Heuristic Container Placement Algorithms

By

Burak Galip ASLAN

A Dissertation Submitted to the
Graduate School in Partial Fulfillment of the
Requirements for the Degree of

MASTER OF SCIENCE

Department: Computer Engineering
Major: Computer Software

Izmir Institute of Technology
Izmir, Turkey

July, 2003

We approve the thesis of **Burak Galip ASLAN**

Date of Signature

--- 31.07.2003

Prof.Dr. Halis PÜSKÜLCÜ

Supervisor

Department of Computer Engineering

--- 31.07.2003

Assist.Prof.Dr. Tuğkan TUĞLULAR

Department of Computer Engineering

--- 31.07.2003

Assist.Prof.Dr. Mustafa Arslan ÖRNEK

Department of Mechanical Engineering

--- 31.07.2003

Prof.Dr. Sıtkı AYTAÇ

Head of Department

ACKNOWLEDGEMENTS

The author would like to express his sincere gratitude to his thesis adviser, Prof. Dr. Halis PÜSKÜLCÜ, for his encouragement, guidance and support throughout the development of this thesis.

The author also would like to express his great thanks to İsmail BERKAN on behalf of Bimar Bilgi İşlem Hizmetleri A.Ş. for their help and support during this study.

The author would especially like to thank Lütfü SÜVARİ for his support and cooperation in every single step of this study.

The author would like to extend his thanks to Oğuz TÜMİŞ for his kindness and support during the analysis phase of this study.

The author would also like to thank his colleague and friend, Mustafa Özgür TUTUM for his inspiring ideas that enlightened his studies.

The author would like to express his gratitude to his dear wife and parents for their love and patience.

ABSTRACT

With the growth of transportation over sea; defining transportation processes in a better way and finding ways to make transportation processes more effective have become one of the most important research areas of today. Especially in the last quartet of the previous decade, the computers had become much powerful tools with their impressive amount of data processing cababilites. It was imminent that computers had begun taking serious roles in the system development studies. As a result; constructing models for the processes in container terminals and processing the data with the computers create opportunities for the automation of various processes in container terminals. The final step of these studies is the full automation of terminal activities with software packages that combine various functions focused on various processes in a single system.

This study is about a project that had been made for a container terminal owned by a special company. During this study; there had been discussions with experts about the subject, and container handling processes in the terminal had been analyzed in order to define the main structure of the yard management software to be created.

This study focuses on the container handling activities over the yard space so as to create a basis for a computer system that will take part in the decisions during the container operations. Object oriented analysis and design methods are used for the definition of the system that will help the decisions in the yard operations. The optimization methodology that will be the core of the container placement decisions is based on using different placement patterns and placement algorithms for different conditions. These placement patterns and algorithms are constructed due to the container handling machinery that was being used in the terminal that this study has been made for.

“ Buluşsal (Heuristic) Konteyner Yerleştirme Algoritmaları ”

ÖZ

Deniz nakliyatının her geçen gün yaygınlaşması nedeniyle nakliyat süreçlerinin nasıl daha iyi tanımlanabileceği ve daha verimli hale getirilebileceği sorularına yönelik yanıtlar üzerindeki çalışmalar günümüzde önemli araştırma konularından biri haline gelmiştir. Özellikle geçtiğimiz yüzyılın son çeyreğinden itibaren, varolan ve yeni oluşturulan sistemlerin iyileştirme çalışmalarında, gün geçtikçe artan bilgi işleme yetenekleri ile bilgisayarların da devreye girmesi kaçınılmaz olmuştur. Bu bağlamda konteyner terminallerindeki işlemlerin modellerinin oluşturularak bilgisayar sistemleri yardımıyla verilerin işlenmesi, terminallerdeki çeşitli süreçlerin otomasyonu ve en son aşamada tüm terminal işlemlerinin bilgisayar sistemleri yardımıyla otomatik olarak yapıldığı yazılım paketleri olarak karşımıza çıkmaktadır.

Bu çalışma özel bir şirkete ait bir konteyner terminali için yapılmış olan bir projeyi kapsamaktadır. Çalışma süresince terminaldeki konteyner hareket süreçleri incelenmiş, konunun uzmanları ile görüşülmüş ve bir sonraki adımda oluşturulacak saha operasyon yazılım paketinin genel yapısı ortaya konulmaya çalışılmıştır.

Bu çalışmanın konteyner terminalindeki tüm süreçler içinde hedef aldığı nokta saha üzerindeki konteyner elleçleme işlemlerinin tanımlanması ve konteyner hareketlerine karar verme sürecinde bir bilgisayar sistemini devreye alacak yapıyı oluşturmaktır. Saha operasyon işlemlerinin karar verme sürecinde yardımcı olması beklenen sistem nesneye dönük analiz ve tasarım yöntemi kullanılarak oluşturulmaya çalışılmıştır. Sistemin konteyner yerleştirme ve karar verme sürecinde temel alacağı optimizasyon yöntemi ise farklı durumlar için farklı yerleştirme desenlerini ve algoritmalarını kullanacak şekilde oluşturulmuştur. Bu yerleştirme desenleri ve algoritmaları projenin yürütüldüğü terminaldeki istif sahasında kullanılan konteyner elleçleme makinelerinin çalışma prensipleri gözönüne alınarak oluşturulmuştur.

TABLE OF CONTENTS

Page

Page

APPENDICES

LIST OF TABLES

LIST OF FIGURES

Page

CHAPTER I

INTRODUCTION

Containers are large boxes used to transport goods from one destination to another. With containers, a bulk unit can be created out of the individual pieces of freight. As a result, containerization can be defined, according to The Containerization Institute, as the utilization, grouping or consolidating of multiple units into a larger container for more efficient movement. Compared to conventional bulk, the use of containers has several advantages, namely less product packaging, less damaging and higher productivity [Agerschou, 1].

Containers were used for the first time in the mid-fifties. Through the years, the proportion of cargo handled with containers has steadily increased. As a result of the enormous growth, the capacity of ships has been extended from 400 TEUs to 4000 TEUs and more [Rath, 2].

The dimensions of containers have been standardized. The term TEU (twenty-feet-equivalent-unit) is used to refer to one container with a length of twenty feet. A container of 40 feet is expressed by 2 TEUs. Several transportation systems can be used to transport containers from one destination to another. Transport over sea is carried out by ships while trucks and trains can be used to transport containers over land. To transfer containers from one mode of transportation to another, ports and terminals can be used. For example a container can be taken off a truck and placed on a ship [Vis, 3].

1.1 Container Terminals

Container terminals are the places where one sometimes has to make hundreds of decisions in a few hours. When there is no computer assistance available experienced captains and experienced terminal operators are the people who are responsible for the operation decisions over the terminal space. These decisions are mostly based on previous experiences of experts and financial expectations of the terminal managers. Each decision means resource allocation, operation time, and cost that have direct effect on customer service quality. Because of the huge number of movements being done for 24 hours a day, and the running costs of the heavy machinery that is used for container operations, even a little efficiency increase in a single operation can result in huge benefits per year considering the endless repetition of the tasks. The more efficient movements in a terminal, the more competent the terminal is. Because, the final costs that are related to the customers, are strictly dependant on the operational costs of the terminal tasks.

Another alternative is that instead of putting the decision making pressure on people, one can build a decision support system that is based on optimizing the port activities and resources with the help of optimization algorithms. While the proper placement of containers is influenced by many factors, an experienced person can make good decisions when there are a lot of suitable storage alternatives on the yard and low container traffic. The need for a decision support system arises especially when there are yard space limitations and heavy container traffic. The major ports are forced to use decision support systems, because the conditions and alternatives become far too complex for the people to make suitable decisions in a short time.

With the introduction of larger ships, small terminals have grown into large terminals. To ensure a fast transshipment process at large terminals information technology and automated control technology can be used [Johansen, 4]. To use these kinds of technologies large investments have to be made and ongoing database management is required. The application of information technology can result in more efficiency and a higher performance [Wan, 5]. In order to achieve an improvement of productivity and reduction in investment costs, an advanced automated control technology is a necessary condition [Leeper, 6].

Using efficient optimization algorithms can lead to the automation of container terminal activities. However in this case, the decision support system should not work only for the container placement operations, but for the resource allocation and customer services, too. This study is about the container placement operations in a container terminal.

1.2 Armaport

The study was especially made for a private container terminal, Armaport, and all the analyses and design studies will be used in developing the decision support system for Armaport and other private ports operated by MARPORT Liman İşletmeleri ve Ticaret Sanayi A.Ş.[1] .

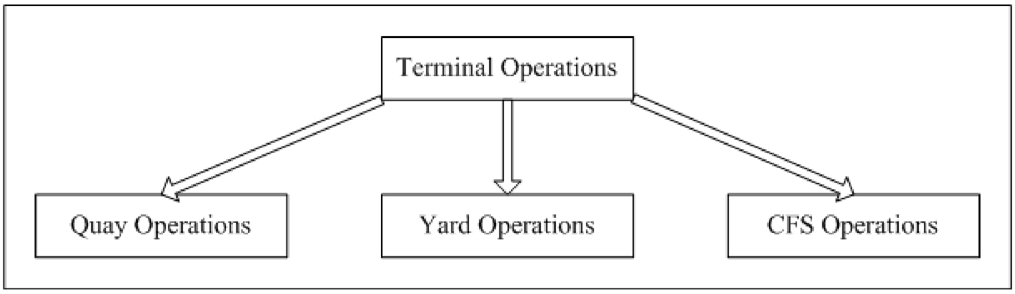


Fig. 1.1. The overview of terminal operations in Armaport

As seen in Fig. 1.1, there are mainly three areas for container operations in Armaport. The quay operations are related with the loading/unloading of the ship, and the container freight station (CFS) operations are related with loading/unloading of containers. The yard operations are related with the distribution and placement of container over the yard space, which is the focus of this study. All these operations are made in different areas of the terminal, so the terminal consists of three main areas, namely quay, yard and CFS area.

1.3 Optimization Criteria

The managers of Armaport focus on two main criteria for the terminal operations. These are the minimization of time spent for the loading and unloading of the berthing ships, and the minimization of the unnecessary shifting operations that is done during the container withdrawal stage following the customer requests. The minimization of time spent for the berthing ships is also related to the quay operations of the terminal and it is out of the scope of this study. However, the storage of the containers is handled over the yard space, and

---

[1] MARPORT Liman İşletmeleri Sanayi ve Ticaret A.Ş. is a subsidiary company of ARKAS Holding.

minimization of the unnecessary shifting operations that is done during the container withdrawal stage is directly related to our study.

One of the most critical problem domains in the terminal is the management of the yard space where the containers are stored. The problem is based on trying to find an answer to the question of how to implement a decision support system that uses a 3D placement optimization algorithm to store containers over the yard space in a commercial port.

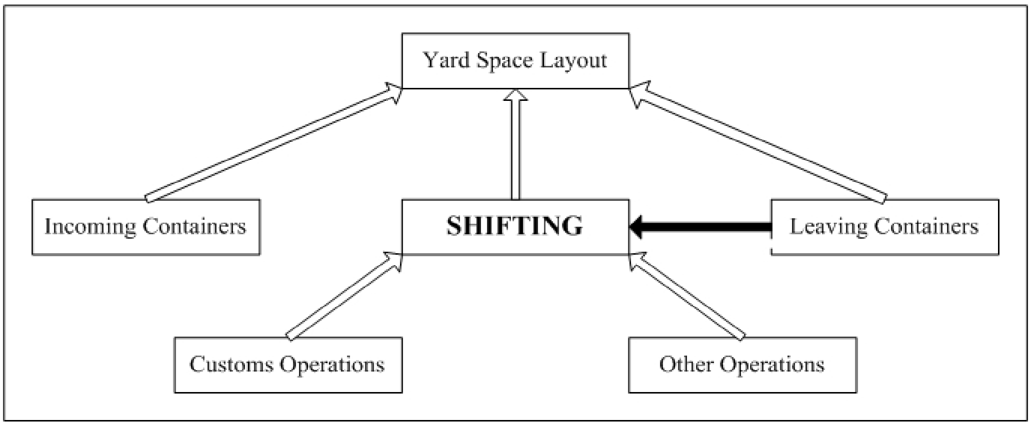


Fig. 1.2. The container operations affecting the yard space layout

As shown in Fig. 1.2, there are mainly three operations which cause shifting in the yard space. The customs operations and other unavoidable operations may directly cause in shifting of containers because they have external sources of our system, and we can not avoid them. The main criterion of our optimization procedure is the proper placement of containers over the yard space at the first time in order to minimize the number of unnecessary shifting during container withdrawal stage.

Our study is about the analysis and design of a software package which uses an optimization algorithm to find suitable places for the incoming containers and handle the shifting operations that may occur during the container withdrawal. Resource allocation, ship stowage, hardware installation and database design are out of the scope of this study.

CHAPTER II

RELATED WORK

Before we proceed further into to the previous studies on container terminal operations, it may be useful to define the processes and problems of the container terminals that the studies are focused on.

2.1 The Processes at Container Terminals

The process of unloading and loading a ship at a container terminal can be divided in to basic operations for better understanding as depicted in Figure 2.1.

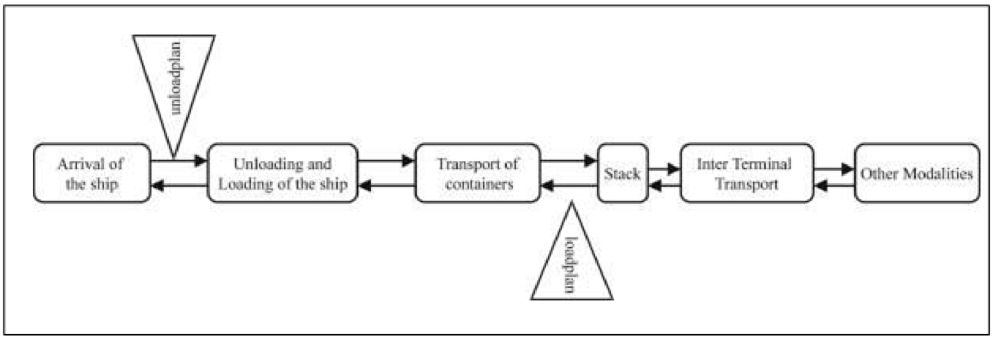


Fig. 2.1. Processes at a container terminal

When a ship arrives at the port, the containers have to be taken off the ship. This is done by manned quay cranes, which take the containers from the ship's hold and the deck. Next, the quay cranes put the containers on vehicles, like automated guided vehicles or manned vehicles. After receiving a container, the vehicle moves to the stack. This stack consists of a number of lines where containers can be stored for a certain period. These lines are served by automatically controlled automated stacking cranes or manned cranes. When a vehicle arrives at a line, the stacking crane takes the container off the vehicle and stores it in the stack. (Several terminal equipment figures can be seen at Appendix C) After a certain period the containers are retrieved from the stack by the stacking cranes and transported by the vehicles to other transportation modes such as deep-sea ships, trucks or trains. This process is also executed in reverse order, to load containers on a ship [Vis, 3].

2.1.1 Unloading and Loading of the Ship

Automated and manned terminals both use quay cranes. Quay cranes are manned because automation of this process encounters practical problems, like exact positioning of containers. The quay cranes 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 pick up device attached to the trolley, picks the containers. The quay cranes move on rails to the different holds to take/put containers off/on the deck and holds. It can occur that at the same moment one quay crane is unloading containers while another quay crane is loading containers.

The number of import containers that has to be unloaded at the terminal is in practice usually known shortly before the arrival of the ship. At the operational level an unloading and loading plan have to be made [Vis, 3].

2.1.2 Container Transport from Ship-to-Stack and Stack-to-Ship

For the transport of a container at a manned terminal, vehicles like forklift trucks, reach-stackers, yard trucks or straddle carriers can be used. Straddle carriers, reach-stackers, and forklift trucks can pick up containers from the ground. A crane is needed to put the container on the yard truck. For the transport of multiple containers, multi-trailer systems can be used.

At an automated container terminal automated guided vehicles are used for the internal transport. Automated guided vehicles are robotic vehicles which travel along a predefined path. The road system consists of electrical wires in the ground, or a grid of transponders, that control accurately the position of the vehicle. Currently, an automated guided vehicle can carry only one 20 feet or 40 feet container. Automated guided vehicles are only practical in ports with high labor costs because of the high initial capital costs. In ports with low labor costs, the system of manned vehicles is preferred [Vis, 3].

2.1.3 Stacking Containers

There are two ways of storing containers. Storing on a chassis and stacking on the ground. With a chassis system each container is individually accessible. With stacking on the ground containers can be piled up, which means that not every container is directly accessible. As a consequence of limited storage space, stacking on the ground is the most common way of storing containers.

The stack is the place where import and export containers can be stored for a certain period. The stack is divided into multiple blocks/lines, each consisting of a number of rows. The height of stacking varies per terminal between two and eight containers high. At the end of each lane a transfer point might be situated. At this point the crane takes/places the container off/on the vehicle that transports the container. Empty containers are usually stored separately.

Most of the described terminal operations have their origin and destination at the stack, for example the transport of containers from the stack to the ship and vice versa. The process of storing and retrieving containers should be executed such that the remaining operations in the terminal can be carried out effectively [Vis, 3].

2.1.4 Inter Terminal Transport and Other Modes of Transportation

Containers have to be transported from the stack to other modes of transportation, like barges, rail and road. It is expected that, with the growth of terminals in the future, this inter terminal transport will become more and more important. New concepts and technologies have to be developed to handle the large numbers of inter terminal container transports expected in the future. Furthermore, research has to be done to the various transport systems by which containers can be transported between the terminals [Van Horssen, 7].

Multi-trailer systems and automated guided vehicles can carry out this inter terminal transport. In certain terminals it is possible that containers are put directly on such as trains without using transport vehicles. For example, one way of transporting containers to other destinations is by rail while another way of transporting containers to other destinations is on the road by trucks [Vis, 3].

2.2 Decision Problems at Container Terminals

We can distinguish between three planning and control levels in making decisions to obtain an efficient terminal: the strategic level, the tactical level, and the operational level for problem domains in container terminals. For example, at the strategic level it is decided which layout, material handling equipment and ways of operation are used. The time-horizon of decisions at this level covers one to several years. These decisions lead to the definition of a set of constraints under which the decisions at the tactical and operational level have to be made [Vis, 3].

2.2.1 Arrival of the Ship

At the strategic level, when a ship arrives at the port, it has to moor at the quay. For this purpose, a number of berths are available. The number of berths that should be available at the quay is one of the decisions that have to be made at the strategic level [Vis, 3].

There have been studies on analyzing the data from several container terminals to acquire the terminal parameters in order to calculate the optimum terminal parameters for the capacity of a new terminal considering the future changes [Yenel, 8], focusing on the bottlenecks for berth and yard allocation in order to determine the physical plan of the terminal for future [Mit, 9].

There are also studies about the the effect of ship sizes, quay length, the number loading/unloading vehicles and their capacities from the economic aspects [Güler, 10], and optimizing the building costs for the design of a container terminal to achieve an optimum terminal size for a specific trading post [Bakalım, 11; Yadipour, 12].

At the operational level, the allocation of a berth to the ship has to be decided on. There are studies on how to allocate berths to ships while optimizing the berth utilization. On one hand optimal berth allocation can be obtained by minimizing the sum of port staying times. This leads to ships mooring at the quay according to the first come first served principle. On the other hand berths can be allocated, without consideration of ship's arrival order, by allocating ships at a berth closest by the area in the stack in which most containers for this specific ship are located.

As a result, the resulting terminal utilization will be maximal, but ship owners will be dissatisfied by the long waiting times of the ships. Consequently, a trade-off exists between the total staying time in the port and the dissatisfaction of ship owners caused by the order in which ships are berthed. The berth allocation problem could be considered as a machine scheduling problem [Imai, 13]. Based on a queuing network model of the logistics activities related to the arrival, berthing, and departure processes of vessels at a container terminal, a process view approach to the simulation of logistics activities related to a vessel's arrival and departure has shown to be an effective method to develop a practical tool for berthing decisions [Legato, 14].

2.2.2 Unloading and Loading of the Ship

At the strategic level, one of the questions arising is the determination of the type of material handling equipments which will be used for the unloading and the loading of containers from the ship. Automated and manned terminals both use quay cranes.

At the tactical level, the number of quay cranes have to be determined that work simultaneously on one ship. One of the objectives is to minimize the staying time of ships at the terminal. The most general case of the crane scheduling problem is the case in which ships arrive at different times in the port and queue for berthing space if the berths are full. The objective in this case is to serve all the ships while minimizing the total delay of the ships.

At the operational level, the unloading plan indicates which containers should be unloaded and in which hold they are situated in the ship. Successively, these containers are unloaded. In a hold the crane driver is almost free to determine the order in which the containers are unloaded. The unloading time of a container depends on its place in the ship. In contrast with the unloading process, there is hardly flexibility in the loading process. A good distribution of containers over the ship is necessary. The stowage planning is made at the operational level. A stowage plan indicates for each container the exact place in the ship. Containers with the same destination, category, weight, size, contents and so on, belong to the same category. Sometimes, only for each category the positions in the ship are given. Locations of containers belonging to the same category can be exchanged between containers of this category. In making the stowage planning attention should be paid to the order in which containers need to be unloaded. Unnecessary moves should be avoided by placing containers designated for a terminal visited later during the journey on top of containers designated for the terminals visited earlier [Vis, 3].

In near future, container ports will no longer be able to expand into surrounding land and will thus be unable to meet the storage requirements due to the boom in the world trade. A solution to this problem is to increase the container throughput of the port by reducing the amount of time necessary to load and unload a ship. A distributed multi-agent architecture for dockyard operations which is based upon elements from both centralized and decentralized strategies can provide a feasible optimization to the problem due to its inherent complexity [Thurston, 15].

The problems of resource allocation and scheduling of loading and unloading operations in a container terminal can be solved hierarchically by using two different but strictly interconnected modules. The resource allocation module remarkably reduces the number of resources typically required by the terminal. The optimized plans tend to exploit the resources to their limit, so that only an effective scheduler can cope with the increased complexity of this task. The scheduler achieves such a result by reducing the conflicts of the yard cranes, allowing their throughput to be dramatically increased. The complexity of the scheduling problem requires local-search techniques and taboo search. The local search techniques based on a set of neighborhood solutions can be computed very efficiently [Gambardella, 16].

2.2.3 Container Transport from Ship-to-Stack and Stack-to-Ship

At the strategic level, one of the decisions about the design of a container terminal concerns the type of material handling equipment that takes care of the transport of containers. After the decision about which system will be used has been made, one of the problems at the tactical level that has to be solved is the determination of the necessary number of transport vehicles to transport all containers in time. At the operational level it should be decided which vehicle transports which container and which route is chosen. Objectives are, for example, to minimize empty-travel distances, to minimize the delay of the ship or to minimize the total travel time of the vehicles [Vis, 3].

In the last four decades the container, as an essential part of a unit-load-concept, has scored a great success in international sea freight transportation. With increasing containerization the number of seaport container terminals and competition has increased considerably. One of the success factors of a terminal is related to the time in port for container vessels and the transshipment rates the ship operators have to pay.

Reducing the time in port for container vessels has become one of the key elements for operators of the main container terminals worldwide. Following this theme, one of the most crucial problems arising within the development of intercontinental container transport is improving the productivity of the gantry cranes in order to gain a reduction of the time in port. Preliminary results show that such problems may be treated mainly in two ways. First, a careful reorganization or reengineering of the processes and strategies on the terminal may result in considerable improvements, and second, algorithmic enhancements for certain combinatorial optimization problems may help to support the planer and allow for further improvements. A dynamic strategy may help to improve the use of gantry cranes and straddle carriers and their interplay when loading and unloading container vessels that is based on the effects on the berth time derived from the reorganization of the transshipment process. Second, simple evolutionary algorithms may successfully be applied to respective real-world problems. The use of a genetic algorithm combined with the reorganization for a minimization of the container vessel processing time shows good performance. It has been shown reasonable improvements for the gantry crane productivity are obtainable. Major improvements are due to the reorganization even simple genetic algorithms are able to gain additional improvements [Böse, 17].

A dynamic model can also be used for optimizing the flows of flatcars that considers explicitly the broad range of complex constraints that govern the assignment of trailers and containers to a flatcar. The problem can be formulated as a logistics queueing network which can handle a wide range of equipment types and complex operating rules. The complexity of the problem prevents a practical implementation of a global network optimization model. The logistics queueing network model has room for considering a range of real-world details, returns integer solutions, and can be applied in a real-time environment [Powell, 18].

High-density, automated container terminals are currently considered as a candidate to improve the performance of container terminals. The overhead grid rail offers the advantages of high storage density, fast loading/unloading, flexibility and reliability and no interference between manual and automated operations. Moreover, contrary to other automated container concepts, the simplicity of grid rail operations makes the development of optimal or nearly optimal dispatching algorithms possible. Designing the operations within the grid rail units, minimum number of shuttles are used to serve the ship and gate buffers. The dispatching algorithm synchronizes the motion of the shuttles in order to serve the buffers of the grid rail units, minimize delays and maximize throughput. Simulations demonstrate that the grid rail unit concept can be used in container terminals to improve productivity and reduce cost while utilizing much less land than conventional container terminals [Kosmatopoulos, 19].

2.2.4 Stacking Containers

A decision at the strategic level that has to be made is choosing the type of material handling equipment that will take care of the storage and retrieval of containers in and from the stack. The efficiency of stacking depends among other things on the stack height and strategies for storage and retrieval planning of import and export containers. Consequences of higher stacking are a higher number of reshuffles/shifting. To reach a specific container it can be necessary to shift containers that are placed on top of the demanded container. To minimize delay by removing containers, reshuffling of the stack can be done in advance. On the other hand, the higher the stacking the less ground space is needed for the same number of containers. Obviously, one of the problems at the strategic level is to determine a good stack layout. At the tactical level the number of transfer cranes has to be determined necessary to ensure an efficient storage and retrieval process. If straddle carriers take care of the storage and retrieval of containers from the stack, it has to be decided at the operational level how to route straddle carriers through the stack.

Another typical problem for a container terminal is that containers have to be stored and retrieved at two sides of the stack, namely seaside (to/from the ship) and landside (to/from other modalities). This can be done by the same yard crane/automated stacking crane. Some of the decisions that have to be made to ensure an efficient process are: which side has the highest priority (commonly seaside) and how long containers can wait before they are stored or retrieved. The problem to decide which automated stacking crane carries out which job, can be examined in two ways. If every container is treated as an individual (Quay crane asks for a specific container from the stack), then it is clear which automated stacking crane should carry out the job. However, one can also distinguish container categories in a stack. This holds especially for empty containers. For example, containers with the same destination, the same weight, contents and size belong to the same category.

Operational questions concerning storage planning are, for example, where an incoming container is stored, in which order containers are stored, when a container is repositioned and in which way and which crane handles which container. For retrieval planning it has to be decided in which order containers are retrieved and which crane handles the request [Vis, 3].

Considering the configuration of the container stack and the weight distribution of containers in the yard-bay, a dynamic programming model can be formulated to determine the storage location to minimize the number of relocation movements expected for the loading operation. The decision tree developed from the set of the optimal solutions supports real time decisions. The performance of the decision tree can be evaluated by the number of decisions that are different from the optimal solutions of the slower dynamic programming solution method. The dynamic programming model is formulated to determine the storage location of an arriving export container. The objective is to minimize the number of relocation movements that occur during the loading operations of a containership. The relocation movements occur when lighter containers are stacked on top of heavier containers in a yard, since the heavier ones are usually loaded first to the ship. The classification procedure is also developed to obtain a decision tree for making real-time decisions. The classification procedure includes the selection criteria of the key attribute for branching, a pruning rule, and a simplification method. Although it can be assumed that heavier containers are always loaded before lighter ones, it is often possible to avoid relocation movements by changing the sequence during loading operations [Kim, 20].

The storage space allocation problem in the storage yards of terminals is related to all the resources in terminal operations, including quay cranes, yard cranes, storage space, and internal trucks. The problem can be solved by using a rolling horizon approach. For each planning horizon, the problem is decomposed into two levels and each level is formulated as a mathematical programming model. A complex situation is considered in which inbound and outbound containers are mixed in the storage blocks in the yard. At the first level, the total number of containers to be placed in each storage block in each time period is determined to balance the workloads among blocks in each period. The second level determines the number of containers associated with each vessel that constitutes the total number of containers in each block in each period, in order to minimize the total distance to transport the containers between their storage blocks and the vessel berthing locations [Zhang, 21].

2.3 Complete Container Terminal Studies

It has been researched that after 50000 TEUs per year a terminal requires an information system for management. A container terminal planning model is suggested for a container terminal which is handling at least 50000 TEUs per year, which is based on a multi-agent system which consists of four global agents (ship, berth, yard, and gate) and three utility agents (crane, transtainer, and transport). The system provides dynamic yard allocation, dynamic berth allocation and reduces the idle time of transport vehicles [Henesey, 22].

System architecture can be based upon the multi-agent system paradigm for solving complex problems. The architecture is applied to solve the port container terminal management problem, and specifically to solve the automatic container allocation. Five agents (ship, stevedore, service, transtainer, gate) are used in the architecture to solve the automatic allocation problem. The architecture provides a maintenance of the necessary co-operation in order to minimize the time the ships are in the container terminal [Rebollo, 23].

The variety of decisions to be made in the course of daily operations at a container shipping terminal are inter-related. The ultimate goal is to make all these decisions in order to minimize time taken to process the vessels, minimize the resources used to handle the workload, minimize the wait time of customer trucks, minimize the congestion on the roads inside the terminal, and finally to make the best possible use of the storage space available. Given the scale and complexity of these decisions, to further enhance the operational efficiency of container shipping terminals, it is essential to bring in decision support tools [Murty, 24].

Another decision support system for improving the management of intermodal container terminals is implemented as a modular architecture which integrates a forecasting model, a planner and a simulation module [Bontempi, 25]. While the forecasting module estimates container traffic, the planning module uses this information to generate efficient policies for storage, resource allocation and scheduling. The performance of management policies is assessed via computer simulation. Genetic algorithms, taboo search, and dynamic programming techniques are used to implement management policies. An intermodal terminal is a complex dynamic system characterised by an high level of uncertainty and non-stationarity. Thus, a unique model representation is not able to properly describe the reality and does not adequately support the definition of efficient control policies. A problem decomposition on different time scales which allows for an easier description and the adoption of different formalisms at the different problem levels. The terminal activities can be managed with success only if we split the whole problem on different time horizons. An integrated approach based on optimisation techniques and simulation methods may lead to a decision support system which is sufficiently robust to be effective in a non stationary environment [Bontempi, 25].

2.4 Off-the-Shelf Container Terminal Management Tools

New generations of management software for marine terminals will blend artificial intelligence with graphics to optimize operations at each terminal without requiring the creation of a single life of terminal-specific code [Dougherty, 26]. The key enabling technology is the use of rule-based expert systems. An expert system is a branch of artificial intelligence that captures in computer usable form the knowledge of human experts skilled in specific tasks. The various expert system applications embedded in the software will allow local terminal management to fine tune the operating rules to make the best use of the assets and other characteristics of a specific terminal. In this way, each terminal has its own personality reflective of the local management style while the software remains identical from installation to installation, greatly minimizing development time and software maintenance costs. By swapping or modifying the rule sets used by the expert systems, the terminal will be able to adapt to changing conditions.

While the expert systems work in the background to provide advice on running the terminal efficiently, graphical tools will provide an easy means for the users to set up, modify, and monitor the terminal inventory. Using graphical methods, the user will be able to install the software, set up the expert system, rules, and define the physical and logical layout of the terminal. For example, the user would employ drag and drop methods to define and position parking areas for containers or trailers, stacked areas for containers, gate facilities, ship berths, etc. Although it will appear to the user to be a simple graphical layout tool, the system configuration expert module will actually configure the database and other portions of the system. The physical layout of the system will be automatically mapped to the data structures required to implement the business rules and maintain the inventory of the software. Another important graphical module within the new generation terminal operating system is a visualization system that offers a graphical view of the inventory in a very useful and intuitive manner. The visualization system provides a straightforward form of virtual reality, allowing the user to walk through the terminal via graphic display.

With its expert systems and graphical tools, the new intelligent terminal systems will provide users with the ability to improve productivity and service quality, improve the ability to track and control inventory, promote more efficient terminal management, and provide high level of reliability, fault tolerance, scalability, and ease of use [Dougherty, 26].

The software companies which develop terminal management software usually combine multi-functional terminal operation modules into centralized software packages. The order management, gate control, container logistics, reports, vessel planning, transporter control, berth planning, signaling and yard planning modules all together forms terminal management software package that uses a central database for all of the operations. Some well-known off-the-shelf container management tools are; Container Terminal Control System of Cosmos (Belgium); Sparcs, PowerStow, and Express of Navis (USA); TrakTainer of Trak Systems (Australia), Smart Port e-Commerce Solutions of The Open Consultancy Network Ltd. (UK); Mach Planning of Ports & Cargo CMC Limited (India).

CHAPTER III

ANALYSIS AND DESIGN

In this chapter, we try to find a methodology for the implementation of a decision support system that uses the 3D placement optimization algorithm developed in this study to store containers over the yard space in a container terminal. The first step is the analysis and design of the yard management system for the container operations.

3.1 Object Oriented Analysis and Design of the Container Terminal

The essence of object oriented development is the identification and organization of application domain concepts, rather than their final representation in a programming language whether it is object oriented or not. Object oriented development is a conceptual process independent of a programming language until the final stages. Focusing on implementation issues too early restricts design choices and often leads to inferior product. Design flaws that surface during implementation are more costly to fix than those that are found earlier. An object oriented development approach encourages software developers to work and think in terms of application domain through the software engineering life cycle [Rumbaugh, 27].

Considering the multiple functions that are expected from the software, object oriented analysis and design approach is used to make the system more suitable for future changes that may occur in the terminal operation business. The yard optimization software acts as a decision support system for the container placement operations over the yard area. The software is planned to be used by a single person called "operator". The users with different authorities are out of the scope of the study. The yard optimization software will accept the operator requests, and return operation suggestions to the operator. It was planned to keep a separate database for the optimization software due to network traffic. Because of the continuous container operations over the yard space, keeping the optimization database and central database up-to-date with periodic replication could reduce the load on the network instead of accessing the central database for every single operation request; however the database design is out of the scope of the study. When a container is processed by the system and the output is generated, it is assumed that the operation command is submitted to the machinery on the yard. If a problem occurs during yard operations, the operator accesses the system to edit the data for correcting the error. Unified modeling language (UML) is used for object oriented analysis and design of the yard optimization software. The strongest reason for using UML is that it has become a de facto standard for object oriented modeling. If it is necessary to involve a team of developers or to convey to the information in models to other people, UML is the obvious choice as it will facilitate communication among participants [Bennett, 28].

3.2 The Use-Case Diagram

When investigating an organization's requirements for a new information system, we can use several fact finding techniques like interviews, observation etc. These are used to gain an understanding of the current system and its operation, of the enhancements the user require to the current system and of the new requirements that users have for the new system. Using agreed standards to document the requirements allow us to communicate these requirements to other professionals and to the users. Use-case diagrams are one diagramming technique that is used to summarize the users' functional requirements in a high level overview of the way that new system will be used [Bennett, 29]. So, the first step taken in this study was the creation of the use-case diagram to focus on what the user behaviors are in the system as shown on Fig. 3.1.

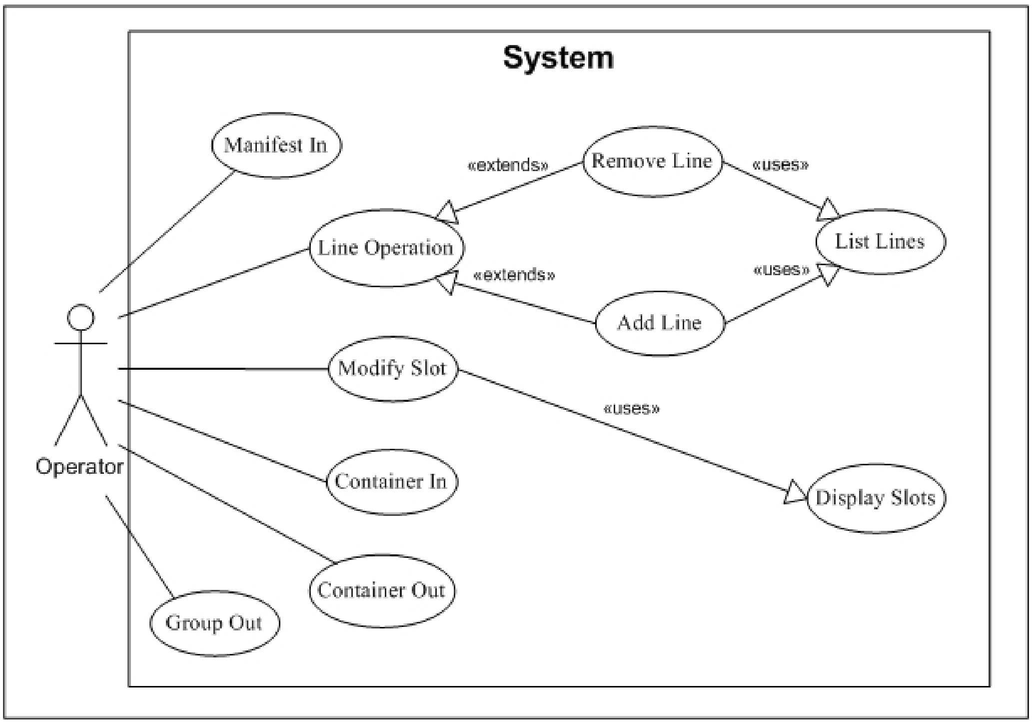


Fig. 3.1. The use-case diagram

3.2.1 The Use-Case Definitions

Use-cases are supported by behavior specifications. These specify the behavior of each use-case either using UML diagrams, such as collaboration diagrams or sequence diagrams, or in text form as use-case descriptions. Textual use-case descriptions provide a description of the interaction between the users of the system, termed actors, and the high level functions within the system, the use-cases [Bennett, 29]. These use-case descriptions can be in summary form, so the corresponding use case definitions are created as follows:

Operator: The operator is the user of the optimization software. He has access to the graphical user interface and creates inputs for the program to perform various tasks over the yard.

Manifest In: When a ship is known to disembark containers to the port, the operator submits the manifest of the ship to the system. The manifest of the ship includes the group of containers to be disembarked. The system returns possible options for this incoming scenario.

Container In: When a container handled by a stacker is to be placed, a worker on the yard signals the operator, or the worker directly signals the system so that the container info is submitted to the optimization routine to acquire a suitable slot for placement

Container Out: When a container must be taken out without any customer request, the operator submits the task to the optimization routine so that the system returns a procedure to carry out the desired task.

Group Out: When a group of containers for a customer must be taken out, the operator submits the task to the optimization routine.

Remove Line: When a line is to be removed from the yard, the operator submits it to the system, so the optimization routine prepares a suitable procedure for the line removal.

Add Line: When there is a place reserved for a line on the yard, the operator creates and adds the line into the system, so that the optimization routine can include this new line as a placement option.

Modify Slot: When any placement slot on the yard must be modified, the operator accesses the data of that slot and makes the desired changes.

3.2.2 The Use-Case Breakdowns

These use-case descriptions can be represented in a more detailed form in which the interaction between the actor and use-case is described in a step-by-step way, so the corresponding use-case breakdowns are detailed as follows:

Manifest In:

1. The user enters the name of the ship. (Actor)
2. The system prints “Busy…” to the user screen. (System)
3. The system prints reservation blocks to the screen. (System)
4. The operator confirms the reservations. (Actor)
5. The system reserves the slots. (System)
6. The system refreshes the screen. (System)

Container In:

1. The operator enters the container ID. (Actor)
2. The system prints “Busy…” to the user screen. (System)
3. The system returns a coordinate to the screen. (System)
4. The pointer confirms the coordinate. (Actor)
5. The system reserves the slot. (System)
6. The system refreshes the screen. (System)

Container Out:

1. The operator enters the container ID. (Actor)
2. The system prints “Busy…” to the user screen. (System)
3. The system prints the container ID and destination coordinate to the screen (System)
4. The operator confirms the action. (Actor)
5. The system reserves the slot. (System)
6. The system checks whether more actions present and repeats the steps 3-5. (System)
7. The system refreshes the screen. (System)

Group Out:

1. The operator enters the group number. (Actor)
2. The system prints “Busy…” to the user screen. (System)
3. The system prints the container IDs and destination coordinates to the screen. (System)
4. The operator confirms the actions. (Actor)
5. The system reserves the slots. (System)
6. The system refreshes the screen. (System)

Remove Line:

1. The system displays the list of lines. (System)
2. The operator chooses a line and confirms the selection. (Actor)
3. The system records the change. (System)
4. The system refreshes the screen. (System)

Add Line:

1. The user enters the line ID, line types and line dimensions. (Actor)
2. The system records the change. (System)
3. The system refreshes the screen. (System)

Modify Slot:

1. The operator enters the destination coordinate. (Actor)
2. The system displays the slot info. (System)
3. The operator edits the slot info and submits. (Actor)
4. The system records the change. (System)
5. The system refreshes the screen. (System)

3.3 The Class Diagram

To move from an initial use-case diagram ultimately to the implementation of software that adequately fulfills the requirements identified by the use-case involves at least one iteration through all of the development activities, from requirements modeling to implementation.

Entity classes are used to model the information and associated behavior of some phenomenon or concept such as an individual, a real-life object, or a real-life event. As a general rule, entity classes represent something within the application domain, but external to software system, about which the system must store some information. Instances of an entity class will often require persistent storage of information about the things that they present. Entity classes often represent the more permanent aspects of an application domain. Through successive iterations, the class diagram provides a high-level basis for system architecture, and low-level basis for the allocation of data and behavior to individual classes and object instances, and ultimately for the design of the program code that implements the system. Given the iterative nature of the object oriented approach, it is not essential to get this right on first attempt [Bennett, 29]. The yard optimization system class diagram is constructed for the basic container operations as depicted in Fig. 3.2. The class diagram consists of the main entity classes, their multiplicities and relationships with each other.

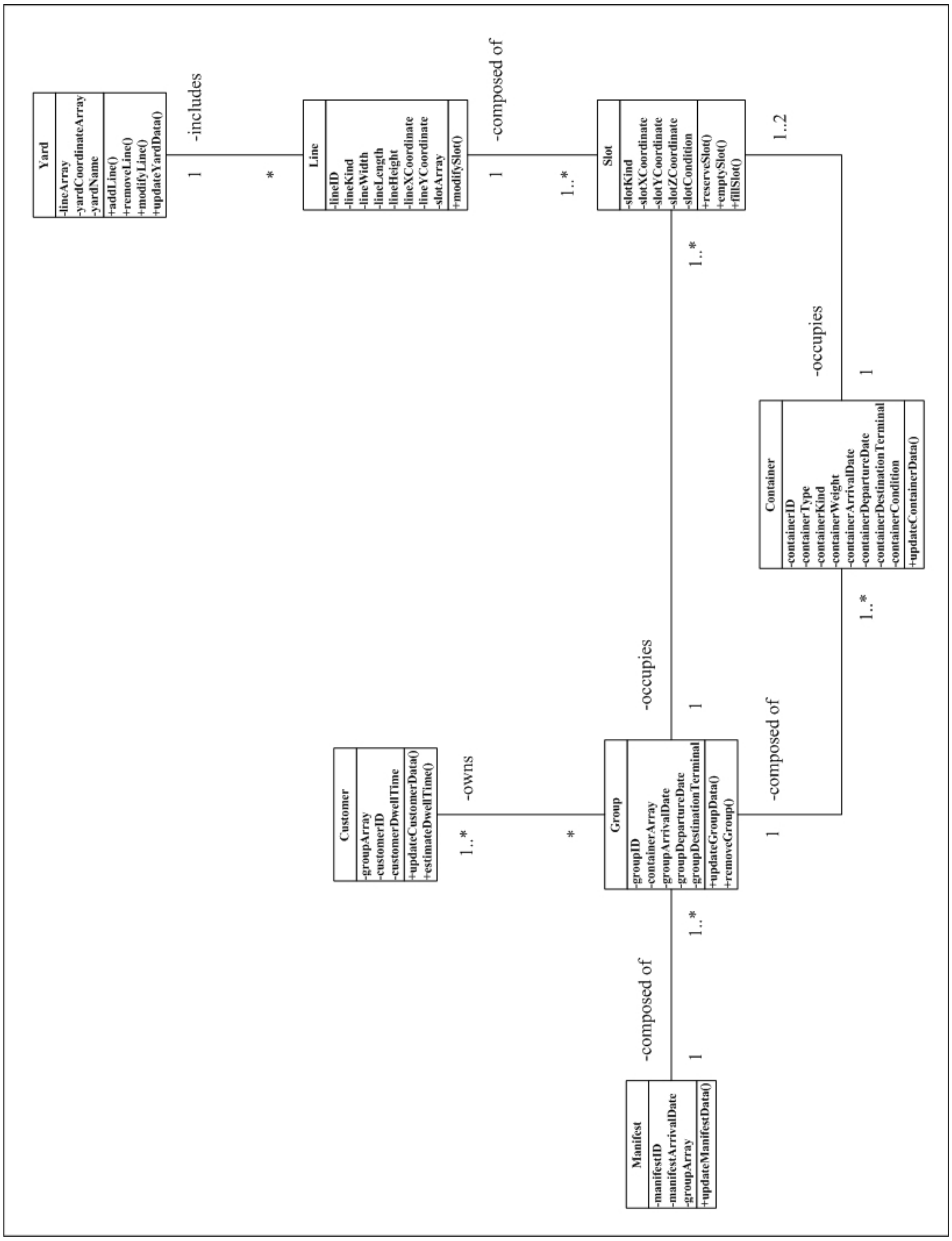


Fig. 3.2. The class diagram

3.4 The Sequence Diagrams

The boundary classes model the interaction between the system and its actors. Since they are part of the requirements model, boundary classes are relatively abstract. They do not directly represent all the different sorts of interface widget that will be used in the implementation language. They represent the interaction with the user, and the entity classes represent the behavior of things in the application domain and storage of information that is directly associated with those things.

Boundary classes represent the relatively stable aspects of the way that the software is intended to operate. As long as the user requirements for the system do not change, the same boundary classes can model the operation of the software and its interaction with the users. Thus the description of each class is also relatively stable, and will probably do not change frequently. By contrast, object instances often change frequently, reflecting the need for the system to maintain an up-to-date picture of a dynamic business environment. Instances of boundary classes are particularly volatile, so they have short lifetimes and are subject to frequent creation and destruction.

The focus of a model of object interaction is to determine the most appropriate scheme of messaging between the objects in order to support a particular user requirement. Each use-case can be seen as a dialogue between an actor and the system that results in objects performing tasks so that the system can respond in the way that is required by the actor. For this reason many interaction diagrams explicitly include objects to represent the user interface with boundary objects.

A sequence diagram shows an interaction between objects arranged in a time sequence. Sequence diagrams can be drawn at different levels of detail and to meet different purposes at several stages in the development life cycle. The most common application of a sequence diagram is to represent the detailed object interaction that occurs for one use-case. When a sequence diagram is used to model the dynamic behavior of a use-case it can be seen as a detailed specification of the use-case [Bennett, 29]. The corresponding sequence diagrams are generated for each use-case of the yard management system and they can be seen in detail in the appendix B.

CHAPTER IV

OPTIMIZATION ALGORITHM

The second step taken after the analysis and design of the yard management system is the development of an optimization algorithm that combines with the yard management system. The system acts as a decision support system, and the optimization algorithm uses several placement patterns for container placement operations over the yard space.

4.1 A Brief Introduction to Operations Research

Many components of organizations tend to grow into relatively autonomous empires with their own goals and value systems, thereby losing sight of how their activities and objectives mesh with those of the overall organization. What is best for one component frequently is detrimental to another, so the components may end up working at cross purposes. As the complexity and specialization in an organization increase, it becomes more and more difficult to allocate the available resources to the various activities in a way that is most effective for the organization as a whole. Operations research involves research on operations. Thus, operations research is applied to problems that concern how to conduct and coordinate operations within an organization. A large amount of computation is usually required to deal most effectively with the complex problems typically considered by operations research. Therefore the development of electronic digital computers, with their ability to perform arithmetic calculations thousands or even millions of times faster than a human being can, was tremendous boom to operations research [Hillier, 30].

Today, the term “operations research” or often called as “management science” means scientific approach to decision making, which seeks to determine how best to design and operate a system, usually under conditions requiring the allocation of scarce resources [Winston, 31]. It is quite often associated with the use of mathematical techniques to model and analyze decision problems. Although mathematics and mathematical models represent a cornerstone of operations research, there is more to problem solving than construction and solution of mathematical models. Specifically, decision problems usually include important intangible factors that can not be translated directly in terms of the mathematical model. Foremost among these factors is the presence of the human element in almost every decision environment. In some cases, the effect of human behavior has so influenced the decision problem that the solution obtained from the mathematical model is deemed impractical.

As a problem solving technique, operations research must be viewed as both a science and an art. The science aspect lies in the providing mathematical techniques and algorithms for solving appropriate decision problems. Operations research is an art because success in all the phases that precede and succeed the solution of a mathematical model largely depends on the creativity and personal ability of the decision-making analysts [Taha, 32].

A common theme in operations research is the search for an optimal, or best, solution. Many procedures have been developed for finding such solutions for certain kinds of problems. However, it needs to be recognized that these solutions are optimal only with respect to the model being used. Since the model necessarily is an idealized rather than an exact representation of the real problem, there can not be any utopian guarantee that the optimal solution for the model will prove to be the best possible solution that could have been implemented for the real problem. According to Hillier; eminent management scientist and Nobel Laureate in economics Herbert Simon points out that satisficing is much more prevalent than optimizing in actual practice. The term satisficing can be realized as a combination of satisfactory and optimizing together. The managers tend to seek a solution that is good enough for the problem at hand. Rather than trying to develop an overall measure of performance optimally reconcile the conflicts between various desirable objectives, a more pragmatic approach may be used. Goals may be set to establish minimum satisfactory levels of performance in various areas, based perhaps on past levels of performance or on what the competition is achieving. If a solution is found that enables all these goals to be met, it is likely to be adopted without further ado. Such is the nature of satisficing. The distinction between optimizing and satisficing reflects the difference between theory and the realities frequently faced in trying to implement that theory in practice. One of England's operations research leaders Samuel Eilon remarks that; while optimizing is the science of the ultimate, satisficing is the art of the feasible. Operations research teams attempt to bring as much of the "science of the ultimate" as possible to the decision-making process. In addition to pursuing the science of the ultimate, the team should consider the cost of the study and the disadvantages of delaying its completion and then attempt to maximize the net benefits resulting from that study. In recognition of this concept, operations research teams occasionally use heuristic procedures to find a good sub-optimal solution. This is most often the case when the time or cost required to find an optimal solution for an adequate model of the problem would be very large [Hillier, 30].

4.2 Container Terminal

Although the analysis and optimizing method developed in this study is general and can be applied to any container terminal, the specific cases and examples are taken from Armaport container terminal. Therefore following details are specific to that terminal.

Armaport is a stacker-operated container terminal with its increasing container traffic every year which is active 24 hours a day and 7 days a week like every other container terminal. The area of the container terminal is not stable yet. Expansions to both land side and sea side are planned in a few years. With the current condition and container traffic of the terminal, a yard management system is not a must yet. But considering the yearly growth of the container terminal, the managers are after the implementation of a yard management system which is planned to act as a decision support system for container operations. With the increase in container traffic, another change is required about the upgrading of machinery being used for the container operations. Because there will be a need to stack more containers in the same yard space, this will welcome the use of alternative stacking equipment. While the main structure of the container terminal can somehow be adapted to the new stacking equipment, the optimization algorithm developed in this study is only suitable for reach-stacker operations as it is now in Armaport.

4.2.1 Yard Layout

The yard can roughly be divided into 3 main areas. The quay side consists of 2 vertical berths for ships to embark/disembark their goods. Container stacking over the quay side is not available unless there are extreme conditions. There is a storage area and container freight station near the management buildings, so those places are not available for container stacking neither unless there are extreme conditions. The stacking are can also be considered in 3 main parts; export container area, import container area, and empty container area. The stacking area approximately can store up to 4500 containers at maximum. There is also another huge empty container stacking area right outside the yard.

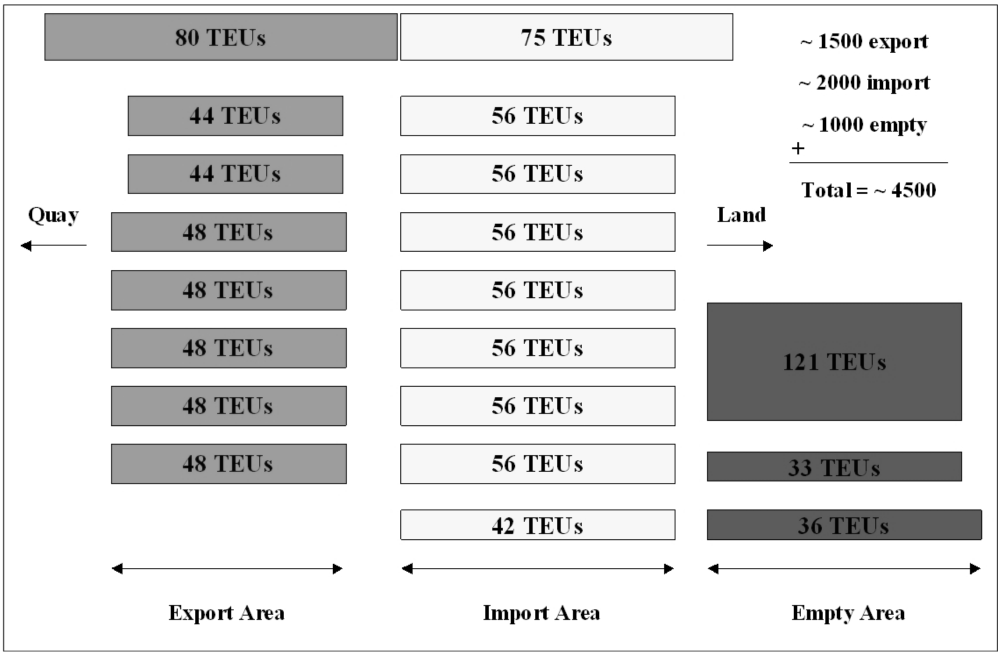


Fig. 4.1. Top view of Armaport yard layout

As seen in Fig. 4.1, the export area is closer to the quay and the import area is closer to the yard. The container export is done from the quay side and the customers take their goods from the land side. This separation is because of preventing the trucks intercepting each other while carrying containers between the lines. The empty container area is located closer to the container freight station in order to reduce the time to bring empty containers to the station for loading the goods into empty containers and to send empty containers to stack after their unloading. In addition to this layout, there is also a reefer line which is closer to electrical power supplies for the freezers, and there is also an imco line which is used for stacking the containers that carry dangerous materials in a reserved space. All the containers must be handled with care because of the size and risks of the heavy equipment, but the containers which carry dangerous materials should be handled much more carefully because of a disaster risk.

The above explanations do not define the exact yard layout of the container terminal. The yard operators create, enlarge or reduce lines in a free manner all the time. They can even deploy lines in the quay area, in front of the management buildings and so on. We tried to model the flexibility of this dynamic yard layout as explained in chapter 3. Considering all the yard and area parameters are open to changes, the system can be easier to adapt to different container terminals either.

4.2.2 Lines

We can physically consider the container stacking lines as rectangular prisms which are made of container slots. The containers are stacked over each other in the lines. One basic constraint for container stacking is that you can not place any 20'-containers over a 40'-container. This is because of the risk of a loaded 20' container may break into the 40'-container from the middle. It is dramatic to say that even this constraint can be violated in several circumstances in reality. But these cases are very rare, so we assumed that no 20'-containers can be placed over a 40'-container. That is why the lines are created to include only a single type of container, whether 20'-container or 40'-container. However a 40'-container can be placed over 20'-containers without any problem. The lines are usually deployed about 10-12 TEUs long while any other length can be applied at any time. In this study, we assumed that the lines are about 10-12 TEUs long as common. Another point that determines the type of stacking we are using on the yard space is the type of the container handling machinery.

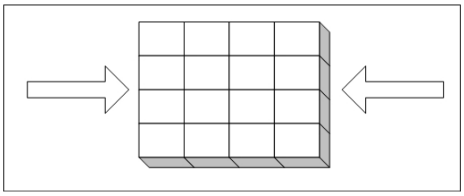


Fig. 4.2. Stacker operation from side view

As seen in Fig. 4.2, the reach stackers approach the lines from the sides and they can only hold the containers horizontally from top. The standard operating height for reach stackers is up to 4 containers height for safe container moves. Above that level a loaded container may cause the back of the stacker to lift up and create dangerous situations. However there are some cases where 5th level of container stacking is allowed where the subject container is very light or empty. Thus, the container stacking is done up to 5 containers height in empty container areas, too. Considering these cases are rather unusual, we assumed that the lines can be stacked up to 4 containers height in this study.

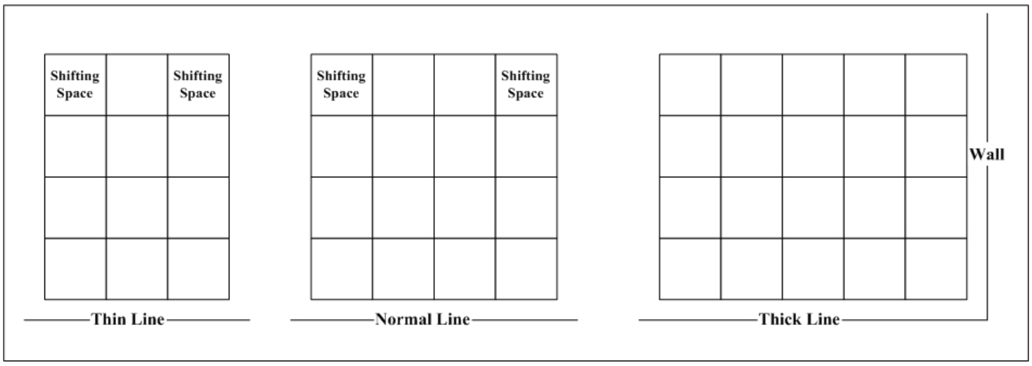


Fig. 4.3. Side view of import line stacking

As shown in Fig. 4.3, there are 3 main types of stacking for the import containers. Thick lines are deployed next to barriers and they allow stacker operations only from one side. They can be considered as simple stacks. The main use of the thick lines is for the placement of large container groups. It is wise to stack the large groups against the barriers in order not to fill up the valuable central yard space with huge groups. The normal lines are the most common type of import lines which are widely used for medium sized container groups, and the thin lines are practically used for smaller container groups. Both the thin lines and normal lines allow stacker operation from both sides. So these lines can not be considered directly as simple stacks like we do in thick lines. The shifting spaces are reserved for creating space for any shifting operation that may occur during the container withdrawal stage. One of the problems about the flexibility of the real life operations is that yard operators sometimes decide to deploy temporary lines which are only two containers thick. These lines are especially created for the large transport ships that bring hundreds of containers to the terminal at once. At those times, it is crucial to decrease the unloading time of the ship instead of keeping the line standards at the yard area. We tried to model the flexibility of this dynamic line creations and destructions as much as possible in the previous chapter. However we assumed that there are only 3 kinds of import lines as explained above for container stacking, because the other type of lines are rarely used and not really efficient for reach stacker operations.

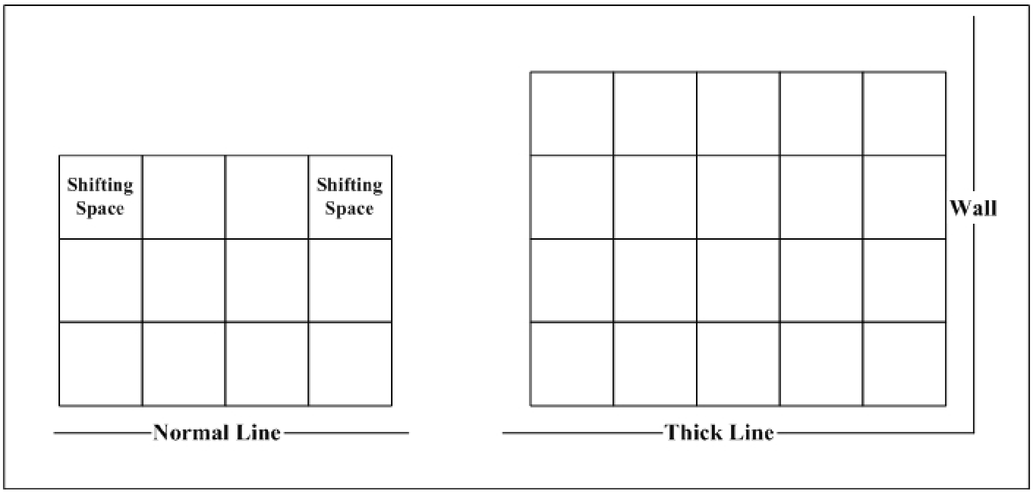


Fig. 4.4. Side view of export line stacking

As depicted in Fig. 4.4, the stacking procedure in the export area is slightly different than the import area. The reason of using the thick lines is similar to the usage of thick lines in the import area. Large groups of containers have similar attributes and can be stacked by the wall. However the common container groups are stacked in normal export lines which are not more than 3 containers height. The reason for keeping the line height lower is for minimizing the risk of making a high number unnecessary shifting. When loading the ships, it is crucial to transport the containers to the ship as soon as possible in order to minimize the time that ships spend for berthing. That is why somewhat smaller lines are preferred for stacking the export containers. The 3 containers height line types are used as the export lines in this study.

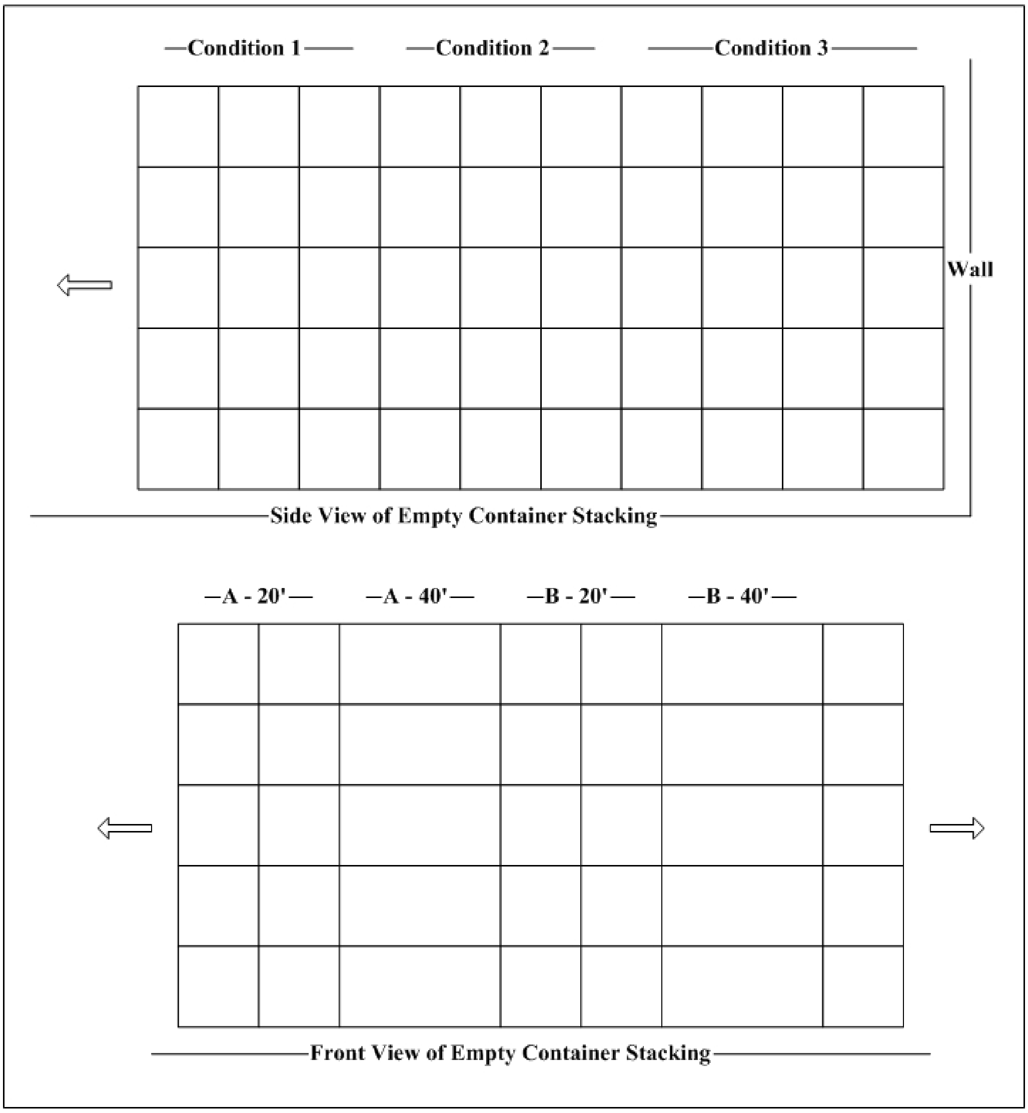


Fig. 4.5. Side and front views of empty container stacking

As seen in Fig. 4.5, the stacking of empty containers is a different story. As the containers are empty, they can be stored up to 5 containers height. The collapsible flat racks can even be stacked over each other up to 20 of them. And there is no general width or length of line for empty container lines. They can reach up to 10 containers thick and about 10 TEUs long. The empty container stacking is based on getting the desired container type with the desired container condition which is owned by a specific trading company. While the containers can travel all over the world, their owners are the trading companies that manufactured them.

4.3 Terminal Data Analysis

It is better to define the container movement processes on the yard space before we proceed further into the analysis of the real terminal data. Basically, a container can enter the yard space whether from sea or land. It is the same case when a container is leaving the yard space.

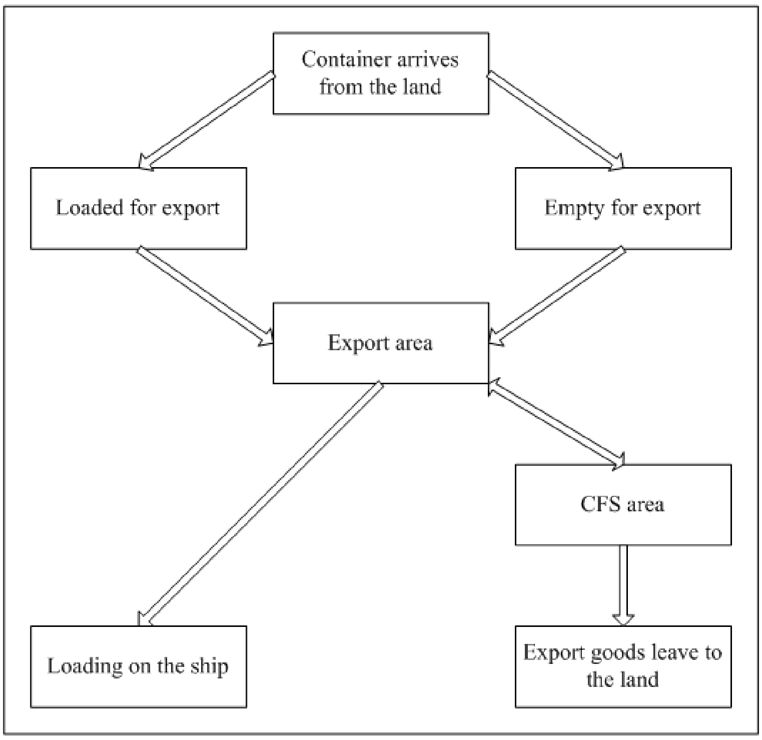


Fig. 4.6. Container arrival from the land

As shown in Fig. 4.6, both the empty containers and the full containers can be exported. The empty containers to be exported are directly stacked on the yard space. The export container can be filled outside the terminal, so when it arrives the terminal, it is directly stacked on the yard space, too. If the goods are to be loaded in the terminal, first the goods are loaded in containers at the container freight station area, then they are stacked on the yard space. There are also some cases when the goods to be exported return back to the land on trucks.

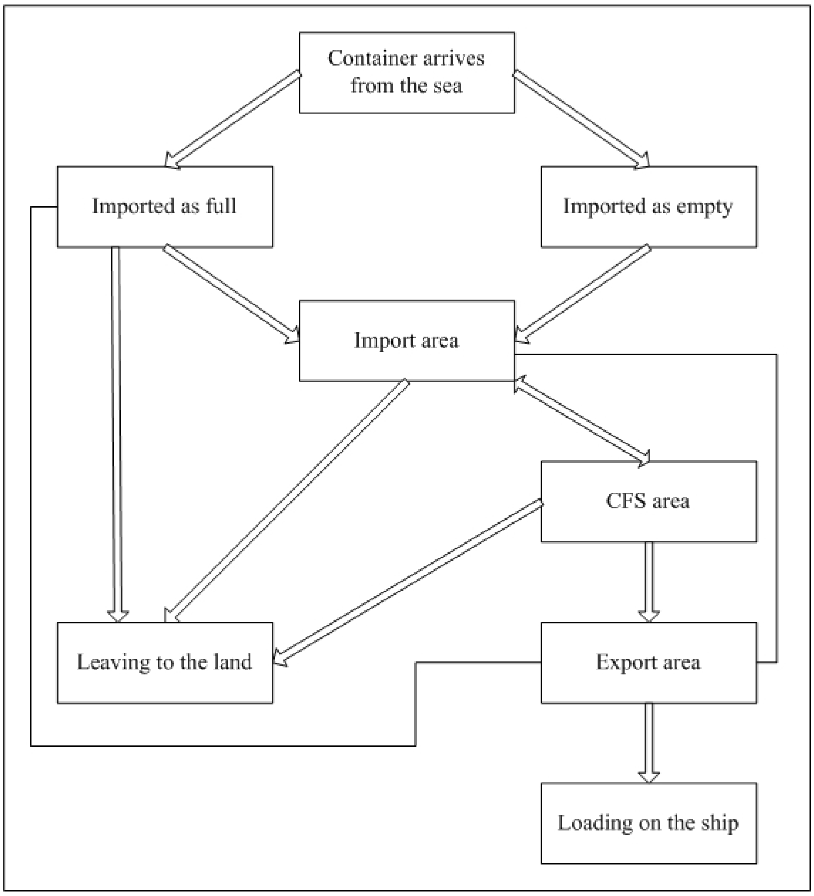


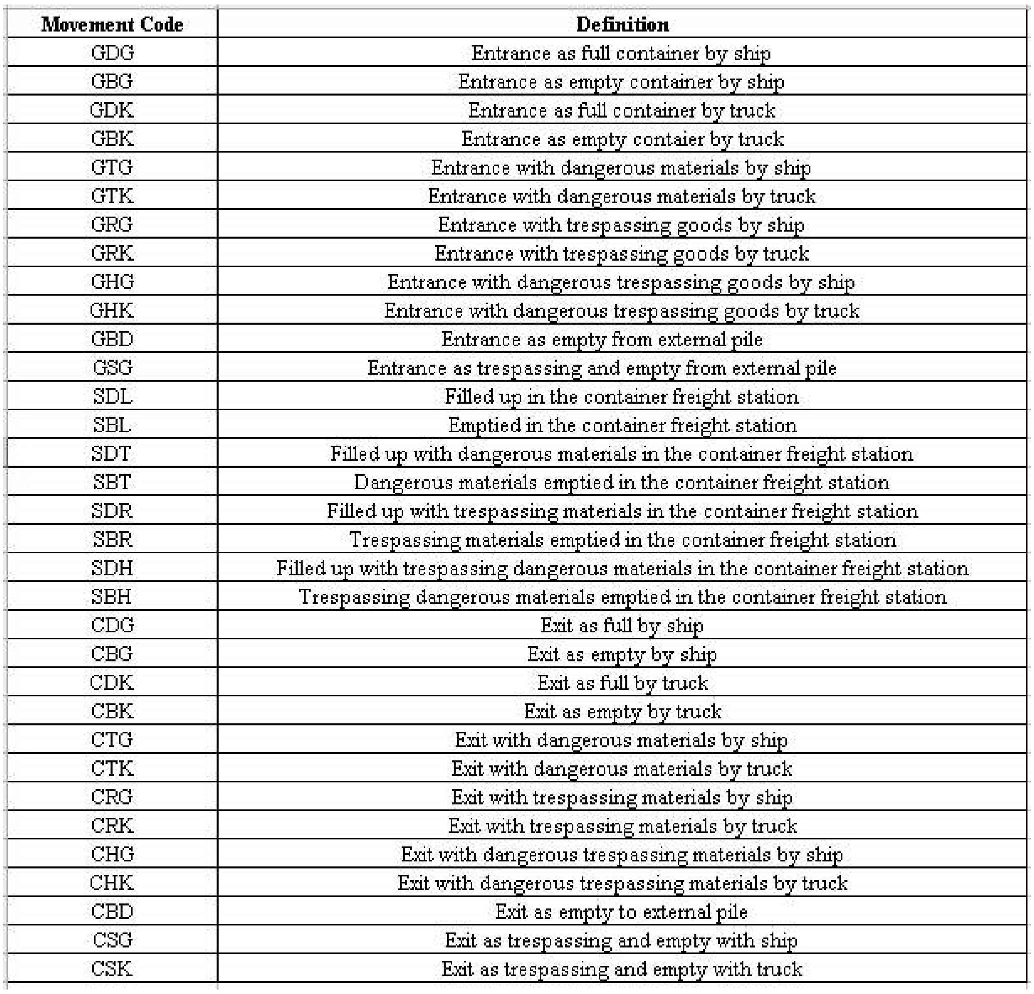
Fig. 4.7. Container arrival from the sea

As depicted in Fig. 4.7, the containers can be imported either as empty or full. The imported full containers can be directly placed in the stack over the import area, or can be placed in the stack over the export area to be loaded in another ship or can directly leave to the land bypassing the yard area. The last one is especially the case for the import of very dangerous materials which are immediately put on a leaving truck by the quay crane. The full containers can also be emptied in the container freight station area and the goods can leave the terminal on trucks. The imported empty containers can be stacked in the yard space or they can be sent to the container freight station area to be filled and exported by ship in the next step.

When an import container is to be taken from the ship and placed on the yard, or when an export container is to be loaded to ship; the placement optimization developed in this study will be in charge.

4.3.1 Container Handling Data Analysis

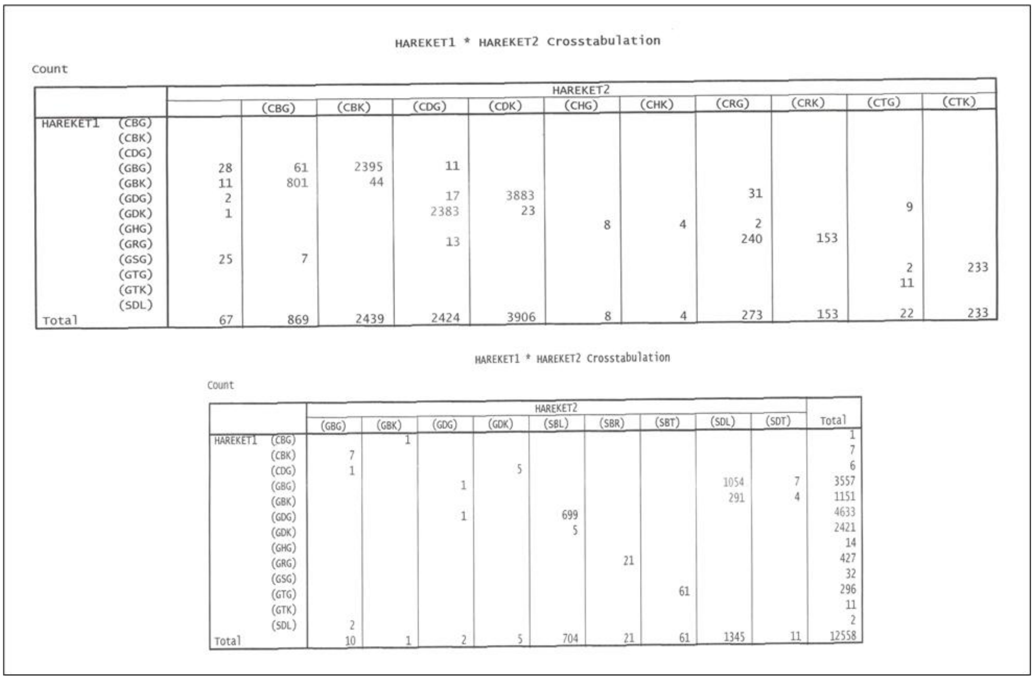
The data about container movements over the terminal in year 2001 is analyzed to filter the most critical operations in the yard area and focus on them. Before we proceed further into the data, it is better to take a look at the container movement codes in the terminal as shown in Table 4.1.



Movement Code	Definition
GDG	Entrance as full container by ship
GBG	Entrance as empty container by ship
GDK	Entrance as full container by truck
GBK	Entrance as empty contaier by truck
GTG	Entrance with dangerous materials by ship
GTK	Entrance with dangerous materials by truck
GRG	Entrance with trespassing goods by ship
GRK	Entrance with trespassing goods by truck
GHG	Entrance with dangerous trespassing goods by ship
GHK	Entrance with dangerous trespassing goods by truck
GBD	Entrance as empty from external pile
GSG	Entrance as trespassing and empty from external pile
SDL	Filled up in the container freight station
SBL	Emptied in the container freight station
SDT	Filled up with dangerous materials in the container freight station
SBT	Dangerous materials emptied in the container freight station
SDR	Filled up with trespassing materials in the container freight station
SBR	Trespassing materials emptied in the container freight station
SDH	Filled up with trespassing dangerous materials in the container freight station
SBH	Trespassing dangerous materials emptied in the container freight station
CDG	Exit as full by ship
CBG	Exit as empty by ship
CDK	Exit as full by truck
CBK	Exit as empty by truck
CTG	Exit with dangerous materials by ship
CTK	Exit with dangerous materials by truck
CRG	Exit with trespassing materials by ship
CRK	Exit with trespassing materials by truck
CHG	Exit with dangerous trespassing materials by ship
CHK	Exit with dangerous trespassing materials by truck
CBD	Exit as empty to external pile
CSG	Exit as trespassing and empty with ship
CSK	Exit as trespassing and empty with truck

Table 4.1. Container movement codes

12558 container movements in the terminal are analyzed for the year of 2001. For each container; group number, arrival voyage number, container owner, container type, container kind, container weight, departure voyage number, target terminal, and container movement codes and dates are included in the data. Our first approach was finding the relation between two consecutive moves in the yard right after the arrival of the container in the terminal. The first containers movement versus second container movement cross tabulation is given in the Table 4.2.



HAREKET1 * HAREKET2 Crosstabulation

Count

		HAREKET2									
			(CBG)	(CBK)	(CDG)	(CDK)	(CHG)	(CHK)	(CRG)	(CRK)	(CTG)	(CTK)
HAREKET1	(CBG)											
	(CBK)											
	(CDG)											
	(GBG)	28	61	2395	11							
	(GBK)	11	801	44								
	(GDG)	2			17	3883			31			
	(GDK)	1			2383	23					9	
	(GHG)						8	4	2			
	(GRG)				13				240	153		
	(GSG)	25	7									
	(GTG)										2	233
	(GTK)										11	
	(SDL)											
Total		67	869	2439	2424	3906	8	4	273	153	22	233

HAREKET1 * HAREKET2 Crosstabulation

Count

		HAREKET2									Total
		(GBG)	(GBK)	(GDG)	(GDK)	(SBL)	(SBR)	(SBT)	(SDL)	(SDT)	
HAREKET1	(CBG)		1								1
	(CBK)	7									7
	(CDG)	1			5						6
	(GBG)			1					1054	7	3557
	(GBK)								291	4	1151
	(GDG)			1		699					4633
	(GDK)					5					2421
	(GHG)										14
	(GRG)						21				427
	(GSG)										32
	(GTG)							61			296
	(GTK)										11
	(SDL)	2									2
Total		10	1	2	5	704	21	61	1345	11	12558

Table 4.2. First movements versus second movements cross tabulation

The results show that the most frequent terminal entrance first movements of containers are GDG (Entrance as full container with ship) with 4633, GBG (Entrance as empty container with ship) with 3557, GDK (Entrance as full container with truck) with 2421, and GBK (Entrance as empty container with truck) with 1151. The combination of those 4 first movements make a total of 11762 moves out of 12558 moves, and that is approximately 94% of all the first movements.

After determining the main first movements, then the corresponding second movements are analyzed due to the most frequent first moves. The results show that:

1054 SDL, and 2395 CBK moves are made after GBG moves. In other words, approximately 30% of the empty containers that arrive with ships are filled in the container freight station of the terminal while approximately 67% of them leave the terminal with trucks so as to be filled outside the terminal area.

291 SDL, and 801 CBG moves are made after GBK moves. In other words, approximately 25% of the empty containers that arrive with trucks are filled in the container freight station of the terminal while approximately 70% of them leave the terminal with ships.

699 SBL, and 3883 CDK moves are made after GDG moves. In other words, approximately 15% of the full containers that arrive with ships are emptied in the container freight station of the terminal while 84% of them leave the terminal with trucks.

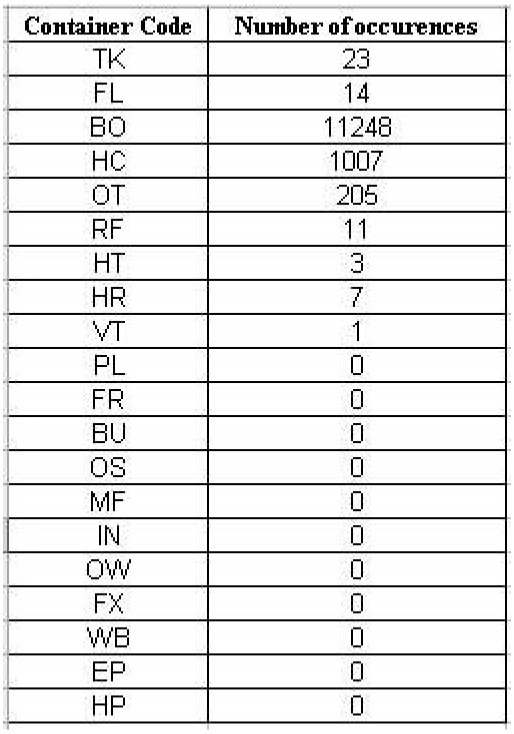
2383 CDG moves are made after GDK moves. In other words, approximately 98% of the full containers that arrive with trucks leave the terminal with ships.

After defining the main container movement activities in terminal, we focused on the analysis of the most frequent container types. Before we proceed further into the data, it is better to take a look at the container codes in the terminal as depicted in Table 4.3.

Container Code	Definition
TK	Tank (TA)
FL	Flat
BO	Box
HC	High Cube
OT	Open Top
RF	Reefer (RE)
HT	Hard Top
HR	High CubeReefer
VT	Ventilated (VE)
PL	Platform
FR	Flat Rack
BU	Bulk
OS	Open Side
MF	Mafi
IN	Insulated
OW	Over Wide
FX	FlexiTank
WB	Widbady
EP	Europalet
HP	High Cube Palet

Table 4.3. Container codes

12558 containers that had been handled during the year 2001 are analyzed to define the most frequent container types that are handled in the terminal. The container codes and their occurrences in the terminal can be seen in Table 4.4.



Container Code	Number of occurences
TK	23
FL	14
BO	11248
HC	1007
OT	205
RF	11
HT	3
HR	7
VT	1
PL	0
FR	0
BU	0
OS	0
MF	0
IN	0
OW	0
FX	0
WB	0
EP	0
HP	0

Table 4.4. Container occurrences

The results show that 11248 standard box type containers and 1007 high cube containers had been handled during the year 2001. The combination of these 2 types of containers creates approximately 98% of the container traffic in the terminal.

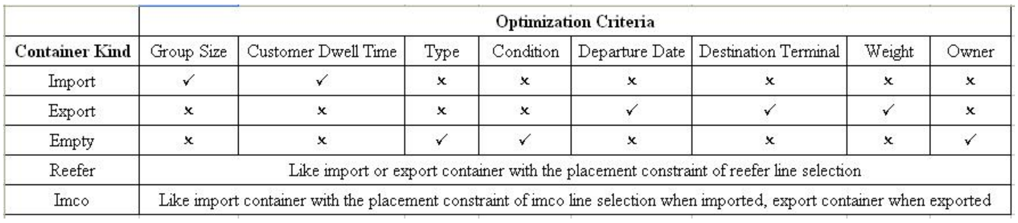
4.3.2 The Evaluation of Analysis Results

The analysis results show that; although there are various containers manufactured for transportation, the most common containers that are being used are by far the standard box containers. It is known that the stacking of high cube containers is no different than the stacking of box containers. Because the high cube containers are just a few inches higher than the box containers and it does not cause any problem to stack them over each other as usual. The open top containers can be treated as box containers while stacking again unless there is nothing in the open top container that creates an extra height. The reefer containers also have similar dimensions like the box containers. Thus, we assumed all the containers as standard box containers during the development of the system in this study. The other containers that are being handled are treated as out of the system, and they are advised to be placed manually.

The analysis results also show that; the container importing activity is significantly higher than the container exporting activity. As a result, we first focused our studies on the ship unloading and import container placement patterns over the yard space. However, the importing and exporting ratio is not stable and highly dependent on the economical situation of the country, so we had to bear in mind that the condition could be vice versa at any time. It is also obvious from the data that the companies usually do not prefer loading and unloading their containers in the container freight station.

4.4 Container Placement Algorithms

Considering that different criteria applies for the different lines over the yard space. We tried to separate the yard area into regions as well as lines. The regions can be defined as import, export and empty basically, and each of them has a different approach for stacking containers. The imco and reefer containers are also stacked with slight exceptions and that will be explained after the main placement algorithms. The placement algorithms can also be divided into 3 main parts; one of them, and the most critical one from the operational point of view, is the unloading phase of the ship in combination with space reservation over the yard space. The other two can be defined as container insertion to stack and container removal from stack processes basically. The different stacking criteria for different areas of the yard space can be seen in Table 4.5.



	Optimization Criteria							
Container Kind	Group Size	Customer Dwell Time	Type	Condition	Departure Date	Destination Terminal	Weight	Owner
Import	✓	✓	×	×	×	×	×	×
Export	×	×	×	×	✓	✓	✓	×
Empty	×	×	✓	✓	×	×	×	✓
Reefer	Like import or export container with the placement constraint of reefer line selection							
Imco	Like import container with the placement constraint of imco line selection when imported, export container when exported							

Table 4.5. Optimization Criteria

- Manifest Submission and Slot Reservation Algorithm
- Container Insertion Algorithm
  - Import Container Insertion Algorithm
  - Export Container Insertion Algorithm
  - Empty Container Insertion Algorithm
  - Reefer Container Insertion Algorithm
  - Imco Container Insertion Algorithm
- Container Removal
  - Container Removal Algorithm
  - Group Removal Algorithm
  - Shifting

4.4.1 Manifest Submission and Slot Reservation Algorithm

Before a ship arrives to a terminal, the terminal operators can obtain the manifest of the ship at least a few hours before the arrival. The manifest of the ship has the detailed info about the containers being transported, and having this information before the arrival of the ship is crucial in order to prevent from the surprises that may occur when the ship arrives. When the groups of containers that are arriving at the terminal are known, we can check the current state of the yard space and make some reservations and arrangements in our lines so that the larger container groups are sent to their reserved slots without scattering them over them lines.

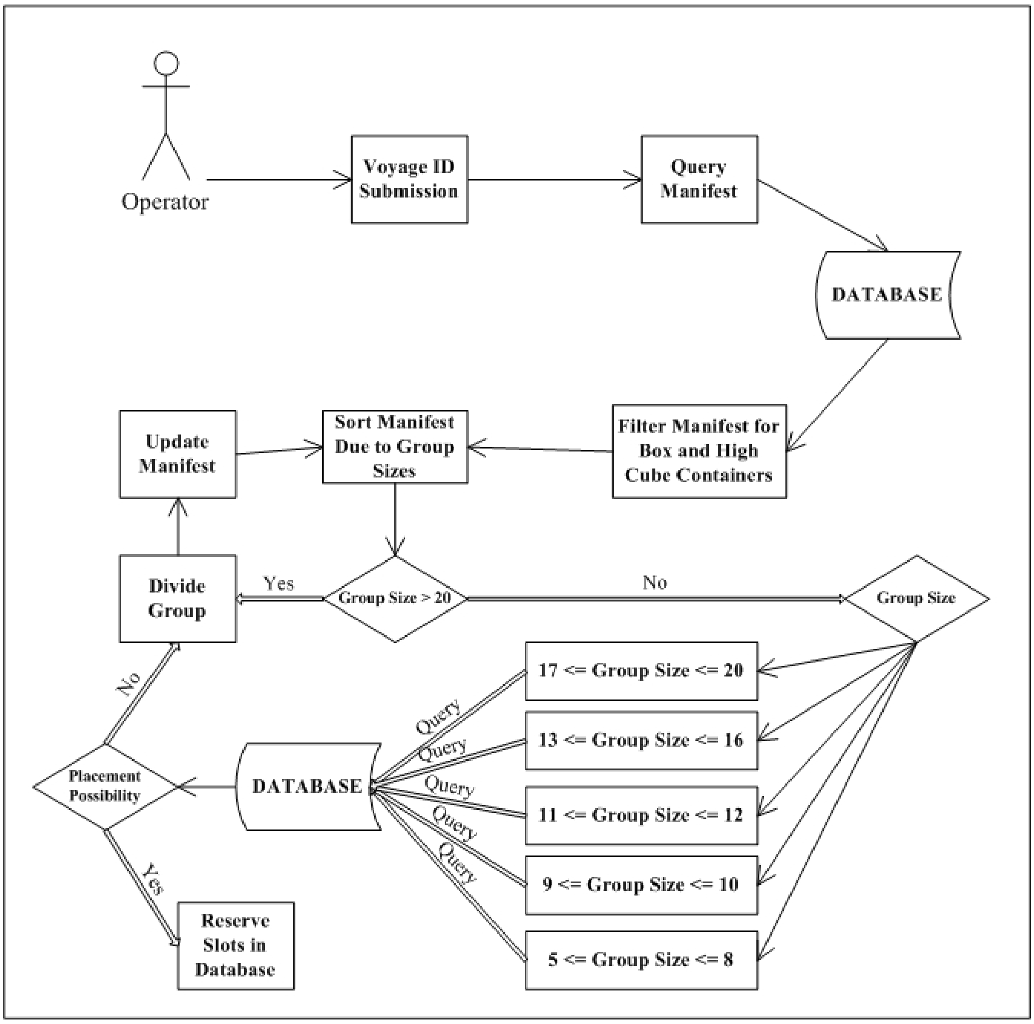


Fig. 4.8. Manifest submission and slot reservation algorithm

As seen on Fig. 4.8, the operator submits the next voyage that is expected to arrive at the terminal to the yard management system, and that creates a query to extract the manifest of that ship from database. The manifest of the ship is first filtered from the containers that are not box or high cube containers. The other containers are left out of the optimization algorithm, so that they should be placed manually on the yard space when the quay cranes handle them. After the filtering, we now have only the box and high cube containers as we desire, and the next step is sorting the container groups due to their group sizes and the update of the manifest. The group sizes are sorted in descending order because the larger the group size is, the more the group is a possible victim of scattering. So the system tries to find and reserve spaces for larger groups as soon as possible. The basic operation unit of a line is its slice because considering the reach stacker operation;

slices are somehow independent from each other. So the biggest capacity of a slice on the yard is the slice of a thick line with 20 slots. That also means that any group size over 20 containers will uneventfully face scattering. That is why; the next step is the division of container groups into smaller ones which have no more than 20 containers. The division is simply made on a basis that taking the mod of the group size, and the manifest is updated. For example a group of 25 containers becomes a group of 20 containers plus a group of 5 containers.

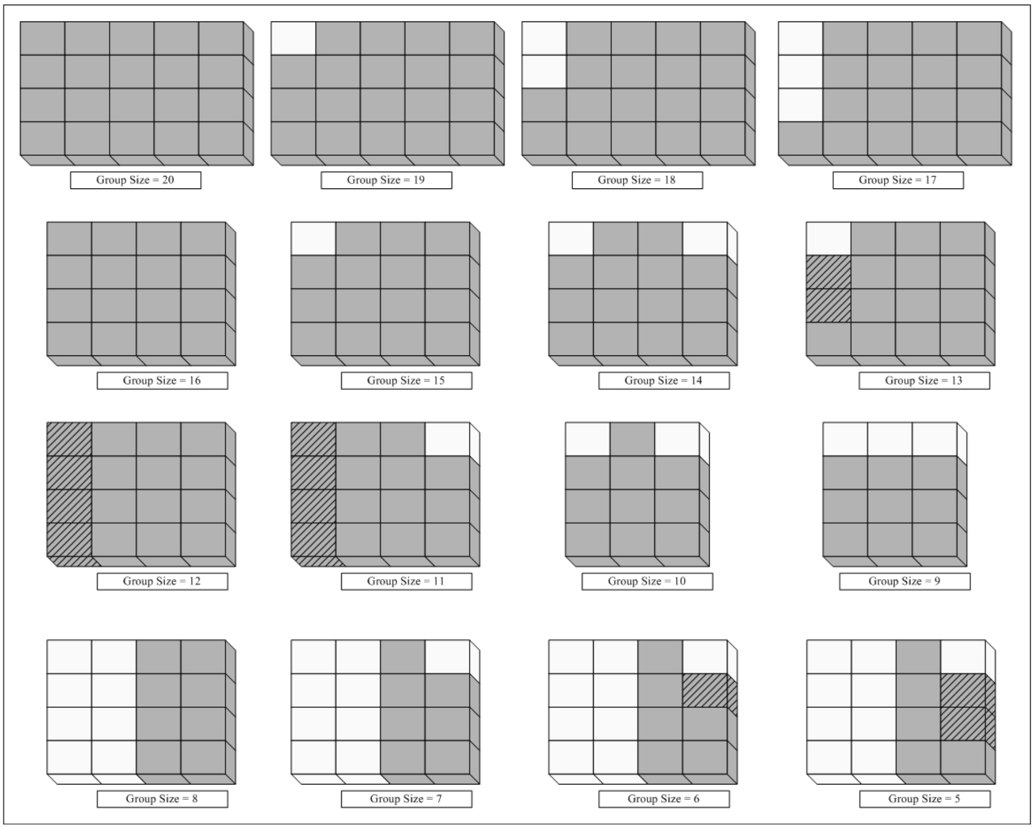


Fig. 4.9. Group placement patterns

When all the group sizes are not more than 20 containers in the manifest, different placement patterns are applied to different container group sizes in order to reserve suitable places for the groups. The details of the placement patterns can be seen in Fig. 4.9 above.

There are 5 main placement patterns to reserve slots for large container groups, the container groups which have 17 to 20 containers are directly sent to the thick line empty slices to fill a slice with a single group. The container groups which have 13 or 16 containers are sent to the normal line empty slices to fill a slice with a single group.

As it can be seen from the stacking patterns shown in Fig. 4.9, it is tried to create empty slots at the top corners of the slices which are also called shifting spaces. The shifting spaces are quite handy when a container removal process creates extra shifting. The shifted containers can be placed easily to the neighbor shifting spaces without carrying them between the lines to find a space to place them. The spaces between two lines are at optimum for the operation of reach stackers and trucks with ease, so a reach stacker operator tries to avoid putting containers on the ground as much as possible.

The placement patterns of container groups which have 11 or 12 containers are another story. There are 2 different possibilities for stacking them. One of them is stacking them in empty normal line slices, while the other one is stacking them in empty thin line slices when there is no available empty normal line slice. The container groups which have 9 or 10 containers are perfect fit for the empty slices in the thin lines which also create double shifting spaces in a slice. Finally, the container groups which have 5 to 8 containers are placed at the half slices of normal lines. Considering the normal line slices consist of 2 half slices which are mirror image of each other, stacks of two container groups which have 5 to 8 containers in the same slice are independent from each other in reach stacker operation point of view.

No reservation patterns are suggested for container groups which have 4 or fewer containers because there is no practical benefit for making reservation for them. They are processed instantly by the system when the quay cranes handle them, and they are most possibly sent to the outer slots of the slices in order to avoid shifting and scattering problems that may occur when they are stuck in inner slots of the slices.

4.4.2 Container Insertion Algorithm

When a container is handled by a quay crane or is brought to the yard on a truck, the system processes the container information and suggests a suitable place for placement. However, depending on the details and the next movement of the container, the placement algorithm to be used differs.

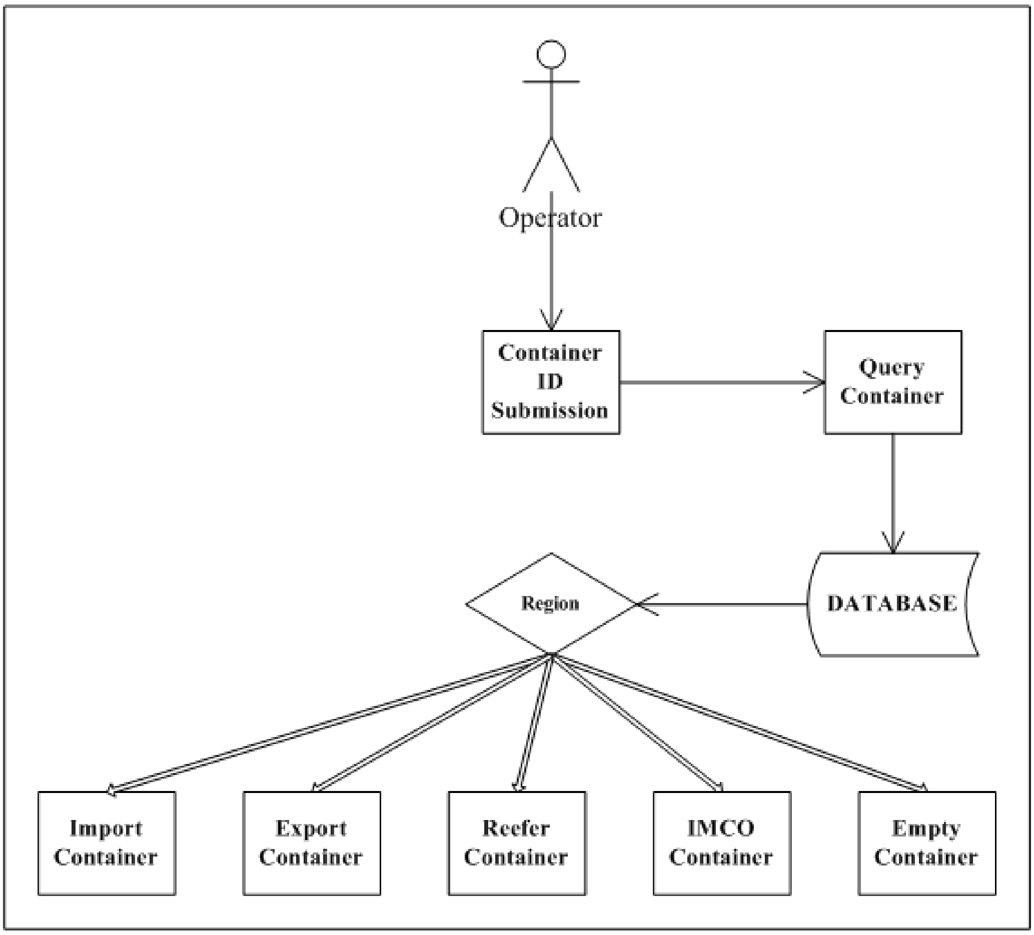


Fig. 4.10. Container insertion algorithm

As depicted in Fig. 4.10, the operator submits the container to the system and creates a query to extract the information about the container from database. Depending on the region that this container belongs to, the optimization algorithm directs the request to the related sub container insertion algorithm as seen in the figure above.

4.4.2.1 Import Container Insertion Algorithm

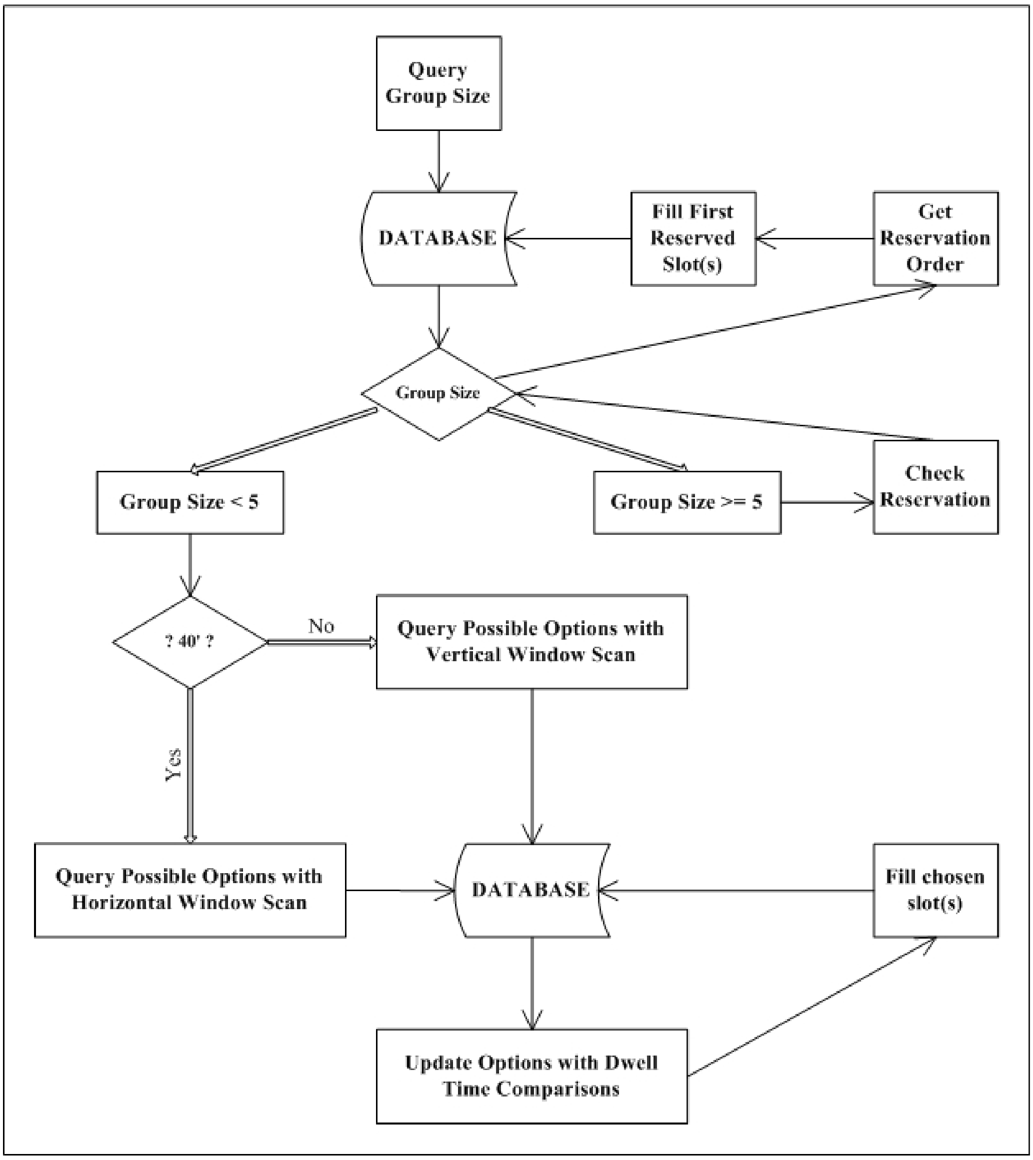


Fig. 4.11. Import container insertion algorithm

As shown in Fig. 4.11, after the optimization algorithm directs the request to the import container insertion algorithm, the first step is querying the group size of the group that the container belongs to. If the group size is bigger than 4 containers, it also means that probably a slot reservation had been already made in the manifest submission and slot reservation algorithm. So the system checks the reserved slots from database and inserts the incoming container to the first slot of the reservation space depending on the order of slot reservation. If the group size is less than 5 containers, the system checks whether the container is a 40’ container of 20’ container. Depending on the type of the container, the system creates a query for finding available spaces on the suitable slices. The horizontal window scan is used for the 40’ containers and it means that when an available slot is found, the neighbor slot is immediately checked to know whether it is empty or not. The vertical window scan is used for the 20’ containers and it means that when an available slot is found, the slot just below the available slot is checked to know whether it is occupied by a 20’ container or a 40’ container. After several possible options are achieved, the dwell time comparisons made due to the slots occupied by previous containers just below the incoming container. The idea is not to place a container over other containers which are expected to be removed earlier than the new one on top. The most suitable option is chosen due to the comparison of dwell times and the slots are updated in database as the container is placed in its location.

4.4.2.2 Export Container Insertion Algorithm

As shown in Fig 4.12, after the optimization algorithm directs the request to the export container insertion algorithm, the first step is the querying of departure date, destination terminal and weight of the incoming container. The group identity is not important in the case of exporting because it is the responsibility of the receiving terminal to keep the group together when the ship is unloaded there. The terminal which unloads the containers has to deal with keeping the container groups together. After getting the information about the container to be exported, depending on whether the container is a 40’ or 20’ container suitable spaces are searched from database. The most suitable options are calculated through comparisons of departure dates, destination terminals and weights of the containers. The containers leaving with the same voyage are kept close to each other in a line while the ordering of the slices in these lines is dependant on the destination terminals. Finally, heavier containers should be placed on top of the stacks while the lighter ones should remain at the lower levels of the slices.

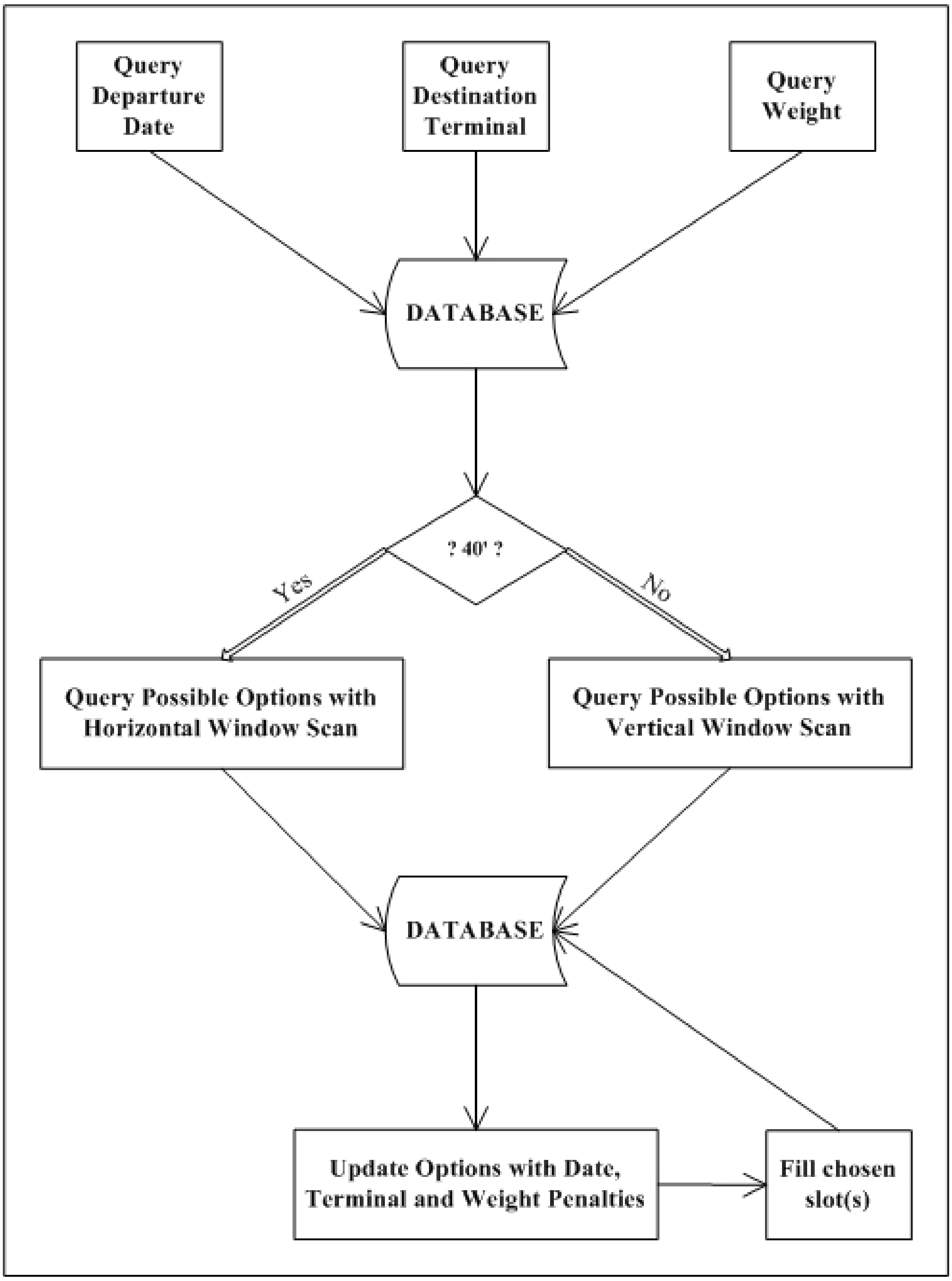


Fig. 4.12. Export container insertion algorithm

4.4.2.3 Empty Container Insertion Algorithm

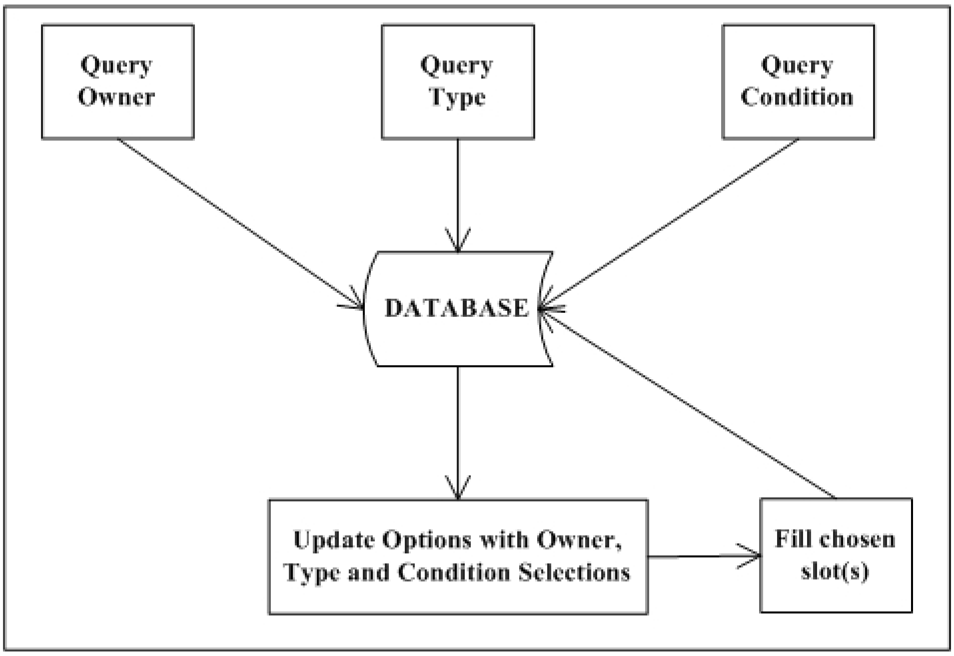


Fig. 4.13. Empty container insertion algorithm

As depicted in Fig. 4.13, after the optimization algorithm directs the request to the export container insertion algorithm, the first step is querying the owner, type and condition of the incoming container from database. The organization of empty slices is based on dividing slices due to container manufacturers/owners with 20' and 40' container slices for each owner. The containers are also stacked with their condition order in the slices. The condition of the containers are ranked with numbers each with meanings like “in good condition”, “worn”, “dirty”, “clean”, etc. The idea is keeping the empty containers with the same attributes close to each other, so it can be easier to find and remove a specific pack of empty containers from a stack in a short time. After getting the information about the incoming container, the possibilities are compared to each other depending on the container owner, condition and type attributes and the most suitable slots are updated at database while placing the containers.

4.4.2.4 Reefer Container Insertion Algorithm

The reefer container insertion algorithm does not have any different logic than then import and export container insertion algorithm. But it has the mixture of the properties of import and export container algorithms. The reefer containers must always be supplied with electrical power in order to maintain their refrigerating functions whether they are to be imported or exported. The reefer containers are sent to their special line with electrical power, and this line consists of two regions to store reefer containers for import or export.

4.4.2.5 Imco Container Insertion Algorithm

The imco containers to be exported are treated with exactly the same export container insertion algorithm while the ones to be imported are treated with almost the same import container insertion algorithm with one exception. The idea is keeping them in a single line in the yard so that the dangerous materials will be handled with extra care over there. So the import container insertion algorithm is used exactly just by adapting it so that the output location always directs the incoming imco container to the imco line.

4.4.3 Container Removal

Depending on the placement decisions we made in the previous sections, the container removal algorithm tries to remove the desired containers from the stack with the minimum number of shifting. The usage of container placement patterns comes in handy during the process of the container removal as the containers of the same group are tried to be kept together. The import area shifting is more critical than the shifting in the export area. Because whatever the shifting is, the containers to be exported are loaded on to the ship in a short period of time; however the shifted containers in the import area have to be stacked again to wait for the customers to take their groups from the terminal.

4.4.3.1 Container Removal Algorithm

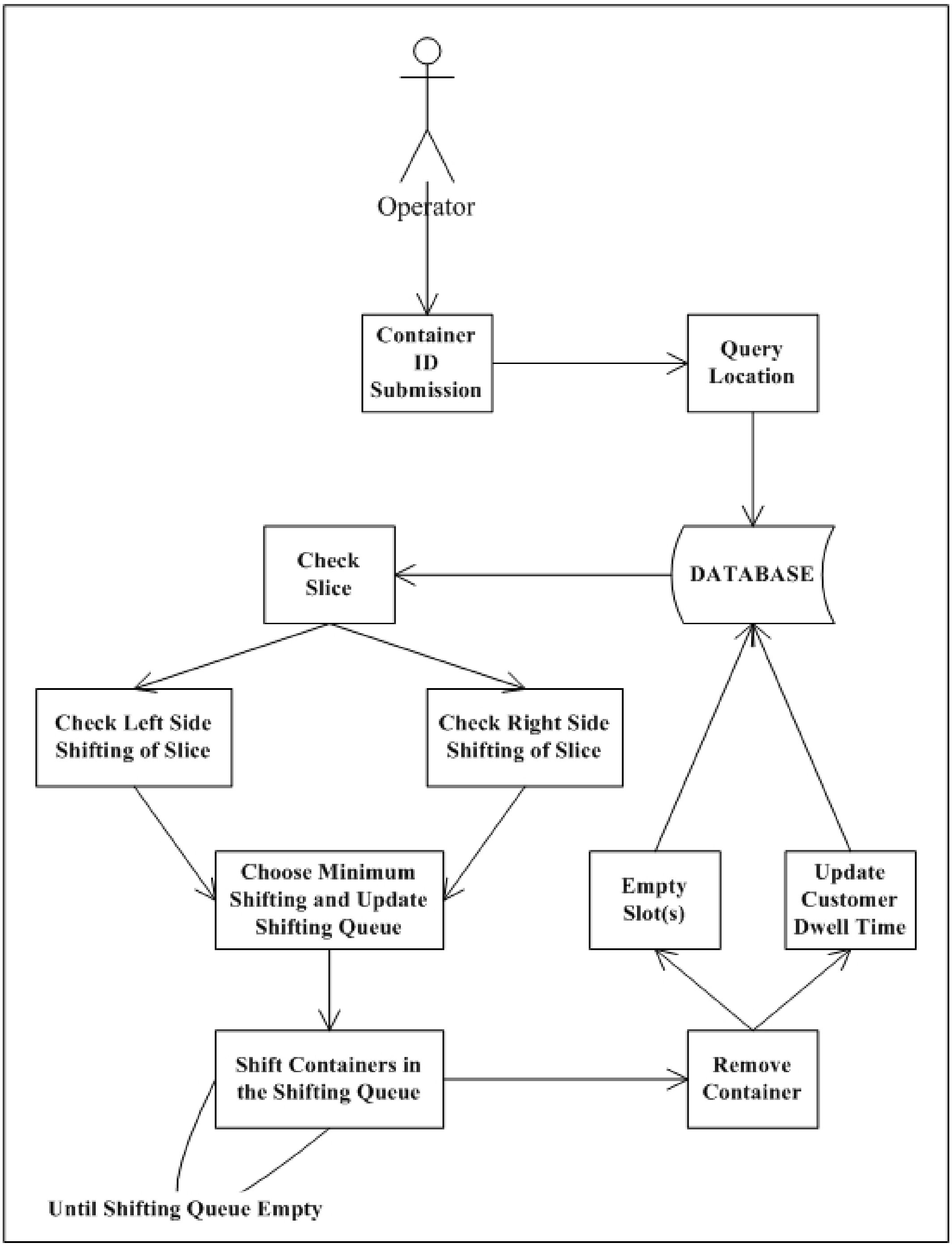


Fig. 4.14. Container removal algorithm

As seen in Fig. 4.14, the operator submits the container to the system and creates a query to find the location of the container in the yard from database. When the slot of the slice that container is assigned is found, container removal procedure from both sides of the stack are calculated and the one that causes minimum shifting is chosen for the removal of the desired container. If there are containers to be shifted above the container to be removed, a shifting queue is created and the containers in the shifting queue are firstly handled and shifted before the removal operation. When the shifting queue is empty the container is removed and the slot becomes free. If it is the container of a customer then dwell time of this container is calculated and updated in the customer dwell time.

4.4.3.2 Group Removal Algorithm

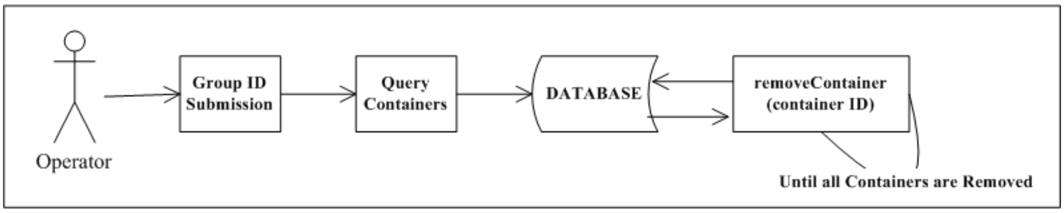


Fig. 4.15. Group removal algorithm

As shown in Fig. 4.15, the operator submits the group ID to the system and creates a query for the containers to be removed. The group removal procedure uses the container removal algorithm consequently. The containers to be removed are sent as input to the container removal algorithm so that shifting will be handled first if there are any.

4.4.3.3 Shifting

When shifting occurs the containers to be shifted are treated as an input to the related container insertion algorithm, such as import, export, empty, reefer, or imco. One exception that becomes a property in the shifting that is trying to place the container to the nearest possible slice. The idea is placing the container as near as possible in order not to make the reach stacker carry the shifted container around.

CHAPTER V

CONCLUSION AND FUTURE WORK

This study consists of mainly two parts. The first part of the study is the analysis and design of a yard management system that will computerize and visualize the yard space layout and operations with software. The second part of the study is related to the development of a 3D optimization algorithm to act as a decision support system for the yard operators during container operations. The optimization routine is planned to accept requests from the operator via the yard management system described above.

5.1 Analysis and Design of the Yard Management System

During the study, it was seen that any computerized system which is planned to handle any kind of decision related to the terminal operations must be designed flexible enough as far as the application is concerned. As this study was focused on the optimization of container operations over the yard space, we tried to construct the model of the yard operations as flexible as possible using the object oriented approach during the analysis and design phase of the yard management system. The yard space is designed so that it will enable the operator to change many features of the yard, including its size, lines, buildings, etc. In addition to this, all those lines and building blocks are shown with their coordinates in order to adapt the program for the unexpected physical changes in future.

While the management system is easier to adapt to any logical change in container operations by its flexible built, the placement optimization algorithm developed in this study should be reconfigured to meet the new requirements of the new condition. For example if rubber tyred cranes are being used as container handling machinery, the 3D placement algorithms and patterns should be reconfigured because they have been designed for those container terminals that use reach stackers for container operations.

5.2 Optimization and Placement Algorithms

The optimization algorithm is designed to consist of several placement algorithms specifically constructed for different area of the yard space. The difference of stacking criteria for each area in the yard avoids the usage of a single placement algorithm that covers every aspect of the container handling operations. For instance, the stacking procedure for an export line is widely different than the stacking procedure of an import line. Thus, the optimization algorithm delivers incoming container operation requests to the related placement algorithms in order to apply the suitable placement criteria for the requests.

The more flexible our system is the less standardization we can have in the container operations over the yard space. Standardization is another key factor that determines the efficiency of our algorithms. We tried to balance these two factors in this study by trying to keep the system as flexible as possible as well as making several assumptions from the terminal data analysis results to focus on the main container activities in the yard space in order have more stable placement algorithms.

5.3 Placement Patterns

Using of placement patterns for container handling focuses on finding suitable places for the large container groups at the first hand. There is not a big problem for a group of two or three containers to be scattered over the yard, however the same case can lead to disastrous results for a group of 15 containers for example. The system uses well known placement patterns and tries to reserve the spaces for the large groups by checking the available lines. Another point is that system does not have to calculate the best position for each container of the same group and just sends the containers to their reserved slots in the stack when they are handled.

5.4 Future Work

The placement patterns and the optimization algorithm can be integrated with the yard management system defined as an application to observe the real behavior of the system. In addition to this, the optimization algorithm and yard management system could be constructed so that it allows multiple machinery usage for container operations.

The yard management system could also be designed so that it does not only act as a decision support system for container operations; it also dynamically configures the yard layout and line structure of the yard as an expert system.

Different optimization approaches to container stacking algorithms can be introduced and the results of these different optimization algorithms can be compared to each other to define a general placement optimization algorithm which can adapt to any kind of yard configuration for each container handling machinery.

The study presented in this thesis is about the yard management system for a container terminal. However every area of operation in container terminals is a study of its own; such as resource allocation, berth planning, signaling, ship stowage etc. The software for these packages can be created and integrated in a single multi-functional software system that combines many aspects of terminal operations for the full automation of container terminal.

REFERENCES

[1] Agerschou, H., Lundgren, H., Sörensen, T., Ernst, T., Kosgaard, J., Schmidt, L.R., Chi, W.K. (1983), *Planning and Design of Ports and Marine Terminals*, John Wiley & Sons, Chichester.

[2] Rath, E. (1973), *Container Systems*, John Wiley & Sons, New York.

[3] Homepage van Iris en Kees Jan, http://www.ikj.nl/.

[4] Johansen, R.S. (1999), *Gate Solutions*, Containerport and Terminal Performance Conference, Amsterdam.

[5] Wan, T.B., Wah, E.L.C., Meng, L.C. (1992), *The Use of Information Technology by the Port of Singapore Authority*, World Development 20(12), 1785-1795.

[6] Leeper, J.H. (1988), *Integrated Automated Terminal Operations*, Transportation Research Cicular 33, 23-28.

[7] Van Horssen, W. (1996), *The Trouble with Growth*, Cargo Systems February, 51-52.

[8] Yenel, C. (1998), *Port Parameters and Optimization: Case Studies*, Master of Science Thesis, METU, Ankara.

[9] Mit, K.U. (1994), *a Computer Program for Testing Port Parameters and a Tool for Port Master Planning*, Master of Science Thesis, METU, Ankara.

[10] Güler, I. (1996), *a Computer Simulation Model for Seaport Planning*, Doctor of Philosophy Thesis, METU, Ankara.

[11] Bakalım, T. (1993), *Computer Simulation Model for a Container Terminal*, Master of Science Thesis, METU, Ankara.

[12] Yadipour, S. (1993), *a Computer Program for Harbor Layout Optimization*, Master of Science Thesis, METU, Ankara.

[13] Imai, A., Nagaiwa, K., Tat, C.W. (1997), *Efficient planning of berth allocation for container terminals in Asia*, Journal of Advanced Transportation 31(1), 75-94.

[14] Legato, P., Mazza, R.M. (2000), *Berth Planning and Resources Optimization at a Container Terminal via Discrete Event Simulation*, European Journal of Research 133, 537-547.

[15] Thurston, T., Hu, H. (2002), *Distributed Agent Architecture for Port Automation*. University of Essex, Colchester.

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

[17] Böse, J., Reiners, T., Steenken, D., Voß S. (2000), *Vehicle Dispatching at Seaport Container Terminals Using Evolutionary Algorithms*, Precedings of the 33th Hawaii International Conference on System Sciences.

[18] Powell, W.B., Carvalho, T.A. (1998), *Real-Time Optimization of Containers and Flatcars for Intermodal Operations*, Statistics and Operations Research Technical Report SOR-96-05, Princeton University, Princeton.

[19] Kosmatopoulos, E.B., Liu, C., Ioannou, P. (2002), *Design and Optimization of a Conceptual Automated Yard Using Overhead Grid Rail System*, University of Southern California, Los Angeles.

[20] Kim, K.H., Park, Y.M., Ryu, K.R. (1999), *Deriving Decision Rules to Locate Export Containers in Container Yards*, European Journal of Operations Research 124, 89-101.

[21] Zhang, C., Liu, J., Wan Y.W., Murty, K.G., Linn, R.J. (2002), *Storage Space Allocation in Container Terminals*, The Hong Kong University of Science and Technology, Kowloon.

[22] Henesey, L., Wernstedt, F., Davidson, P. (2002), *a Market Based Approach to Container Port Terminal Management*, Blekinge Institute of Technology, Ronneby.

[23] Rebollo, M., Julian, V., Carrascosa, C., Botti, V. (2000), a Multi-Agent System for the Automation of a Port Container Terminal, Universidad Politécnica de Valencia, Valencia.

[24] Murty, K.G., Liu, J., Wan, Y.W., Zhang, C., Tsang, M.C.L., Linn, R. (2000), *Decision Support Systems for Operations in a Container Shipping Terminal*, Univeristy of Michigan, Michigan.

[25] Bontempi, G., Gambardella, R.M., Rizzoli, A.E. (2000), *Simulation and Optimization for Management of Intermodal Terminals*, Istituto dalle Molle di Studi sull'Intelligenza Artificiale, Lugano.

[26] Dougherty, E. (1999), *Intermodal Intelligence*, August Design, Inc.

[27] Rumbaugh, J., Blaha, M., Premerlani, W., Eddy, F., Lorensen, W. (1991), *Object Oriented Modeling and Design*, Prentice-Hall, New Jersey.

[28] Bennett, S., Skelton, J., Lunn, K. (2001), *UML*, McGraw Hill, Vincenza.

[29] Bennett, S., McRobb, S., Farmer, R. (2002), *Object Oriented Analysis and Design Using UML*, McGraw Hill, Berkshire.

[30] Hillier, F.S., Lieberman, G.J. (1995), *Introduction to Operations Research*, McGraw Hill, Singapore.

[31] Winston, W.L. (1991), *Operations Research: Applications and Algorithms*, PWS-KENT Publishing Company, Boston.

[32] Taha, A.H. (1987), *Operations Research: an Introduction*, Macmillan Publishing Company, New York.

APPENDICES

APPENDIX A

SEQUENCE DIAGRAMS

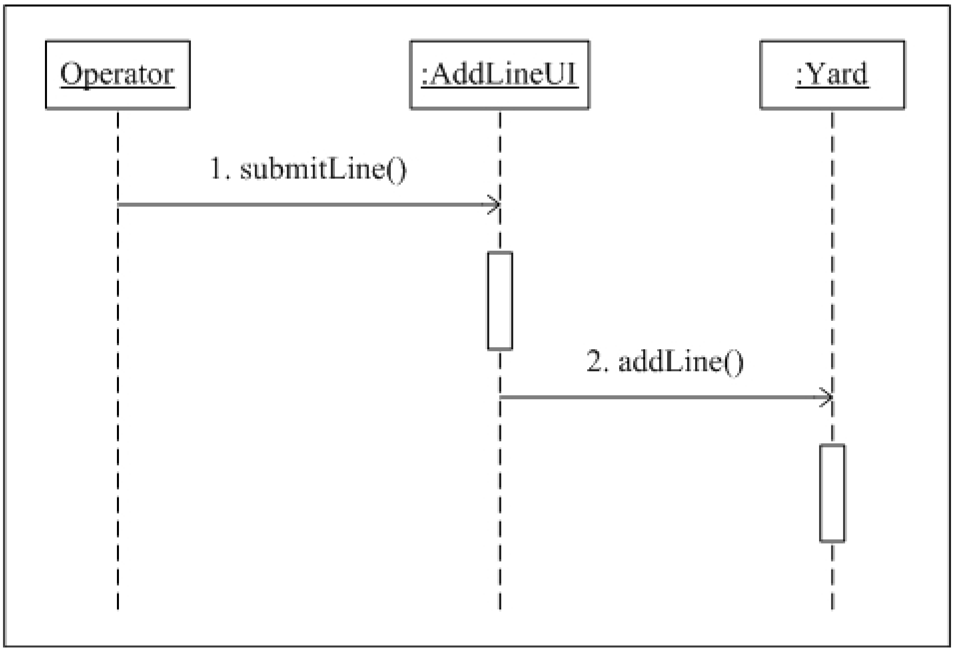


Fig. A.1. Sequence Diagram for the use-case Add Line

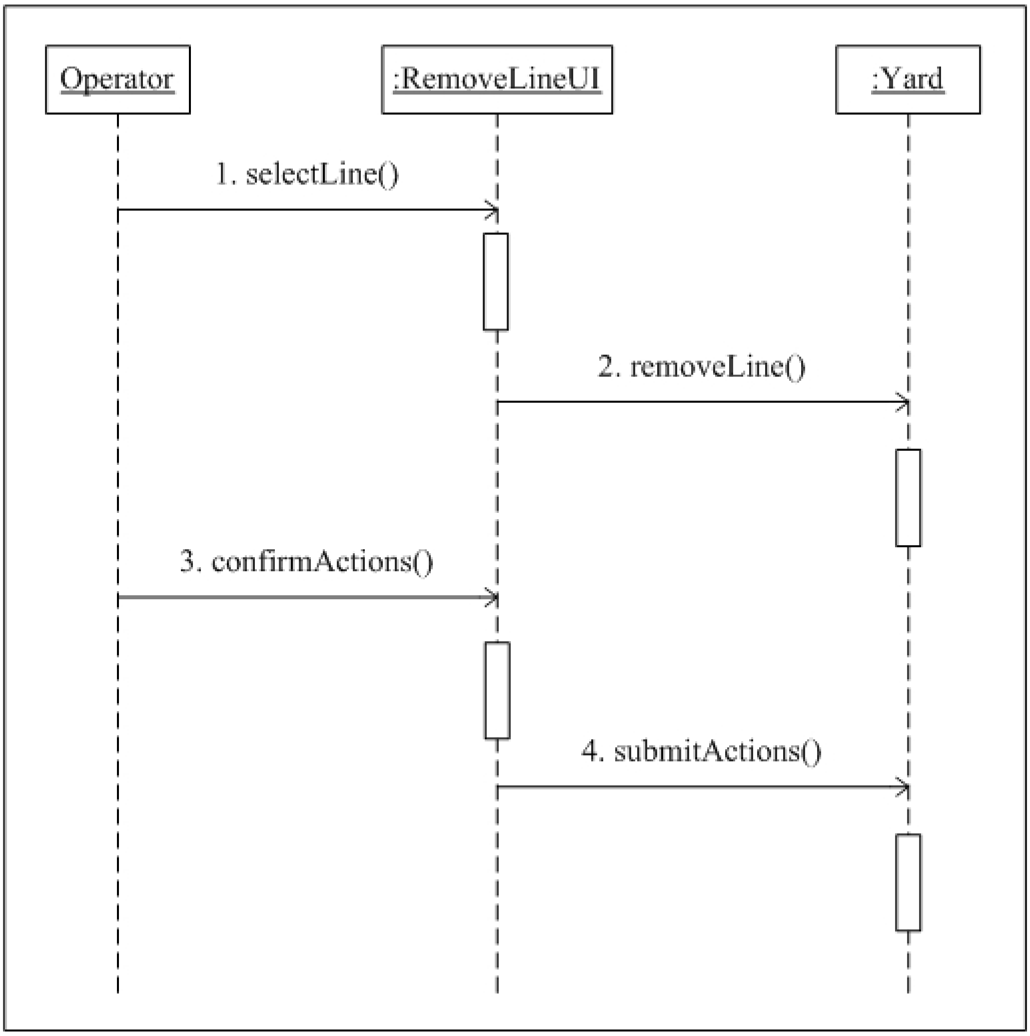


Fig. A.2. Sequence Diagram for the use-case Remove Line

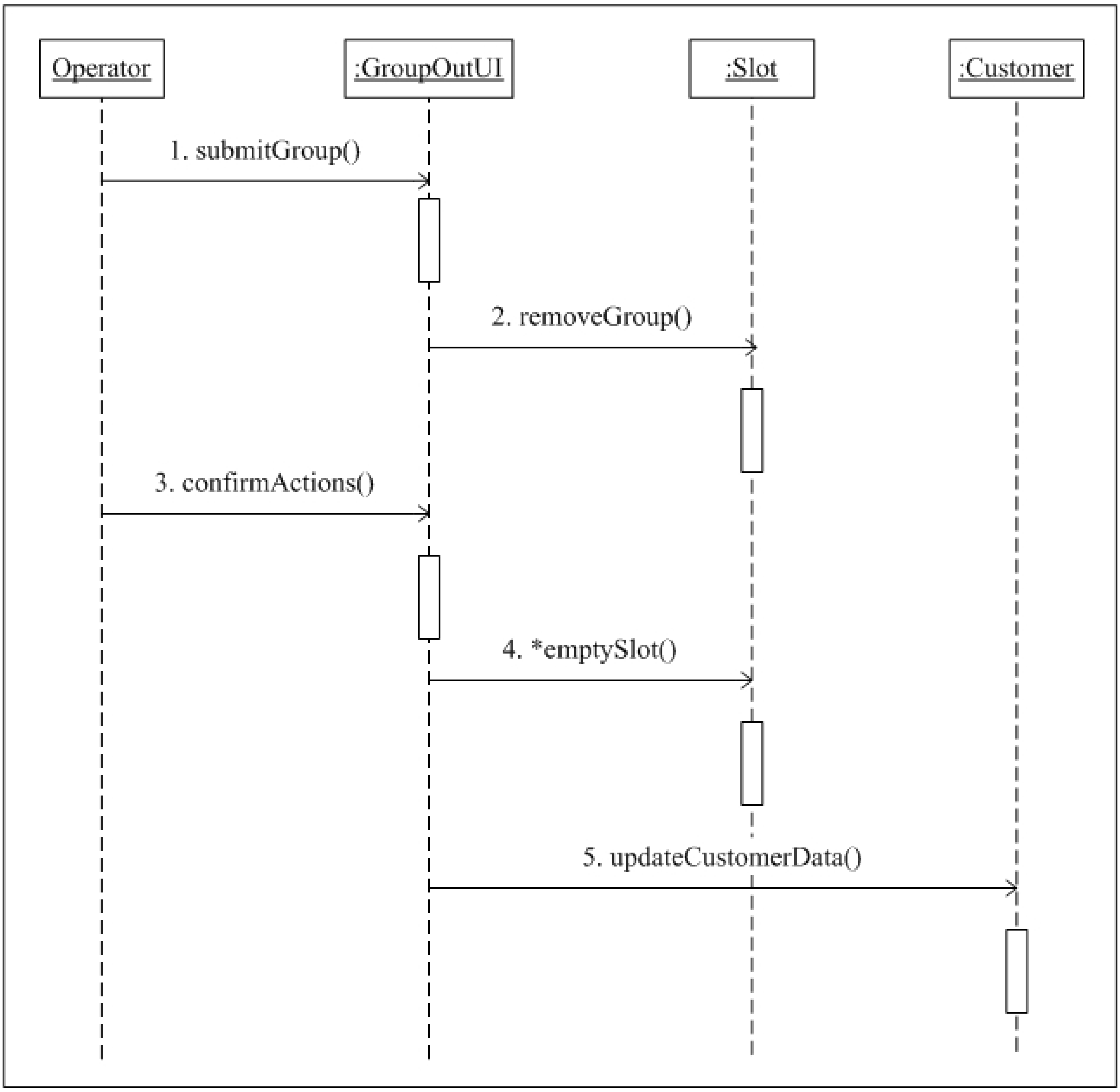


Fig. A.3. Sequence Diagram for the use-case Group Out

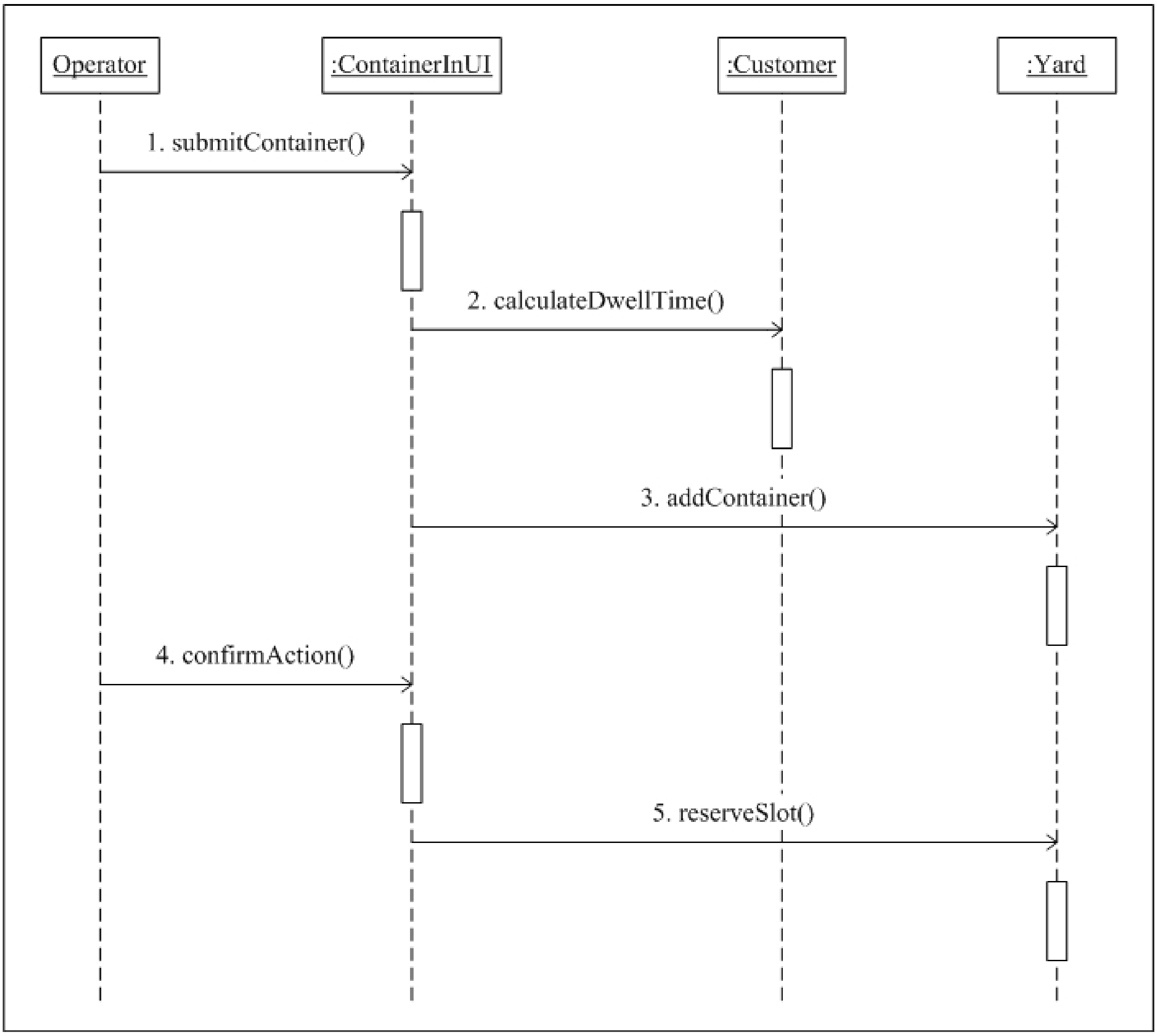


Fig. A.4. Sequence Diagram for the use-case Container In

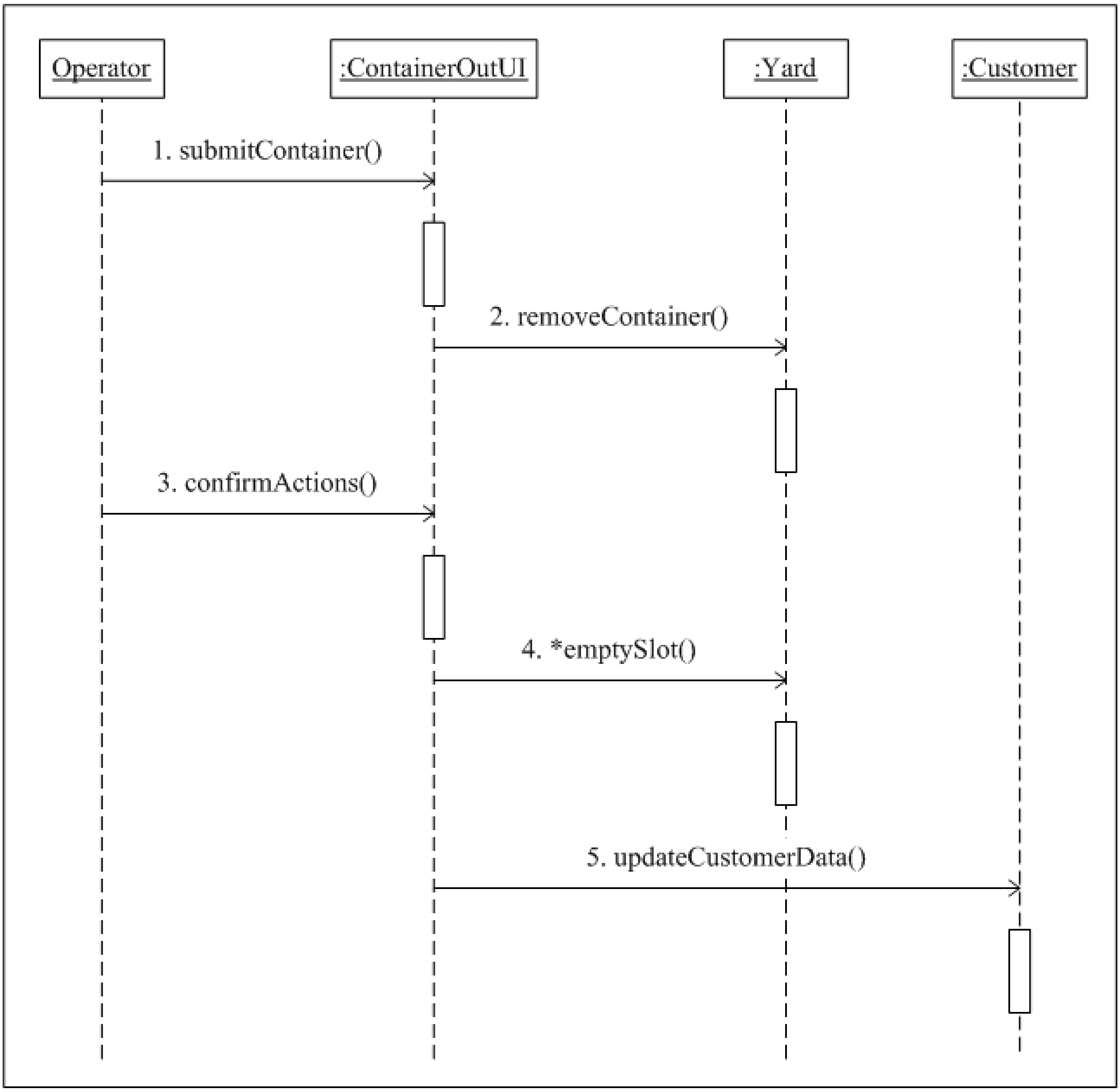


Fig. A.5. Sequence Diagram for the use-case Container Out

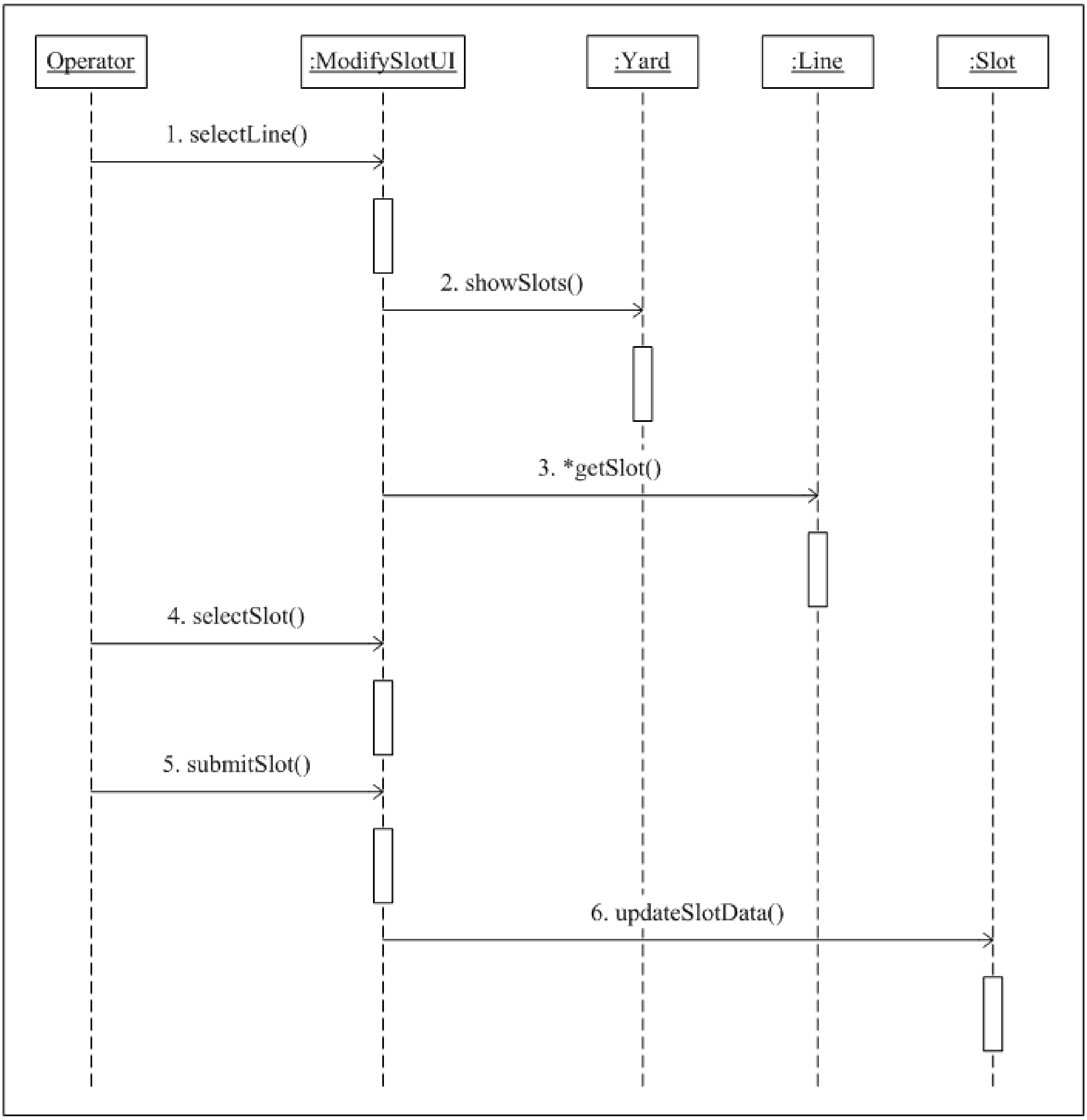


Fig. A.6. Sequence Diagram for the use-case Modify Slot

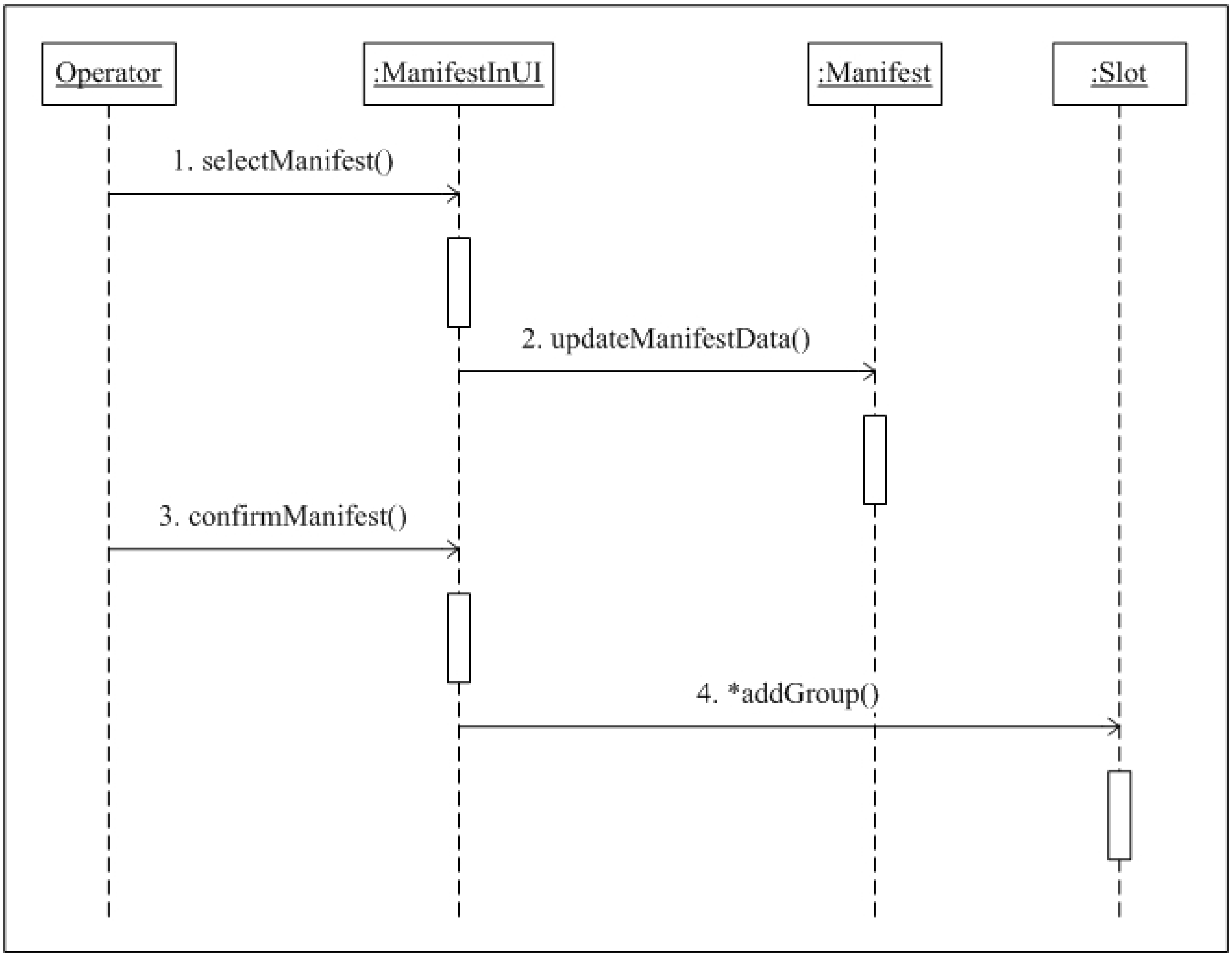
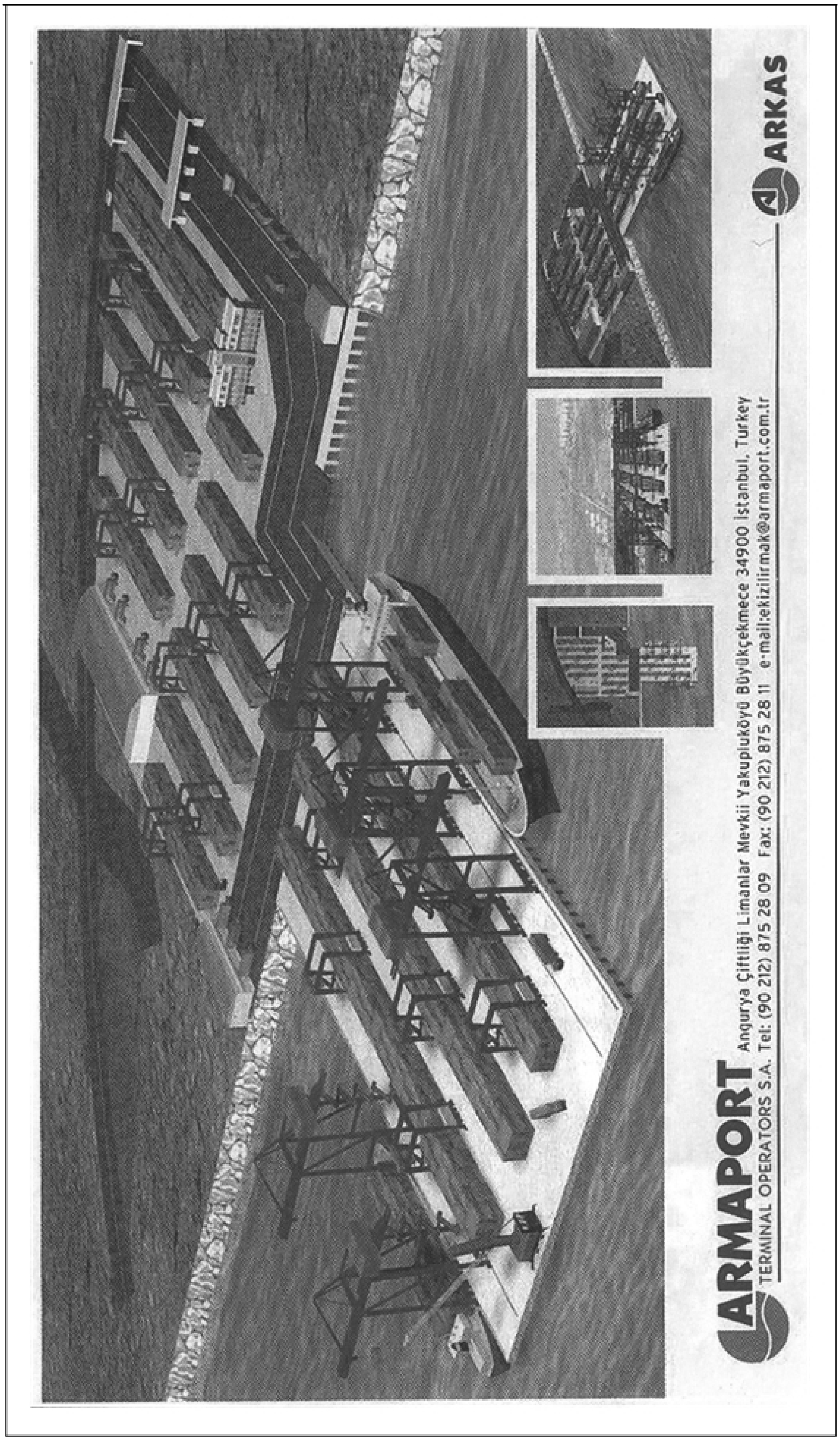


Fig. A.7. Sequence Diagram for the use-case Manifest In

APPENDIX B

TERMINAL FIGURES



ARMAPORT
TERMINAL OPERATORS S.A.
Angurya Çiftliği Limanlar Mevkii Yakuplukoyü Büyükçekmece 34900 İstanbul, Turkey
Tel: (90 212) 875 28 09 Fax: (90 212) 875 28 11 e-mail:ekizilirmak@armaport.com.tr

ARKAS

Fig. B.1. A futuristic view of Armaport Container Terminal

Fig. B.2. A Quay Crane

Fig. B.3. A Quay Crane and Ship Crane at operation

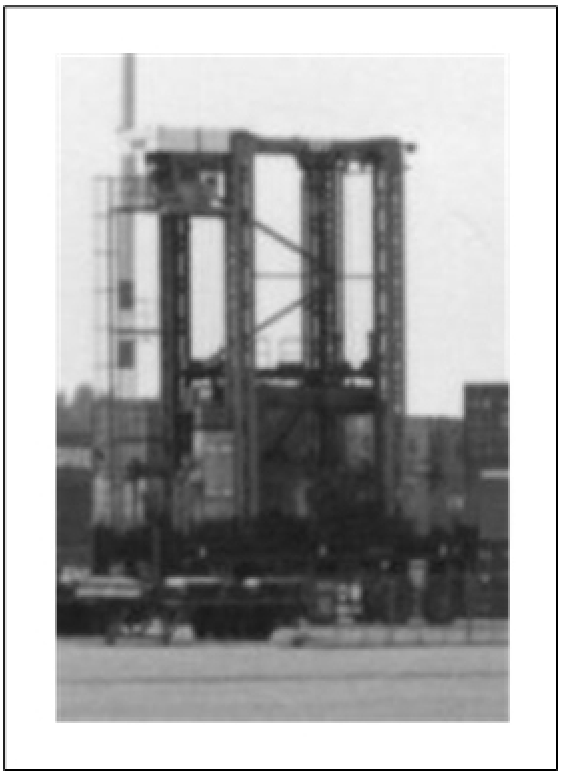


Fig. B.4. A Straddle Carrier

Fig. B.5. Forklift Trucks at operation

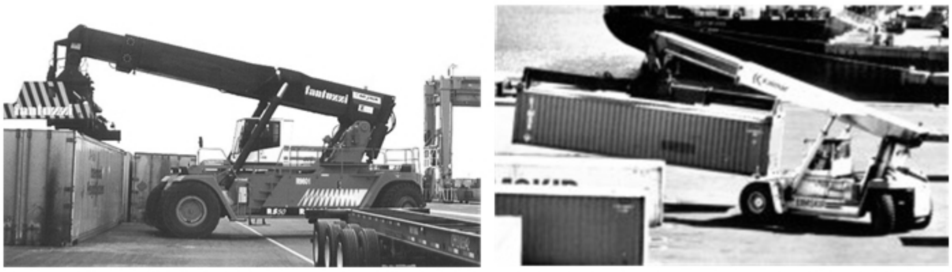
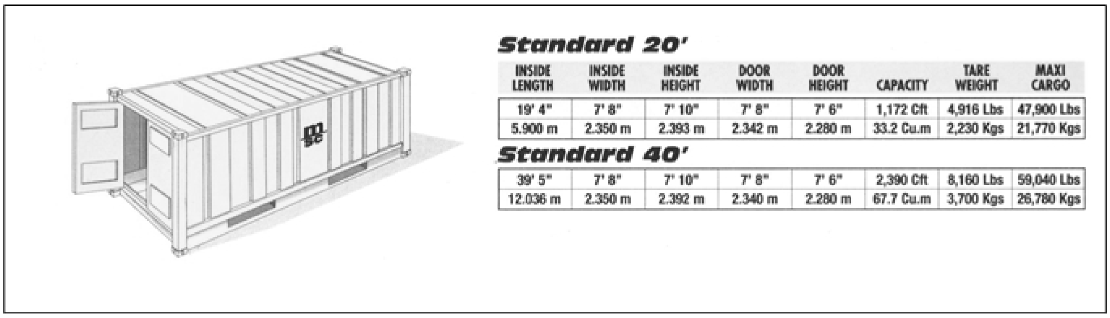


Fig. B.6. Reach Stackers at operation

Fig. B.7. Rubber Tyred Cranes at operation

Fig. B.8. A Container Truck at operation



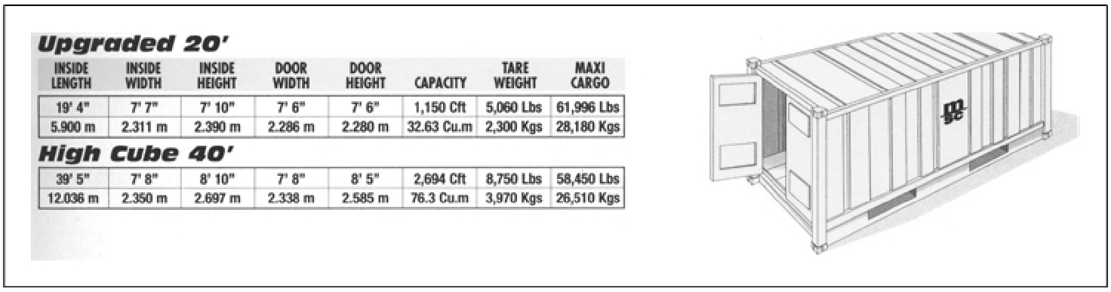
Standard 20'

INSIDE LENGTH	INSIDE WIDTH	INSIDE HEIGHT	DOOR WIDTH	DOOR HEIGHT	CAPACITY	TARE WEIGHT	MAXI CARGO
19' 4"	7' 8"	7' 10"	7' 8"	7' 6"	1,172 Cft	4,916 Lbs	47,900 Lbs
5.900 m	2.350 m	2.393 m	2.342 m	2.280 m	33.2 Cu.m	2,230 Kgs	21,770 Kgs

Standard 40'

39' 5"	7' 8"	7' 10"	7' 8"	7' 6"	2,390 Cft	8,160 Lbs	59,040 Lbs
12.036 m	2.350 m	2.392 m	2.340 m	2.280 m	67.7 Cu.m	3,700 Kgs	26,780 Kgs

Fig. B.9. Standard Container



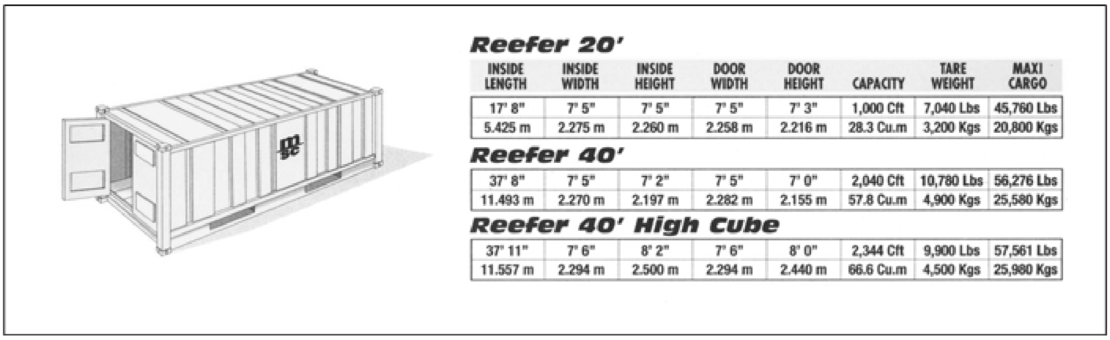
Upgraded 20'

INSIDE LENGTH	INSIDE WIDTH	INSIDE HEIGHT	DOOR WIDTH	DOOR HEIGHT	CAPACITY	TARE WEIGHT	MAXI CARGO
19' 4"	7' 7"	7' 10"	7' 6"	7' 6"	1,150 Cft	5,060 Lbs	61,996 Lbs
5.900 m	2.311 m	2.390 m	2.286 m	2.280 m	32.63 Cu.m	2,300 Kgs	28,180 Kgs

High Cube 40'

39' 5"	7' 8"	8' 10"	7' 8"	8' 5"	2,694 Cft	8,750 Lbs	58,450 Lbs
12.036 m	2.350 m	2.697 m	2.338 m	2.585 m	76.3 Cu.m	3,970 Kgs	26,510 Kgs

Fig. B.10. Upgraded and High Cube Containers



Reefer 20'

INSIDE LENGTH	INSIDE WIDTH	INSIDE HEIGHT	DOOR WIDTH	DOOR HEIGHT	CAPACITY	TARE WEIGHT	MAXI CARGO
17' 8"	7' 5"	7' 5"	7' 5"	7' 3"	1,000 Cft	7,040 Lbs	45,760 Lbs
5.425 m	2.275 m	2.260 m	2.258 m	2.216 m	28.3 Cu.m	3,200 Kgs	20,800 Kgs

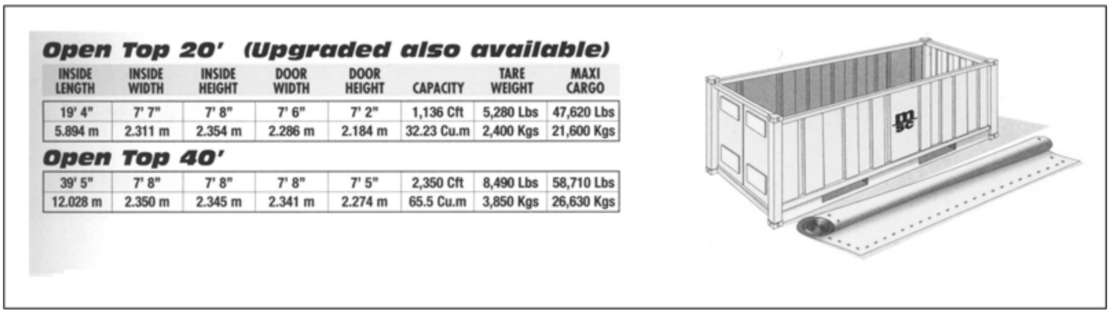
Reefer 40'

37' 8"	7' 5"	7' 2"	7' 5"	7' 0"	2,040 Cft	10,780 Lbs	56,276 Lbs
11.493 m	2.270 m	2.197 m	2.282 m	2.155 m	57.8 Cu.m	4,900 Kgs	25,580 Kgs

Reefer 40' High Cube

37' 11"	7' 6"	8' 2"	7' 6"	8' 0"	2,344 Cft	9,900 Lbs	57,561 Lbs
11.557 m	2.294 m	2.500 m	2.294 m	2.440 m	66.6 Cu.m	4,500 Kgs	25,980 Kgs

Fig. B.11. Reefer and Reefer High Cube Containers



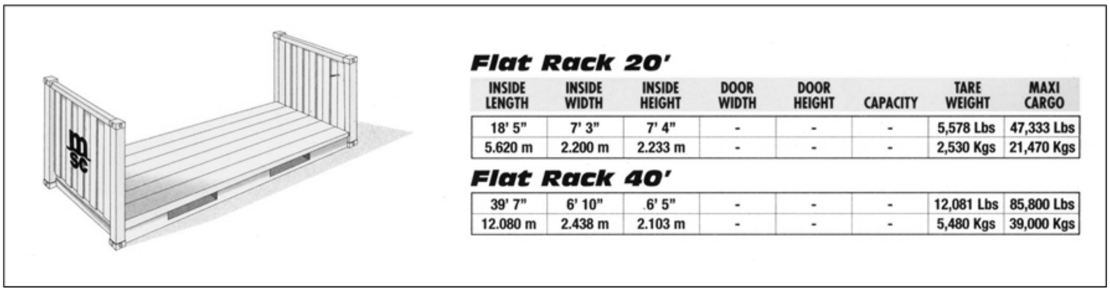
Open Top 20' (Upgraded also available)

INSIDE LENGTH	INSIDE WIDTH	INSIDE HEIGHT	DOOR WIDTH	DOOR HEIGHT	CAPACITY	TARE WEIGHT	MAXI CARGO
19' 4"	7' 7"	7' 8"	7' 6"	7' 2"	1,136 Cft	5,280 Lbs	47,620 Lbs
5.894 m	2.311 m	2.354 m	2.286 m	2.184 m	32.23 Cu.m	2,400 Kgs	21,600 Kgs

Open Top 40'

39' 5"	7' 8"	7' 8"	7' 8"	7' 5"	2,350 Cft	8,490 Lbs	58,710 Lbs
12.028 m	2.350 m	2.345 m	2.341 m	2.274 m	65.5 Cu.m	3,850 Kgs	26,630 Kgs

Fig. B.12. Open Top Container



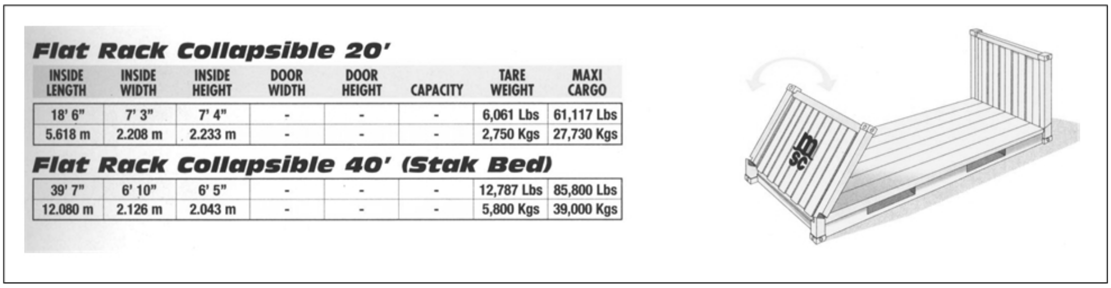
Flat Rack 20'

INSIDE LENGTH	INSIDE WIDTH	INSIDE HEIGHT	DOOR WIDTH	DOOR HEIGHT	CAPACITY	TARE WEIGHT	MAXI CARGO
18' 5"	7' 3"	7' 4"	-	-	-	5,578 Lbs	47,333 Lbs
5.620 m	2.200 m	2.233 m	-	-	-	2,530 Kgs	21,470 Kgs

Flat Rack 40'

INSIDE LENGTH	INSIDE WIDTH	INSIDE HEIGHT	DOOR WIDTH	DOOR HEIGHT	CAPACITY	TARE WEIGHT	MAXI CARGO
39' 7"	6' 10"	6' 5"	-	-	-	12,081 Lbs	85,800 Lbs
12.080 m	2.438 m	2.103 m	-	-	-	5,480 Kgs	39,000 Kgs

Fig. B.13. Flat Rack



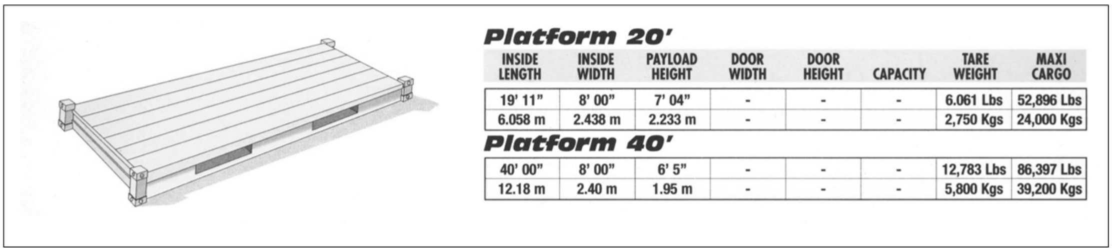
Flat Rack Collapsible 20'

INSIDE LENGTH	INSIDE WIDTH	INSIDE HEIGHT	DOOR WIDTH	DOOR HEIGHT	CAPACITY	TARE WEIGHT	MAXI CARGO
18' 6"	7' 3"	7' 4"	-	-	-	6,061 Lbs	61,117 Lbs
5.618 m	2.208 m	2.233 m	-	-	-	2,750 Kgs	27,730 Kgs

Flat Rack Collapsible 40' (Stak Bed)

INSIDE LENGTH	INSIDE WIDTH	INSIDE HEIGHT	DOOR WIDTH	DOOR HEIGHT	CAPACITY	TARE WEIGHT	MAXI CARGO
39' 7"	6' 10"	6' 5"	-	-	-	12,787 Lbs	85,800 Lbs
12.080 m	2.126 m	2.043 m	-	-	-	5,800 Kgs	39,000 Kgs

Fig. B.14. Flat Rack Collapsible



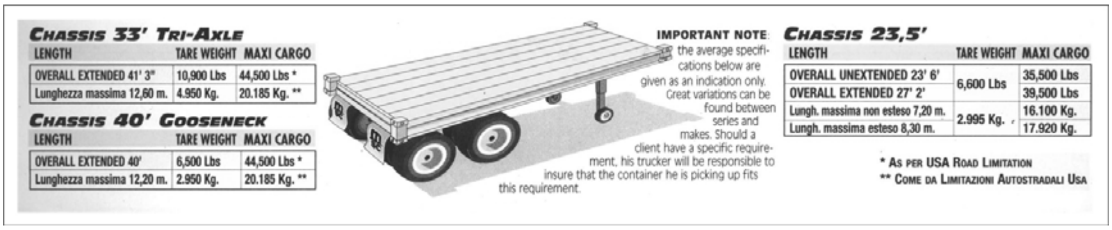
Platform 20'

INSIDE LENGTH	INSIDE WIDTH	PAYLOAD HEIGHT	DOOR WIDTH	DOOR HEIGHT	CAPACITY	TARE WEIGHT	MAXI CARGO
19' 11"	8' 00"	7' 04"	-	-	-	6.061 Lbs	52,896 Lbs
6.058 m	2.438 m	2.233 m	-	-	-	2,750 Kgs	24,000 Kgs

Platform 40'

INSIDE LENGTH	INSIDE WIDTH	PAYLOAD HEIGHT	DOOR WIDTH	DOOR HEIGHT	CAPACITY	TARE WEIGHT	MAXI CARGO
40' 00"	8' 00"	6' 5"	-	-	-	12,783 Lbs	86,397 Lbs
12.18 m	2.40 m	1.95 m	-	-	-	5,800 Kgs	39,200 Kgs

Fig. B.15. Platform



CHASSIS 33' TRI-AXLE

LENGTH	TARE WEIGHT	MAXI CARGO
OVERALL EXTENDED 41' 3"	10,900 Lbs	44,500 Lbs *
Lunghezza massima 12,60 m.	4.950 Kg.	20.185 Kg. **

CHASSIS 40' GOOSENECK

LENGTH	TARE WEIGHT	MAXI CARGO
OVERALL EXTENDED 40'	6,500 Lbs	44,500 Lbs *
Lunghezza massima 12,20 m.	2.950 Kg.	20.185 Kg. **

IMPORTANT NOTE: the average specifications below are given as an indication only. Great variations can be found between series and makes. Should a client have a specific requirement, his trucker will be responsible to insure that the container he is picking up fits this requirement.

CHASSIS 23,5'

LENGTH	TARE WEIGHT	MAXI CARGO
OVERALL UNEXTENDED 23' 6'	6,600 Lbs	35,500 Lbs
OVERALL EXTENDED 27' 2'		39,500 Lbs
Lungh. massima non esteso 7,20 m.	2.995 Kg.	16.100 Kg.
Lungh. massima esteso 8,30 m.		17.920 Kg.

* AS PER USA ROAD LIMITATION

** COME DA LIMITAZIONI AUTOSTRADALI USA

Fig. B.16. Tri-Axle and Gooseneck