arrival-departure period pairs and corresponding number of containers for each pair. Thus, the exact information regarding each container is the duration of stay.

Simple modeling case assumes that there exists just one type of container handled in the terminal. In the extended modeling scenario, we consider three types of containers: transshipment, export and import containers. Let us assume that berth allocation has been already done and the berth plan for the planning horizon has been given. Arriving and departing ships of transshipment type containers are known as well as the corresponding arrival and departure periods from the predetermined berth plan. A departing ship and an arrival period is given for an export container. Also, an arriving ship and departure period is known for the import containers. Therefore the corresponding parameters for each type of containers will be as follows:

$TRA_{ad}$: Number of transshipment containers arriving with ship $a$ and departing with ship $d$

$EXP_{td}$: Number of export containers arriving at period $t$ and departing with ship $d$

$IMP_{ak}$: Number of import containers arriving with ship $a$ and departing at period $k$

On the other hand, the parameters associated with the simple modeling case will be as follows:

$A_{tk}$: Number of containers arriving at period $t$ and departing at period $k$

Generation procedures for each of the first three sets of parameters are explained below and the $A_{tk}$ values associated with the simple case are also generated arbitrarily with respect to the similar assumptions.

**Assumption 1**: All set of parameters are IID uniform random variables, which are generated arbitrarily before the numerical experiments and set to be constant throughout the computational runs.

### i. Generating $TRA_{ad}$

Arrival period of ship $a$: $A_a$

Departure period of ship $d$: $D_d$

If $D_d \leq A_a$ then $TRA_{ad}=0$

Otherwise $TRA_{ad}=U(x,y)$ (uniformly distributed)

Hence, parameter matrix is upper triangular.

We assume that storage times of containers in the terminal are at most a couple of days and the number of containers retrieved shortly are greater than the number of container waiting more than a few days.

**Assumption 2**: The number of containers belonging to an arrival-departure pair increases as the number of periods between the arrival and departure decreases. The expected number of containers for such a pair is a monotone decreasing linear function of the number of periods between arrival and departure.

**Assumption 3**: The parameters representing the number of containers are uniform random variables such as *U(x,y)*, where *x* and *y* are the functions of lower and upper levels of the number of containers, time between arrival and departure, and the scaling values as described below, respectively.

$x = CTL - (D_d - A_a)L_{low}$ such that $x \geq 0$

*CTL*: lower level of transshipment containers

$L_{low}$: scaling parameter for lower level

$y = CTU - (D_d - A_a)L_{up}$ such that $y \geq 0$

*CTU*: upper level of transshipment containers

$L_{up}$: scaling parameter for upper level

For instance, we would like to generate $TRA_{34}$, where $A_3$=2 and $D_4$=5

*Input values:* *CTL*=10, *CTU*=100, $L_{low}$=1 and $L_{up}$=10

*Output value:* *x*=10-3*1

*y*=100-3*10=70

Hence, *U(7,70)* returns a value for $TRA_{ad}$

### ii. Generating $EXP_{td}$

Arrival period of containers: $t$

Departure period of ship d: $D_d$

If $D_d \leq t$ then $EXP_{td}$ =0

Otherwise $EXP_{td}$= *U(x,y)*

Assumptions are the same as the preceding case.

$x = CEL - (D_d - t)L_{low}$ such that $x \geq 0$

*CEL*: lower level of export containers

$L_{low}$: scaling parameter for lower level

$y = CEU - (D_d - t)L_{up}$ such that $y \geq 0$

$CEU$: upper level of exports containers

$L_{up}$: scaling parameter for upper level

For instance, we would like to generate $EXP_{48}$, where $D_8$=5

*Input values*: $CEL$=10, $CEU$=100, $L_{low}$=1 and $L_{up}$=10

*Output Values:* $x$=10-1*1=9  $y$=100-1*10=90

Hence, *U(9,90)* returns a value for $EXP_{td}$

### iii. Generating $IMP_{ak}$

Arrival period of ship $a$: $A_a$

Departure period of containers: $k$

If $k \le A_a$ then $IMP_{ak}$ =0

Otherwise $IMP_{ak}=U(x,y)$

Assumptions are the same as the preceding cases.

$x = CIL - (k - A_a)L_{low}$ such that $x \ge 0$

$CIL$: lower level of import containers

$L_{low}$: a scaling parameter for lower level

$y = CIL - (k - A_a)L_{up}$ such that $y \ge 0$

$CIU$: upper level of import containers

$L_{up}$: a scaling parameter for upper level

For instance, we would like to generate $IMP_{4,10}$, where $A_4$=2

*Input values*: $CIL$=10, $CIU$=100, $L_{low}$=1 and $L_{up}$=10

Output values:

$x$=10-8*1=2

$y$=100-8*10=20

Hence, *U(2,20)* returns a value for $IMP_{ak}$

# 4. STORAGE SPACE ALLOCATION

## 4.1. Problem Description

In order to sustain efficient container terminal operations, determining the storage space for each and every container is a substantial task that dramatically influences the effectiveness of many preceding and succeeding operations. Determining the locations to assign containers discharged from ships and containers arriving from other modalities are operational level issues. Deciding on the storage location of a container depends upon numerous parameters such as container type and space limitations. Since the arrival and departure period of each ship as well as each container are known well before the operations, planning space allocation for such systems in a rolling horizon seems to be reasonable. Therefore, the space allocation decisions are made via updated information that will be on hand throughout the rolling horizon. In this chapter, an aggregate level storage space allocation in a rolling horizon for two different scenarios is proposed.

As mentioned earlier, the ultimate objective of the container terminal operations is minimizing the ship turnaround times. The mathematical models proposed in this section assign containers or groups of containers to the storage locations by trying to minimize the total distance traveled. We construct a minimum cost network flow model for a considerable planning horizon, i.e. 7 or 10 periods, so as to minimize the total distance traveled by containers in the terminal area. Intuitively, it can be said that such an objective does not conflict with the overall objective of a container terminal, which is minimizing the ship turnaround times. Note that the time periods can be set to days, shifts, or hours, depending on the traffic and capacity of container terminal. Moreover, arrival and departure times of each ship can initiate the beginning or the end of a period in the rolling horizon since the remaining containers to the next period still occupy the space. In such a methodology, the number of containers currently seizing a location is an input for each and every period; the number of containers to be added on and the number of containers to be picked up from such a location are the decision variables. Eventually, the remaining containers to the next period are the output of this equilibrium. A brief illustration of the allocation problem as a network is depicted in Figure 4.1. There exist nodes for each location and each period as well as the source and sink nodes correspond to the supply and demand, respectively. The flows on arcs represent the decision variables and linkage among the nodes for a particular location in consecutive time periods.

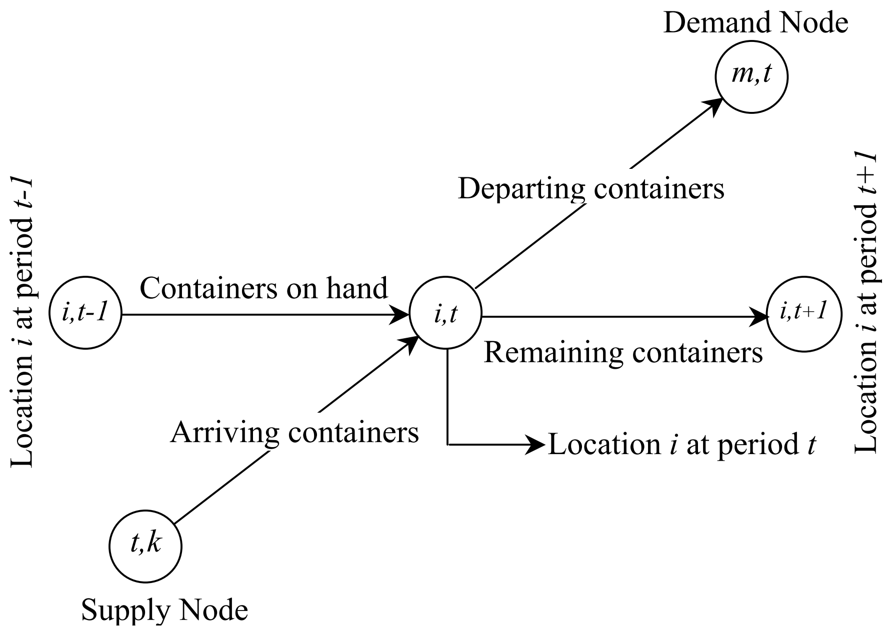



Figure 4.1 Nodes and arcs of the network model

The incoming arcs to such a location-period node from supply nodes demonstrate the number of containers added on the location whereas outgoing arcs to the demand nodes represent the number of containers to be uploaded from this location. The incoming arc from the location's preceding period demonstrates the remaining containers to the current period, whereas the outgoing arc to the location's succeeding period represents the remaining containers to the next period. The arcs illustrating decision variables are not capacitated, but the flow of each location-period node has a deterministic value since each location has a capacity. Thus, the arc capacities cannot exceed the flow limitation of the node that they can enter or leave. The remaining nodes of the network are the arrival and departure nodes as supply and demands nodes in transshipment network, respectively. The arrival/departure parameters should be duplicated for the corresponding arrival and departure periods so as to complete the flow of network. A simple network sketch given below illustrates the network structure for first three periods. Since the initial period consists of the containers on hand and the final period returns the remaining containers, the following network is just for two periods planning.

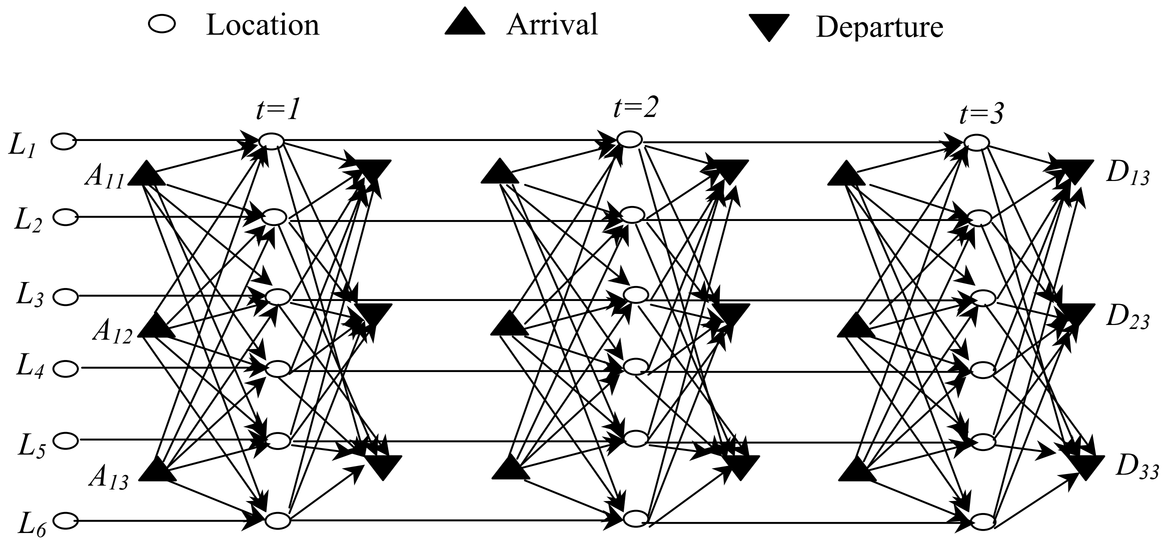



Figure 4.2. Network representation of the allocation model

In order to analyze this network modeling approach, two different allocation scenarios have been developed. The first one is a simple allocation model where the variations among container types and berths at the quayside do not exist. Simply a single commodity is allocated over a rolling horizon, in which the reference point is also set to be unique as the center of gravity of the berth area. The second model is more detailed, and it covers three different container types as import, export, and transshipment containers as well as several berth locations available for ships.

## 4.2. Simple Allocation Model

In the simple allocation model, we consider only one type of container to be unloaded from the ship, stored until the departure, and picked up after a certain time period. In this case, we have a single decision variable representing the number of containers stored at location *i*, which have arrival and departure times' of *t* and *k,* respectively. A reference point called "origin" is selected as the center of gravity at the quayside and it is assumed that all containers are discharged and uploaded from this point. The distance of a location from the quayside reference point is calculated with the parameters defined in the Section 3.2.

The stack height for each location is limited, so each location has a capacity of *C*, which is equal to the height of stack. Generating some buffer space at the yard area may result in efficiency in further re-handling operations in the yard area. Thus, some portion of this capacity could be reserved for buffer space. For the sake of simplicity, we suppose that the capacity of each location is the effective capacity on hand and it can be used without any re-handling concerns.

The number of arriving and departing containers are given in the ship schedules for each time period of the planning horizon. Since the actual information regarding the arrival and departure schedules is known only for a certain number of periods, the number of periods to be considered as the planning horizon is limited. For each update of the arrival and departure schedules, the network model based planning should be repeated. The arrival and departure schedules illustrate the arrival/departure parameters, $A_{tk}$, which denote the number of containers that arrived to the terminal in period *t* and will be departing in period *k* where $k > t$. It is assumed that each container is stored at the terminal for at least one period. If we illustrate the parameters as a table, we have an upper triangular matrix in which the values of diagonal and lower triangle are zero. Associated with each element of this parameter matrix, we have an arrival/departure constraint to compensate the total number of containers scattered over yard to the given parameters. Flow conservation constraints for each location and every period fulfill the balance of the network as mentioned earlier. Integrality constraints conclude the model.

The corresponding sets, parameters, decision variables, objective function, and constraints of the allocation model are described below.

*Notation*

$T$: Set of periods in the planning horizon, $t,k \in T$ $\quad T = \{1, 2, ..., T_{max}\}$

$L$: Set of locations, $i \in L$ $\quad L = \{1, 2, ..., L_{max}\}$

*Parameters*

$d_i$ : Distance between the origin of berth area and yard location $i$

$A_{tk}$: Number of containers arriving at period $t$ and departing at period $k$, where $k > t$ and $t,k \in T$

$C$: Capacity of each location in the yard, i.e. $C=5$ for all locations.

*Decision Variables*

$X_{it}$ : Number of containers located on $i$ during period $t$

$CON_{itk}$ : Number of containers loaded on $i$ in period $t$ and unloaded from $i$ in period $k$

*Objective Function*

$$Min\ Z = \sum_i \sum_t \sum_k d_i CON_{itk} \tag{2.1}$$

*Capacity Constraint*

$$X_{it} \le C \qquad \forall (i,t)\ i \in L,\ t \in T \tag{2.2}$$

*Arrival/Departure Constraints*

$$\sum_i CON_{itk} = A_{tk} \qquad \forall \{(t,k) \in T : t < k\},\ i \in L \tag{2.3}$$

*Flow Conservation Constraints*

$$X_{it-1} + \sum_{k>t} CON_{itk} = X_{it} + \sum_{k<t} CON_{ikt} \qquad \forall (i,t)\ i \in L,\ t \in T \tag{2.4}$$

*Integrality Constraints*

All decision variables are nonnegative integers.

$$X_{it} \in \mathbb{Z}^+ \cup \{0\}$$
$$CON_{itk} \in \mathbb{Z}^+ \cup \{0\} \qquad \forall \{(t,k) \in T : t < k\},\ i \in L \qquad (2.5)$$

Equation (2.1) is the objective function of the model, the total distance of the storage locations of containers handled throughout the planning horizon, which has to be minimized. The capacity constraint (2.2) indicates that the number of containers at a particular location in a particular time period cannot exceed the capacity of a location, *C*. Arrival-departure constraints in (2.3) guarantee that all containers arriving and departing should be scattered over the yard area. Flow conservation constraints of the network model are given in equation (2.4), which indicates that total number of containers at the beginning of period *t* in location *i* and containers loaded in *i* during *t* should be equal to the total number of containers located in *i* at the end of *t* and containers unloaded from *i* at *t*. Non-negative integrality constraints in (2.5) conclude the model. Fortunately, the integrality constraints can be omitted due to the totally unimodular property of the network flow model.

### 4.2.1. Side Constraints for Work Load Balancing

As illustrated in the terminal framework, unloading/loading operations at each block of the yard area are executed via dedicated yard cranes. In order to balance the workloads of these cranes and cumulative traffic around each block, we can introduce some balancing constraints. With such additional constraints, yard crane allocation can also be completed via a storage space allocation model. Since both unloaded and loaded containers exist for all blocks and all time periods, the total workload of a block in a given time period is the total number of containers processed. However, summing up number of containers added on a particular location and picked up from that location for any time period destroys the network structure since these constraints conflict with the balance of incoming and outgoing arcs in the network model. In what follows, the workload balancing constraints added to the network model are demonstrated where $\mu_{up}$ and $\mu_{low}$ denotes the upper and lower workload factors, respectively. The parameters $\mu_{up}$ and $\mu_{low}$ are set with respect to the density of terminal traffic and possible deadlocks due to crowding.

$B$: Set of blocks in the storage area, $b \in B \qquad B = \{1, 2, ..., B_{max}\}$

*Workload Balancing Constraints*

$$\sum_{i \in b} \sum_{k > t} CON_{itk} + \sum_{i \in b} \sum_{k < t} CON_{ikt} \leq \mu_{up} (\sum_{k > t} A_{tk} + \sum_{k < t} A_{kt}) \qquad \forall (b,t) \; b \in B, \; t \in T \qquad (2.6)$$

$$\sum_{i \in b} \sum_{k > t} CON_{itk} + \sum_{i \in b} \sum_{k < t} CON_{ikt} \geq \mu_{low} (\sum_{k > t} A_{tk} + \sum_{k < t} A_{kt}) \qquad \forall (b,t) \; b \in B, \; t \in T \qquad (2.7)$$

$$0 \leq \mu_{up(low)} \leq 1 \qquad (2.8)$$

Constraints (2.6) and (2.7) indicate that the total number of containers processed at each block in a period should be within the upper and lower total workload limitations.

Without loss of generality, we assume that the allocation model consists of *capacity*, *arrival/departure* and *flow conservation* constraints. *Workload balancing* constraints are supposed to be the side constraints for network model and these extensions will be analyzed in further studies. Lagrangian Relaxation may be a good candidate solve this extended model since the relaxed workload balancing constraints give us a network flow model.

### 4.3. Extended Storage Space Allocation Model

The containers handled in an intermodal terminal can be classified in several ways according to their size, content, origin and destination, arrival and departure ships, or owner cargo companies. In a typical intermodal container terminal, containers are transferred from ship to any mode at the landside, from landside to ship, or from ship to ship, and vice versa. Hence, there exist three different types of containers based on the origin-destination modes of transportation in practice; import containers, export containers and transshipment containers. The import containers arrive with an arriving ship, after having been stored for a particular time period until departure, they will be transferred to some other modality at the hinterland of the terminal via the vehicles of that other modality such as rail or trucks. The export containers are brought via the vehicles of other modalities from the landside; they stay at the yard until the departure of the ship and will depart after a certain time period with the associated departing ship. The transshipment containers arrive and depart via different ships at different time periods, so they stay at the yard between arrival and departure.

As mentioned earlier, we are attempting to allocate storage space at the yard area of the container terminal so as to minimize the total distance traveled by containers in the terminal area throughout the predetermined planning horizon. In the container terminal area, both the vehicles of the terminal, called internal trucks, and the vehicles of other modalities, called external trucks, can handle containers. The internal trucks perform the ship to yard and yard to ship transfers. The external trucks are utilized to retrieve import containers from the terminal and to transfer export containers to the terminal. Therefore, the export and import containers are handled by both internal and external trucks, whereas the transshipment containers occupy only internal trucks for transfers.

In general, operations handled by internal trucks play a more significant role when we consider the efficiency of container terminal operations since they are the major resources of container handling system at the terminal area. Recall that the major concern of both terminal management and ship liners is the ship turnaround times, which are mainly inferred from the operational effectiveness of internal handling equipment since the external trucks are almost out of control of the terminal. Inter terminal transfer times of external trucks are disregarded in most cases; however, terminal operators should take them into account so as to keep up an efficient terminal traffic mechanism. For instance, the intensity of external trucks traffic in the terminal area may not be desirable while ensuring the smooth flow of internal trucks. In our models, transfer distances or corresponding times spent inter terminal by both internal and external trucks are considered with different weights, where the weight associated with an internal truck distance is at least as great as the weight associated with the travel distance of an external truck. Although traffic control of vehicles is much more of an operational issue for container terminals, aggregate space allocation models should deal with the utilization of both internal and external trucks in order to offer considerable results for succeeding decisions. Thus, considering the weighted amounts of distances traveled by vehicles while assigning containers to the locations might be reasonable.

We assume that the arrival and departure periods for each ship are given as in the simple allocation model. The berth allocations are already done, so the devoted berth for each ship is also known. As explained in the simple model case, each location has a *capacity* of *C*. For each type of container, we have an upper triangular parameter matrix. Corresponding to each cellular value of these matrices, we have *arrival/departure* constraints to represent the equivalence of the total number of containers scattered over the yard to the available value on the right-hand side. The balance constraints that illustrate the equilibrium of incoming and outgoing arcs for each location at every period are called *flow conservation* constraints. The number of containers that are currently staying at a particular location is the first incoming arc and the number of containers that could be added on this location is the sum of other incoming arcs. The number of remaining containers is the first outgoing arc and the number of containers that could be picked up from this location is the total of remaining outgoing arcs. Nonnegative integrality of all decision variables concludes the model. Corresponding sets, parameters and decision variables are listed as the inputs to construct the model below.

*Notation*

$T$ : Planning horizon, $t, k \in T$ $\quad T = \{1, 2, ..., T_{max}\}$

$L$ : Set of locations, $i \in L$ $\quad L = \{1, 2, ..., L_{max}\}$

$S$ : Set of ships, $a, d \in S$ $\quad S = \{1, 2, ..., S_{max}\}$

$AS_t$ : Set of ships arriving at period $t$, $t \in T, AS_t \subset S$

$DS_t$ : Set of ships departing at period $t$, $t \in T, DS_t \subset S$

*Parameters*

$d_{ia}$ : Distance between location $i$ and berth of ship $a$, $i \in L, a \in S$

$d_i$ : Distance between location $i$ and yard gate, $i \in L$

$w_I$ : weight of distance traveled by internal trucks

$w_E$ : weight of distance traveled by external trucks

$w_I + w_E = 1$ and $w_I > w_E$ $\quad$ ($e.g., w_I : 0.75$ and $w_E : 0.25$)

$TRA_{ad}$ : Number of $(a - d)$ type containers (*transshipment* containers)

$t_a$ : arrival period of ship $a$, $t_a \in T$

$k_d$ : departure period of ship $d$, $k_d \in T$

$(a - d)$ type container arrives with ship $a$ and departs with ship $d$; $a, d \in S$ and $k_d > t_a$.

$EXP_{td}$ : Number of $(t - d)$ type containers (*export* containers)

$(t - d)$ type container arrives in period $t$ and departs with ship $d$; $t \in T, d \in S$ and $k_d > t$.

$IMP_{ak}$ : Number of $(a - k)$ type containers (*import* containers)

$(a - k)$ type container arrives with ship $a$ and departs in preriod $k$; $d \in S, k \in T$ and $k > t_a$.

$C_i$ : Capacity of location $i$, $C_i = 10 \quad \forall i \in L$

*Decision Variables*

$X_{it}$ : Number of containers located on $i$ during period $t$, $i \in L$, $t \in T$

$T_{iad}$ : Number of transshipment containers located on $i$, $i \in L$, $a, d \in S$

$E_{itd}$ : Number of export containers located on $i$, $i \in L$, $t \in T, d \in S$

$I_{iak}$ : Number of import containers located on $i$, $i \in L$, $a \in S$, $k \in T$

*Objective Function*

$$Min\ Z = Z^{TRA} + Z^{IMP} + Z^{EXP} \tag{3.1}$$

$$Z^{IMP} = \sum_i \sum_a \sum_k (w_I d_{ia} + w_E d_i) I_{iak} \tag{3.2}$$

$$Z^{TRA} = \sum_i \sum_a \sum_d w_I (d_{ia} + d_{id}) T_{iad} \tag{3.3}$$

$$Z^{EXP} = \sum_i \sum_t \sum_d (w_I d_{id} + w_E d_i) E_{itd} \tag{3.4}$$

*Capacity Constraints*

$$X_{it} \le C_i \qquad \forall i \in L, t \in T \tag{3.5}$$

*Arrival / Departure Constraints*

$$\sum_i T_{iad} = TRA_{ad} \qquad \forall a, d \in S \tag{3.6}$$

$$\sum_i E_{itd} = EXP_{td} \qquad \forall t \in T, d \in S \tag{3.7}$$

$$\sum_i I_{iak} = IMP_{ak} \qquad \forall a \in S, k \in T \tag{3.8}$$

*Flow Conservation Constraints*

$$X_{it-1} + Inflow = X_{it} + Outflow \qquad \forall i \in L, t \in T \tag{3.9}$$

$$\sum_{a \in AS_t} \sum_{d} T_{iad} + \sum_{d} E_{itd} + \sum_{a \in AS_t} \sum_{k} I_{iak} = Inflow \tag{3.10}$$

$$Outflow = \sum_{d \in DS_t} \sum_{a} T_{iad} + \sum_{d \in DS_t} \sum_{k} E_{ikd} + \sum_{a} I_{iat} \tag{3.11}$$

*Integrality Constraints*
All decision variables are nonnegative integers.

The objective function given in equation (3.1) is the summation of total weighted distances for three types of containers. The capacity constraint for each location over all periods is denoted in (3.5). Arrival and departure constraints in (3.6), (3.7) and (3.8) guarantee that the total number of each type of containers stored on the yard area is equal to the number of containers given as the parameters for transshipment, export and import containers, respectively. As illustrated in simple allocation case, flow conservation constraints satisfy the balance equation for each location and every period in (3.9). The integrality constraints can be removed due to the totally unimodular property of the network flow model.

## 4.4. Numerical Experiments

### 4.4.1. Optimum Allocation vs. Random Allocation

The results of storage space allocation model proposed in this chapter will be used as the inputs of the location matching models for both simple and extended scenarios. Bish (2003) proposes a solution for the assignment and matching problem without solving storage space allocation model to decide on the set of candidate locations for assignment. The results given in Bish (2003) for the assignment and matching problem assume that storage space allocation is performed randomly. So we will compare our model with this case, i.e. random storage space allocation.

Since the ultimate goal of a container terminal is minimizing the ship turnaround times, the objective function values of the models should be consistent with it. As mentioned earlier, the Manhattan Distances are used as the cost coefficients of the objective function for the storage space allocation model. Although this choice is intuitive, we attempt to show the suitability of this cost coefficient via comparing the results of location matching problem with respect to the random allocation and optimal allocation inputs. At the very beginning, this surrogate objective function seems to be appropriate since it is composed of the total of distances traveled over the planning horizon, which is parallel to the objective function of the location matching problem.

Let us assume that we would like to implement the optimal allocation strategy and obtain excellent results. However, a decision maker can question the accuracy of this allocation. In order to measure the performance of random allocation, containers should be scattered randomly over the yard for a sufficient number of times. Note that when we have $n$ random allocation experiments and $n$ goes to infinity, we reach the exact performance of random allocation. Let $Z^R$ denote the ultimate objective function of random allocation model, which is equal to the arithmetic mean of $n$ experimental objective function values of random allocation when $n$ goes to infinity.

$$\lim_{n\to\infty}\frac{1}{n}\sum_{k=1}^{n} Z_k^R = Z^R \qquad (4.1)$$

If we have $n$ different experiments regarding the random allocation, the probability that any location belongs to the solution space of such an allocation becomes identical to all other locations as $n$ goes to infinity. In other words, each location is equally preferable to be seized in random allocation for a large number of experiments. Thus, the average distance of locations from the berth ($d_i$) is arranged as the cost coefficient for such a strategy.

$$\overline{d_i} = \frac{1}{|L|}\sum_{i\in L} d_i \qquad (4.2)$$

, where $|L|$ is the cardinality of set $L$.

Hence the objective function value for random allocation $Z^R$ is,

$$Z^R = \overline{d_i} \sum_t \sum_k A_{tk} \tag{4.3}$$

Since the random allocation ignores the minimization of distances traveled, the $Z^R$ should be at least as great as the objective function value of optimal allocation model $Z^*$. Thus, the random allocation reveals an Upper Bound for objective function value.

$$Z^* \leq Z^R \tag{4.4}$$

The allocation models are coded in GAMS® IDE and solved via CPLEX® linear programming solver. The computational times vary around 5 minutes in a Celeron® 800 Mhz, 128 MB PC. We have completed 120 runs for four layouts, three weighted cost coefficients, and 10 different parameter sets. The generation procedures of the arrival and departure parameters are explained in Section 3.3.1.

Computational results regarding optimal and random allocation scenarios for extended allocation model show that there exist a robust improvement in the objective function value. Although the numerical experiments are conducted for four different layout types, the percentages of reduction in terms of objective function values are identical, due to the use of Manhattan Distance metric as the cost coefficients. Since we have an equal number of locations for each layout alternative, each location of a particular layout can be mapped to an identical location in terms of the distances from berths in other layout alternatives. Therefore the layout alternatives are indifferent for allocation models in terms of the cost coefficients.

Table 4.1Weight Parameters

| | *Weights 1* | *Weights 2* | *Weights 3* |
|---|---|---|---|
| $w_I$ | 1 | 0.75 | 0.5 |
| $w_E$ | 0 | 0.25 | 0.5 |

Table 4.2 Optimal vs. Random Allocation Results for Extended Allocation Model

| | Improvement % (Optimal vs. Random) | | |
|---|---|---|---|
| | $w_1$ | $w_2$ | $w_3$ |
| Problem Sets | 42 | 38 | 33 |
| | 42 | 38 | 32 |
| | 44 | 39 | 33 |
| | 41 | 36 | 30 |
| | 41 | 37 | 32 |
| | 45 | 39 | 33 |
| | 43 | 38 | 32 |
| | 41 | 37 | 31 |
| | 45 | 40 | 34 |
| | 42 | 37 | 32 |
| Average | 43 | 38 | 32 |

Since the improvement for this surrogate objective function is at least 30%, it can be said that the outputs of the storage space allocation model will be the reasonable inputs of the location matching models for further improvements.

# 5. LOCATION MATCHING

## 5.1. Problem Description

The routes of the vehicles to be dispatched in order to transfer containers from the berth to the yard area and vice versa during unloading and loading operations can be determined after obtaining the exact information regarding the locations of containers that will be stacked or retrieved. Upon completion of the space allocation models without side constraints for both simple and extended cases, loading locations and unloading locations at the storage yard for each time period throughout the planning horizon can be derived. The next step of our methodology is finding appropriate pairs of locations such that first element of a pair is an unloading location and the second one is a loading location so as to combine the trips of trucks in the terminal area. In other words, a dispatched vehicle carries an unloaded container from ship to the predetermined location at the yard area, goes to another location to pick up a container awaiting to be loaded on to the ship, and transfers the container to the ship. The presence of simultaneous unloading and loading operations at different quay cranes at different ships or different holds of a ship is the main motivation behind this cyclic vehicle orientation. Although there may exist other constraints ignored in simultaneous loading and unloading operations, we assume that they work in ideal cases. In the most recent study of Bish (2003) combining trips of vehicles and minimizing empty trips that occur between a loading and picking up location is the starting point. The Multiple Crane Constraint Vehicle Scheduling Location (MVSL) problem defined by Bish (2003) assigns each unloaded container to a particular location at the yard area and matches this location with the location of a container waiting to be loaded onto the ship. The transshipment type model proposed by Bish (2003) performs both assignment and matching tasks together within considerable times due to the totally unimodular structure of the integer programming model. With such a modeling approach, empty trips of vehicles at the yard area are minimized via matching loading locations with unloading locations. In Bish's (2003) models and analysis, each container listed in the unloading sequence has a number of potential locations and the locations of containers to be uploaded are given regardless of any precedence constraints. Thus, the containers waiting to be retrieved are identical in terms of the order of operation, which does not reveal the actual cases where some containers precede others in the uploading sequence due to the ship stowage plans.

Our methodology in this section is more aggregate than the model proposed by Bish (2003). Since the storage space allocation is performed on a rolling horizon, the location matching model is considered over the same horizon so as to integrate both models and compare the results regarding any allocation vs. optimal allocation. The process of matching predetermined loading and unloading locations over the rolling horizon constitutes the well-known transportation problem, which is illustrated in Figure 5.1 for period *t*.

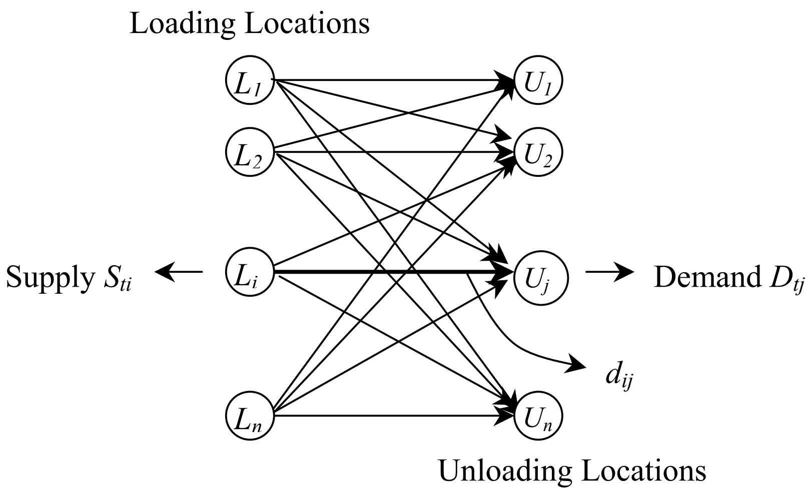



Figure 5.1 Balanced transportation network for period *t*

Although the transportation model for the whole planning horizon can be constructed, the decomposition of the main model into *t* identical transportation models does not influence the overall objective since these distinct models are independent of each other in terms of input parameters. Of course, decomposition of the model into *t* identical models reduces the computational complexity in terms of the size of the models.

The number of loading and unloading containers may not be equal for a time period due to the random arrival/departure parameters, so the transportation model is unbalanced for many cases. In order to remove this imbalance, dummy locations are created for each unloading and loading location. A location-dummy matching has a cost of $2d_i$, where $d_i$ is the distance of traveling from berth to location $i$, 0 for $i$ to dummy, and $d_i$ for dummy to berth. Location-dummy pairs are not restricted with the number of supply/demand slacks. Thus, there may exist three types of cyclic trips at the terminal area as follows:

- Supply-Demand Match; corresponding cost coefficient is $d_i + d_{ij} + d_j$
- Dummy-Demand Match; corresponding cost coefficient is $2d_j$
- Supply-Dummy Match; corresponding cost coefficient is $2d_i$

The figures below depict the routes for each type of matching respectively.

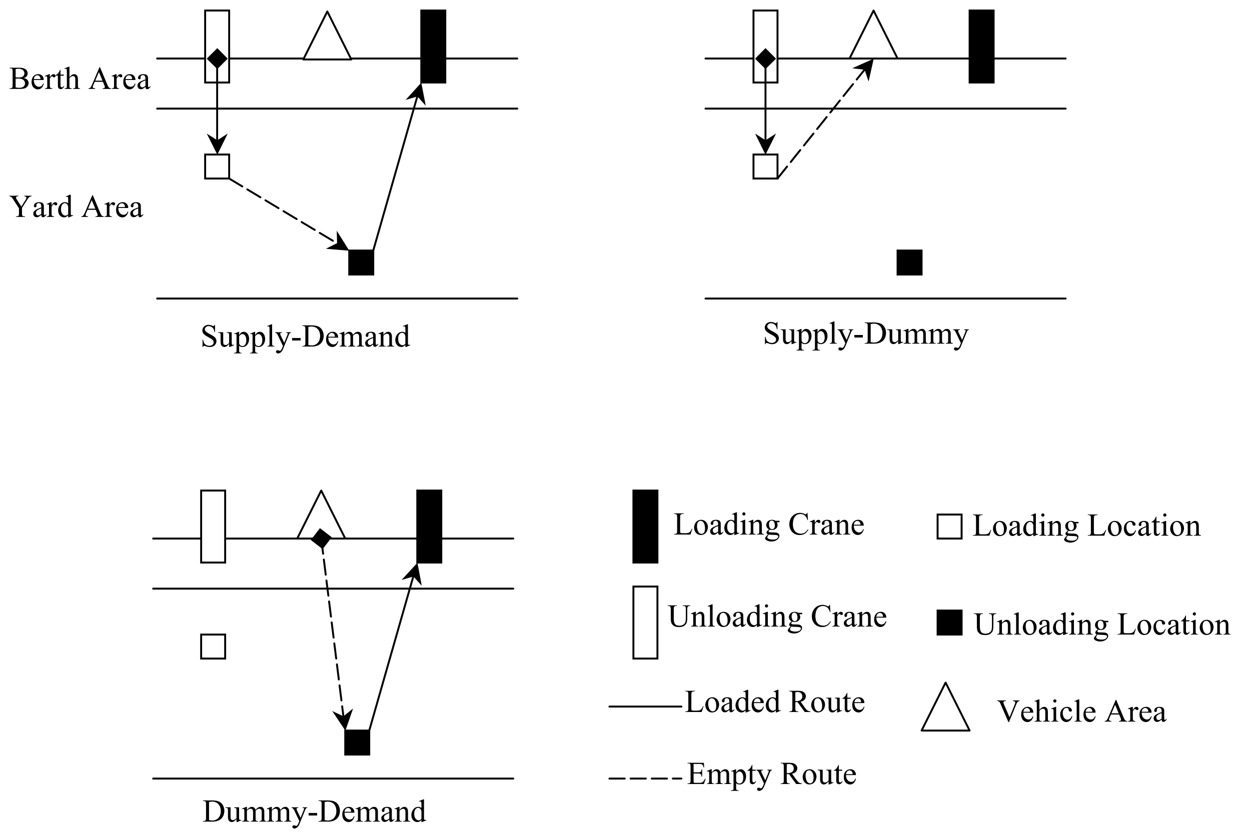



Figure 5.2 Possible Routes of Trucks

Note that the storage space allocation models proposed in Chapter 4 can be expanded so as to include location matching. Although expressing these two optimization purposes in a single mathematical model is theoretically possible, the explosion of model sizes in terms of decision variables and constraint sets forces us to decompose the combined model into the space allocation and the location matching models. Thus, the output of the storage space allocation model, which offers the possible loading and unloading locations with respect to the objective function based on Manhattan distances, reveals an input for the matching problem as the supply and demand parameters. The combined allocation-matching model is described in the next section to illustrate the complexity of such an integrated issue. The succeeding location matching models for simple and extended allocation cases are proposed in the following sections.

## 5.2. Combined Allocation-Matching Model

Performing allocation and matching throughout the planning horizon is theoretically possible. As proposed in Bish (2003), the assignment of containers to the locations and matching these locations so as to minimize the total turnaround time is one alternative. However, our planning methodology regarding the space allocation is over a rolling horizon. Thus, the integrated allocation-matching model for our case should also be considered over such a rolling horizon. Although the location matching step can be divided into $t$ separate models, combining these two models necessitates dealing with them simultaneously.

The combined allocation-matching model determines the storage locations for each type of arriving-departing container group and matches the loading and unloading locations over the planning horizon simultaneously so as to minimize the total distance traveled by trucks to transfer containers during loading and unloading operations. Therefore, the objective function is composed of the summation of total distances associated with those pairs of locations that are matched. The set of constraints can be grouped into two categories such as the *allocation* related constraints and the *matching* related constraints. The allocation constraints ensure the capacity limitations for each location and each period, the arrival departure constraints, and the equilibrium that represents the balance of incoming and outgoing arcs in the network as illustrated in the storage space allocation models. The matching constraints guarantee that the total number of containers loaded on a particular location at a period is equal to the number of matching originated from that location and the number of supply-dummy matching associated with that location. Moreover, the number of containers uploaded from a particular location should be equal to the number of matching finalized at that location and the demand-dummy matching associated with that location for all time periods.

The integrality constraints conclude the model. They cannot be omitted since this model does not have totally unimodularity property. Thus, computational time to solve this model is extremely longer than the total time consumed for the decoupled models. The combined allocation-matching model over the planning horizon is given as follows.

*Notation*

$T$ : Set of periods $\quad T = \{1, 2, ..., T_{max}\}$

$L$ : Set of locations $\quad L = \{1, 2, ..., L_{max}\}$

*Parameters*

$d_{ij}$ : Distance between location $i$ and location $j$, $i, j \in L$

$dis_{ij}$ : Total distance corresponding to the $i$ - $j$ pair, $i, j \in L$

$dis_{ij} = d_i + d_{ij} + d_j$

$C$ : Capacity of a storage location

$A_{tk}$ : Number of containers arriving at period $t$ and departing at period $k$

*Decision Variables*

$CON_{itk}$ : Number of $t$ - $k$ type containers located on $i$

$X_{it}$ : Number of containers located on $i$ at period $t$

$M_{tij}$ : Number of $supply - demand$ matching between $i$ and $j$ in period $t$

$SD_{it}$ : Number of $supply - dummy$ matching in period $t$

$DD_{jt}$ : Number of $dummy - demand$ matching in period $t$

*Objective Function*

$$Min\ Z = \sum_t \sum_i \sum_j dis_{ij} M_{ijt} + 2\sum_t \sum_i d_i SD_{it} + 2\sum_t \sum_j d_j DD_{jt} \qquad (2.1)$$

*Allocation Constraints*

$$X_{it} \le C \qquad \forall i \in L,\ t \in T \qquad (2.2)$$

$$\sum_i CON_{itk} = A_{tk} \qquad \forall \{t, k \in T : t < k\} \qquad (2.3)$$

$$X_{it-1} + \sum_{k>t} CON_{itk} = X_{it} + \sum_{k<t} CON_{ikt} \qquad \forall\ i \in L,\ t \in T \qquad (2.4)$$

*Matching Constraints*

$$\sum_j M_{ijt} + SD_{it} = \sum_{k>t} CON_{itk} \qquad \forall\ i \in L,\ t \in T \qquad (2.5)$$

$$\sum_i M_{ijt} + DD_{jt} = \sum_{k<t} CON_{jkt} \qquad \forall\ j \in L,\ t \in T \qquad (2.6)$$

*Integrality Constraints*

All the decision variables are nonnegative integers.

The objective function value in (2.1) denotes the total distance traveled by the containers in the terminal area throughout the planning horizon. The set of constraints is divided into two groups, namely allocation and matching constraints. The allocation constraints ensure the capacity limitation for each location at every period as in (2.2), the equality of total number of containers to the parametric values in (2.3) as well as the balance of inflow and outflow in (2.4). The matching constraints in (2.5) and (2.6) guarantee the equilibrium for total supply and total demand values for each location at every period. The nonnegative integrality constraints conclude the model.

### 5.3. Location Matching for Simple Allocation

As mentioned earlier, we decouple the combined allocation-matching model as two models, namely the storage space allocation and the location matching in order to obtain results in a reasonable amount of time. After solving the storage space allocation models provided in the preceding chapter, we are ready to construct the location matching models. Since the solution set of the space allocation model for each time period is an input as the supply and demand parameters, the location matching model should be introduced in a lower hierarchy. Each storage space location loaded with a number of containers at a particular time period represents a source node with a supply of the number of containers loaded on that location at that period. Similarly, each storage location emptied by a number of containers at a particular time period represents a sink node with a demand of the number of containers uploaded from that location at that period. Therefore, the loading and unloading locations, which may overlap, are determined after completion of the storage space allocation model. The overlapping of supply and demand locations can be prevented via additional constraints. Without loss of generality, we assume that each location can be a loading or an unloading location, or both of them at a particular time period. Matching these supply and demand nodes for each time period is the well-known transportation problem, where the cost coefficients are the distances associated with each matching as described before. Let us assume that we have $T$ time periods for planning; with the initial and remaining periods, it will be $T+2$; thus, there exist $T+2$ identical transportation problems each having specific input values determined in the allocation phase. If the initial distribution of containers is given, we have $T+1$ models.

### 5.3.1. Model

In order to formulate the location matching model, we have to determine the parameters and decision variables. The supply-demand, supply-dummy and dummy-demand routes are the decision variables for this problem. The costs associated with each type of route (matching) are described in Section 5.1. The objective function value is the sum of the total costs associated with each type of matching. Note that a supply-demand pair handles two containers throughout a single route, whereas the supply-dummy and dummy-demand pairs handle just one container throughout the route. The constraints regarding the transportation problem ensure that the total number of containers supplied should equal the total number of supplies as well as that the total number of containers demanded should equal the total demand. The integrality of the decision variables is automatically satisfied because of the totally unimodular structure of the transportation model. The sets, parameters, decision variables, objective function, and constraints of the location matching model for the whole planning horizon are described below.

*Notation*

$T$ : Set of periods $\quad T = \{1, 2, ..., T_{max}\}$

$L$ : Set of locations $\quad L = \{1, 2, ..., L_{max}\}$

$I_t$ : Set of loadable locations for period $t$, $I_t \subseteq L, t \in T$

$J_t$ : Set of unloadable locations for period $t$, $J_t \subseteq L, t \in T$

*Parameters*

$d_{ij}$ : Distance between location $i$ and location $j$, $i, j \in L$

$dis_{ij}$ : Total distance corresponding to the $i$ - $j$ pair, $i, j \in L$

$dis_{ij} = d_i + d_{ij} + d_j$

$S_{ti}$ : Total number of containers loaded on $i$ in period $t$

$D_{tj}$ : Total number of containers unloaded from $j$ in period $t$

*Decision Variables*

$X_{tij}$ : Number of $supply - demand$ matching between $i$ and $j$ in period $t$

$SD_{ti}$ : Number of $supply - dummy$ matching in period $t$

$DD_{tj}$ : Number of $dummy - demand$ matching in period $t$

*Objective Function*

$$Min\ Z = \sum_t \sum_i \sum_j dis_{ij} X_{tij} + 2\sum_t \sum_i d_i SD_{ti} + 2\sum_t \sum_j d_j DD_{tj} \qquad (3.1)$$

*Balance Constraints (Supply/Demand)*

$$\sum_j X_{tij} + SD_{ti} = S_{ti} \qquad \forall t \in T, i \in I_t \qquad (3.2)$$

$$\sum_i X_{tij} + DD_{tj} = D_{tj} \qquad \forall t \in T, j \in J_t \qquad (3.3)$$

*Integrality Constraints*

All the decision variables are nonnegative integers.

The objective function in (3.1) expresses the total distance traveled for all available matches, which should be minimized to reduce ship turnaround times. Since there is no limitation on the number of supply-dummy and dummy-demand matches, the exact distance values are put on the objective function value rather than sufficiently large numbers. However, this kind of model tends to minimize the number of dummy originating or terminating routes so as to decrease the empty travels. The supply and demand constraints satisfy the characteristic transportation model constraints or balance of inflow and outflow for each location and at every period as denoted in (3.2) and (3.3) respectively. Due to the totally unimodularity, the integrality constraints are omitted.

### 5.3.2. Comparison of Results of Random and Optimal Allocation Scenarios

In order to analyze the improvement gained through optimal allocation, we have run the matching models with CPLEX® linear programming solver for the results of optimal and random allocations. The model is coded in the GAMS® IDE, which converts the codes into the CPLEX® solver input format. Although there exist several algorithm options for CPLEX® such as *Dual Simplex*, *Barrier*, and *Network Simplex*, we prefer the *Barrier* algorithm since it is recommended to solve the problems with sparse constraint matrices faster. The run times vary around 30 minutes, depending on the parameter set in a Celeron® 800 Mhz, 128 MB PC. The computational times are a magnitude shorter, i.e. 5 minutes, in a Xeon® 1Ghz, 1GB Workstation.

The arrival and departure parameters are set arbitrarily as explained in Section 3.3.1. For 10 arbitrarily generated arrival/departure parameter sets, the objective function values of matching model (total distance traveled by vehicles) are listed in the table below. There exists a considerable reduction in objective function value, changing within the range of 32%-40% and average reduction in objective function value is 34.52% for these instances. The amount of time consumed during the numerical experiments varies within 20-40 minutes depending on the complexity of the problem parameters, but they are extremely short compared to the typical planning period, such as 10 days. Although making decisions on a rolling horizon requires reasonable updates and distinct runs after each update, we claim that the overall performance will be the same since we do not have any chance to update parameters.

Table 5.1 Comparison of Random vs. Optimal results

| Parameter $A_{tk}$ | (Random + Matching) /(Optimal + Matching) | Reduction % |
|---|---|---|
| Set 1 | 6679060 / 4455940 | 33.28 |
| Set 2 | 6088520 / 3898560 | 35.97 |
| Set 3 | 5944840 / 3810440 | 35.9 |
| Set 4 | 6280150 / 4121610 | 34.37 |
| Set 5 | 7201690 / 4750790 | 34.03 |
| Set 6 | 6446890 / 4222030 | 34.51 |
| Set 7 | 5684080 / 3397380 | 40.23 |
| Set 8 | 6774010 / 4693190 | 30.72 |
| Set 9 | 6630050 / 4555010 | 31.9 |
| Set 10 | 5353270 / 3484570 | 34.91 |
| | Average Reduction | 32.58 |
| | Standard Deviation | 2.58 |

It can be concluded that the results of the allocation model offer more reasonable inputs than those of random allocation case for the location matching model. In other words, it can be claimed that the cost coefficients considered in the objective function values of the both modeling steps are consistent.

### 5.4. Location Matching for Extended Allocation

The succeeding location matching model also exists for extended storage space allocation scenario. The general approach is similar to the previous case, but the problem sizes are larger due to the extensions. The output of the storage space allocation model for extended case as described in Section 4.3 constitutes the input for the location matching model. However, the location matching for extended allocation only deals with the routes associated with internal trucks. Since there exist three different weight categories defining surrogate objective functions in storage space allocation models and four different layout alternatives a detailed analysis regarding the solutions can be done at the end of this chapter.

### 5.4.1. Model

Similar to the location matching model for the simple allocation case, the location matching model associated with the extended allocation is also in the form of a transportation model. However, determining the supply and demand values is more complicated since in the former model we have considered only the travels of internal trucks. The locations in which a number of containers are added on via internal trucks represent the supply nodes. Similarly, the locations where a number of containers are uploaded from via internal trucks represent the demand nodes. Therefore, the arriving export containers and departing import containers are ignored while computing the supply and demand values. The supply and demands values can be calculated as follows:

*Notation*

$T$ : Set of periods $\quad T = \{1, 2, ..., T_{max}\}$

$L$ : Set of locations $\quad L = \{1, 2, ..., L_{max}\}$

$S$ : Set of ships, $a, d \in S$ $\quad S = \{1, 2, ..., S_{max}\}$

$AS_t$ : Set of ships arriving at period $t$, $t \in T, AS_t \subset S$

$DS_t$ : Set of ships departing at period $t$, $t \in T, DS_t \subset S$

$I_t$ : Set of loadable locations for period $t$, $I_t \subseteq L, t \in T$

$I_t = I_t^{tra} \cup I_t^{imp}, \ \forall t \in T$

$I_t^{tra}$ : Set of loadable locations for transshipment type containers for period $t$

$I_t^{imp}$ : Set of loadable locations for import type containers for period $t$

$J_t$ : Set of unloadable locations for period $t$, $J_t \subseteq L, t \in T$

$J_t = J_t^{tra} \cup J_t^{exp}, \ \forall t \in T$

$J_t^{tra}$ : Set of unloadable locations for transshipment type containers for period $t$

$J_t^{exp}$ : Set of unloadable locations for export type containers for period $t$

*Parameters*

$S_{ti}^{tra}$ : Total number of transshipment type containers loaded on $i$ in period $t$

$S_{ti}^{imp}$ : Total number of import type containers loaded on $i$ in period $t$

$S_{ti}$ : Total number of containers loaded on $i$ in period $t$

$$S_{ti} = S_{ti}^{tra} + S_{ti}^{imp}$$

$D_{tj}^{tra}$ : Total number of transshipment type containers unloaded from $j$ in period $t$

$D_{tj}^{exp}$ : Total number of export type containers unloaded from $j$ in period $t$

$D_{tj}$ : Total number of containers unloaded from $j$ in period $t$

$$D_{tj} = D_{tj}^{tra} + D_{tj}^{exp}$$

The distances from berth to any location and vice versa vary over the planning horizon since the berth locations of discharging and uploading ships change by period.

$d_i^a$ : Distance between discharging berth of ship $a$ and location $i$ , $a \in S, i \in L$

$d_j^d$ : Distance between uploading berth of ship $d$ and location $j$ , $a \in S, j \in L$

$d_{ij}$ : Distance between location $i$ and location $j$, $i, j \in L$

$dis_{ij}$ : Total distance corresponding to the $i$ - $j$ pair, $i, j \in L$

$$dis_{ij} = d_i^a + d_{ij} + d_j^d$$

*Decision Variables*

The decision variables are the same as the location matching for simple model.

$X_{tij}$ : Number of $supply - demand$ matching between $i$ and $j$ in period $t$

$SD_{ai}$ : Number of $supply - dummy$ matching for ship $a$, $a \in AS_t$

$DD_{dj}$ : Number of $dummy - demand$ matching for ship $d$, $d \in DS_t$

*Objective Function*

$$Min\ Z = \sum_{t}\sum_{i}\sum_{j} dis_{ij} X_{tij} + 2\sum_{a}\sum_{i} d_i^a SD_{ai} + 2\sum_{d}\sum_{j} d_j^d DD_{dj} \tag{4.1}$$

*Balance Constraints (Supply/Demand)*

$$\sum_{j} X_{tij} + \sum_{a \in AS_t} SD_{ai} = S_{ti} \qquad \forall t \in T, i \in I_t \tag{4.2}$$

$$\sum_{i} X_{tij} + \sum_{d \in DS_t} DD_{dj} = D_{tj} \qquad \forall t \in T, j \in J_t \tag{4.3}$$

*Integrality Constraints*

All decision variables are nonnegative integers.

The objective function in equation (4.1) denotes the total distance traveled throughout the planning horizon. The balance constraints in (4.2) and (4.3) guarantee that the number of containers supplied and demanded are equal to the existing supply and demand values, respectively. As mentioned earlier, the integrality of the decision variables can be omitted because of the transportation model structure. Although additional constraints to remove overlapping of unloading and loading can be added to refine the modeling, we considered the most basic structure.

### 5.4.2. Comparison of Results of Random and Optimal Allocation Scenarios

In order to compare the results of matching models with random allocation and optimal allocation inputs, we have solved the models with CPLEX® linear programming solver. As mentioned in the simple case, models are coded in GAMS® IDE environment. The linear programming method option is set to the *Barrier Algorithm*. The computational times are extremely small compared to the planning horizon and almost same as the values given for the simple case. Since we have constructed models for four different layout alternatives and three different weighted cost coefficients for storage space allocation models, we have 12 different input sets for location matching models. Additionally, the random allocation outputs constitute a matching input for four different layout types. Thus, we have 16 different inputs for each of the 10 data sets and 160 numerical experiments to cover all combinations.

As mentioned earlier, the arrival and departure parameters are set arbitrarily as explained in Section 3.3.1. For 10 arbitrarily generated arrival/departure parameter sets, we can find the total distances traveled subject to the matching model constraints. The numerical values listed in the tables below represent the percentage of reduction in the objective function value when we use the optimal allocation strategy rather than random inputs.

Table 5.2 Comparison of Random vs. Optimal results (Layouts 1&2)

| Improvement % ( Random + Matching ) vs. ( Optimal + Matching ) | | | | | | |
|---|---|---|---|---|---|---|
| | Layout 1 | | | Layout 2 | | |
| Set | $w_1$ | $w_2$ | $w_3$ | $w_1$ | $w_2$ | $w_3$ |
| 1 | 15.2 | 14.5 | 14.0 | 18.1 | 13.9 | 13.2 |
| 2 | 16.2 | 15.3 | 12.9 | 17.4 | 16.7 | 14.3 |
| 3 | 18.2 | 17.5 | 14.3 | 20.4 | 19.7 | 16.6 |
| 4 | 18.2 | 17.7 | 15.4 | 19.7 | 19.2 | 16.8 |
| 5 | 16.7 | 16.5 | 12.9 | 18.8 | 18.6 | 14.8 |
| 6 | 16.8 | 16.1 | 14.6 | 19.4 | 18.9 | 16.9 |
| 7 | 17.3 | 16.1 | 13.7 | 18.0 | 17.1 | 14.8 |
| 8 | 14.8 | 14.7 | 12.5 | 16.6 | 16.4 | 13.9 |
| 9 | 16.1 | 15.4 | 13.1 | 17.5 | 16.8 | 14.3 |
| 10 | 17.4 | 16.4 | 13.8 | 18.6 | 17.8 | 15.0 |
| Average | 16.7 | 16.0 | 13.7 | 18.5 | 17.5 | 15.1 |
| Std. Dev. | 1.13 | 1.06 | 0.91 | 1.16 | 1.72 | 1.29 |

Table 5.3 Comparison of Random vs. Optimal results (Layouts 3&4)

| Improvement % ( Random + Matching ) vs. ( Optimal + Matching ) | | | | | | |
|---|---|---|---|---|---|---|
| | Layout 3 | | | Layout 4 | | |
| Set | $w_1$ | $w_2$ | $w_3$ | $w_1$ | $w_2$ | $w_3$ |
| 1 | 17.8 | 17.0 | 13.2 | 15.0 | 14.9 | 12.5 |
| 2 | 16.1 | 15.4 | 13.4 | 13.5 | 13.3 | 11.6 |
| 3 | 18.0 | 16.5 | 13.2 | 14.8 | 14.9 | 12.1 |
| 4 | 18.2 | 17.4 | 14.4 | 13.0 | 13.8 | 11.7 |
| 5 | 18.4 | 18.0 | 13.0 | 14.9 | 15.0 | 12.0 |
| 6 | 18.6 | 16.6 | 13.5 | 15.9 | 15.9 | 12.6 |
| 7 | 17.2 | 16.3 | 13.7 | 15.7 | 14.9 | 13.7 |
| 8 | 17.3 | 16.8 | 12.4 | 13.4 | 13.4 | 10.8 |
| 9 | 16.1 | 15.0 | 12.1 | 13.7 | 13.6 | 11.7 |
| 10 | 17.1 | 16.1 | 12.6 | 14.5 | 15.0 | 12.6 |
| Average | 17.5 | 16.5 | 13.2 | 14.4 | 14.5 | 12.1 |
| Std. Dev. | 0.88 | 0.88 | 0.68 | 0.98 | 0.86 | 0.77 |

The results show that there exists an improvement of at least 10% and at most 20% for all instances. Although, the percentage of improvement varies within this range, the fourth layout alternative is more costly than the others since mixing the orientation of blocks necessitate to travel longer distances to turn around blocks. Hence, the results of random and optimal allocation for such mixed layouts get close to each other. The weights of cost coefficients of storage space allocation model are other significant parameters to discuss. Since the objective function value of the location matching model is the total distance traveled by the internal trucks, the most consistent surrogate objective function for the storage space allocation model should emphasize the distances traveled by internal trucks. As seen in the tables above, the greatest improvements are gained in the $w_1$ category, which ignores the distances traveled by external trucks. The $w_3$ results are the smallest ones since the weights of distances belonging to this type are identical for external and internal trucks.

Compared to the results of the simple matching case, it can be concluded that the results of the extended matching model are less robust. However, the distances traveled by the internal trucks are taken into account while formulating the extended matching case. Note that the travels of external trucks for both import an export containers are ignored, thus the effect of the surrogate objective function of the storage allocation model on the matching step becomes smaller. Nevertheless, 10% improvement for the worst case in such an aggregate modeling is quite impressive.

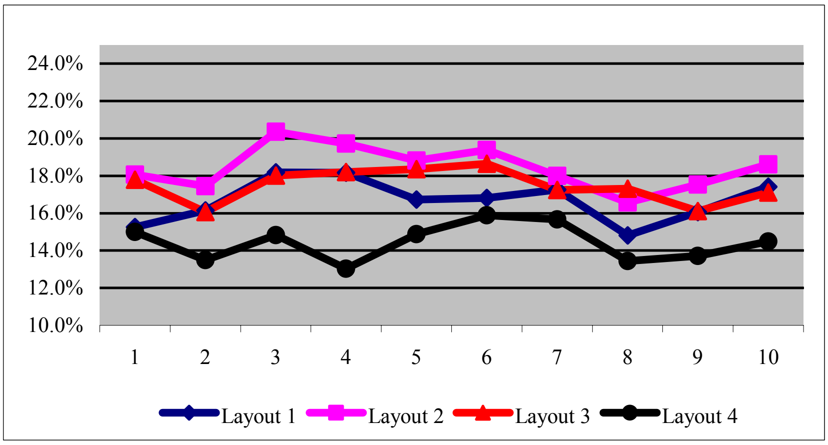



Figure 5.3 Improvement $w_1$

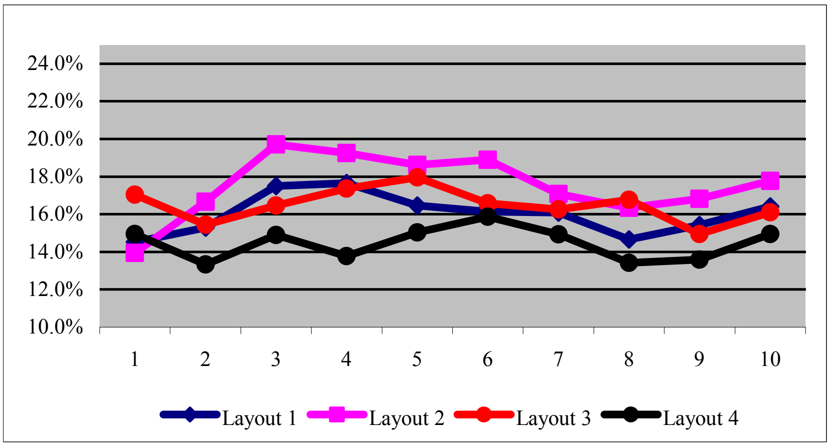



Figure 5.4 Improvement $w_2$

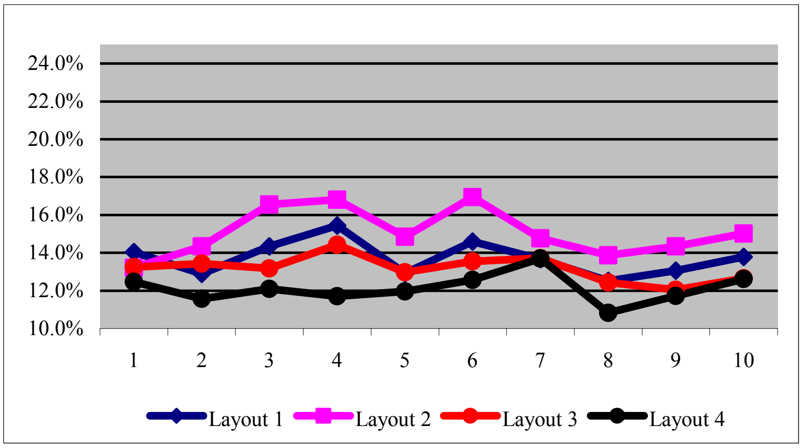



Figure 5.5 Improvement $w_3$

# 6. VEHICLE SCHEDULING

As defined at the beginning of this thesis, the turnaround time of a ship is equal to the total unloading and loading time. Since the ultimate objective of a container terminal is minimizing ship turnaround time, unloading and loading operations should be performed as efficient as possible. Thus, the time consumed during unloading and loading operations should be reduced so as to minimize ship turnaround time. Completion time of unloading operations for a containership is the time when the last container is picked up from the ship via a quay crane and is loaded on a transfer vehicle. Similarly, ship loading or stowage is completed after uploading the last container in the loading sequence. Since the quay cranes are fed by internal trucks in order to transfer containers from berth to yard and vice versa, the major concern becomes the existence of trucks on time nearby the quay cranes. The completion times associated with the container transfers scheduled on the vehicles lengthen due to the absence of vehicles at the berth, the queues on yard crane, and congestion. Thus, the efficient scheduling of these trucks and other terminal equipment such as the quay cranes and yard cranes is necessary to sustain the smooth flow of containers as well as the ships at the terminal.

In this chapter, we discuss the vehicle scheduling issues in container terminals. As mentioned earlier, internal trucks are the transshipment vehicles between the berth area and the storage yard. Due to the limitations on the number of container handling equipment such as quay cranes, trucks and yard cranes, the implementation of an efficient scheduling mechanism is necessary in order to minimize ship turnaround times while maximizing utilization of terminal facilities.

The aggregate storage space allocation and the location matching have been discussed in Chapter 4 and Chapter 5, respectively. Vehicle scheduling models that will be proposed in this chapter deal with the output of location matching model. Recall that a list of pairs of locations or jobs and corresponding processing times or total distances traveled throughout the cyclic route associated with each pair is the consequence of the location matching problem. Although there may be precedence constraints for the unloading and loading containers in practice, we assume that each job associated with container transfers is identical in terms of the operational sequence.

In Section 6.1, the vehicle scheduling problem is analyzed by assuming yard / quay crane capacity is sufficient. In the most basic case, only vehicles are assumed to be the constraints. The quay crane constraint is appended as the next case. Eventually, the scheduling of quay crane-internal truck-yard crane triple is proposed as the integrated scheduling of terminal equipment.

## 6.1. Vehicle Constraint Scheduling

In this section, we assume that the inter-terminal trucks (vehicles) are the major constraints for the handling of containers. In addition, containers are available to be picked up and loaded at both quayside and yard area for all instances. The most vital impact of these assumptions is the absence of the crane constraint in discharging/uploading operations, which may effect the exact turnaround time of the ships. Also, it is assumed that the yard cranes are available to serve the vehicles on time. We have a list of jobs (pairs of locations) for each time period of the planning horizon and the corresponding processing time for each job. Thus, the vehicle scheduling problem evolves into the parallel machine scheduling problem, where the identical machines are the vehicles and the independent processing times of jobs are the travel times corresponding to the routes.

Since it has already been established in the machine scheduling literature that $P2 || C_{max}$, 2-parallel machines subject to minimize the makespan, is NP-hard, various heuristics have been proposed to get near optimal solutions. The most frequently used heuristic is the Longest Processing Time First (LPT) heuristic in the machine scheduling concept. In our models, each job corresponds to a route traveled by the dispatched vehicle, and the processing time of each job is the total distance of the corresponding route. Since the processing times of yard and quayside loading and unloading operations are assumed to be deterministic and identical for all operations, they are ignored in the calculation of total traveling time of each route. Therefore, the total traveling time of *(i-j)* pair is the sum of traveling time from quayside to location *i*, location *i* to location *j* and location *j* to quayside. If we assume that vehicles move at a constant speed, total traveling distances extracted from the matching problem can be used instead of traveling times. The Longest Traveling Time First (LTT) heuristic, non-optimality, and worst-case analysis of the heuristic are given below.

### 6.1.1. LTT Heuristic

The LTT heuristic assigns at $t=0$ the longest $v$ routes to the first $v$ vehicles, and whenever a vehicle is freed, the longest unscheduled route is put on that vehicle. The process of scheduling via LTT is as follows:

i. Sort the traveling times of pairs (jobs) in an ascending order.
ii. Assign these jobs to the vehicles with respect to the sequence above, where ties are broken arbitrarily.

This heuristic tries to place the shorter jobs towards the end of the schedule where they can be used for the balancing the loads.

### 6.1.2. Non-optimality of the LTT Heuristic

The non-optimality of the LTT heuristic is shown via a simple example as illustrated below.

**Example 6.1.** Let us consider 7 jobs (routes) to be scheduled on 3 identical parallel vehicles. The traveling times associated with each route are denoted as set $J_t:\{3,3,3,5,5,4,4\}$. The schedule found via LTT heuristic will be as follows $S_{LTT}$:{(5,5,3), (5,3), (4,4)}, which has a makespan of 11 time units whereas the optimal schedule having a makespan of 9 time units is $S_{OPT}$:{(5,4), (5,4), (3,3,3)}

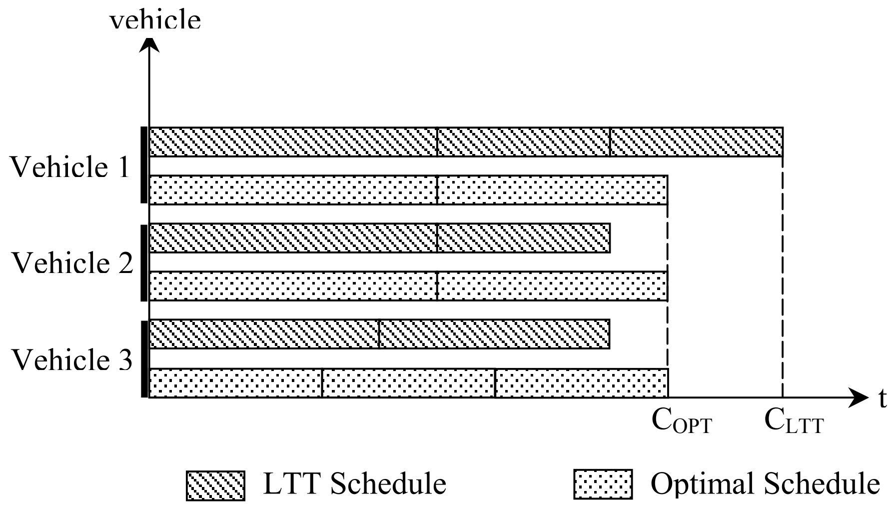



Figure 6.1 Non-optimality of the LTT heuristic

### 6.1.3. Worst-Case Analysis of the LTT Heuristic

In Pinedo (1995), the worst-case performance of the heuristic is analyzed and a proof is completed by contradiction. Thus, the following inequality is concluded for $v$ parallel vehicles, where $C_{max}(LTT)$ denotes the *makespan* of the heuristic and $C_{max}(OPT)$ denotes the *makespan* of an optimal schedule.

$$\frac{C_{max}(LTT)}{C_{max}(OPT)} \leq \frac{4}{3} - \frac{1}{3v}$$

## 6.2. Quay Crane Constraint Scheduling

The vehicle scheduling problem with vehicle constraint proposed in the preceding section seems improper for most of the cases where crane operations and schedules determine the turnaround times of ships. In this section, we illustrate the scheduling problem as a parallel machine scheduling problem with common servers, in which machines are vehicles and servers are quay cranes. The Figure 6.2 denotes the structure of the problem.

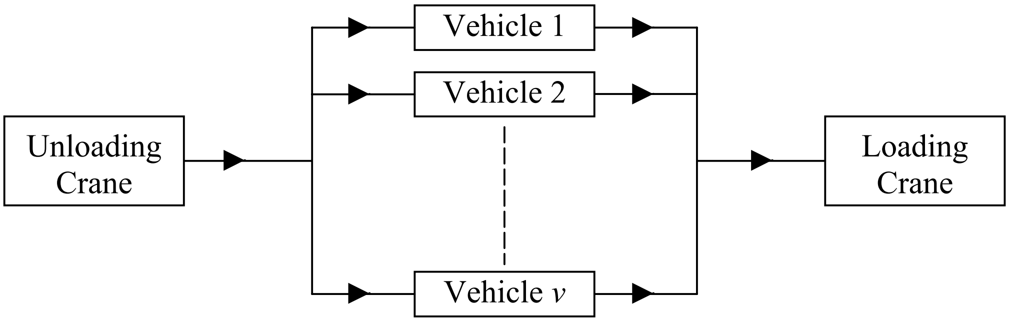



Figure 6.2 Structure of the vehicle scheduling problem with common servers

Yard crane related operational times are assumed to be deterministic and identical for all jobs as discussed in the previous problem. Thus, yard crane operation times are included in the total travel time for each route.

Hall et al. (2000) show that the parallel machine scheduling problem for two machines and a single server is NP-hard. Thus, the quay crane constraint vehicle scheduling problem is also NP-hard. Several list scheduling heuristics are proposed to solve such a problem efficiently. However, determining the appropriate heuristic depends on the objective of the problem. For instance, Bish (2003) proposed Shortest Traveling Time (STT) first rule to minimize the makespan of the unloading crane while scheduling the jobs on cranes. Figure 6.3 denotes the scheduling of routes on the vehicles and quay cranes for the example problem described in the previous case. The quay crane operational times are assumed to be identical and 1 for all routes.

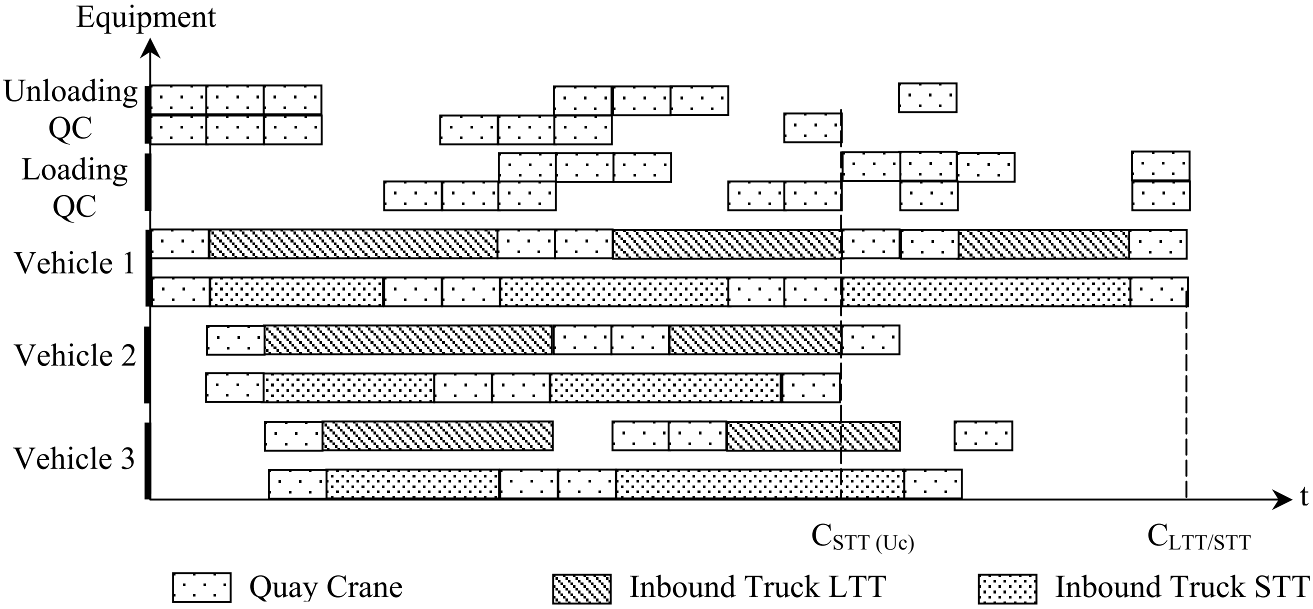



Figure 6.3 Quay crane constraint vehicle scheduling

As shown in the Gantt chart, the LTT rule generates the same makespan for the loading crane than the STT rule, whereas the completion time associated with the unloading crane is less in STT schedule.

### 6.3. Integrated Scheduling of Terminal Equipment

As the most complicated case, the integrating scheduling of terminal equipment is proposed. Terminal equipment refers to the quay crane-internal truck and yard crane system. Scheduling this combined system is more complex than the cases illustrated before. The scheduling commences from the discharging crane(s), a vehicle is dispatched to transfer container to the yard, it serves the stacking yard crane and moves to the retrieving yard crane to take the container to be uploaded, scheduling on the uploading quay crane(s) concludes the process. The structure of such a system is depicted as follows.

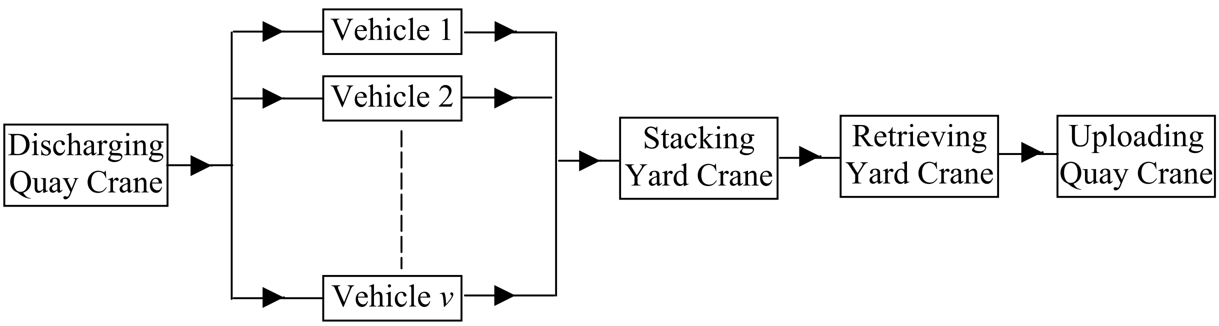



Figure 6.4 Structure of the integrated scheduling of terminal equipment

In a series of research by Meersmans and Wagelmans (2001a and 2001b), integrated scheduling of handling equipment in an automated container terminal for outbound (export) containers was proposed both in static and dynamic nature. QC-AGV-ASC system of the Rotterdam Terminal was analyzed as an integrated system, since modeling and scheduling these handling equipments separately may cause inconsistency, deadlocks and delays, and overall objective is the minimization of the makepan. The solution procedure is the beam search, which is a greedy version of the branch and bound method. Almost all of the studies regarding such systems deal with metaheuristics or simulation.

To illustrate the integrated scheduling of terminal experiment via an example, the problem defined in Example 6.1 is restructured to divide the processing times into travel times and yard crane operational times. QC loading and unloading times are identical and they are arranged same as the previous case.

Table 6.1 Processing times associated with tasks for each job

| Processing time \ Job # | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|
| Berth-Yard | 0.5 | 1 | 0.5 | 2 | 2 | 1 | 2 |
| Yard Crane (Stacking) | 0.5 | 1 | 0.5 | 0.5 | 1 | 1 | 0.5 |
| Inter Yard Travel | 1 | 0 | 0.5 | 0.5 | 1 | 0.5 | 0.5 |
| Yard Crane (Retrieval) | 0.5 | 0.5 | 0.5 | 1 | 0.5 | 0.5 | 0.5 |
| Yard-Berth | 0.5 | 0.5 | 1 | 1 | 0.5 | 1 | 0.5 |
| Total Processing Time | *3* | *3* | *3* | *5* | *5* | *4* | *4* |

The following Gantt chart denotes the schedules generated via STT and LTT rules, where the yard cranes associated with each job are different.

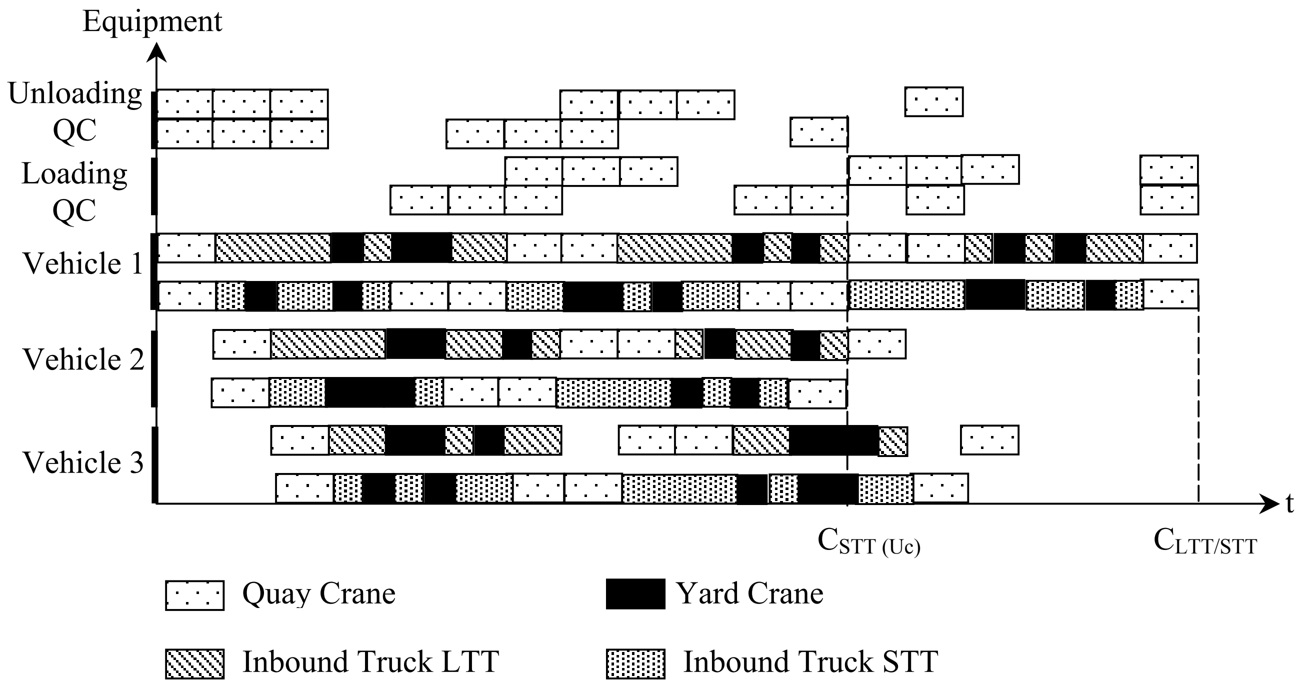



Figure 6.5 Integrated scheduling of terminal equipment

# 7. SIMULTANEOUS VEHICLE DISPATCHING WITH PRECEDENCE CONSTRAINTS

## 7.1. Problem Description

So far, we have proposed models to determine the storage locations for containers so as to find the minimum route distances for the dedicated vehicles and to schedule these jobs (routes) on the vehicles and cranes. The location matching model illustrated in Chapter 5 combines a loading location with an unloading location over a rolling horizon. This model assumes that the loading and unloading operations at the yard area or discharging and uploading operations at the quayside are simultaneously executed. Since more than one ship with multiple cranes may be berthed and operated in a time period, simultaneous operations should be employed to reach relevant results. In the recent studies of Chen et al. (1999) and Li and Vairaktarakis (2001), the vehicle dispatching problem with prespecified locations as well as the processing sequences has been analyzed. The main motivation of their studies is to match the *terminal* discharging containers with the *leading* uploading containers in order to minimize the makespan of quay cranes, where the discharging and uploading operations in a ship are separately and consequently maintained.

Bish et al. (2001) extend the problem and analyze the vehicle scheduling location problem, in which assigning a container to a location among a set of alternative locations and dispatching vehicles to containers are done at the same time. An assignment based heuristic algorithm is developed and analyzed for asymptotic and absolute worst-case performances. Bish’s most recent study focuses on combining vehicle trips so as to decrease overall handling times, which also considered as the principal motivation of location matching models on a more aggregate basis in our study.

Here, we define the problem for simultaneous loading and unloading operations, which better fits our modeling approach as indicated in prior stages of this study. In practice, there exist crane job sequences in which the containers are listed with respect to the precedence. Of course, the containers at the top level on the ship should be discharged first, and then the containers at lower levels become accessible so as to be uploaded and transferred to the yard. As mentioned in the very beginning of this study, the order of containers that are discharged via manned Quay Cranes merely depends on the intuition of the crane operator. There exist precedence relationships due to the stowage position on ship, but the crane operator chooses the container to be discharged among a set of accessible containers at a time. Thus, the order of discharging containers is flexible and there exist less precedence than a strict ordering of uploading containers. The uploading operation is hardly flexible because there exist limited storage location on the ship and several constraints so as to maintain the ship's stability. Also, an efficient stowage planning should be made to satisfy the productivity of further discharging/uploading operations. As a result, the precedence relationships between containers occur more frequently in an uploading job sequence. Therefore, the problem of vehicle dispatching for simultaneous discharging and uploading operations has to contain precedence relationships.

Let us assume that set $D$ and set $U$ denote the set of discharging and uploading containers, respectively. For the sake of simplicity, assume that both sets have the cardinality of $n$ as illustrated below.

$$D = \{d_1, d_2, ..., d_n\} \qquad U = \{u_1, u_2, ..., u_n\}$$

The elements of the discharging and uploading job sets given above are ordered with respect to the precedence of containers, i.e. $d_2$ succeeds $d_1$. Therefore, jobs should be scheduled with respect to the predetermined sequence in order to minimize the makespan of each quay crane, which leads to achieving our overall objective in terms of reducing the total turnaround times of ships. In other words, in turn the completion time of $u_n$ is the makespan that we have to minimize.

### 7.2. Nonlinear Mixed Integer Programming Model

We have formulated a nonlinear mixed integer programming model so as to minimize the makepan of simultaneous unloading and loading operations. We assume that there exist an unloading quay crane and a loading quay crane, which share a set of identical trucks. The time consumed for unloading and loading a container from / to ship is deterministic and equal to $q$ for all containers. Similarly the intertravel time between quay cranes is $b$ for all instances. The storage location of each unloaded container and the existing locations of containers to be uploaded are also known, and thus the travel times between these pairs of locations. The mathematical model illustrated for simultaneous vehicle dispatching with precedence constraints assumes that a truck loaded by a discharging quay crane goes to the final destination of the container at the storage area, and then goes to the location of an uploading container and transfers it to the loading quay crane. Thus, the routing procedure is similar to the previous cases. Each unloading container (job) has a release or ready time to be taken by a truck and each loading container has a completion time corresponding to the end of the quay crane uploading operation. The nonlinear mixed integer programming model described in this section assigns each unloaded container to an uploading container so as to minimize the completion time of the last uploading container. Let us assume that the terminal has $v = |V|$ identical vehicles where $|V|$ is the cardinality of set of vehicles $V$. The model assumes that the $(i+v)^{th}$ unloading container is taken by the truck that transfers the $j^{th}$ uploading container where the order of $i$ is equal to the order of $j$. The parameters, decision variables, objective function and constraints are listed below.

*Notation*

$D$: Set of discharging containers $\quad D = \{d_1, d_2, ..., d_n\}$

$U$: Set of uploading containers $\quad U = \{u_1, u_2, ..., u_n\}$

*Parameters*

$q_i$ : quay crane time for container $i$ $\quad q_i : q \quad \forall i$

$b_i$ : berth travel time for container $i$ $\quad b_i : b \quad \forall i$

$t_{ij}$ : travel time between locations of container $i$ and $j$

*Decision Variables*

$R_i$ : release time of $i^{th}$ discharging container

$C_j$ : completion time of $j^{th}$ uploading container

$$X_{ij} = \begin{cases} 1 \text{ if container } i \text{ is matched with container } j \\ 0 \text{ otherwise} \end{cases}$$

*Objective Function*

$$\text{Min } Z = C_{\max} \qquad (1.1)$$

$$C_{\max} = C_n \qquad n = |U| \qquad (1.2)$$

*Scheduling Constraints*

$$C_j \geq C_{j-v} + q_j \qquad \forall j \in U \qquad (1.3)$$

$$C_j \geq \sum_i (R_i + t_{ij}) X_{ij} + q_j \qquad \forall j \in U \qquad (1.4)$$

$$R_i \geq R_{i-1} + q_i \qquad \forall i \in D \qquad (1.5)$$

$$R_i \geq C_{j-v} + b_i + q_i \qquad \forall i \in D,\ i = j \qquad (1.6)$$

*Assignment Constraints*

$$\sum_j X_{ij} = 1 \qquad \forall i \in D \qquad (1.7)$$

$$\sum_i X_{ij} = 1 \qquad \forall j \in U \qquad (1.8)$$

The objective function value in Equation (1.1) denotes the makespan for the uploading crane, which is equal to the completion time of the last uploading container. The scheduling constraints in (1.3) and (1.4) guarantee that the completion time of a particular uploading container is the maximum of the completion time of preceding container plus the quay crane operational time and the total time consumed from the release of the unloading container that was assigned. Similarly, Equations (1.5) and (1.6) satisfies that the release of a particular unloading container can be done after completing the unloading of the preceding container or after completing the uploading of the container that was $v$ before in the sequence. The assignments constraints in (1.7) and (1.8) ensures the assignment of each and every unloading container to an uploading container.

## 7.3. Clustering Heuristic

In order to remove the constraints creating nonlinearity, we propose an assignment problem based solution procedure for the simultaneous vehicle dispatching problem with precedence constraints. Let us assume that we have a set of $v$ $(v = |V|)$ vehicles to transship containers between quayside and yard areas. Clustering heuristic divides the sets $D$ and $U$ into $m = \left\lceil \frac{n}{v} \right\rceil$ subsets, the first $m-1$ subsets have the cardinality of $v$ and the last subset has the cardinality of $[n - v(m-1)]$. After clustering the discharging and uploading container sets in separate groups, the matching problem is solved for $m$ times regarding each subset. Eventually, we have separately matched containers and associated traveling times for each combination (job). The mixed integer programming model and the iterative assignment problem based approach are proposed as follows:

### 7.3.1. Mixed Integer Programming Model

In addition to the sets and parameters defined for the nonlinear mixed integer programming model, the release times of discharging containers are assumed to be known since the whole set of containers are divided into clusters and the mixed integer programming (MIP) model will be solved for each cluster. Thus, the solution of MIP for a cluster reveals the release times of discharging containers for the next cluster's MIP.

*Notation*

$G$: Set of clusters $G = \{g_1, g_2, ..., g_m\}$

$D_c$: Discharging containers for cluster $c$, $c \in G$

$U_c$: Uploading containers for cluster $c$, $c \in G$

*Parameters*

$$R_i = \max\{R_{i-1} + q_i, C_{j-v} + b_i + q_i\} \qquad \forall i \in D_c,\ i = j \tag{2.1}$$

*Objective Function*

$$\text{Min } Z = C_{\max} \tag{2.2}$$

$$C_{\max} = C_n \qquad n = |U_c| \tag{2.3}$$

*Scheduling Constraints*

$$C_j \geq C_{j-v} + q_j \qquad \forall j \in U_c \tag{2.4}$$

$$C_j \geq \sum_{i \in D_c} (R_i + t_{ij}) X_{ij} + q_j \qquad \forall j \in U_c \tag{2.5}$$

*Assignment Constraints*

$$\sum_j X_{ij} = 1 \qquad \forall i \in D_c \tag{2.6}$$

$$\sum_i X_{ij} = 1 \qquad \forall j \in U_c \tag{2.7}$$

$$X_{ij} \in \{0,1\} \qquad \forall i \in D_c\ \forall j \in U_c \tag{2.8}$$

### 7.3.2. Iterative Solution Procedure

After clustering the sets of discharging and uploading containers into clusters, identical MIPs are solved for each cluster iteratively.

---

Set the iteration count, *k=1*

Set the initial parameters, $R_i = \max\left\{R_{i-1} + q_i, C_{j-v} + b_i + q_i\right\}$,

where $C_k$ is the makespan for iteration(cluster) *k*

Solve the MIP model $P_k$

If the last unloading-loading containers are matched, terminate

Otherwise, Go to step *k+1*

---

### 7.3.3. Worst-Case Analysis of the Iterative Solution Procedure

To illustrate the worst-case performance of the iterative solution procedure, we propose an extreme instance regarding the solutions. Assume that the terminal has *v* identical vehicles, and *n* discharging and uploading containers. Moreover, there exist a strict precedence relationship between both discharging and uploading containers.

$$D = \{d_1, d_2, ..., d_n\} \qquad U = \{u_1, u_2, ..., u_n\} \qquad V = \{1, 2, ..., v\}$$

As an extreme case, let us assume that the travel times between the locations of containers are defined as follows:

$$t_{ij} = \begin{cases} t_{min} & \text{for } i = \{1, 2, ..., v-1\} \wedge j = \{n-v+2, n-v+3, ..., n\} \\ t_{min} & \text{for } i = \{v, v+1, ..., n\} \wedge j = \{1, 2, ..., n-v+1\} \\ t_{max} & \text{for all other possible pairs} \end{cases}$$

$t_{min}$ : travel time for minimum possible distance at the terminal area

$t_{max}$ : travel time for maximum possible distance at the terminal area

$t_{max}$ is sufficiently large compared to the $t_{min}$ and $q$, and $b$ is assumed to be 0.

Since $t_{max}$ is sufficiently large compared to other parameters, the solution for the optimal allocation will be as follows:

$$S^* : \left\{ (i,j) : t_{ij} = t_{min} \ , \forall (i,j) \right\}$$

whereas the cluster heuristic divides the set of containers into *m* subsets and for each subset a unique assignment solution exists due to the fact that all cost coefficients are equal to $t_{max}$.

$C^H$ : Heuristic makespan $\qquad$ $C^*$ : Optimal makespan

$C^H : \frac{n}{v}(2q + \frac{n}{v} t_{max}) + (v-1)q$ $\qquad$ $C^* : 2nq + (n - v + 1)t_{\min}$

$\lim_{n \to \infty} \frac{C^H}{C^*} = \frac{1}{2v} + \frac{t_{max}}{2vq}$, where $t_{max} > 2vq$ and $t_{max} >> t_{min}$

The upper bound for the performance of cluster heuristic depends on the value of $t_{max}$, $v$ and $q$. Since $t_{max}$ is directly proportional to the terminal size, number of vehicles increases as $t_{max}$ increases for actual cases. Thus, it could be said that there exists a finite upper bound for the performance of heuristic.

# 8. CONCLUSION AND FUTURE RESEARCH DIRECTIONS

After the breakdown of trade barriers among countries the volume of international trade has grown dramatically. Since the greatest portion of the international transportation is overseas, the globalization of trade has increased the importance of issues regarding marine transportation. Containers are the standardized packages that are widely preferred to handle increasing international bulk. After the eighties, tremendous growth in the number of containers, fleet sizes, and container throughput have highlighted the significance of port terminal operations. Container terminals are the interfaces for the container flow from one mode to the other. The pressure of competition and limitation of capacity have forced container terminal managers to focus on operational efficiency.

In this study, we have focused hierarchically on a number of decisions regarding container terminal operations and proposed planning approaches to minimize the ship turnaround times, the ultimate performance criterion for a container terminal.

As the first step of our study, a storage space allocation problem is modeled and analyzed over a rolling horizon for aggregate planning. The aim of this model is to find appropriate locations for each group of containers so as to minimize the total distance traveled by the containers over the planning horizon. A network flow model is proposed and solved for two different scenarios regarding the types of containers. In the simple allocation case, it is assumed that just one type of container exists to locate. In the extended case, three types of containers such as transshipment, export and import containers have been taken into account. The parameters regarding the numerical experiments are generated arbitrarily. The numerical experiments show that the optimal allocation generates significantly better solutions than any other allocation. Although benchmarking the results of optimal and random allocation at this stage seems to be irrelevant, such a process motivates us to deal with more detailed models.

The output of the storage space allocation models constitutes an input for the next level in our hierarchy, namely location matching. In the location matching step, the locations to be seized and emptied are matched for each time period of planning horizon so as to minimize the total distance traveled by vehicles to transfer containers from ship to yard area and vice versa. The location matching phase is also considered for both simple and extended cases. After completing the numerical experiments, we show that the inputs gained through optimal space allocation result in quite efficient solutions compared to those of random allocation. It can be concluded that the surrogate objective function proposed for the storage space allocation is consistent with the ultimate objective and leads considerable reduction in terms of total distances traveled by trucks.

Vehicle scheduling is discussed as the next level of our modeling hierarchy. There exist three different cases for this step and each case is analogically similar to an NP-hard problem already proposed in the scheduling literature. The Longest Traveling Time (LTT) first heuristic provides reasonable solutions for vehicle constraint case. The performance of list heuristics that are appropriate for quay crane constraint scheduling depends on the objective that the decision maker takes into account. As a more complicated problem, the integrated scheduling of terminal equipment can be executed via several metaheuristics for a more general framework in the literature. Due to the absence of detailed terminal operating data, we have preferred discussing the problems and offering practical solution procedures rather than investigating the problems deeply.

Lastly, the simultaneous vehicle dispatching problem with precedence constraints is described and analyzed. The motivation behind such a modeling overlaps with the main motivation of the location matching problem, which is combining trips of vehicles so as to minimize empty trips. However, the models and analysis so far ignore the existence of precedence between operations, especially for the container uploading. We have proposed a cluster heuristic to solve the nonlinear mixed integer programming model iteratively.

This thesis proposes a set of problems that are interrelated sequentially. Although we analyze the model performances subject to a general container terminal framework, our models and assumptions can be extended and implemented in more specific instances. The investigation of an actual container terminal based on our modeling methodology can be a further direction for applied research, due to the additional constraints that are specific to the terminal.

The workload balancing constraints are introduced but omitted in the analysis of storage space allocation models. Analysis of these models with side constraints by Lagrangian Relaxation technique is another interesting direction to obtain better results for the storage space allocation. With such an extension, assigning limited number of yard cranes to the storage blocks over a rolling horizon may be completed.

The introduction of precedence constraints to the location matching problem expresses real life cases more accurately since both loading and unloading operations are inflexible in terms of the order of operations. We have briefly introduced such constraints in the last chapter and they can be expanded for the location matching case. This can be another area for further research, which probably may discuss more complicated problems compared to polynomial time solvable network flows.

Further research directions for container terminal operations are said to be unlimited in the complexity and number of operations that could be handled in strict time frames. Here, we simply depict the issues that can be considered as the extensions of our hierarchical modeling and analysis.

## 9. REFERENCES


1. Ahuja, R.K., Magnanti, T.L., Orlin, J.B., *Network Flows: Theory, Algorithms and Applications*, Prentice Hall, Englewood Cliffs, New Jersey, 1993.

2. Bish, E.K., *Theoretical analysis and practical algorithms for problems in a mega container terminal*, Ph.D. Dissertation, Northwestern University, Evanston, IL, 1999.

3. Bish, E.K., 'A multiple-crane-constrained scheduling problem in a container terminal', *European Journal of Operational Research*, 144, pp. 83-107, 2003.

4. Bish, E.K., Leong, T., Li, C., Ng, J.W.C., Simchi-Levi, D., 'Analysis of a new vehicle scheduling and location problem', *Naval Research Logistics*, Vol.48, No.5, pp. 363-385, 2001.

5. Canadian Transportation and Logistics, http://www.ctl.ca

6. Cao, B., Uebe, G., 'Solving transportation problems with nonlinear side constraints with Tabu Search', *Computers and Operations Research*, Vol.22, No.6, pp. 593-603, 1995.

7. Chamber of Shipping, http://www.chamber-of-shipping.org.tr

8. Chen, F.Y., Bish, E.K., Leong, Y.T., Liu, Q., Nelson, B.L., Ng J. W. C., Simchi-Levi, D., 'Dispatching vehicles in a mega container terminal', *Working Paper*, Northwestern University, 1999.

9. Cheung, R.K., Li C-L., Lin W., 'Interblock crane deployment in container terminals', *Transportation Science*, Vol.36, No.1, pp. 79-93, 2001.

10. Daganzo, C.F., 'The crane scheduling problem', *Transportation Research B*, Vol.23, No.3, pp. 159-175, 1989.

11. De Castilho, B., Daganzo, C.F., 'Handling strategies for import containers at marine terminals', *Transportation Research B*, Vol.27, No.2, pp. 151-166, 1993.

12. Edmong, E.D., Maggs, R.P., 'How useful are queue models in port investment decisions for container berths', *Journal of Operational Research Society*, Vol.29, No.8, pp. 741-750, 1978.

13. Evers, J.J.M., Koppers S.A.J., 'Automated guided traffic control at a container terminal', *Transportation Research A*, Vol.30, No.1, pp. 21-34, 1996.

14. Gambardella, L.M., Rizzoli, A.E., Zaffalon, M., 'Simulation and planning of an intermodal container terminal', *Simulation*, Vol.71, No.2, pp. 107-116, 1998.

15. Gambardella, L.M., Mastrolilli, M., Rizzoli, A.E., Zaffalon, M., 'An optimization methodology for intermodal terminal management', *Journal of Intelligent Manufacturing*, Vol.12, No.5-6, pp. 521-534, 2001.

16. GAMS® IDE, *Geometric and Algebraic Modeling System Integrated Development Environment*, http://www.gams.com

17. Hall, N.G., Potts, C.N., Sriskandarajah, C., 'Parallel machine scheduling with a common server', *Discrete Applied Mathematics*, Vol.102, No.3, pp. 223-243, 2000.

18. Imai, A, Nagaiwa, K., Tat, C.W., 'Efficient planning of berth allocation for container terminals in Asia', *Journal of Advanced Transportation*, Vol.31, No.1, pp. 75-94, 1997.

19. Informa Maritime and Transportation, http://www.informamaritime.com

20. Kim, K.H., 'Evaluation of number of re-handles in container yards', *Computers and Industrial Engineering*, Vol.32, No.4, pp. 701-711, 1997.

21. Kim, K.H., Bae, J.W., 'Re-marshalling export containers in port container terminals', *Computers and Industrial Engineering*, Vol.35, No.3-4, pp. 655-658, 1998.

22. Kim, K.H., Bae, J.W., 'A dispatching method for automated guided vehicles to minimize the delays of containership operations', *International Journal of Management Science*, Vol.5, No.1, pp. 1-25, 1999.

23. Kim, K.H., Kim, H.B., 'The optimal determination of the space requirement and the number of transfer cranes for import containers', *Computers and Industrial Engineering*, Vol.35, No.3-4, pp. 673-676, 1998.

24. Kim, K.H., Kim, H.B., 'Segregating space allocation models for container inventories in port container terminals', *International Journal of Production Economics*, Vol.59, No.1-3, pp. 415-423, 1999a.

25. Kim, K.H., Kim, H.B., 'An optimal routing algorithm for a transfer crane in port container terminals', *Transportation Science*, Vol.33, No.1, pp. 17-33, 1999b.

26. Kim, K.H., Kim, H.B., 'A routing algorithm for a single straddle carrier to load export containers onto a containership', *International Journal of Production Economics*, Vol.59, No.1-3, pp. 425-433, 1999c.

27. Kim, K.H., Kim, H.B., 'Routing straddle carriers for the loading operation of containers using a beam search algorithm', *Computers and Industrial Engineering*, Vol.36, No.1, pp. 109-136, 1999d.

28. Kim, K.H., Park, K.T., 'A note on a dynamic space-allocation method for outbound containers', *European Journal of Operational Research,* Vol.148, No.1, pp. 92-101, 2003.

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

30. Korea Maritime Institute, http://www.kmi.re.kr

31. Kozan, E., 'Increasing the operational efficiency of container terminals in Australia', *Journal of Operational Research Society*, Vol.48, No.2, pp. 151-161, 1997.

32. Kozan, E., Preston, P., 'Genetic algorithms to schedule container transfers at multimodal terminals', *International Transactions in Operational Research*, Vol.6, No.3, pp. 311-329, 1999.

33. Kozan, E., Preston, P., 'An approach to determine storage locations of containers at seaport terminals', *Computers and Operations Research*, 28, pp. 985-995, 2001.

34. Kozan, E., 'Optimising container transfers at multimodal terminals', *Mathematical and Computer Modeling*, 31, pp. 235-243, 2000.

35. Legato, P., Mazza, R.M., 'Berth planning and resources optimization at a container terminal via discrete event simulation', *European Journal of Operational Research*, 133, pp. 537-547, 2001.

36. Li, C., Vairaktarakis, G.L., 'Loading and unloading operations in container terminals', *Technical Memorandum*, No.745, Department of Operations, Weatherhead School of Management, Case Western Reserve University, 2001.

37. Liu, J., Zhang, C., Wan, Y., Linn, R., 'Dynamic crane deployment in container storage yards', *Transportation Research B*, Vol.36, No.6, pp. 537-535, 2002.

38. Meersmans, P.J.M., Wagelmans, A.P.M., 'Effective algorithms for integrated scheduling of handling equipment at automated container terminals', *ERIM Research Series Research in Management*, June 2001a.

39. Meersmans, P.J.M., Wagelmans, A.P.M., 'Dynamic scheduling of handling equipment at automated container terminals', *ERIM Research Series Research in Management*, November 2001b.

40. Meersmans, P.J.M., Dekker, R., 'Operations Research supports container handling', *Econometric Institute Report*, Erasmus University, Rotterdam, November 2001.

41. Murty, K. G., Liu, J., Wan, Y., Zhang, C., Tsang, M., Linn, R., 'DSS (Decision Support Systems) for operations in a container shipping terminal', *Proceedings of the First Gulf Conference on Decision Support Systems*, pp. 189-208, 6-8 November 2000, Kuwait.

42. Narasimhan, A., Palekar, U.S., 'Analysis and algorithms for the transtrainer routing problem in container port operations', *Transportation Science*, Vol. 36, No.1, pp. 63-78, 2002.

43. Penn, M, Avriel, M., Shpirer, N., 'Container ship stowage problem: complexity and connection to the coloring of circle graphs', *Discrete Applied Mathematics*, 103, pp. 271-279, 2000.

44. Peterkofsky, R.J., Daganzo, C.F., 'A branch and bound solution method for the crane scheduling problem', *Transportation Research B*, Vol.24, No.3, pp. 159-172, 1990.

45. Pinedo, M., Scheduling: *Theory, Algorithms and Systems*, Prentice Hall, New Jersey, 1995.

46. Shabayek, A.A., Yeung, W.W., 'A simulation model for the Kwai Chung container terminals in Hong Kong', *European Journal of Operational Research,* 140, pp. 1-11, 2002.

47. Shields, J.J., 'Container Stowage: A computer aided pre-planning system', *Marine Technology*, Vol.21, No.4, 1984.

48. Steenken, D., 'Fahrwegoptimierung am container terminal under echtzaitbedingungen', *OR Spektrum*, Vol.14, No.3, pp. 161-168, 1992.

49. Steenken, D., Henning, A., Freigang, S., Voss, S., 'Routing of straddle carriers at a container terminal with the special aspect of internal moves', *OR Spektrum,* Vol.15, No.3, pp. 167-172, 1993.

50. Taleb-Ibrahimi, M., De Castilho, B., Daganzo, C.F., 'Storage space vs. handling work in container terminals', *Transportation Research B*, 27, pp. 13-32, 1993.

51. Teo, C., Cheng, Y., Sen, H., Natarajan, K., Tan, K., 'Dispatching automated guided vehicles in a container terminal', *Working Paper*, Singapore MIT Alliance Program, November 2002.

52. Van der Meer, R., 'Operational Control of internal transport', *ERIM Ph.D. Series Research in Management*, No.1, 2000.

53. Veras, V.J., Diaz, J.S., 'Optimal pricing for priority service and space allocation in container ports', *Transportation Research B,* Vol.33, No.2, pp. 81-106, 1999.

54. Vis, I.F.A., Koster, R., de Roodbergen, K.J., Peeters, L.W.P., 'Determination of number of automated guided vehicles required at a semi-automated container terminal', *Journal of the Operational Research Society*, 52, pp. 409-417, 2001.

55. Vis, I.F.A., Koster, R., 'Transshipment of containers at a container terminal: an overview', *European Journal of Operational Research,* Vol.147, No.1, pp. 1-16, 2003.

56. Vis, I. F. A., *Container Logistics*, http://www.ikj.nl/container

57. Wilson, I.D., Roach, P.A., 'Container stowage planning: a methodology for generating computerized solutions', *Journal of the Operational Research Society*, 51, pp. 1248-1255, 2000.

58. Zhang, C., Liu, J., Wan, Y., Murty, K.G., Linn, R.J., 'Storage space allocation in container terminals', (To appear in *Transportation Research B*), 2001.

59. Zhand, L., Ye, R., Huang, S., Hsu, W., 'Mixed integer programming models for dispatching vehicles at a container terminal', *Working Paper*, National Technology University, Singapore, 2002.