A SYSTEM DYNAMICS SIMULATION APPROACH TO CONTAINER TERMINAL MANAGEMENT

CHENG JACK KIE

DOCTOR OF PHILOSOPHY

UNIVERSITI UTARA MALAYSIA

JUNE 2010

Kolej Sastera dan Sains
(UUM College of Arts and Sciences)
Universiti Utara Malaysia

PERAKUAN KERJA TESIS / DISERTASI
(Certification of thesis / dissertation)

Kami, yang bertandatangan, memperakukan bahawa
(We, the undersigned, certify that)

CHENG JACK KIE

calon untuk Ijazah **PhD**
(candidate for the degree of)

telah mengemukakan tesis / disertasi yang bertajuk:
(has presented his/her thesis / dissertation of the following title):

"A SYSTEM DYNAMICS SIMULATION APPROACH TO CONTAINER TERMINAL MANAGEMENT"

seperti yang tercatat di muka surat tajuk dan kulit tesis / disertasi.
(as it appears on the title page and front cover of the thesis / dissertation).

Bahawa tesis/disertasi tersebut boleh diterima dari segi bentuk serta kandungan dan meliputi bidang ilmu dengan memuaskan, sebagaimana yang ditunjukkan oleh calon dalam ujian lisan yang diadakan pada : ***21 Januari 2010.***
That the said thesis/dissertation is acceptable in form and content and displays a satisfactory knowledge of the field of study as demonstrated by the candidate through an oral examination held on:
January 21, 2010.

Pengerusi Viva: *(Chairman for Viva)*	**Prof. Dr. Mohd. Zaini Abdul Karim**	Tandatangan *(Signature)*
Pemeriksa Luar: *(External Examiner)*	**Assoc. Prof. Dr. Abas Md Said**	Tandatangan *(Signature)*
Pemeriksa *Dalam*: *(Internal Examiner)*	**Assoc. Prof. Dr. Engku Muhammad Nazri Engku Abu Bakar**	Tandatangan *(Signature)*
Nama Penyelia/Penyelia-penyelia: *(Name of Supervisor/Supervisors)*	**Assoc. Prof. Dr. Ang Chooi Leng**	Tandatangan *(Signature)*
Nama Penyelia/Penyelia-penyelia: *(Name of Supervisor/Supervisors)*	**Prof. Dr. Razman Mat Tahar**	Tandatangan *(Signature)*

Tarikh:

(Date) ***January 21, 2010***

A SYSTEM DYNAMICS SIMULATION APPROACH TO CONTAINER TERMINAL MANAGEMENT

A Thesis submitted to the College of Arts and Sciences in full fulfillment of the requirements for the degree of the Doctor Philosophy Universiti Utara Malaysia

By

Cheng Jack Kie

PERMISSION TO USE

In presenting this thesis in fulfillment of the requirement for a postgraduate degree from Universiti Utara Malaysia, I agree that the University Library may make it freely available for inspection. I further agree that permission for copying of this thesis in any manner, in whole or in part, for scholarly purpose may be granted by my supervisor(s) or, in their absence by the Dean of Research and Graduate Studies. It is understood that any copying or publication or use of this thesis or parts thereof for financial gain shall not be allowed without my written permission. It also understood that due recognition shall be given to me and to Universiti Utara Malaysia for any scholarly use which may be made of any material from my thesis.

Requests for permission to copy or to make other use of materials in this thesis, in whole or in part, should be addressed to

Dean of Research and Graduate Studies
College of Arts and Sciences
Universiti Utara Malaysia
06010 UUM Sintok
Kedah Darul Aman

ABSTRAK

Terminal kontena beroperasi di persekitaran yang dinamik dan penuh persaingan di mana setiap terminal kontena sentiasa mencari jalan untuk meningkatkan daya saing masing-masing. Salah satu daya saing adalah kebolehan untuk mengendalikan kapal kontena dalam masa yang tersingkat. Tetapi, kebolehan ini amat bergantung kepada keefisienan seluruh operasi terminal kontena itu sendiri. Pengurusan, pembuatan keputusan serta operasi dermaga dan gudang penyimpanan sementara yang cekap adalah sangat penting untuk memastikan keseluruhan terminal kontena beroperasi dengan efisien. Operasi dalaman sesebuah terminal kontena adalah sangat komplex dan rumit, ini menyebabkan perancangan dan pengurusan dermaga serta gudang penyimpanan sementara adalah mencabar. Penyelidikan ini mengaplikasikan kaedah simulasi sistem dinamik untuk memodelkan hubungan serta interaksi di antara operasi di dermaga dengan operasi di gedung penyimpanan sementara. Daripada model sistem dinamik ini, didapati faktor kelajuan kren dermaga memindahkan kontena dan jarak perjalanan *Prime Movers* di antara dermaga dan gudang penyimpanan sementara memainkan peranan yang penting ke atas kadar penggunaan dermaga. Selain daripada itu, model sistem dinamik ini juga boleh digunakan dalam pengurusan kapasiti menerusi experimentasi seperti menguji apakah impak terhadap operasi terminal kontena jika berlakunya pertambahan pada jumlah kontena yang dikendalikan, bilangan kapal yang berlabuh serta peningkatan pada saiz kapal kontena. Penyelidikan ini menyumbang dalam menghubungkan jurang di antara literatur melalui pembinaan sebuah model yang berupaya untuk memodelkan hubungan dan interaksi di antara operasi di dermaga dan operasi di gedung penyimpanan sementara; dan pada masa yang sama berupaya untuk menggabungkan isu-isu di peringkat operasi dan strategik. Selain daripada itu, penyelidikan ini juga memanfaatkan pihak pengurus terminal kontena menerusi pembinaan *Microworlds*. *Microworlds* berupaya untuk membantu pengurus terminal kontena dalam aspek pengurusan dan pembuatan keputusan serta berfungsi sebagai alat pembelajaran di mana pengurus terminal kontena boleh mendalami serta memahami kekompleksitian operasi dalaman sesebuah terminal kontena.

ABSTRACT

The container terminal operates under a competitive and dynamic environment where every container terminal continuously seeks to secure a competitive advantage. One of the competitive advantages is the ability to turnaround vessels within the shortest time period. However, this ability very much depends on the overall efficiency of the container terminal operations itself. The planning, decision making and operation of the berth and container yard are crucial in order to ensure the whole container terminal operates in an efficient and timely manner. However, due to the complexity of the container terminal operation, decision making and planning in the berth and yard subsystems are very challenging. This research presents the application of system dynamics simulation into capturing the relationship and interdependency between the berth and yard operation. The system dynamics model reveals that both quay crane moves and prime mover traveling distances have an impact on the berth occupancy rate. Besides that, the system dynamics model also provides capacity planning by allowing the experimentation of the impact on the increase in container throughput, vessel arrival and vessel size on the container terminal operation. This research contributes at bridging the gap between the literatures by developing a model that is capable of capturing the relationship and interdependency between the berth and yard operation as well as incorporating both operational and strategic level issues at the container terminal. This research also benefits the container terminal management through the development of Microworlds. Microworlds is capable of aiding terminal managers on planning and decision making as well as serving as a learning tool where the managers can gain insight to the complexity of the terminal operations.

ACKNOWLEDGMENTS

This work simply could not have been possible without the assistance and encouragement from many others. First and foremost, I would like to thank two most important people that make this thesis possible. They are none other than my supervisors, Professor Dr. Razman Mat Tahar and Associate Professor Dr. Ang Chooi Leng. I sincerely thank both of them for their mentor and guidance throughout the journey of my studies.

I would like to acknowledge the support from the officers at the case study container terminal, especially Mr. Arulmaran Ramasamy who willing to spare his valuable time to assists me throughout my two-weeks stay at the terminal. I am also grateful to all the officers from the operational team who spare their time in explaining the operational processes of the case study container terminal, especially to Mr. Chua Cheng Hock who spend numerous hours organizing the data for me.

I am also grateful to the management of the Physical Sciences Division for providing a comfortable working space for the postgraduate students. Besides that, I would also like to extend my gratitude to my fellow colleagues for being with me throughout the ups and downs in these four years. To Kak Nor, we finally made it, all our sweat and tears finally paid off and to Chia Yin, you can do it too. Also to my fellow colleagues who are still pursuing their PhD, I believe that all of you will make it as well.

A special thank to my dad, mum and brother for their unwarranted belief in me and for putting up with my constant emotional rollercoaster. Also to all my friends who supported me along the way and showered me with words of encouragement, thank you so much.

TABLE OF CONTENTS

CHAPTER FOUR: SITE DESCRIPTION

CHAPTER FIVE: MODEL CONSTRUCTION

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 Introduction

This chapter starts with the introduction on the current phenomena of the global container terminal industry; followed by a brief review on the history of containers. Problems faced by modern container terminals are presented next followed by a discussion on the container terminal industry in Malaysia as well as the background of the case study container terminal. The major motivation on why this research was conducted and the objectives of this research are presented subsequently. The choice of method used to conduct this research and the assumptions of the developed model are also discussed in detail in the later section of this chapter. This chapter finally briefly summarizes the organization of this thesis.

1.2 Introduction to Container Terminal

The market environment in which container terminals operate is changing rapidly due to globalization and the adoption of containerization since late 1960's. Container terminals have evolved from being simply loading and unloading points to serving as crucial hubs in an industrial center. Today, a container terminal acts as an interface between production and consumption centers, eliminating the discontinuity between sea and land transport (Moglia and Sanguineri, 2003), thus integrating the entire supply chain.

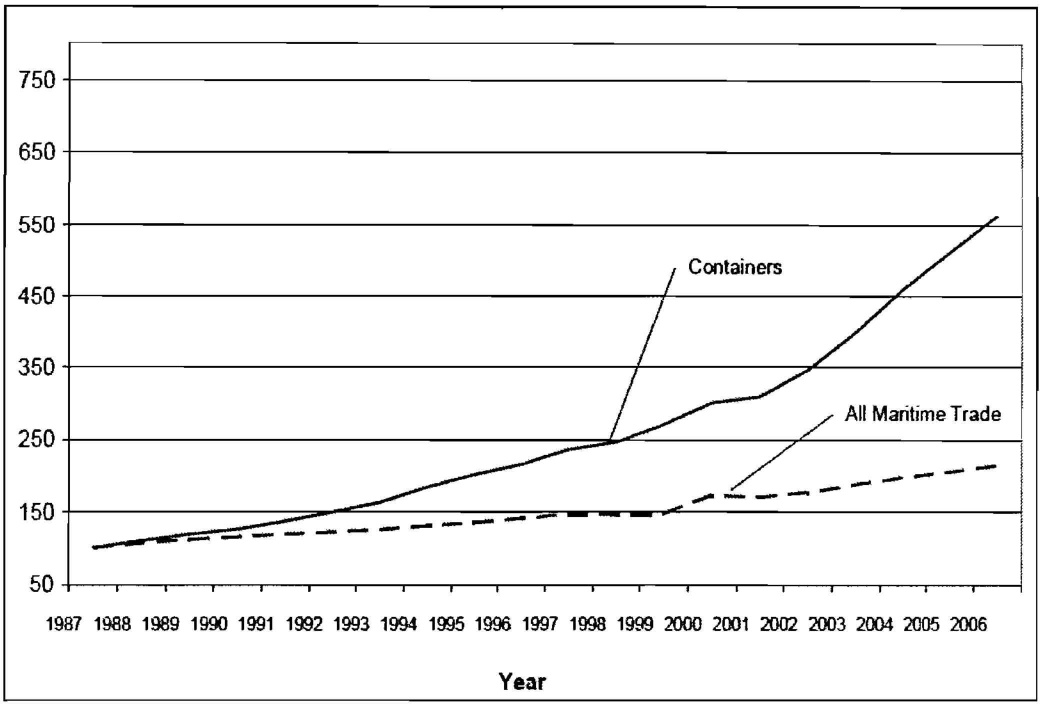


Figure 1.1: Growth of world maritime trade (1987-2006) (INDEX: 1987=100) (UNESCAP, 2007).

The usage of containers has gained enormous importance in worldwide trade and transportation of goods due to the increase in consumer goods. Figure 1.1 exhibits the positive growth of the world maritime trade from the year 1987 to 2006. With the increase in trade volumes as well as the effect of deregulation and globalization, containerized shipping is expected to increase in importance along the transport chains and therefore become the backbone of international trade (Schinas and Papadimitriou, 2003).

Figure 1.2 further justifies the importance of the container trade by exhibiting the forecasted trend of global container volume until the year 2015. The total number of full containers shipped internationally is expected to grow to 235.7 million TEUs (Twenty-Feet-Equivalent-Unit) by 2015, from 113.6 million TEUs in 2005 (the base year for cargo forecast) and the compound growth rate during the period of 2005 to 2015 is expected to be 7.6 per cent per annum, decreasing from 9.0 per cent per annum in the period of 1980 to 2005 (UNESCAP, 2007).

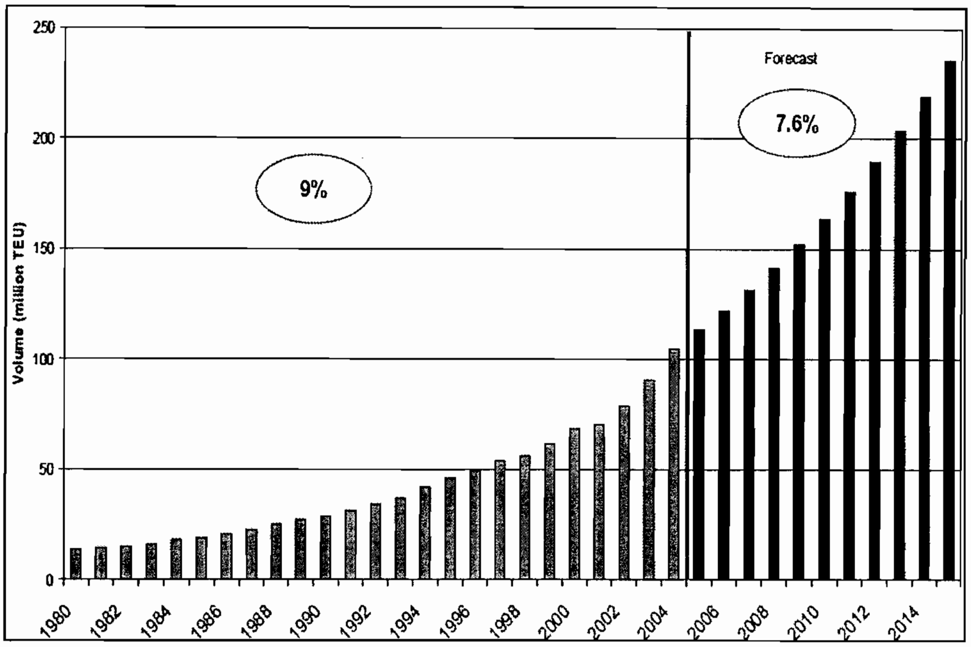


Figure 1.2: Past and forecast global container volume (1980-2015) (UNESCAP, 2007).

As the growth of global container volume becomes evident, the role of a container terminal, being the node that facilitates containers transfer becomes more prominent. Table 1.1 unveils the ranking of world ports in terms of container traffic for the year 2007 where Singapore Port and Shanghai Port both ranked first and second respectively while Malaysia's Port Klang and Port of Tanjung Pelepas both ranked 16th and 18th respectively (UNCTAD, 2008).

Table 1.1: Top 20 world port ranking in terms of container traffic for the year 2007 (UNCTAD, 2008).

RANK	PORT	COUNTRY	TEUs (000s)
1	Singapore	Singapore	27,932
2	Shanghai	China	26,150
3	Hong Kong	China	23,881
4	Shenzhen	China	21,099
5	Busan	South Korea	13,270
6	Rotterdam	Netherlands	10,790
7	Dubai	United Arab Emirates	10,653
8	Kaohsiung	Taiwan	10,256
9	Hamburg	Germany	9,900
10	Qingdao	China	9,462
11	Ningbo	China	9,360
12	Guangzhou	China	9,200
13	Los Angeles	United States	8,355
14	Antwerp	Belgium	8,176
15	Long Beach	United States	7,312
16	*Port Klang*	*Malaysia*	*7,120*
17	Tianjin	China	7,103
18	*Tanjung Pelepas*	*Malaysia*	*5,500*
19	New York/New Jersey	United States	5,400
20	Bremen/Bremerhaven	Germany	4,892

1.3 History of Containers

On 26 April 1956, in Port Newark, New Jersey, Malcom P. McLean, an entrepreneur, loaded a partially converted World War II vintage T-2 tanker *Ideal X* with fifty eight 35-foot containers (Kendall, 1986). The tanker arrived in Houston, six days later where the containers were off-loaded and hauled away by waiting trucks (The Economist, 2006). Thus, the era of 'container revolution' began (Kendall, 1986) and container shipping eventually replaced the traditional "break-bulk" method of handling crates, barrels and bags, and stowing them loose in a ship's hold.

Containers are actually uniform boxes that are used to transport goods from one destination to another. Standardization of the container boxes leads to fast and easy handling of goods, simplified scheduling and controlling while giving protection against weather and pilferage (Steeken et al., 2004). The boxes are so designed so that they can be rapidly shifted between modal carriers without having their contents unpacked and repacked. The term Twenty-Feet-Equivalent-Unit (TEU) is used to refer to standard containers with the length of twenty feet. A container of 40 feet is expressed by 2 TEUs.

McLean's conception of an integrated transportation system has therefore dramatically reduced shipping costs, reinvigorated markets and fuelled the world economy (Raine, 2006). With the significant drop in the cost of transportation, containerization was able to transform global transportation, commerce and manufacturing (Donovan, 2004). Meersmans and Dekker (2001) also expressed that globalization would have been impossible without containers.

1.4 Problems Faced by Modern Container Terminal

The port industry has always been competitive and every port seeks to secure a competitive advantage as any cost advantage or service efficiency improvement could be translated into a greater market share. However, modern container terminals are subjected to pressure from numerous external developments which increase the uncertainty (Slack, 2001) and complexity of the industry itself. Hence, only the best container terminals can survive while the others may face the risk of extinction.

In order to take advantage of the positive growth of the global container trade, global liner operators keep introducing larger vessels that are capable of carrying more containers and at the same time, achieving economies of scale. Figure 1.3 shows the past and forecasted increase in the size of containerships. Currently, most of the vessels in service are with capacities of above 7,000 TEUs, and according to The Economist (2006), the trend of increasing vessel size is expected to continue up to "Malacca-Max" which has the capacity of 18,000 TEUs, the maximum size of a vessel passing through the Straits of Malacca, the shipping lane between Malaysia and Indonesia.

The introduction of larger vessels has put an enormous pressure on the container terminal as a larger vessel requires a deeper draft, more equipment to unload and load the containers from the vessel and also a larger container yard for temporary storage of the containers. Without adequate capacity to accommodate the mega vessels, the huge number of containers discharged from these vessels will create congestion in the container terminal especially at the container yard as there is very limited space left to store the containers.

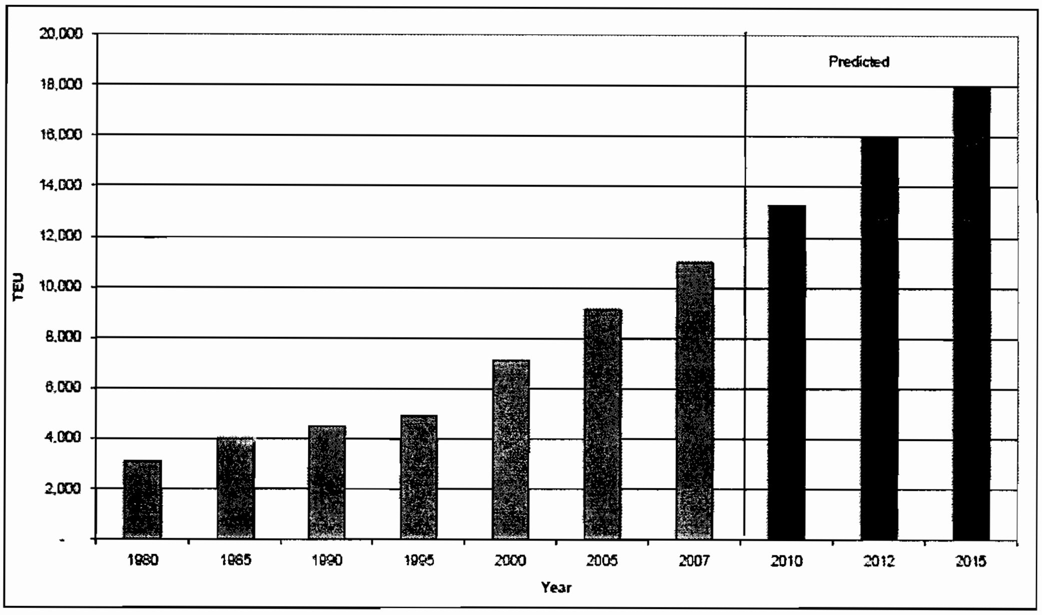


Figure 1.3: Increase in containership size (1980-2015) (UNESCAP, 2007).

The over congested yard will disrupt the overall terminal operations, thus reducing terminal efficiency and attractiveness. Besides that, insufficient berth space also will lead to a longer queue of vessels waiting to be served, thus increasing the vessel turnaround time. From the perspective of a liner operator, it is of paramount importance that the vessel turnaround time (duration spent in the container terminal) is minimized to save on the terminal costs.

In order to attract and serve these gigantic vessels as well as maintain current customers, the container terminal must either expand its facilities or improve the efficiency of the current terminal operations. However, investment on terminal expansion and on new infrastructures is not only expensive but also requires detailed planning and forecasting of future container growth. This is because building and purchasing new facilities without first understanding the terminal's current and future container growth is risky and it may damage the position of existing facilities by generating a significant surplus capacity.

Modification of existing terminals may also result in lengthy interruptions of service as these interruptions can result in intangible costs such as decreased productivity and customer dissatisfaction (Carpenter and Ward, 1990). Moreover, these mega vessels do not guarantee a visit even if the terminal is well equipped with state-of-the-art superstructures.

Congestion problem can seriously affect the terminal operations and damage the terminal reputation if the terminal fails to deliver its service efficiently and effectively. Congestion will also lead shipping lines to consider alternative routings or choose other neighboring ports to berth. A container terminal that is incapable of accommodating massive container volumes and serving large vessels may not only lose its customers but also face the risk of extinction.

1.5 Container Terminal in Malaysia

Malaysia is one of the developing countries in Southeast Asia. There are two distinct parts of Malaysia; Peninsular Malaysia to the west and East Malaysia to the east. Peninsular Malaysia is positioned strategically along the Straits of Malacca, one the most important shipping lanes in the world. Having the benefit of natural geographical location, international trade therefore plays a large role in Malaysia's economy. As of 2007, Malaysia ranked 19th in the world leading exporters with a share of 1.3% of global trade and ranked 25rd in the world leading importers with a share of 1.0% of global trade (WTO, 2008).

Ports in Malaysia are either under the supervision of the Federal Government or the State Government. Those administered by the Federal Government can be categorized into major and minor ports, where the major Federal ports include Port Klang, Penang Port, Bintulu Port, Kuantan Port, Kemaman Port, Johor Port and Port of Tanjung Pelepas, all regulated by port authorities (Nazery, 2005).

Over the years, huge amounts of investments have been poured into developing Malaysian ports, its shipping sector and the ancillary services to handle greater volume of trade and to cater to greater trade growth in the future (Nazery et al., 2007). The growth in container throughput of Malaysian seaports especially, has been tremendous, parallel with the growth of world trade. Table 1.2 captures the positive growth in container throughput of Malaysia ports from 1998 to 2008.

Table 1.2: Total container throughput in Malaysia, 1998-2008 (MOT, 2008).

	EXPORT	IMPORT	TRANSSHIPMENT	TOTAL (Mil. TEUs)
1998	1,574,307	1,489,764	-	3,064,071
1999	2,015,833	1,974,182	-	3,990,015
2000	2,441,514	2,361,132	-	4,802,646
2001	2,723,926	2,578,728	-	5,302,654
2002	5,860,683	3,199,638	-	9,060,321
2003	2,179,407	2,015,859	5,858,501	10,248,742
2004	2,560,831	2,310,110	6,470,330	11,341,271
2005	2,513,180	2,389,272	7,141,777	12,044,229
2006	2,713,388	2,514,860	8,261,487	13,489,735
2007	2,936,873	2,726,380	9,507,643	15,170,896
2008	3,123,313	2,924,030	10,229,089	16,276,432

The total container throughput in Malaysia for the year 1998 was 3,064,071 TEUs and this value increased at the average of 20% per year for the next three years to 5,302,654 TEUs in year 2001. In the year 2002, the total container throughput increased drastically to 9,060,321 TEUs and for the subsequent six years, the total container throughput in Malaysia increased at the average of 10% per year to 16,276,432 TEUs in year 2008.

As containerization has been the leading way to transport cargo around the world, the Malaysian government has taken various initiatives to enhance the competitiveness and attraction of the Malaysian container terminal. One of the initiatives was the designation of Port Klang as the national load center in 1993 to serve a hinterland with a large cargo base as well as the designation of the Port of

Tanjung Pelepas as a transshipment hub port (Nazery et al., 2007). As of 2007, both Port Klang and Port of Tanjung Pelepas ranked 16^{th} and 18^{th} respectively, in world port ranking, in terms of containers handled (UNCTAD, 2008).

1.6 Background of the Case Study Container Terminal

The case study container terminal is located on the southwestern tip of the Malaysian peninsular. Like the Port of Singapore Authority or PSA, the case study container terminal is located close to some of the world's major shipping lanes, an ideal position to capture transshipment trade volumes.

The emergence of this container terminal also caused the first serious challenge to PSA's dominance as the main port in Southeast Asia. According to Lam and Yap (2007), the case study container terminal is expected to pose the strongest challenge to Singapore's transshipment hub ambitions. Figure 1.4 exhibits the forecast distribution of Asian transshipment volume for the year 2015 where Malaysia holds 15.4% of Asian transshipment volume (UNESCAP, 2007).

The case study container terminal is scheduled to be developed in five phases and will be completed by the year 2020. Currently, there are ten berths of 360m each operating while the container yard is capable of handling a total of 4.5 to 6 million TEUs annually. The container yard has a total ground slots of 22,120 TEUs and a storage capacity of 151,200 TEUs with 2,100 refer points.

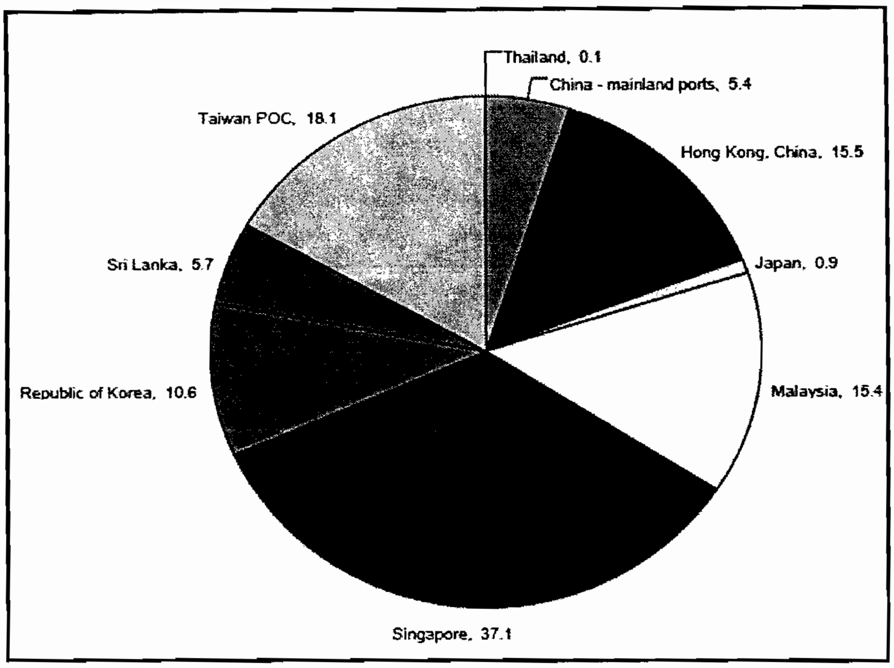


Figure 1.4: Asian transshipment throughput distribution (2015) (UNESCAP, 2007)

1.7 Problem Statement

The case study container terminal is designed as a transshipment hub port. One of the competitive edges of a container terminal that handles high volume of transshipment containers is the ability to turnaround the vessels within a short time (Lee et al., 2006). The ability to service vessels within the shortest time period very much depends on the efficiency of the overall container terminal system. A container terminal consists of many subsystems where the berth and container yard are the two most important subsystems. Therefore, the planning, decision making and operation of the berth and container yard are crucial in order to ensure the whole container terminal operates in an efficient and timely manner.

However, due to the complexity of the container terminal operations, the planning and decision making of the berth and yard are executed separately. Upon receiving the information regarding the estimated time of arrival and number of containers to be unloaded from a vessel, the berth planners will assign an empty berth and adequate quay cranes accordingly. When the berth location is fixed, the berth planners will forward this information to the yard strategist. The yard strategist will then plan where to store the unloaded containers in the container yard according to the berth plan forwarded by the berth planners.

As the individual planning systems are organized within its own operating constraints, there is a lack of general holistic view of the terminal. In the context of the case study container terminal, the berth planners may develop a berth plan that optimizes the utilization of berth space and quay cranes, but due to the fixed berthing location of vessels, the yard strategist may not be able to come up with an optimum yard plan that minimizes the traveling distance of containers between the yard and berth.

This happens because the decision to allocate storing location in the yard to the inbound containers depends on both the berthing position of the vessel carrying these inbound containers as well as the berthing position of the vessel that will pick up the containers. As traveling distance of containers between berth and yard increases, the vessel service time will be affected, thus increasing the vessel turnaround time.

While the internal operation of the container terminal is complex, the external environment where a container terminal operates is also highly complex, competitive and full of uncertainty as well. Berth planners usually schedule berthing location for each vessel according to the estimated time of arrival given by the vessel prior to its arrival. However, the actual arrival time of the vessel usually deviates from the estimated time of arrival due to unforeseen weather change and delay of departure from the previous container terminal.

Additionally, due to the lack of integration between berth and yard planning as well as the understanding of the overall terminal system where complexity is the main culprit, berth planners and yard strategists sometimes fail to see the impact of their decision on other subsystems and on the overall container terminal operations as well. Thus, when a problem arises within the terminal system, it is very difficult to clearly identity the bottleneck that contributes to the inefficiency of the terminal system.

Chen (1999) highlighted that container handling operations consist of different inter-dependent activities and it is very important to investigate the relationship among these terminal subsystem components and the management of each part of the terminal system. Therefore, there is a need for a flexible tool that can aid both berth planners and yard strategists to improve their understanding as well as decision making by providing them with a model that is capable of capturing the interdependency of both berth and yard operations.

1.8 Objectives of Study

The broad aim of this study is to understand, evaluate and model the operational processes in the container terminal while its specific objectives are to:

1. Model the flow of containers across the berth side and yard side to evaluate the system and to measure performance indicators. Four categories of performance indicators that are measured are:
    i. Vessel performance indicators
    ii. Berth performance indicators
    iii. Container handling performance indicators
    iv. Container yard performance indicators
2. Evaluate the relationship and interdependency between the berth and yard operation for improvement creation and understanding of the terminal overall efficiency.
3. Develop a capacity plan to ensure the terminal has sufficient capacity to sustain demand and yet avoid overcapacity.
4. Develop a management flight simulator or Microworlds to allow container terminal managers to test new strategies and policies and reflect on the outcomes before implementing it into the real world.

1.9 Assumptions

There are four assumptions taken into consideration in this study. The first assumption is that the number of quay cranes assigned to each vessel is assumed to be equal for all vessels. This is because the number of quay cranes that are assigned to each vessel at the case study container terminal varies between two to three quay cranes with the majority of two quay cranes per vessel. Therefore, two quay cranes are assigned to each vessel.

The second assumption involves the assignment of prime movers to each quay crane. In this research, the number of prime movers assigned under each quay crane is assumed to be equal as well. According to the yard strategist at the case study container terminal, on the average, seven prime movers are assigned under each quay crane. Therefore, the number of prime movers assigned under each quay crane is assumed to be equal.

The third assumption made is that the quay cranes and prime movers are assumed to be available upon berthing, so that no breakdowns occur. The last assumption is that all container yard blocks are assumed to be 30 percent full at the beginning of the simulation run. This is because during the visit to the case study container terminal, the yard strategist reported that the yard blocks were 30% full at that time.

1.10 Organization of Thesis

This thesis is organized as follows. Chapter Two presents the reviews of literatures that are related to this study while Chapter Three focuses on the presentation of the methodology of this research. The description of the case study container terminal is discussed in Chapter Four while Chapter Five and Chapter Six present the discussion on model construction and management flight simulator respectively. This thesis ends with conclusion and recommendation for future research in Chapter Seven.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

This chapter reviews previous studies related to container terminals and system dynamics. The literature on container terminal covers mainly the research works that have been done in the area of container terminal operations. The literature on system dynamics methodology on the other hand covers the application of system dynamics specifically in the area of industrial management. The knowledge obtained from this literature review not only contributes to the research process but also reveals differences between previous research works and this study, thereby confirming the significance of this study.

2.2 Container Terminal

A container terminal is the place where vessels dock on a berth, and containers are loaded and unloaded (Lee et al., 2006). A container terminal can be ideally divided into two areas, the berth side for berthing vessels and the yard side for storing containers (Vacca et al., 2007 and Lee et al., 2006). The berth side consists of several berths for vessels to be moored. Typically, the berth is linear in shape. On the other hand, the yard side consists of several yard blocks which serve as the storage areas for import, export and transshipment containers. Besides the berth and yard side, there is also the gate side. The gate side consists of several gate lanes where external trucks enter the container terminal to deliver export containers to and pick up the import containers from the yard blocks.

As the container terminal operates under a very complex environment which consists of several important subsystems such as the berth side, yard side and gate side, the operation and performance of one subsystem will influence the performance of the other subsystems. If bottlenecks occur in one part of the subsystem, it will eventually lead to the disruption of the entire operations of the container terminal. In short, the overall performance of the container terminal is highly dependent on the efficiency of each subsystem.

When a vessel arrives at a container terminal, it will be assigned to an empty berth, waiting for the quay cranes to unload or discharge the import containers from the vessel and transfer it to the terminal. Under these quay cranes, prime movers are waiting to receive the import containers from the quay cranes. These prime movers will then transfer the import containers to the container yard for temporary storage, and wait for incoming vessels or external trucks to retrieve them.

Export containers follow the reverse flow of the import containers. Export containers arrive at the container terminal via external trucks or train. Upon arriving at the container port, these export containers will be stored temporarily in the container yard waiting for incoming vessels to collect them. Figure 2.1 shows the flow of containers in a typical container terminal.

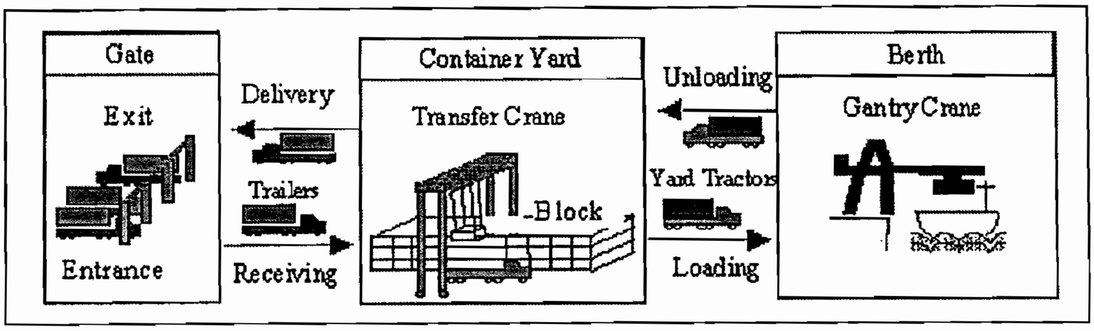


Figure 2.1: Containers flow in a container terminal (Yun and Choi, 1999).

Container activities can be categorized into three types, namely the import, export and transshipment activities (Lee et al., 2006). The import and export container activities have already been described earlier. The process for transshipment container activities is slightly different from the import and export container activities. According to Vacca et al. (2007), recently, container transport tends to develop towards single-mode transportation called transshipment, where containers are exchanged between vessels commonly referred to as mother vessels and feeders.

For the transshipment container activity, the containers will be stored in the storage yard after being unloaded or discharged from the vessel, and finally loaded onto other vessels (Lee et al., 2006). Therefore, most of the transshipment container activities revolve around the berth area as well as the yard area and seldom make it to the gate area.

The number of publications on container terminals is increasing as container terminal operations are becoming important. Steenken, Voss and Stahlbock (2004) and Stahlbock and Voss (2008) presented the most comprehensive overview on container terminal operations. Stahlbock and Voss (2008) provided a detailed description of the structure and handling equipment used in the container terminal and they also reviewed the method of optimizing logistics operations in the terminal, supported by past studies (253 references up to year 2007). Similarly, Vis and De Koster (2003) and Meersmans and Dekker (2001) also provided an overview of relevant literatures on various decision problems in a container terminal. These two papers however divided the decision problems by strategic, tactical and operational levels. Like Steenken, Voss and Stahlbock (2004) and Stahlbock and Voss (2008), Meersmans and Dekker (2001) highlighted the contribution of operational research method in solving problems in the container terminal.

A container terminal can be roughly divided into two main areas, which are the berth side and the yard side (Zhang et al., 2003). The berth side is where vessels are berthed and quay cranes unload inbound or import containers and load outbound or export containers to vessels. The yard side is a storage area made up of blocks of containers where yard cranes are used to handle the containers in the storage blocks. Many past studies on container terminals revolve around solving the problems of these subsystems separately. Steenken, Voss and Stahlbock (2004) and Stahlbock and Voss (2008) both highlighted the lack of much needed research on integrating problems from different subsystems in a container terminal. According to Vis and De Koster (2003), in order to be an efficient terminal, it is necessary to address all problems as a whole.

Before discussing past studies on the container terminal as a whole, work which investigated berth side as well as the yard side independently is presented. The purpose of including this individual subsystem studies is to highlight what previous researchers have tried to solve by breaking the container terminal into smaller parts and solving them separately.

2.2.1 Berth Allocation

Among all the resources in the container terminal, berths are the most important and an effective berth allocation is critical to the efficient management of container traffic flow (Guan and Cheung, 2004). Effective berth allocation also improves customers' satisfaction, and increases port throughput, which leads to higher revenues for a container terminal (Kim and Moon, 2003). Nevertheless, the cost of constructing a berth is very high compared to the investment costs of other facilities in the terminal (Park and Kim, 2000; Legato and Mazza, 2001). Activities at a berth include the arrival of vessels as well as the unloading and loading of containers from and onto vessels. The task of the berth allocation or planning system involves allocating the limited berth space, organized in a number of slots, among incoming vessels (Legato and Mazza, 2001).

Numerous studies have been conducted on berth allocation problems (e.g. Imai, Nishimura and Papadimitriou, 2001; Nishimura, Imai and Papadimitriou, 2001; Imai, Nishimura and Papadimitriou, 2003; Kim and Moon, 2003; Guan and Cheung, 2004; Dai et al., 2004 and Moorthy and Teo, 2006). Although they may differ in the assumptions made, these literatures share the same aim of proposing the best berth allocation model to maximize berth utilization and at the same time minimize the vessel turnaround time. Kim and Moon (2003) developed a mixed-integer-linear-programming model and later applied simulated annealing algorithm to solve the berth-scheduling problem. In this study, the authors aim to determine both the berthing times of the vessel as well as locate its best berthing position to maximize berth utilization.

Similarly, Guan and Cheung (2004) proposed a model to allocate berth space to vessels but with the objective of minimizing the vessels' total weighted flow time. The flow time is defined as the sum of waiting time and processing time of a vessel while the weights reflect the relative importance of the vessel. Their model allowed multiple mooring per berth, taking into account the vessel arrival time. The authors used heuristics technique to generate solutions from the constructed model.

There are also studies that attempted to find a tradeoff between maximizing the berth utilization and minimizing the vessel turnaround time, such as by Dai et al. (2004) and Moorthy and Teo (2006). Dai et al. (2004) developed a heuristic technique to construct a berthing plan of allocating berthing space to vessels in real time to ensure most of the vessels can be berthed-on-arrival and be allocated a berthing space close to their preferred locations within the terminal. Moorthy and Teo (2006) also used heuristic approach to study the economic impact of preferred berthing location template design problem on container terminal operations. The authors tried to generate solutions while finding the trade-off between the service (waiting time for vessels) and cost (movement of containers between berth and yard).

Imai, Nishimura and Papadimitriou (2001), on the other hand, proposed an algorithm for solving dynamic berth allocation problem where vessels may arrive while work in berth is still in progress. This problem was solved using Lagrangian Method. Nishimura, Imai and Papadimitriou (2001) used genetic algorithm to further extend the previous dynamic berth allocation problem for multi-water depth configuration. Genetic Algorithm technique was applied again by Imai, Nishimura and Papadimitriou (2003) to modify the existing formulation of berth allocation problem in order to treat calling vessels at various service priorities.

The literatures discussed so far had either assumed that the vessel staying time in the container terminal is deterministic or the vessel processing time is constant. In reality, the duration of berthing for each vessel also depends on the number of cranes assigned to the corresponding vessel (Kim and Moon, 2003). Dragovic et al. (2005) agreed that the service time of a vessel depends on its berthing point and is a function of number of quay cranes assigned to it. When the number of cranes assigned to a vessel increases, the duration of berthing of the vessel can be reduced (Park and Kim, 2003).

There are also studies that simultaneously addressed the berth allocation and crane assignment such as by Dragovic et al. (2005), Park and Kim (2003), Guan et al. (2002), and Legato and Mazza (2001). The primary objectives of these literatures were mainly to maximize berth utilization and minimize the vessel turnaround time. By including crane assignment into berth allocation problems, the model developed and solutions derived from these literatures appeared to be more realistic than solutions derived from literatures that only considered berth allocation problem independently.

With the aim of scheduling berth and quay crane simultaneously, Park and Kim (2003) developed their model in two phases. The first phase used sub-gradient optimization technique to determine the berthing times and positions of vessels while the second phase used dynamic programming technique to determine a detailed operating schedule for individual crane. Guan et al. (2002) considered the berth allocation problem as a multiprocessor task scheduling problem where the objective was to allocate vessels (jobs) to a berth with multiple quay cranes (processor) while minimizing the total weighted completion time of the jobs (vessels). Heuristic technique and worst-case analysis were used to solve the problem.

Dragovic et al. (2005) and Legato and Mazza (2001) demonstrated the use of simulation for analyzing berth planning subsystem. Dragovic et al. (2005) constructed a ship-berth link model to investigate the efficiency of operations and processes in the ship-berth link through the inspection of basic operating parameters such as berth utilization, average number of ships in waiting line, average time that a ship spends in waiting line, average service time of a ship, average total time that a ship spends in port, average quay crane productivity and average number of quay cranes per ship.

The model developed by Legato and Mazza (2001) assigned vessels to the berth according to priorities. Mainline vessels were given higher priority to berth first compared to feeder ships. The model also allowed the testing of the impact in increasing and decreasing number of quay cranes on the vessel waiting and service time. Razman and Khalid (2000) also developed a model to improve the logistic processes at the container terminal where the model simulated all processes such as berth allocation, crane and prime mover assignment. Like Legato and Mazza (2001), Razman and Khalid (2000) also assigned berthing priorities to both mainlines and feeder ships.

The inclusion of crane assignment into the berth allocation model proposed by Dragovic et al. (2005), Park and Kim (2003), Guan et al. (2002), Legato and Mazza (2001) and Razman and Khalid (2000) did generate a more realistic solution but, they only focused on one part of the container terminal system, namely the berth planning subsystem and neglected the other subsystem such as the yard side. Chen (1999) highlighted that the efficiency and quality of management in container yard operations will also influence all the terminal operations. Therefore, the next section is dedicated to the discussion on literature about yard planning.

2.2.2 Yard Planning

The yard operation is the most complex part at a terminal because it simultaneously handles both inbound and outbound container flows (Zhang, 2002). In the terminal operations, container yard plays a vital role and acts as the nerve centre in the operations as most of the terminal operations either originate from or are destined to the container yard. The container yard functions as the storage area, where the export containers are stored to be loaded and import boxes are stacked (Chen, 1999). Therefore, it is the duty of the yard planner to assign optimal allocation of storage areas for import, export and transshipment containers (Yun and Choi, 1999). A problem that is usually faced by a yard planner is the inadequate storage space in the yard to accommodate all the containers. Therefore, it is crucial for yard planners to make best use of the available yard space.

The objective of minimizing the utilization of yard space has become a major priority for yard planners in order to gain more storage space for incoming containers. Chen et al. (2004) proposed a model for the central allocation process which minimizes the utilization of yard space while satisfying space requirement. Their paper used different metaheuristic tools such as tabu search, simulated annealing, squeaky wheel optimization and genetic algorithm to generate feasible solutions which were later compared and the best results among all selected. In order to minimize the utilization of yard space, containers are usually stacked in the yard to generate more space for the incoming containers. Containers must be stacked in such a way that the stacking capacity is maximized while the response time is minimized.

Besides the attempt to minimize yard utilization to generate more space, there are also literatures that report on determining the best storage location of the containers in the yard to minimize the total distance to transport the containers between their storage blocks and vessel berthing locations such as in Preston and Kozan (2001) who used genetic algorithm to solve the problem. Zhang et al. (2003) addressed the similar problem using a mathematical programming model which was capable of determining the number of containers to be placed in each storage block to balance the workloads among blocks.

Besides the containers storage location factor, yard crane assignment also influences the vessel turnaround time. Zhang et al. (2002) addressed the crane deployment problem where the objective was to find the time and routes of Rubber Tyred Gantry (RTG) crane movements among container blocks so that the total delayed workload in the yard is minimized. This problem was formulated using mixed integer programming and solved using Lagrangian relaxation.

Some of the literature such as by Kim and Kim (2002) only focused on the inbound or import containers where they proposed various cost models for determining the space requirement as well as the number of transfer cranes needed in import container yard with the objective of minimizing the terminal operational costs. In addition, there are also studies that only focused on the allocation of outbound or export containers. One such study is by Kim and Park (2003) where the authors focused on how to allocate outbound containers when they arrive at the container yard while utilizing the space efficiently and making the loading operations more efficient.

Two major subsystems of the container terminal have been discussed, but all of these literatures only addressed the problems of the berth side or yard side independently. As a container terminal is composed of several subsystems, any study that investigates a subsystem in isolation would fail to provide a satisfactory understanding of the overall operations in the container terminal.

Results generated from these studies although contributing to the development of new ideas, somehow failed to represent the whole container terminal as it did not take into account all the processes in the terminal. In recent years, simulation has become an important tool to improve terminal operation and performance (Steenken, Voss and Stahlbock, 2004). The next section focuses on the application of simulation in modeling the whole container terminal system.

2.2.3 Application of Simulation in Modeling A Container Terminal System

This section focuses on studies that demonstrate the application of simulation techniques on modeling the complete container terminal. This is because the direction of this research is towards developing a system dynamics simulation model of the container terminal which integrates both the berth and yard sides. This section begins with the discussion on the strength of simulation technique in solving problems in container terminal compared to other analytical methodology.

Complexity of different container terminal operations often results in difficulties in using analytical tools as a method of investigation because analytical modeling tends to simplify the real situation which results in the model developed to lose out in terms of detail and flexibility (Dragovic et al., 2005). Yi et al. (2002) agreed that most analytical models only take few variables into account and numerous constraints have to be satisfied before results can be applied in practice. On the other hand, simulation modeling is better than analytical models in representing the random and complex environment of a container terminal (Dragovic et al., 2005).

Besides that, the simulation model is also a powerful tool for evaluating the performance of a proposed plan, choosing an appropriate design and making changes to it before committing it to operation (Razman and Khalid, 2000). The simulation model also provides a platform to evaluate the dynamic processes of container terminal and to identify potential bottlenecks as it allows generating and analyzing statistics such as average productivity and average waiting time (Hartmann, 2004).

Simulation technique is widely used to analyze the performance of the existing terminal such as in Ramani (1996), Yun and Choi (1999), Shabayek and Yeung (2002), Liu, Jula and Ioannou (2002), Duinkerken et al. (2006), and Ma and Hadjiconstantinou (2008), or to test newly proposed operating strategy such as in Kia, Shayan and Ghotb (2002)., Simulation technique is also commonly used to evaluate the construction of a new container terminal as demonstrated in Ottjes et al. (2006).

Ramani (1996) developed a simulation model to support the logistics planning of container operations. This model allowed testing on various operating strategies and provided estimates for port performance indicators such as berth occupancy, vessel output, vessel waiting and turnaround time. The developed model allowed users to view terminal activities berth-wise, ship-wise and shift-wise on a real time basis. This paper however, does not elaborate clearly on the type of operating strategies that the author intended to test with the developed simulation model.

In order to investigate whether the existing container terminal was efficient enough to handle large container streams and whether the usage of transfer cranes and quay cranes would be more effective, Yun and Choi (1999) constructed a simulation model to address those issues. Besides providing estimates for terminal performance indicator as in Ramani's (1996), the developed model also provided estimates for container handling equipment performance indicators such as quay cranes and transfer cranes utilization and container yard occupancy. The authors however only used a reduced model of the terminal understudy to conduct the simulation test due to the complexity and massive data requirement if the whole terminal is simulated.

Therefore, the developed model did not accurately reflect the actual terminal understudy. Furthermore, the proposed model did not provide scenario testing such as the impact of adding extra number of transfer cranes or quay cranes to the overall terminal operations.

The simulation model developed by Shabayek and Yeung (2002) allowed planning for future additional berths. As done by Ramani (1996) and Yun and Choi (1999), Shabayek and Yeung (2002) also developed a simulation model to simulate the current performance of the container terminal. The developed model was used to estimate the improvements in performance of the terminal when the handling capacity of the container terminal varied.

Simulation technique was also used to evaluate different scenario testing such as in Liu, Jula and Ioannou (2002), Duinkerken et al. (2006), and Ma and Hadjiconstantinou (2008). Liu, Jula and Ioannou (2002) used simulation to design, analyze and evaluate four different Automated Container Terminal (ACT) concepts. The four concepts included automated container terminals based on the use of Automated Guidance Vehicles (AGVs), a Linear Motor Conveyance System (LMCS), an overhead Grid Rail (GR) system, and a high-rise Automated Storage and Retrieval Structure (AS/RS). Performance indicators such as ship turnaround time, container throughout and container dwell time of each terminal system for the same operational scenario were evaluated to determine the best terminal system.

On the other hand, Duinkerken et al. (2006) developed a simulation model of the container terminal which included container handling and container transport at the container terminal. The model also analyzed the effectiveness and efficiency of three possible transport systems, i.e. the present Multi-Trailer System (MTS), a system based on Automated Guided Vehicles (AGVs) and a system based on Automated Lift Vehicles (ALVs). Ma and Hadjiconstantinou (2008) presented a simulation model of complete terminal operations that allowed the terminal operator to evaluate and validate the operational plans resulting from an optimization model of the combined container assignment and yard crane deployment problem. The model was used to analyze different proposed operational scenarios such as the impact of sharing and no sharing of yard cranes on the yard crane utilization rate, and the average truck waiting time at the gate, yard and quay.

Simulation technique is also capable of assisting container terminal managers to evaluate new terminal operating strategies. As it is very risky and expensive to conduct the proposed plan, a simulation model provides the opportunity to experiment the impact of the proposed strategies on the terminal performance before implementing it for real. Kia, Shayan and Ghotb (2002) successfully demonstrated the application of simulation technique in evaluating new operating strategy.

Kia, Shayan and Ghotb (2002) proposed a model where a large portion of imported containers were taken away via rail upon arrival of the imported containers as opposed to the current system where upon arrival at the terminal, the imported containers were stored temporarily in the container yard. The aim of the proposed system was to identify and reduce the terminal congestion thereby increasing the terminal capacity. Besides generating a huge amount of yearly savings, the proposed method created more space for container stacking, reduced congestion within terminal, reduced turnaround time and increased berth availability. The comparison of the two operational systems led to significant savings in port expansion, vessel turnaround time and inventory cost on cargo.

There are also studies that constructed simulation model to evaluate the construction of a new container terminal, such as in Ottjes et al. (2006). This research was conducted to determine the requirements for quay length, stacking capacity, handling and transport equipment as well as the Inter-Terminal Transport (ITT). This research served as the starting point for the overall study of the newly proposed terminal. Similar to Ottjes et al. (2006), the research of Hartmann (2004) also acted as an input generator of containers for future studies.

Hartmann (2004) introduced an approach for generating scenarios which consisted of the arrivals of large vessels, feeder ships, train and truck together with the lists of containers to be loaded and unloaded. The generated scenarios could be used as input data for simulation model as well as to test data for algorithm to solve optimization problems in a container terminal. In this paper, the author used the implemented generator into a newly built terminal to support a simulation study on the strategies for selecting yard blocks and slots for arriving containers with the purpose of testing, improving and parameterizing the stacking strategies. Hence, the simulation led to improved stacking strategies before the terminal started operations.

This section presents the application of simulation methodology in addressing problems in the container terminal. The simulation technique applied in the discussed literatures so far are all of discrete-event simulation. Table 2.1

summarizes the studies which focused on the berth allocation, berth allocation with crane assignment, yard planning and whole container terminal as was discussed earlier on.

Table 2.1: Summary of studies conducted on berth allocation, berth allocation with crane assignment, yard planning and whole container terminal.

SCOPE	STUDY	TECHNIQUE
Berth Allocation	Imai, Nishimura and Papadimitriou (2001)	Lagrangian Method
	Nishimura, Imai and Papadimitriou (2001)	Genetic Algorithm
	Imai, Nishimura and Papadimitriou (2003)	Genetic Algorithm
	Kim and Moon (2003)	Mixed Integer Linear Programming and Simulated Annealing
	Guan and Cheung (2004)	Heuristics
	Dai et al. (2004)	Heuristics
	Moorthy and Teo (2006)	Simulated Annealing
Berth Allocation with Crane Assignment	Razman and Khalid (2000)	Discrete Event Simulation
	Legato and Mazza (2001)	Discrete Event Simulation
	Guan et al. (2002)	Heuristics
	Park and Kim (2003)	Optimization and Dynamic Programming
	Dragovic et al. (2005)	Discrete Event Simulation
Yard Planning	Preston and Kozan (2001)	Genetic Algorithm
	Zhang et al. (2002)	Integer Programming
	Kim and Kim (2002)	Cost Model
	Kim and Park (2003)	Heuristics

	Zhang et al. (2003)	Mathematic Programming
	Chen et al. (2004)	Heuristics
Complete Container Terminal	Ramani (1996), Yun and Choi (1999), Shabayek and Yeung (2002), Liu, Jula and Ioannou (2002), Kia, Shayan and Ghotb (2002), Hartmann (2004), Duinkerken et al. (2006), Ottjes et al. (2006), Ma and Hadjiconstantinou (2008).	Discrete Event Simulation

Our research focuses on modeling the container flows across the berth side and yard side of a container terminal. It is similar to Legato and Mazza (2001), which aimed to identify potential bottlenecks in the container flow that contributed to container terminal congestion. As opposed to the study by Legato and Mazza (2001) which only concentrated on modeling the berth side, our research sets to integrate both the berth side and yard side. Our research also shares the same interest of Yun and Choi (1999) in investigating whether the existing container terminal is efficient enough to handle large container streams and whether the usage of current terminal facilities can be more efficient. In addition, our research further enhances the study of Yun and Choi (1999) as the simulation model developed in our research is capable of providing scenario testing such as the impact of adding extra number of quay cranes and berths to the overall terminal operations.

Estimates for terminal performance indicators such as vessel performance indicators, berth performance indicators, container handling performance indicators and container yard performance indicators are generated from the simulation results.

These performance indicators are somewhat similar to Ramani (1996) and Yun and Choi (1999). The developed simulation model is also used to predict the future capacities of existing terminal facilities and identify when new terminal facilities will be needed to ensure the terminal has enough capacity to sustain demand and minimize the risk of overcapacity.

2.3 System Dynamics

This section provides discussion on the strength of system dynamics methodology compared to other conventional tools as well as the application of system dynamics to problems in various areas that operate under complex and uncertain conditions.

2.3.1 The Definition and Strength of System Dynamics Modeling

System dynamics is a methodology for analyzing complex systems and problems with the aid of computer modeling and simulation software. System dynamics approach originated from the research of Professor Jay W. Forrester at Massachusetts Institute of Technology in the late 1950s. System dynamics has long been used to help management teams to formulate strategy and improve individual, team and organization learning (Warren and Langley, 1999).

Maani and Cavana (2000) defined the word *system* as a collection of parts that interact with one another to function as a whole and a *system* is more than the sum of its parts, it is the product of their interactions. The word *dynamics* is defined as changes in demand and supply over time as the components are constantly evolving, as a result of previous actions (Ruth and Hannon, 2004). Richardson and Pugh III (1981) defined system dynamics as a methodology for understanding certain kinds of complex problems while Sterman (2000) gave a detailed definition of system dynamics by explaining it as partly, a method for developing management flight simulators, often computer simulation models, to help us learn about dynamic complexity, understand the sources of policy resistance and design more effective policies.

While traditional analysis approaches focus on breaking down problems to smaller parts and solving it separately, system dynamics approach, however, involves a broader view, looking at possible interactions among the subsystems to create a better understanding of the big picture. According to Richardson and Pugh III (1981), the problems that one addresses from the perspective of system dynamics has at least two features in common. The first is that they are dynamic and involve quantities which change over time and secondly, they involve the notion of feedback.

The need for system dynamics arises as many times, our best efforts to solve a problem actually make it worse because our well intentioned efforts to solve pressing problems lead to policy resistance where policies are delayed, diluted or defeated by the unforeseen reactions of other people or nature (Sterman, 2000). Policy resistance arises because the full ranges of feedbacks operating in the system are difficult to understand. The difficulty in operating an organization is directly related to the complexity of individual interconnections and to the number of interconnections that must be managed.

According to Ruth and Hannon (2004), both the number and complexity of the interconnections have changed over time because of growth in business size, globalization, pressure to improve efficiency, growing competition while trying to live up to customers' expectations and technological advances. System dynamics therefore is a powerful method to gain useful insight into situations of dynamic complexity and policy resistance. It is increasingly used to design more successful policies in companies and public policy settings (Sterman, 2000).

As analytical method offers no means to capture interdependency (Warren and Langley, 1999), it is the nature of system dynamics that captures the interdependencies between all subsystems that make up the whole. System dynamics also combines qualitative and quantitative aspects and aims to enhance understanding of a system and the relationships between different system components (Brailsford et al., 2004).

Many discussions were made regarding the differences between the Discrete-Event Simulation (DES) technique and system dynamics simulation technique. The aim of system dynamics is to provide an understanding of the modes of behavior (basic trends) thus, system dynamics model tends to be more holistic, with aggregate flows while the aim of DES is more quantitative with the models being repeatedly re-run to provide prediction (Taylor and Dangerfield, 2005).

Koelling and Schwandt (2005) added that DES works well for issues concerned with transaction, processing and the flow of individual entities through a system and that DES is more commonly associated with operational level types of problems. However, according to Koelling and Schwandt (2005), system dynamics is commonly used to model the relationships between system variables, rate of change over time and explicit feedback, rather than to focus on individual transaction in the system. It is often associated with higher level types of problems, especially consideration of the impact of policy and strategy decision.

An additional benefit of system dynamics modeling is the policy analysis where the type of policies that system dynamics considers are of higher level, focusing more on the effect of the structure and possible changes to that structure as opposed to DES which emphasizes more on evaluating policies such as routing decisions, staff schedules or special queue handing (Koelling and Schwandt, 2005). Lane et al. (2000) added that system dynamics modeling enables users to understand why structure produces behavior (the base case), and how behavior varies under different conditions (the policy analysis).

System dynamics provides a feasible experimental environment as exploring the effect of policy changes and experimenting with alternative policy formulation are not feasible in the real world. Unlike forecasting or market research that over-simplify interactions among important variables and parameters to generate results, policies testing presents alternative images about the future, thus helping managers to face their own logic and assumption (Georgantzas, 2003).

Another major advantage of system dynamics is the development of Microworlds or management flight simulator. Microworlds is a dynamic learning laboratory which allows players to test new strategies and policies and reflect on the outcomes before implementing it into the real world. Furthermore, Microworlds offers the opportunity for role playing and role switching to explore differences in critical assumptions and also helps players to cope with uncertainty, complexity and ambiguity. The Microworlds learning experience has been considered superior to more traditional training alternatives in organizations favoring experiential hands-on learning (Hirsch and Immediato, 1999).

System dynamics modeling technique demonstrates many advantages and it is well suited to capture the complexity of the operation in the container terminal. However, up to date, there are very limited studies conducted on container terminal using system dynamics simulation modeling technique although this technique has been used widely in other areas such as health care, education, manufacturing and transportation.

2.3.2 Application of System Dynamics

There are very limited literatures that discuss the application of system dynamics in container terminal. To our knowledge, currently there is only one related literature, by Choi et al. (2007) which discussed the application of system dynamics in solving issues of the container terminal.

Choi et al. (2007) used system dynamics to determine what factors contribute to the reliability and competitiveness of a container terminal in the long run. The study revealed that many existing literatures that discussed port competitiveness emphasize on environmental change such as geographical features, location, service, convenience and expenses of the container terminal and neglect the changes of the container terminal itself. Therefore their study aimed on analyzing factors from the operational point of view. A system dynamics model was developed to analyze first, the impact of handling volume of quay cranes, transfer equipment, yard crane and gate and second, the impact of vessel turnaround time on the reliability of a container terminal.

The study by Choi et al. (2007) focused on solving issues at the operational level. As was discussed in the previous section, a container terminal consists of many subsystems such as the berth and the yard. Choi et al. (2007) however, did not incorporate the interdependency and relationships between various subsystems at the container terminal.

As there are very few studies on the application of system dynamics in container terminal, the discussions of literature on the application of system dynamics to solve problems include other business sectors where the operating environment of these areas are complex and interdependent, somehow similar to the operating environment of the container terminal.

Homer and Hirsch (2006), Dangerfield (1999) and Royston et al. (1999) compiled all previous studies conducted in the area of health care using system dynamics. Given one of the strengths of system dynamics is its ability to solve complex problems, therefore this tool is well suited to solve health care problems as health care system is not only large and complex, but also not easy to analyze, design or even understand (Koelling and Schwandt, 2005). The application of system dynamics revolved around solving the problem of increasing demand on health care, increase in patients' waiting time and waiting lists and resources allocation problems.

The growing number of patient waiting lists has always been a challenge to the health care management. In order to determine why the number of patients waiting for emergency care is increasing, Brailsford et al. developed a system dynamics model of the entire health care system in the city of Nottingham, England to simulate patients' flows and identify the potential bottleneck in the health care system. From the developed model, appropriate solution to address the problem was identified and scenario such as maintaining current growth in demand without adding additional resources was tested to see its impact on the system. Besides this, a variety of possible alternatives were generated from the system dynamics model.

Similarly, Gonzalez-Busto and Garcia (1999) and Ackere and Smith (1999) also addressed the problem of increase in patients' waiting lists using system dynamics. Gonzalez-Busto and Garcia (1999) modeled the patients' waiting lists of Spanish public hospitals. They investigated how patients' waiting lists behave over time and analyzed the long term effectiveness of policies that were currently being applied. Policies that were analyzed included subcontracting part of the hospitals' activities, extending working day until afternoon and this extra work being remunerated with a fee-for-service system, depending on the number of patients treated. The third policy was constantly updating the patients' waiting lists to determine the exact number of patients waiting for treatment. Besides analyzing these policies, the system dynamics model was used to review new recommendation such as the decision on capacity investment and the effect of changes in staffs' vacation periods on patients' waiting time.

Ackere and Smith (1999), on the other hand, focused on developing a macro model of UK National Health Service to investigate the reason for increase in patients' waiting lists for elective surgery by dissecting the problem into the demand side which emanated from the patients and the supply side which emanated from the physicians.

There are also studies that investigate both the problem of growing patient lists and waiting time such as the study by Wolstenholme (1999). The system dynamics model developed in this research was used to address both problems of growing waiting lists for elective surgery in acute hospitals and problem on the growing rate of non-elective hospital admissions simultaneously. This model also allowed policies testing such as evaluating the use of 'intermediate care facilities' to release the used resources by preventing patients needing hospital treatment or continuing community care in order to reduce the length of patients' stay in the hospitals.

In order to determine whether shifting services such as implying stricter clinical guidelines and increasing capacity can stimulate demand, Taylor and Dangerfield (2005) developed a framework to support this hypothesis. The authors successfully translated the whole health care system into their system dynamics model to study the interactions of each element in the health care system. Study of two different cases of the shift in Cardiac Catheterization (CC) services was presented. From the study, a number of recommendations were derived; first was that increasing capacity is not necessarily the most effective way of improving access. Secondly, by focusing on isolated events, short term results and single performance measure could lead to ineffective policies and misleading conclusion and thirdly, patient pressure is a challenge to the delivery of health care which cannot be ignored in realistic model.

Lane et al. (2000) however focused on the micro level of the health care system by modeling patients' flow at the Accident and Emergency (A&E) department to investigate the sensitivity of waiting time to hospital bed numbers. In other words, the authors were interested to see whether bed reduction contributed to the increase of waiting time in A&E. The developed model was used to explore scenarios which involved changes in bed capacity as well as in A&E demand.

Besides the health care sector, system dynamics is also capable of solving problems in numerous highly complex industrial management areas such as transportation management (Liehr et al., 2001 and Mayo et al., 2001), tourism management (Georgantzas, 2003), insurance management (Barlas et al., 2000), university management (Barlas and Diker, 1996), television program management (Mukherjee and Roy, 2006) and supply chain management (Kumar and Yamaoka, 2007).

Mayo et al. (2001) demonstrated the application of system dynamics in London Underground Limited (LUL). Initially, the research intended to solve two major challenges faced by LUL. The first was to determine how different LUL restructuring options would affect their major stakeholders, while the second was to foresee the potential changes that would accompany restructuring. The initial analysis was then shifted to identifying effective actions that would ensure a successful implementation of the Public-Private Partnership (PPP) between LUL and potential private companies. From the system dynamics model, the authors uncovered three critical actions important to the success of the PPP implementation and shared these critical factors with potential bidders to improve the chances of successful PPP implementation and to avoid any future major pitfalls. At the last stage, when the bidders submitted their tender, LUL used the developed system dynamics model to evaluate the bidders' performance to determine which bidders would be the best to work with.

Besides being capable of solving transportation problems faced by the underground train, Liehr et al. (2001) demonstrated the usage of system dynamics model to identify key variables and leverages for cyclical management strategies in the airline market. The authors revealed that most of the airline managers assumed that the cycles in the airline market are a response to fluctuations in the evolution of the gross domestic product (GDP) that lie beyond the sphere of the industry's influence. Thus, there is a lack of cyclical management strategies to smooth the oscillations and reduce their negative impact on the carriers' profitability. The developed system dynamics model however demonstrated that the cyclical behavior of the airline industry's performance was to a significant degree caused by their decision rules and not by exogenous factors. From the model, the author identified several leverage points to stabilize the system's behavior in the process of aircraft ordering, network planning and in adding flexibility to existing capacity, especially through leasing and retirement policies.

As Mayo et al. (2001) and Liehr et al. (2001) discussed the application of system dynamics in underground train and railway system respectively, these two literatures share the similarity where both focused on the macro level and long term issues.

Georgantzas (2003) conducted a study on Cyprus' hotel value chain within the island's tourism customer-supplier value chain. Multiple scenarios on what might happen to Cyprus' tourism over the next 40 years were computed using system dynamics model. These strategic scenarios were capable of assessing Cyprus' hotel value-chain sensitivity, tourism growth and hotel price seasonality as well as their potential effects on Cyprus' hotel profitability.

The system dynamics model developed in Barlas et al. (2000) aimed at tackling two potential problems that occurred in strategic insurance management. The first problem was that the company exhibited a fast growth between 1988 and 1993, followed by persistent stagnation and a slight decline. The second problem was that, in spite of continuous stagnation or decline in policy sales, there was a persistent demand from various departments for additional employees. Numerous simulation experiments were carried out on the model and the simulation results revealed the potential cause of stagnation and provided remedies to avoid similar decline in the future.

An interactive simulation model using system dynamics was constructed in Barlas and Diker (1996) to address a wide range of problems concerning the academic aspects of a university. Long-term strategic problems such as growing student-faculty ratios, poor teaching quality and low research productivity were analyzed and certain policies for overcoming these problems were tested and compared. Apart from developing a system dynamics model to understand and identify solutions to the above problems, an interactive dynamic simulation game was also constructed. Players consisted of faculty members, teaching assistants and students playing the role of university policy makers in this game. The comparison of the players' game results revealed that players with different orientations focused on different performance measures.

System dynamics modeling technique was also used by Mukherjee and Roy (2006) to understand the dynamics of brand management in a television game show. The purpose of their research was to explore why some entertainment products such as *Kaun Banega Crorepati* (KBC), an Indian version of Britain's *Who Wants to Be a Millionaire,* succeeded while other similar television game shows failed. The analysis of the developed model revealed that the right mix of host popularity, channel popularity and prize money had enabled KBC to achieve unrivaled success. The authors highlighted that the developed model was capable of helping managers in making decisions on the optimal number of episodes to air, expected revenue stream, choice of show host and the right channel selection.

Kumar and Yamaoka (2007) developed a system dynamics model to analyze the closed loop supply chain or reversed supply chain design of the Japanese automotive industry. The relationships between reducing, reusing, recycling and disposal were explored with base case scenario analysis using the car consumption and forecast. The developed model was then tested under extreme conditions for the purpose of model validation. Dynamic analysis of different market scenario on the Japanese car industry's reverse supply chain was analyzed to understand how different logistics elements were impacted by government regulations on a long term basis.

Again, all the discussed application of system dynamics model to address problems in the area of service management only focused on the strategic level and long term issues. Much emphasis is given to determine the future growth of the related area at the service industry.

Table 2.2: Summary of studies conducted at numerous areas using system dynamics.

AREA	SCOPE	STUDY
Container Terminal Management	Operational decision-making	Choi et al. (2007)
Health Care Management	Operational decision-making	Lane et al. (2000)
	Strategic decision-making	Gonzalez-Busto and Garcia (1999), Ackere and Smith (1999), Wolstenholme (1999), Brailsford et al. (2004), Taylor and Dangerfield (2005).
Transportation management	Strategic decision-making	Liehr et al. (2001); Mayo et al. (2001)
Tourism management	Strategic decision-making	Georgantzas (2003)
Insurance management	Strategic decision-making	Barlas et al. (2000)
University management	Strategic decision-making	Barlas and Diker (1996)
Television program management	Strategic decision-making	Mukherjee and Roy (2006)
Supply chain management	Strategic decision-making	Kumar and Yamaoka (2007)

Table 2.2 summarizes the literatures that were discussed in this section according to the respective area of application and scope of study. Of all the literatures, only Lane et al. (2000) and Choi et al. (2007) used system dynamics to model the problems at the operational level. The application of system dynamics modeling in strategic decision making has been widely recognized. However, studies on application of system dynamics models to support operational decision-making are still relatively few.

As has been said in (Wolstenholme, 1999), system dynamics modeling technique has demonstrated itself in a number of industries as an important framework to improve the understanding of a complex organization. Its use of 'model as learning' involving description of organizations in terms of aggregate processes, policies, organization boundaries, information links and delays can lead to improved communication and sharing of mental models that ultimately break down barriers involving culture, attitudes and beliefs.

The development of system dynamics model in this study is based on the queuing theory principle. According to Warwick (2009), the general framework for the analysis of queues, categorizes queues according to four criteria; the first is the average arrival rate (λ) and pattern of arrivals (usually assumed to be random); the second is the average service rate (μ) and the pattern of service times (usually assumed to be exponential); the third is the queue discipline which is usually first-in-first-out (FIFO) and the fourth is the number of servers assigned to the queue. Warwick (2009) also added that other assumptions of the queuing process are that customers do not leave the queue once joined or that the pool of customers who could join the queue is infinite.

Therefore, it is the aim of this study to bridge the gap by developing a system dynamics model to capture the relationship and interdependency between numerous subsystems at the container terminal and at the same time be able to aid container terminal managers in both operation and strategic decision making.

2.4 Conclusion

This chapter presents a review of previous studies relating to the research topic, i.e. container terminal and system dynamics. The main purpose of reviewing these literatures was to obtain the background information for conducting our research and to find gaps in previous researches. Our research differs from previous researches in three aspects.

1. Our study integrated all the effects generated by the berth side and the yard side in order to understand their interdependency rather than considering them individually. Upon understanding the interdependency of each subsystem, potential bottlenecks can be identified and appropriate measures can be carried out.
2. Up to date, there are very limited studies on the applications of system dynamics in a container terminal. Thus, our research intends to demonstrate the strength of system dynamics and its capability in addressing problems of the container terminal.
3. Many of the applications of system dynamics focus mainly on the strategic level decision making. Our study, however, demonstrates a system dynamics model capable of supporting both strategic and operational level decision making of a container terminal operation.

CHAPTER 3

METHODOLOGY

3.1 Introduction

This chapter begins with the discussion on the source and method of data collection used in this research, followed by a presentation on data analysis. The subsequent section discusses the definition and application of system dynamics modeling, followed by a discussion on the important components of system dynamics modeling. Next, the building block of system dynamics as well as the modeling processes in system dynamics are presented. Before ending this chapter, a discussion on the procedure of validating the developed system dynamics model is presented.

3.2 Data Source and Data Collection

The data collected in this research came from both the primary and secondary data sources. Most of the data were gathered during the two-week visit to the container terminal of study; from 21 January to 1 February 2008. This research employed three methods of data collection, namely observations, interviews and documentation.

3.2.1 Observations

Observations on the real-time operations of the container terminal were conducted. The whole process starting from the moment a vessel berthed at the container terminal, followed by the process of unloading and loading containers from or onto the vessel using quay cranes, transferring containers to container yard blocks using prime movers and finally the departure of the vessel were observed. Besides observing real-time operations, the real-time decision making by officers from the Operations Department was also observed.

3.2.2 Interviews

Numerous interviews were conducted with officers from various teams in the Operations Department during the course of the two-week visit to the case study container terminal. Arrangements and appointments were made with the officers in order to have a one-to-one interview. Officers interviewed were from the berth planning team, cargo control team, yard planning and strategy team, vessel planning team, operation equipment control team as well as the gate planning team.

All of these officers were interviewed once except for officers from the berth planning team and yard strategy team because this research focused on developing a system dynamics model that could capture both the operations of the berth and yard. Therefore, additional clarifications were made with officers from both these teams in order to have a complete picture of the berth and yard operations. Table 3.1 shows the schedule of interviews conducted with the container terminal officers.

Table 3.1: Schedule of interview conducted at the case study container terminal.

DATE	OFFICERS INTERVIEWED
21 January 2008	Berth Planner Officer and Cargo Control Officer
22 January 2008	Yard Strategy Officer and Operation Equipment Control Officer
23 January 2008	Yard Planning Officer and Vessel Planning Officer
24 January 2008	Gate Officer and Berth Planner Officer
25 January 2008	Gate Officer and Yard Strategy Officer

The purpose of the interviews was to understand the operational process conducted by each team and how planning and decision making by these individual teams ultimately influenced the efficiency of the overall terminal operations.

3.2.3 Document Content Analysis

Five months worth of historical daily data, dated from January to May 2008, were provided by the case study container terminal. The data consisted of:

i. The number of arrivals for both mainlines and feeders at the container terminal.

ii. The time of arrivals and departures for both mainlines and feeders.

iii. The number of containers unloaded from mainlines and feeders.

iv. The number of containers loaded onto mainlines and feeders.

v. The number of unloaded containers entering the yard blocks.

vi. The number of loaded containers leaving the yard blocks.

vii. The number of quay crane allocated for each vessel.

viii. The length of mainlines and feeders entering the container terminal.

Besides the historical data, the officer from the Corporate Communications Department also supplied additional documents which included the number and types of quay cranes used in the container terminal. Selected historical data are exhibited in Appendix 1.

Additional annual data from year 2003 to 2008 were retrieved from the website of the Ministry of Transportation, Malaysia. The annual data reports extracted included the number of vessels (mainlines and feeders) calling at the case study container terminal and the total container throughput.

3.3 Data Analysis

System dynamics methodology was used to develop a model of the container terminal that captured the flow of containers across the berth side and the yard side. By modeling the operations of the case study container terminal, four categories of performance indicators, namely the vessel performance, berth performance, yard performance and container handling performance indicators were generated from the developed system dynamics model. These performance indicators were then benchmarked against the maximum internationally acceptable standards for berth occupancy rate according to Ray and Blankfeld (2002).

Once the performance indicators were generated and analyzed, the root cause of terminal inefficiency and bottlenecks were identified by understanding the relationship and interdependency of each subsystem. Subsequently, that particular subsystem could be focused upon to create improvement.

3.4 Components of a System Dynamics Model

The three important components in the development of a system dynamics model are feedback, delay and causal loop diagram. This section details each of these components individually.

3.4.1 Feedback

Feedback is one of the core concepts of system dynamics (Sterman, 2000). Most of the complex behaviors usually arise from the interactions (feedbacks) among the components of the system, not from the complexity of the components themselves (Sterman, 2000). According to Ruth and Hannon (2004), feedback occurred when a variable within a process is used to modify the value of another variable in the process. A feedback loop in a dynamic system can be defined as a closed-loop circle of cause-and-effect in which 'condition' in one part of the system causes 'results' elsewhere in the system, which in turn acts on the original 'conditions' to change them (Deaton and Winebrake, 2000). All dynamics arise from the interaction of just two types of feedback loops, reinforcing feedback and balancing feedback.

A reinforcing loop is a positive feedback system as it causes a variable to increase or decrease in a sustained fashion. If a company's profits go up, then it has more money to invest in research that leads to new products and new sales which then will generate more profit for the company. Similarly, a decrease in profits leads to less investment in research, fewer new products, fewer new sales and hence reduced profits.

Unlike a reinforcing loop, a balancing loop seeks stability which causes a variable to return to a target value if a change has moved it away from target. The balancing loop describes processes that tend to be self-limiting, and that seeks balance and equilibrium (Sterman, 2000). Suppose a manager controls costs closely and if costs rise above plan, the gap is noted and activities are put in place to reduce the costs. If the costs go below the plan, the system might look for ways to use the extra cash for unfunded activities. Balancing feedback is critical to the notion of management controls where such controls are used to keep performance such as budget and project on track (Ruth and Hannon, 2004).

3.4.2 Delay

Delay always occurs between taking an action and seeing the result of this action (Ruth and Hannon, 2004). Short delay means the time lapse between a cause and its effects (Maani and Cavana, 2000). Many people consider delay to be one of the primary difficulties in managing systems. In business systems, delay occurs because information or physical transfers are imbedded in business and production processes.

The problem with such delays is that the underlying cause-and-effect relationships are masked over time and space. System dynamics acknowledges the existence of such delays and identifies them in the modeling process. In the causal loop diagram, the notation '||' on the arrow is used to denote delay in the cause-and-effect relationship. Lack of awareness of system delays causes managers to make erroneous decisions or to intervene unnecessarily and harmfully (Maani and Cavana, 2000).

3.4.3 Causal Loop Diagram

Causal loop is a conceptual tool which reveals a dynamic process in which the chain effect(s) of a cause is/are traced, through a set of related variables, back to the original cause or effect (Maani and Cavana, 2000). Causal loop diagrams are excellent for quickly capturing the cause of dynamics, eliciting and capturing the mental models of individuals or teams and are best for communicating the important feedbacks that are responsible for a problem (Sterman, 2000).

A causal loop diagram consists of variables connected by arrows denoting the causal influences among the variables and the important feedback loops are also identified in the diagram (Sterman, 2000). Figure 3.1 shows an example of causal loop diagram notations. In the example, the birth rate is determined by both population and birth fraction rate. Each causal link is assigned a polarity, either Positive (+) or Negative (-) to indicate how the dependent variable changes when the independent variable changes. The important loops are highlighted by a loop identifier which shows when the loop is positive (reinforcing, R) or negative (balancing, B).

A positive link means that if the cause increases, the effect will increase as well and if the cause decreases, the effect will decrease accordingly. In the example in Figure 3.1, an increase in the fraction birth rate means the birth rate (people per year) will increase and a decrease in the fraction birth rate means the birth rate will fall or decrease as well. Overall, if the average fertility rises, the birth rate, given the population will rise and if fertility falls, the number of births will fall.

A negative link means that if the cause increases, the effect decreases and vice versa. In the example, an increase in the average lifetime of the population means the death rate (in people per year) will fall and a decrease in the average lifetime means the death rate will rise, i.e. if life expectancy increases, the number of deaths will fall and vice versa.

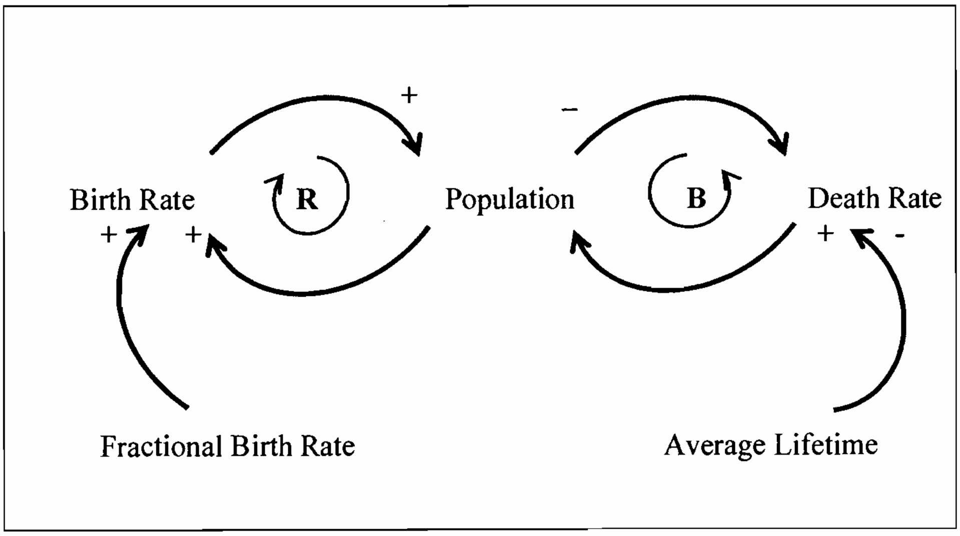
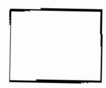
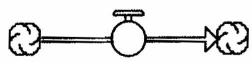


Figure 3.1: Causal loop diagram notation (Sterman, 2000).

3.5 Building Blocks of System Dynamics Model

Causal loop diagrams emphasize the feedback structure of a system while stock and flow diagrams emphasize their underlying physical structure (Sterman, 2000). According to Sterman (2000), system dynamics models are composed of four fundamental entities or building blocks which are stocks, flows, converters and connectors. The four entities described in Table 3.2 are used to build simple as well as complex models using the system dynamics approach.

Table 3.2: Building blocks of system dynamics model (Deaton and Winebrake, 2000).

NAME	DESCRIPTION	SYMBOL
Stock	A component of the system where something is accumulated. The contents of the reservoir or stock may go up or down with time.	Stock 
Flow	Activities that determine the values of reservoirs or stocks.	 Flow
Converters	System quantities that dictate the rates at which processes operate and the reservoirs/stocks change.	 Conv erters
Connecters	Define the cause-effect relationships among the different components of the system.	Connecters

Stocks are accumulated quantities within the system such as cash, population, inventories, level of knowledge or average statistical data such as average sales rate, average cash inflow and average birth rate (Maani and Cavana, 2000). Stocks characterize the state of the system and generate information upon which decisions and actions are based. Stocks are altered by inflows and outflows. They also create delays by accumulating the difference between the inflow to the process and its outflow.

Flows are the changes to the stocks that occur during a period of time. For example, revenue earned during the month or interest earned on a bank account during the quarter. The flows in the system are usually the outcome of decisions by management. Converters are other variable types which include constants, graphical relationships and behavioral relationships. In general, converters convert inputs into outputs. Connectors represent intricate connections among all the components (stocks, flows and converters) of a system.

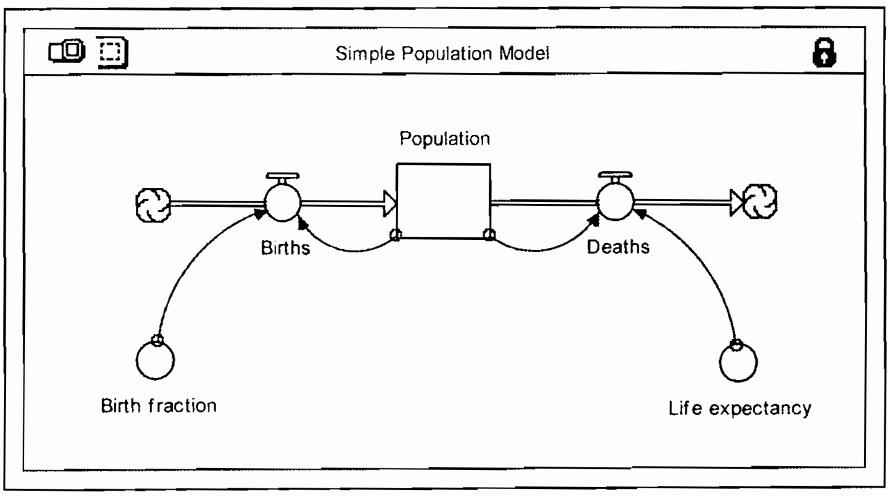


Figure 3.2: Stock flow diagram (Maani and Cavana, 2000).

Figure 3.2 is a simplified stock and flow diagram of a population model developed using *iThink* computer simulation package. In the example, stock represents population, flows represent births and deaths while birth fraction and life expectancy are represented by converters. The size of population (stock) is dependent on the rate of inflow (births) and outflow (deaths). The number of births and deaths is influenced by birth fraction and life expectancy accordingly.

3.6 System Dynamics Modeling Process

System dynamics approach to modeling emphasizes describing (by means of causal loop diagrams) and formulating equations (in a quantitative model) for each cause-and-effect relationship, and is a good example of 'open-box' approach for policy modeling (Maani and Cavana, 2000). The purpose of modeling is not only to gain insight (though insight into the problem is required to design effective policies) but to solve a problem (Sterman, 2000).

Modeling has three possible general uses. Firstly, it provides experiment with models. Secondly, a good model enables prediction of the future course of a dynamic system. Thirdly, a good model stimulates further questions about the system behavior and the applicability of the principles that are discovered in the modeling process to other systems (Ruth and Hannon, 2004).

Although different authors introduce different steps or phases in system dynamics modeling process, the goal of each step is generally the same. In this research, the modeling process suggested by Sterman (2000) was applied. According to Sterman (2000), system dynamics modeling process can be divided into five stages:

1. Problem Articulation (Boundary Selection)
2. Formulation of Dynamic Hypothesis
3. Formulation of a Simulation Model
4. Model Testing
5. Policy Design and Evaluation

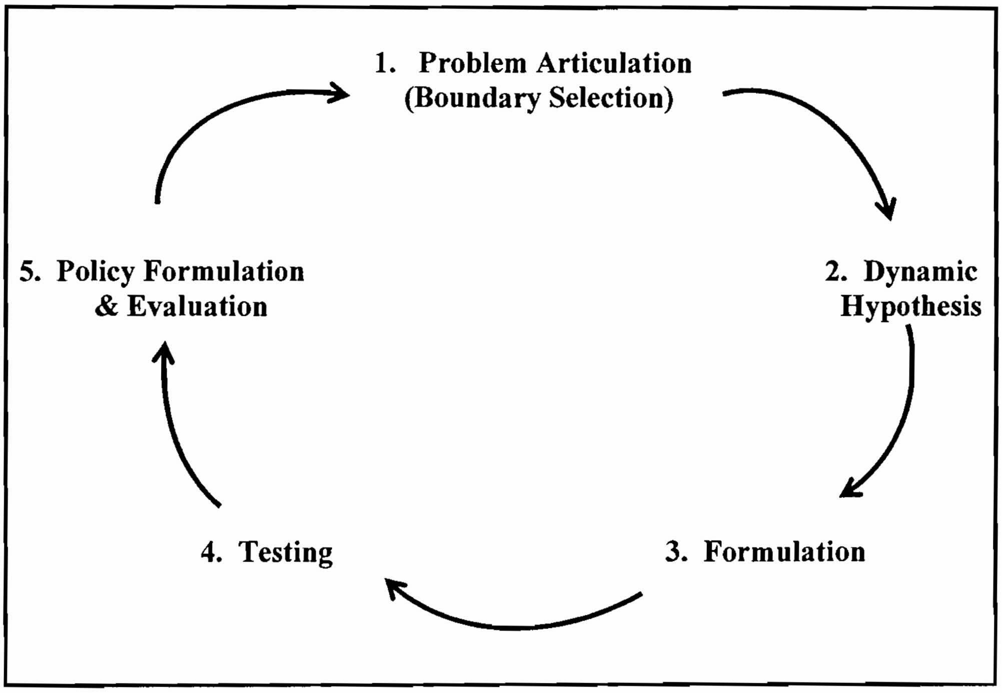


Figure 3.3: Steps of the modeling process (Sterman, 2000).

Modeling is a feedback process, not a linear sequence of steps (Sterman, 2000). Figure 3.3 shows the five steps of modeling process more accurately as an iterative cycle. Iteration can occur from any step to any other step and in any modeling project, one will iterate through these steps many times. The following sections discuss each step in detail (Sterman, 2000).

3.6.1 Problem Articulation

In this phase, the situation or issue at hand is defined and the scope as well as boundaries of the study are identified. A clear purpose is the single most important ingredient for a successful modeling study. This phase includes identifying the problem area or policy issues of concern to management. In order to identify the problem, clear objectives have to be established, taking into account multiple stakeholders and perspectives. Finally, preliminary information and data including media reports, historical and statistical records, policy documents, previous studies and stakeholder interview need to be collected.

3.6.2 Formulating a Dynamic Hypothesis

Once the problem has been identified and characterized, a dynamic hypothesis is developed to account for the problematic behavior. A dynamic hypothesis is a working theory of how the problem arose and understanding the consequences of feedback structure in the problem. Based on the initial hypothesis, key variables, reference models, other available data and map of causal structure are developed using causal loop diagrams and then stock and flow diagrams.

3.6.3 Formulating a Simulation Model

After the initial dynamic hypothesis and conceptual models have been developed, they have to be tested. Sometimes, a dynamic hypothesis can be tested directly through data collection or experiments in the real system. However, the conceptual model is usually complex and its dynamic implications are unclear. Furthermore, human ability to infer correctly the dynamics of a complex model is extremely poor. Therefore, it is much safer to conduct these experiments in a virtual world. In order to do so, the conceptual diagram developed earlier is converted to a fully specified formal model, complete with equations, parameters and initial conditions.

3.6.4 Testing

In the testing phase, the simulated behavior of the model is compared with the actual behavior of the system. Testing involves far more than the replication of historical behavior as every variable must correspond to a meaningful concept in the real world and every equation must be checked for dimensional consistency. The model developed must also be tested under extreme conditions; conditions that may never be observed in the real world. Extreme conditions tests along with other tests of model behavior are critical tools to discover the flaws in the model and set the stage for improved understanding.

3.6.5 Policy Design and Evaluation

After the model has been validated in the testing phase, it can be used to design and evaluate policies for improvement. Policy design is much more than changing the values of parameters as it includes the creation of entirely new strategies, structures and decision rules. The robustness of policies and their sensitivity to uncertainties in model parameters and structure must be assessed, including their performance under a wide range of alternative scenarios. The interactions of different policies must also be considered as real systems are highly nonlinear and the impact of combination policies is usually not the sum of their impact alone.

3.7 System Dynamics Software Application

There are many software applications developed to aid system dynamics analyses such as COSMOS, DYNAMO, DYSMAP2, *iThink*, STELLA, Vensim and Powersim. This research used *iThink* computer simulation package for system dynamics analyses because it is a powerful yet easy-learn-software (Ruth and Hannon, 2004) and it supports users at each stage of the modeling process. Bailsford et al. (2004) stated that *iThink* is a user friendly package with a drag-and-drop user interface which allows the modeler to develop the model without the need of programming.

iThink software is highly flexible and can be used to develop a variety of macro-simulation models and test a variety of scenarios. Besides that, it also offers a powerful combination for improving the quality of our mental models. The biggest advantage of *iThink* is that it is a good interface for presentation to non-modeled users (Intrapairot, 2000).

A brief comparison of some of the existing system dynamics modeling software was compiled by Constanza and Voinov (2001) as quoted in Jutla (2006). Although Jutla (2006) emphasized on the benefits of STELLA, the benefits are also applicable to *iThink*. This is because *iThink* and STELLA were developed by the same developer and possessed the same characteristics. The only difference between *iThink* and STELLA is that STELLA is more suitable for solving social science issues while *iThink* is more suitable for solving industrial and managerial issues (iseesystems, 2009)

Table 3.3: Brief comparison of selected existing system dynamics modeling software (Constanza and Voinov, 2001) in (Jutla, 2006).

Software	Developers	Extendible	User Friendly	Learning Curve
STELLA / *iThink*	High Performance Systems (www.hps-inc.com)	No	5	5
Powersim	Powersim (www.powersim.com)	Somewhat	5	4
Extend	Imagine That (www.ImagineThatInc.com)	Yes	4	3
Simulink	Mathworks (www.mathworks.com)	Yes	3	2
Vensim	Ventana (www.vensim.com)	No	5	5
Model Maker	CS (www.cherwell.com)	No	5	5

Table 3.3 gives a brief comparison between various system dynamics software. Extendible system is explained as the ability of the software to offer tools to incorporate some additional user-defined functionality. On the other hand, user-friendliness (maximum=5) is a subjective estimate on how easy it is to use the software. Lastly, learning curve (maximum=5) is an estimate of how easy it is to use the software based on the number of hours needed to formulate a simple one-variable model and run it starting from scratch. A higher value indicates that the software is more user-friendly and easy to learn compared to a lower value.

From Table 3.3, STELLA or *iThink* scored the maximum value of 5 for both user-friendly and easy to learn. The development of the system dynamics model in our research did not require extendibility to other additional software. Therefore, *iThink* was deemed a suitable software package for this research to support the development of container terminal operations.

3.8 Validation of System Dynamics Models

According to Forrester and Senge (1980) in Maani and Cavana (2000), before a model can be used for policy analysis or any other purpose, the model team (the user or management) must have sufficient confidence in the 'soundness and usefulness' of the model. Verification and validation tests must be performed on the developed system dynamics model in order to test the reliability of the developed model in mirroring the actual container terminal process.

In order to verify the system dynamics model developed in this research, a conceptual model reflecting the operation of the case study container terminal was first constructed. This conceptual model along with its structures and parameters were then verified by the case study container terminal's Yard Strategist Officer as correctly reflecting the terminal operations. After the conceptual model was verified, the actual system dynamics model was developed according to the verified conceptual model. This was to ensure that the developed system dynamics model accurately represented the actual operations in the case study container terminal.

As for the validation test, the developed system dynamics model was run for five months and the results obtained were compared with the five months of actual data supplied by the Statistics Officer at the case study container terminal. Elements that were compared included the number of vessels entering the container terminal, the number of containers unloaded and loaded at each berth, berth occupancy rate and yard occupancy rate. A detailed description on validation test is discussed in Chapter 5.

3.9 Conclusion

This chapter presents a detailed discussion on system dynamics methodology, as system dynamics is used to support the development of this research. Besides that, the discussion on the overview of how this research was carried out is also presented. The next chapter focuses on the description of operations in a typical container terminal as well as the actual terminal operation processes in the case study container terminal.

CHAPTER 4

SITE DESCRIPTION

4.1 Introduction

There are two main parts in this chapter. The first section focuses on the operational processes of a general container terminal system including vessel arrival, unloading and loading containers as well as distribution of containers to the container yard. The later part emphasizes on the detailed operations of the case study container terminal. This section encompasses the chronicles of planning processes carried out by the berth planning team, yard planning team and vessel planning team at the case study container terminal. This chapter aims at providing the fundamental understanding on how the case study container terminal operates before the formal modeling.

4.2 Current Operation in the Case Study Container Terminal

The case study container terminal is a transshipment hub port, where a majority of the operations revolve around the berth and yard sides. At the time of this research being conducted in the year 2008, there were eight berths operating simultaneously with the length of 360 m each. However, recently two new berths have been added, making the total length of berth as 3600 m. There are also a total of 28 quay cranes used for container unloading and loading process.

As for the capacity of the terminal's container yard, currently, there are 144 yard blocks where the maximum capacity for each block is 1,050 TEU (30 rows * 5 height * 7 rows). Each slot in the yard block is designated for both the container size of 20 TEUs and 40 TEUs.

Shipping lines that call in the case study container terminal are grouped according to 'Service' or their sailing route. Currently, there are nearly 70 Services being handled by the container terminal. One Service represents one vessel where this vessel's travelling route is different from the other 69 Services. Each Service calls nearly once a week. The next section focuses on the discussion of the three important planning teams of the case study container terminal, i.e. the berth planning team, yard planning team and vessel planning team.

4.2.1 Berth Planning

The objectives of berth planning is to ensure optimum berthing of vessels, meeting lines requirements, while facilitating high berth utilization and productivity. After receiving the schedule of arrival from the vessel, the berth planners start to plan the berthing location for the vessel.

Each morning, the berth planners will discuss with the shift manager to compare the planned events for the last 24 hours against the last 24 hours of actual events. Berth planners will then update the berthing plan and prepare berthing schedule for the next 24 hours. The berth planners will also liaise with the technical executive to check on whether there are any quay cranes to be taken out for maintenance for the next 24 hours. If there are any changes in the vessels' estimated time of arrival

(ETA) and the berth plan or cranes allocation, the berth planners will inform the shift manager.

Berth planners will also make a continuous liaison with the shipping lines in case of any changes in the arrival of vessels. Besides that, berth planners also have to retrieve the daily schedules from the respective shipping lines and constantly make updates on the berthing plan. Information received from vessels are the expected time of arrival, the expected time of departure as well as the number of containers to be discharged and loaded. The berth planners will use this information to develop the berthing locations for the vessels and decide on the number of quay cranes to be used to load and unload the containers from and onto the vessels.

There are two types of vessels served at the case study container terminal: mainlines and feeder ships. If both of these vessels are scheduled to arrive at the container terminal around the same time, a mainline vessel will be given a higher berthing priority as compared to a feeder ship.

The shipping lines have to communicate with the berth planners preferably 24 to 48 hours prior to the vessel's estimated time of arrival. Once the schedules are confirmed, the berth plan will be finalized. The confirmed schedule will also be uploaded into the database by the berth planners. All the finalized information and data regarding the operations of the container terminal will be stored into this database.

The finalized berth plan contains information such as the berthing location for vessels, their estimated time of arrival and departure and the number of containers that will be unloaded. The confirmed berth plan will then be disseminated to shipping lines and the yard planners.

4.2.2 Yard Planning

The yard planning team is responsible for planning and allocating space in the yard blocks to the containers that will be discharged from vessels while ensuring maximum yard productivity and efficiency. The yard planning team is divided into dynamic team and static team. The dynamic team operates in real-time operation to monitor the actual discharging process from the vessels, while the static team is responsible for planning the storage location for discharged containers prior to the arrival of vessels.

The static team is responsible for the discharge operations, 12 hours before the vessels' arrival and two hours after the vessels depart from container terminal. If the vessel is already on board, the dynamic team will be responsible for any changes in the discharge operations.

4.2.2.1 Discharge Operations

The static team consists of yard planners and yard strategists. A yard strategist plans storage locations for containers that will be discharged from vessels while yard planners in the static team update the data regarding discharged containers from the vessels into the database.

After the berth planners have finalized the berth plan, the yard strategist will be given a copy of the finalized berth plan. The berth plan will serve the yard strategist as a guideline in developing the yard plan. The yard plan is a prescheduled plan for storing containers that will be discharged into the container yard. Normally, the yard strategist will pre-schedule the yard plan before the arrival of the vessels while the yard planners will conduct the actual discharge operations according to the yard plan finalized by the yard strategist.

Discharged containers will be stored in the yard blocks according to Services. Each yard block may contain two to three different types of Services. The decision rules that the yard strategist uses to allocate containers that will be discharged onto the best location in the container yard block is that; firstly, the storage location of the discharged containers must not be more that 800m from the berthing point of the next vessel that will pick up the discharged containers.

Secondly, Services that are being stored in the same yard block must not at the same time be loaded onto different vessels. For example, if Service A in Yard Block 01B is scheduled to be loaded onto Service B on Monday at 7 a.m., then the other Services that will be stored in Yard Block 01B must be Services that do not have the same loading schedule as Service A. In short, no two Services in one yard block can be loaded onto the vessel simultaneously.

The container yard consists of 144 yard blocks and there are nearly 70 Services berthing at the case study container terminal every week. The yard strategist needs to allocate the containers that will be discharged from over 70 Services onto the most optimum storage location among the 144 yard blocks while fulfilling the two decision rules. The finalized yard plan will be used by the yard planners in the dynamic team to conduct and monitor the actual discharging process when the vessel actually arrives at the container terminal. The yard planners of the static team will upload the number of discharged containers from the vessels into the database.

4.2.3 Vessel Planning

The vessel planning team is responsible for the process of loading containers onto vessels. As with the yard planning team, the vessel planning team also consists of two teams. The static team is in charge of the planning procedure 12 hours prior to the arrival of the vessel and two hours after the vessel's departure. On the other hand, the dynamic team is in charge of monitoring the actual container loading process when the vessel arrives at the container terminal.

4.2.3.1 Load Operations

Vessel planners of the static team are in charge of developing loading plan 12 hours before the vessel's estimated time of arrival. Prior to arrival, vessels will provide the vessel planners with number of containers to load onto the vessels as well as the special location slots in the vessel where the loaded containers will be placed when they are loaded onto the vessel. However, it is up to the vessel planners to decide on the stacking position of the loaded containers in the vessel.

Vessel planners usually decide according to the stowage instruction and subject to where the loaded containers will be discharged, the weight of the loaded containers as well as the classification of the containers. The loaded containers that will not be discharged in the next container terminal must not be placed above containers that will be discharged in the next container terminal, because this will cause additional reshuffling when the vessel calls at the next container terminal. Upon the arrival of the vessel at the case study container terminal, the vessel planners of the dynamic team will submit the loading plan finalized by the vessel planners of the static team to the captain of the vessel.

4.3 Conclusion

A container terminal is not merely a simple connection between transportation. The internal operations of a container terminal are particularly complex as it consists of many subsystems. In order for the whole container terminal to perform efficiently, the operation of each sub-system is equally important as well. Besides the berth planning team, yard planning team and vessel planning team, there are also the operational equipment control team and gate team. However, as this research focuses only on understanding the relationship and interdependency of operations between the berth and yard, therefore only the discussion of berth planning, yard planning and vessel planning are emphasized. The next chapter focuses on the construction of a system dynamics model.

CHAPTER 5

MODEL CONSTRUCTION

5.1 Introduction

This chapter focuses on the construction of a system dynamics model to achieve the first three objectives of this research, which are:

1. To model the flow of containers across the berth side and yard side to evaluate the system and to measure performance indicators. The performance indicators that are measured are vessel, berth, container handling and container yard performance indicators.
2. To evaluate the relationship and interdependency between the berth and yard operations for improvement creation and understanding of the terminal overall efficiency.
3. To develop a capacity planning to ensure the terminal has sufficient capacity to sustain demand and yet avoid overcapacity.

This chapter begins with the discussion on the boundary of the developed model, followed by a detailed explanation on model conceptualization. The conceptual model is then translated into a causal loop diagram, which is further discussed in the model realization in the system dynamics section. Model formulation and verification and validation are discussed next. The results generated from the developed system dynamics model are presented next in the data analysis section. This chapter ends with a short conclusion.

5.2 Setting the Model Boundary

The first objective of this research is to model the flow of containers across the berth side and the yard side. As the case study container terminal functions as a transshipment hub port, most of its operations revolve around the berth and yard. Therefore, only these two were considered in this research. When data was collected at the case study container terminal in 2008, there were a total of eight berths with the length of 360m each, 28 quay cranes and 144 yard blocks. Although changes in the number of facilities after the data collection period were not included in the developed system dynamics model, the development of Microworlds allows users to experiment the impact of increasing terminal facilities on the case study container terminal operations. As decision made at the operational level will influence the decision making at the strategic level and vice versa, this research investigated planning and decision making at both the operational and strategic levels.

5.3 Model Conceptualization

Before developing the actual system dynamics model, a conceptual model was first developed. The conceptual model which captures the operating processes at the case study container terminal is illustrated in Figure 5.1. The processes are grouped according to where the processes took place, i.e. at the berth or yard. All the physical processes illustrated in Figure 5.1 are simulated in the developed system dynamics model.

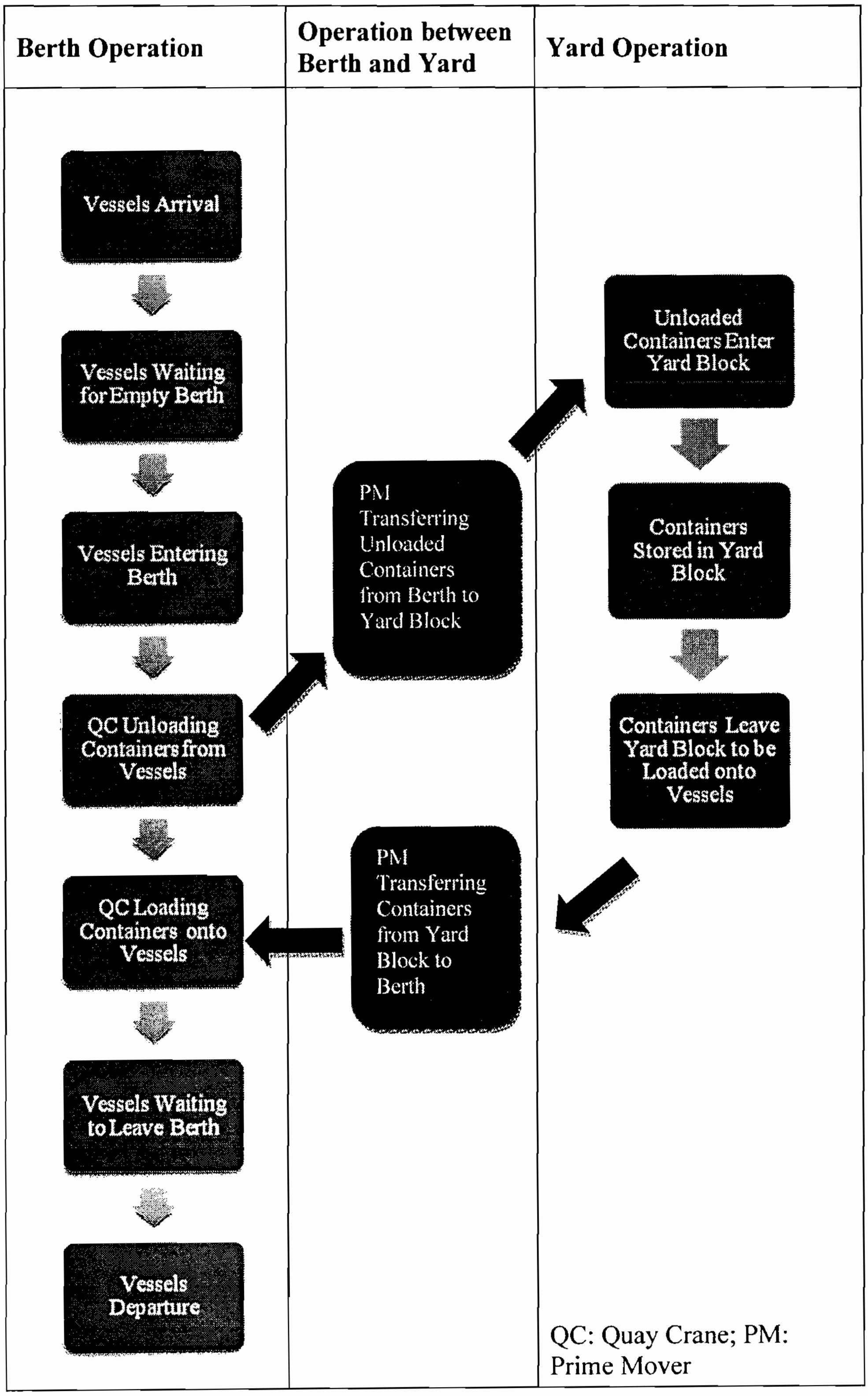


Figure 5.1: Conceptual model of the operation process at the case study container terminal.

A typical berth operation starts off with the arrival of a vessel at the terminal, followed by the vessel waiting for berth, the vessel entering berth, containers unloading and loading of the vessel, its waiting to leave berth and finally the vessel departing from the container terminal. Operations that occurred between the berth and yard are the activity of prime movers transporting containers between these two subsystems. Finally, the containers are stored temporarily at the yard. Detailed discussion on the operation processes conducted at different subsystems was carried out next.

Upon a vessel's arrival at the container terminal, it is directed to its destined berth space. If the berth space is occupied, the vessel has to wait for the berth to be vacant. If the berth space is vacant, the vessel may enter the berth space directly. On average, each vessel takes around 45 minutes to be moored into the berth. Mainline vessels are given a higher berthing priority as compared to feeder ships if both of these vessels arrive within the same time range. Upon entering the berth, containers are unloaded from the vessel using quay cranes. Typically, two to three quay cranes are assigned to each vessel and on average, each quay crane performs an average of 33 moves per hour.

Containers that have been unloaded from the vessel using quay cranes are transferred to the container yard block using prime movers. Seven prime movers are assigned under each quay crane. Prime movers transfer unloaded containers to its predetermined storage location at the yard. Upon completing the unloading process, the container loading process starts. Again, prime movers are used to transport containers from the yard to the berth side. Upon reaching the berth, quay cranes transfer the containers from the prime movers and load them unto the vessel. As the container loading process is completed, the vessel spends around an additional 45 minutes getting ready to leave the berth. Finally, the vessel departs from the container terminal.

Although the conceptual model is useful in describing the flow of processes occurring at each subsystem, it is not capable of showing the relationship and interdependency between the processes and their effect on the overall terminal efficiency. Therefore, there is a need for a tool that is capable of capturing the dynamics of the case study container terminal operational processes.

5.4 Model Realization in System Dynamics: Causal Loop Diagram

The conceptual model which describes the sequences of operations that took place at the case study container terminal as illustrated in Figure 5.1 is translated into the causal loop diagram. A causal loop diagram is capable of capturing dynamic processes of a system by demonstrating the chain effect through a set of related variables, back to the original cause or effect. Figure 5.2 shows the causal loop diagram of the case study container terminal operation process while Table 5.1 demonstrates the simplified notation of feedback loops in the case study container terminal operation process.

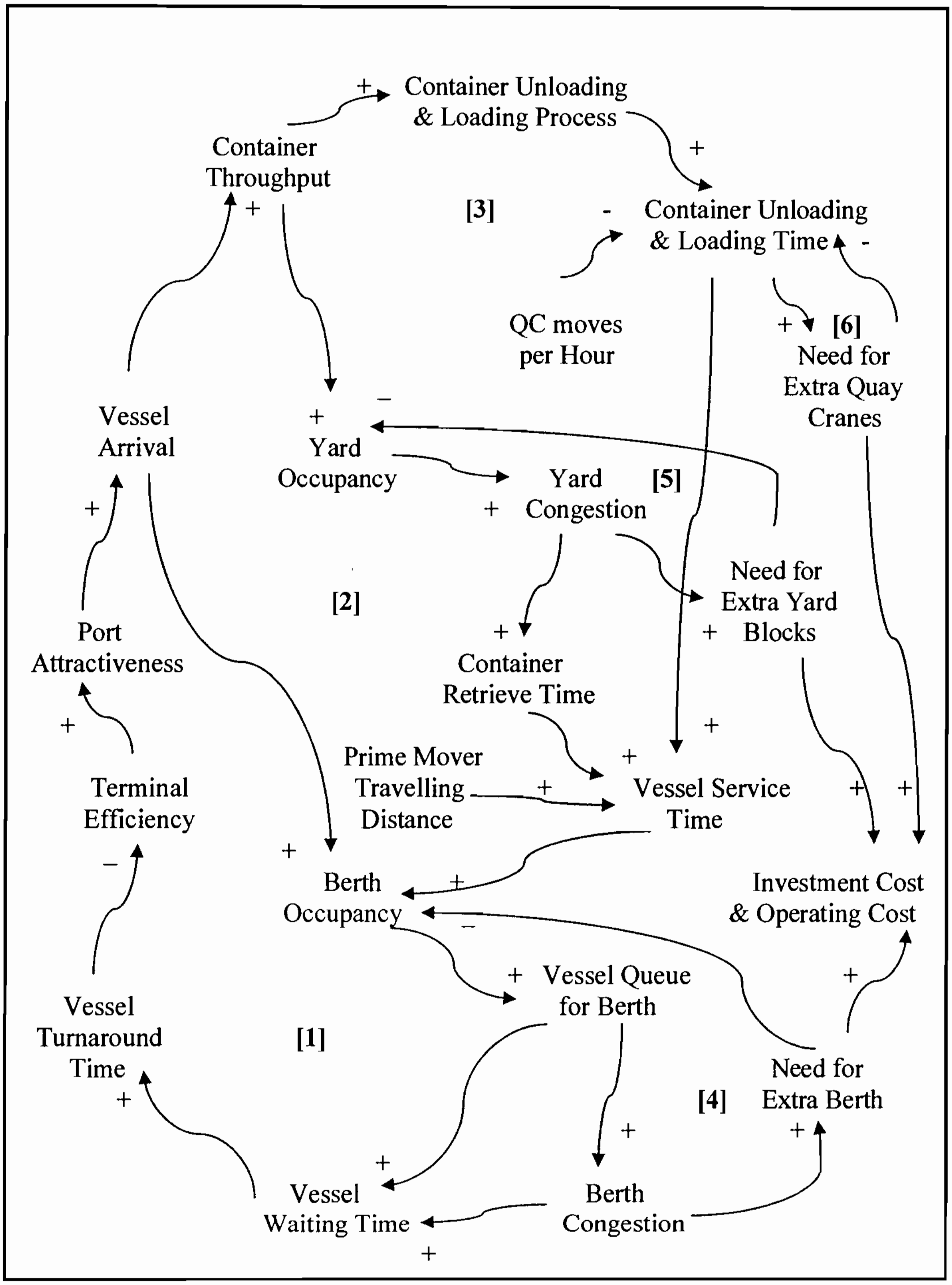


Figure 5.2: Causal loop diagram of case study container terminal operation process.

Table 5.1: Simplified notation of feedback loops in case study container terminal operation process.

Loop	Description	Type of Loop
Loop [1]	Vessel Arrival +> Berth Occupancy +> Vessel Queue for Berth +> Berth Congestion +> Vessel Waiting Time +> Vessel Turnaround Time - > Terminal Efficiency +> Port Attractiveness - > Vessel Arrival	Balancing Loop
Loop [2]	Vessel Arrival +> Container Throughput +> Yard Occupancy +> Yard Congestion +> Container Retrieve Time +> Vessel Service Time +> Berth Occupancy +> Vessel Queue for Berth +> Berth Congestion +> Vessel Waiting Time +> Vessel Turnaround Time - > Terminal Efficiency +> Port Attractiveness - > Vessel Arrival	Balancing Loop
Loop [3]	Vessel Arrival +> Container Throughput +> Container Unloading & Loading Process +> Container Unloading & Loading Time +> Vessel Service Time +> Berth Occupancy +> Vessel Queue for Berth +> Berth Congestion +> Vessel Waiting Time +> Vessel Turnaround Time - > Terminal Efficiency +> Port Attractiveness - > Vessel Arrival	Balancing Loop
Loop [4]	Berth Occupancy +> Vessel Queue for Berth +> Berth Congestion +> Need for Extra Berth - > Berth Occupancy	Balancing Loop
Loop [5]	Yard Occupancy +> Yard Congestion +> Need for Extra Yard Blocks - > Yard Occupancy	Balancing Loop
Loop [6]	Container Unloading & Loading Time +> Need for Extra Quay Cranes - > Container Unloading & Loading Time	Balancing Loop

Loop [1] shows the dynamics of berth operations. An increase in vessel arrival increases the berth occupancy rate and thus, increases the number of vessels queuing for berth. This further increases berth congestion problem, vessel waiting and turnaround time. The increase in vessel turnaround time leads to the decrease of terminal efficiency, port attractiveness and the number of vessels as vessels choose to call on other ports.

Loop [2] on the other hand exhibits the dynamics of yard operations. The increase in vessel arrival also increases the container throughput; and yard occupancy as more space is needed to store the increased number of containers. The increase in yard occupancy leads to the problem of yard congestion and therefore this increases the time needed to retrieve the containers from yard to berth as terminal operators resort to higher stacking of containers in the yard block to save space. When the container retrieval time increases, the vessel service time also increases. This leads to an increase in berth occupancy, number of vessels queuing for berth, berth congestion, vessel waiting and turnaround time, thus decreasing terminal efficiency, port attractiveness and vessel arrival.

Loop [3] presents the effect of an increase in container throughput on the unloading and loading process. As an increase in vessel arrival increases container throughput, the immediate effect is the increase in container unloading and loading process because more containers need to be unloaded and loaded. This therefore increases the container unloading and loading time as additional time is needed to process the containers. This further increases vessel service time, vessel occupancy, vessel queue for berth, berth congestion, and vessel waiting and turnaround time. The increase in vessel turnaround time decreases the terminal efficiency, port attractiveness and finally the number of vessels' arrival due to inefficient terminal services.

Loops [4], [5] and [6] illustrate the need for further investment in additional facilities. In Loop [4], the increase in berth occupancy increases the number of vessels queuing for berth, hence increasing the berth congestion problem. This also increases the need for extra berths. The increase in the number of berth decreases berth occupancy. The same goes for loop [5] where the increase in yard occupancy further increases the yard congestion problem and therefore increases the need for extra yard space. Building extra yard space decreases yard occupancy. As for loop [6], as the container unloading and loading time increases, the need for extra quay cranes also increases. The increase in the number of quay cranes decreases the container unloading and loading time.

The need for extra berths, yard blocks and quay cranes increases the terminal operating and investment costs. The additional two elements that are not part of any loop but that do influence the process are prime mover traveling distance and quay crane moves per hour. An increase in quay crane moves per hour decreases the time used for unloading and loading containers. This also decreases the need for extra quay cranes. Increase in prime mover traveling distance also increases vessel service time as more time is spent to transfer containers from yards to blocks. Therefore, the location of containers stored in the yard block needs to be as near as possible to the container loading point at the berth.

5.5 Model Formulation

The causal loop diagram as discussed in the previous section was translated into the stock and flow diagram which is the essence of system dynamics modeling. The following section discusses the development of system dynamics model using *iThink* software. *iThink* software consists of four layers for constructing a system dynamics model. The first layer is the interface layer. The interface layer is used to develop a user-friendly graphical interface. Users will be able to navigate through the system dynamics model without the need for dealing with the complexity of the model and its equations. The map layer, which is the second layer, is where the system dynamics model is developed. Once the construction of a model is completed in the second layer, numerical value or logical equation is inputted into the model in the model layer or the third layer. The forth layer is the equation layer and this layer holds the mathematical equations of the system dynamics model developed at the map and model layers (see Appendix 8).

The subsequent section discusses the development of the system dynamics model at the model layer. The discussion is divided into two parts, where the first part emphasizes on developing the model at the operation level while the second part describes the construction of a strategic level model. The second part also presents how decision making at the operation level can affect decisions at the strategic level and vice versa.

5.5.1 Operation Level Model Formulation

The purpose of constructing an operation level model is to achieve the first and second objectives of this research:

1. Model the flow of containers across the berth side and yard side to evaluate the system and to measure performance indicators. The performance indicators that are measured are vessel performance indicators, berth performance indicators, container handling performance indicators and container yard performance indicators.
2. Evaluate the relationship and interdependency between the berth and yard operations for improvement creation and understanding of the terminal overall efficiency.

The model at the operation level is run by hours. Here, the model is clustered into two different sectors, namely the berth and yard sectors. These two sectors mirror the actual operations of the case study container terminal.

5.5.1.1 Berth Sector at Operation Level

The berth sector captures operations that take place at the berth. The task of the berth planners is to allocate the limited number of berths to incoming vessels. There were eight berths (linear and of the same size) operating in the case study container terminal when this study was carried out. Decision on how to allocate these eight berths to incoming vessels depends on the size of the vessels against the available berth space. The second rule is that a mainline vessel is given a higher priority to berth first as compared to the feeder ship if both arrive at the same time.

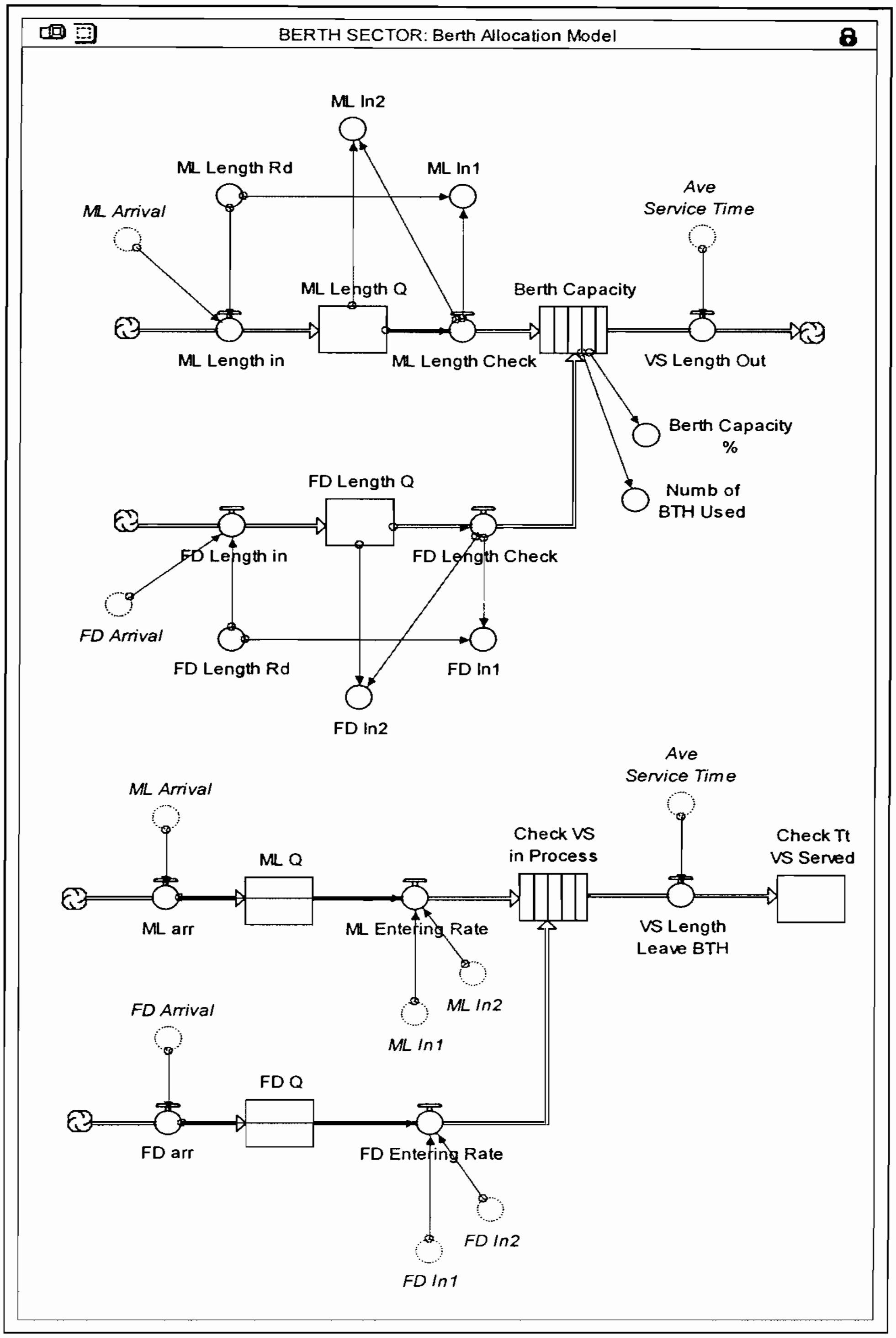


Figure 5.3: The decision model of berth allocation process.

Figure 5.3 exhibits the decision model for berth allocation process. Figure 5.3 consists of two models where the first model decides on whether the vessels may or may not enter the berth by comparing the length of the vessels against the available berth space. The function of the second model is to convert the first model, with unit in "vessel length" to the unit "number of vessel".

The *ML Length in* flow represents the total length of mainlines that arrives at the terminal per day while *FD Length in* flow represents the total length of feeder ships that arrives at the terminal per day. By multiplying the number of vessels arrived per day with the length of vessels arrived per day, the total length of vessels that arrived can be computed. The formulation of *ML Length in* flow and *FD Length in* flow are:

$$\textit{ML Length in } \text{flow} = \textit{ML Length Rd} * \textit{ML Arrival} \qquad (5.1)$$
$$\textit{FD Length in } \text{flow} = \textit{FD Length Rd} * \textit{FD Arrival} \qquad (5.2)$$

Both *ML Length Rd* converter and *FD Length Rd* converter hold the randomization value of mainline vessel's length and feeder ship's length respectively. The formulas of these two converters are:

$$\textit{ML Length Rd } \text{converter} = \text{ROUND (RANDOM (230, 430))} \qquad (5.3)$$
$$\textit{FD Length Rd } \text{converter} = \text{ROUND (RANDOM (128, 240))} \qquad (5.4)$$

The function RANDOM (x, y) generates random number between x and y where x represents the minimum value while y represents the maximum value. Therefore the values 230m and 430m represent the minimum and maximum length of mainlines that enter the case study container terminal, while 128m and 240m represent the minimum and maximum length of feeder ships that enter the case study container terminal respectively. The function ROUND (RANDOM (x, y)) rounds the randomized value to its nearest integer value.

ML Arrival converter and *FD Arrival* converter represent the number of mainlines and feeder ships entering the case study container terminal. These two converters are discussed in detail at the later part of this section. The dotted converter of *ML Arrival* and *FD Arrival* are actually replications of the actual *ML Arrival* converter and *FD Arrival* converter in Figure 5.4 on page 96. This function is called the Ghost Tool in *iThink* software. The purpose of this tool is to make replicas or shortcuts for individual stock, flows and converters. A ghost of an entity has no independent identity as it is simply a replication of the ghosted entity.

ML Length in flow and *FD Length in* flow represent the total length of vessels that arrive per day. If the value of *ML Length in* flow or *FD Length in* flow is smaller than the capacity of *Berth Capacity* conveyor, then the vessels are allowed to enter the *Berth Capacity* stock via *ML Length Check* flow for mainlines and *FD Length Check* flow for feeder ships. *ML Length Check* flow is assigned as priority number 1 while *FD Length Check* flow is assigned as priority number 2 as higher berthing priority is given to mainlines. If the flow of *ML Length in* or *FD Length in* are greater that the *Berth Capacity* conveyor, then the value of these two flows are accumulated respectively in *ML Length Q* stock and *FD Length Q* stock.

The *Berth Capacity* conveyor holds the maximum capacity of the berth. The total capacity of the *Berth Capacity* conveyor is the total length of all available berths. The *Berth Capacity* conveyor is designed to hold up to fifteen berths with the total length of 5400m (each berth is 360m). There were a total of eight berths operating when this research was conducted at the case study container terminal. Therefore, by default, the capacity of *Berth Capacity* conveyor is set to 2880m (8 berths x

360m). As the *Berth Capacity* conveyor is designed to accommodate up to fifteen berths, with the remaining seven berths, users may experiment whether extra berth capacity influences their terminal operations before making the decision to construct new berths. Table 5.2 shows the length of the extra seven berths.

Table 5.2: Number of berths and its length.

Number of Berths	Length
Default 8	2880m
9	3240m
10	3600m
11	3960m
12	4320m
13	4680m
14	5040m
15	5400m

The total vessel length is released from the *Berth Capacity* conveyor via *VS Length Out* flow. This flow is dependent on the *Ave Service Time* dotted converter where this converter represents the vessel's average service time. The *Berth Capacity %* converter holds the current berth capacity rate while the *Numb of BTH Used* converter holds the number of berths currently being used. Formulas for both *Berth Capacity %* converter and *Numb of BTH Used* converter are:

$$\textit{Berth Capacity \%}\text{ converter} = [(\textit{Berth Capacity} / \text{CAP}(\textit{Berth Capacity})) * 100] \quad (5.5)$$

$$\textit{Numb of BTH Used}\text{ converter} = [\text{CAP}(\textit{Berth Capacity})/360] \quad (5.6)$$

The function of *ML In1* converter and *ML In2* converter is to translate the unit *length* into unit *vessel*. For example, if the value of *ML Length Check* flow is 783m at t = 4, this means that at t = 4, total length of mainlines that ought to enter the berth of capacity 2880m is 783m. However, the value 783m only represents the total length of mainlines that enter the berth at that given time, not the number of mainlines that enter the berth at the given time. Therefore, the number of mainlines represented by the value 783m needs to be determined. This representation also applies to *FD In1* converter and *FD In2* converter.

The purpose of the first model is to check the length of the arrived mainlines and feeders against the available berth space; the output from the first model is calculated in the unit of *length* (meters). On the other hand, the purpose of the second model is to translate the unit *length* of the first model into the unit *vessel* (how many vessels arrived, queued for berth and are being processed).

The second model starts with *ML arr* flow which represents the arrival of mainlines in unit *vessel* and *FD arr* flow which represents the arrival of feeders in unit *vessel*. *ML Arrival* converter influences the *ML arr* flow while *FD Arrival* converter influences the *FD arr* flow. The explanation of *ML Arrival* converter and *FD Arrival* converter is discussed at the later part of this section. Upon arrival at the container terminal, vessels enter *Check VS in Process* conveyor to be processed via *ML Entering Rate* flow for mainlines and *FD Entering Rate* flow for feeders. Vessels that are not able to enter the berth due to limited space queue in *ML Q* stock for mainline and *FD Q* stock for feeder, waiting for the *Berth Capacity* conveyer to free some space.

ML Entering Rate flow represents the number of mainlines entering the berth per day. The *ML In1* converter and *ML In2* converter control the number of mainlines that enter *ML Entering Rate* flow. As discussed in the first model, *ML In1* converter and *ML In2* converter convert the total length of mainlines into the estimated number of mainlines that represent the total length. The same goes for *FD In1* converter and *FD In2* converter which control the *FD Entering Rate* flow, representing the number of feeders that enter the berth daily.

Vessels are released from *Check VS in Process* conveyor via *VS Length Leave BTH* flow. *VS Length Leave BTH* flow represents the number of vessels leaving the berth after being processed. This flow is dependent on the *Ave Service Time* converter. The *Check Tt VS Served* stock accumulates the total number of vessels served by the case study container terminal.

While the discussion on Figure 5.3 so far revolved around formulating a model to represent the berth allocation decision process, the next section emphasizes on developing a model that captures the physical activities occurring at the berth. Figure 5.4 exhibits the system dynamics model of vessel arrival, process and departure from the container terminal.

Activities at the berth side start with the arrival of vessels. The data of vessel arrival from January to May 2008 is inputted into Arena Input Analyzer in order to determine the suitable distribution for the vessel arrival. The Arena Input Analyzer is a powerful tool, used to determine the quality of fit of probability distribution functions to input data. From the Arena Input analyzer, Poisson distribution is the best distribution that fits the data of vessel arrival as the square error (0.00163) is the smallest among ten other suggested distributions (see Appendix 2). Therefore, the vessel arrival is generated using the Poisson distribution.

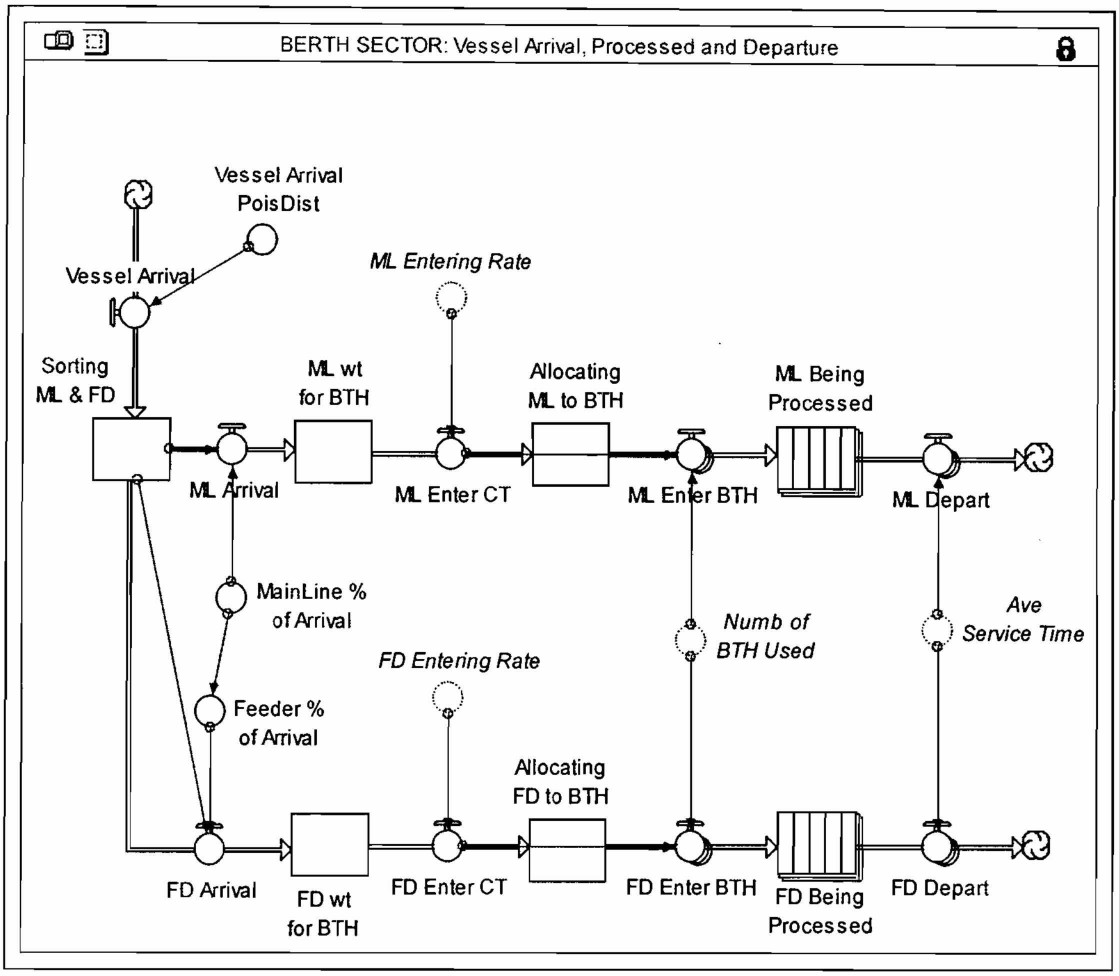


Figure 5.4: The process of vessel arrival, processed and departure.

The *iThink* software provides built-in statistical functions, which includes the Poisson distribution. From the five months of raw data supplied by the statistics officers at the case study container terminal, the mean of vessel arrival was calculated as 9.0855. Using the calculated mean, the vessel arrival was generated using the Poisson distribution function provided by the *iThink* software. The *Vessel Arrival PoisDist* converter holds the formula:

Vessel Arrival PoisDist converter = POISSON (9.0855) (5.7)

This equation generates the vessel arrival rate using Poisson distribution with the mean of 9.0855. *Vessel Arrival* flow which represents the number of vessels arriving at the container terminal is influenced by the value generated by the *Vessel Arrival PoisDist* converter.

The vessels that enter via *Vessel Arrival* flow were sorted according to mainline or feeder ship by the *ML Arrival* flow and *FD Arrival* flow. The breakdown of arrival percentage for these two types of vessels is 54% for mainline and 46% for feeder based on the five-months data provided. These values were captured by *MainLine % of Arrival* converter and *Feeder % of Arrival* converter through the given formulas:

MainLine % of Arrival converter = 54 (5.8)
Feeder % of Arrival converter = (100 – *MainLine % of Arrival*) (5.9)

ML Arrival flow and *FD Arrival* flow execute the sorting process according to the breakdown of arrival percentage defined in *MainLine % of Arrival* converter and *Feeder % of Arrival* converter. *ML Arrival* flow and *FD Arrival* flow can also be seen in Figure 5.3 on page 90 as dotted converter as these two flows represent the number of mainlines and feeders that ought to enter the container terminal. *ML Arrival* flow and *FD Arrival* flow hold the formulas:

ML Arrival flow = ROUND (*Sorting ML & FD* * (*MainLine % of Arrival* / 100)) (5.10)
FD Arrival flow = ROUND (*Sorting ML & FD* * (*Feeder % of Arrival* / 100)) (5.11)

When a mainline or feeder arrives at the container terminal, it is assigned to its berthing location. A mainline or feeder has to queue if the berthing location is occupied. The *ML wt for BTH* stock and *FD wt for BTH* stock represent this process. When the berth space is available, the queued mainline or feeder is allowed to enter the berth space. The rate of these queued vessels entering the berth is controlled by the dotted converter of *ML Entering Rate* and *FD Entering Rate*.

The dotted converter of *ML Entering Rate* and *FD Entering Rate* is the replication of flows with the same name in Figure 5.3. In Figure 5.3, *ML Entering Rate* flow and *FD Entering Rate* flow control the decision of mainlines and feeders entering the berth by judging the available berth length. *ML Enter BTH* flow and *FD Enter BTH* flow both direct mainlines and feeders into the berth. The number of berths that are operating is controlled by the *Numb of BTH Used* dotted converter (discussed in Figure 5.3). The developed operational level model is designed for fifteen berths by using the array function.

Array is another function supported by *iThink* software, which is a very useful mechanism for representing repetitive processes. Only one dimension of array is used in this model to represent the number of berths. Therefore, if any of the variables (stock, flow and converter) in this model is defined as an array, the variable will automatically be transformed into an arrayed variable that represents the process of the 15 berths.

Upon entering the berth, container unloading and loading process starts or in short, mainline and feeder are processed (represented by *ML Being Processed* stock and

FD Being Processed stock). Discussion on container unloading and loading process are presented in the next section. After completing the container unloading and loading process, vessel departs from container terminal via *ML Depart* flow for mainlines and *FD Depart* flow for feeders. The rate of vessel departure is influenced by the *Ave Service Time* converter. *Ave Service Time* converter is discussed in detail in the next section.

After discussing how the system dynamics model is developed to represent the physical activities at the berth and also how mainline and feeder are being assigned to berth, the next section emphasizes on the construction of the model that represents container unloading and loading process at the berth.

As vessel enters the berth, it is processed. A vessel being processed signifies the process of containers being unloaded from and loaded onto the vessel. Figure 5.5 exhibits the container unloading process. Container unloading process begins as the vessel is berthed. The processes in Figure 5.5 consist of 15 arrays, representing 15 berths. Each array describes the process of containers being unloaded from a vessel at each berth individually. The logic or formula for each array however is similar for all 15 berths. By default, only eight berths are operating.

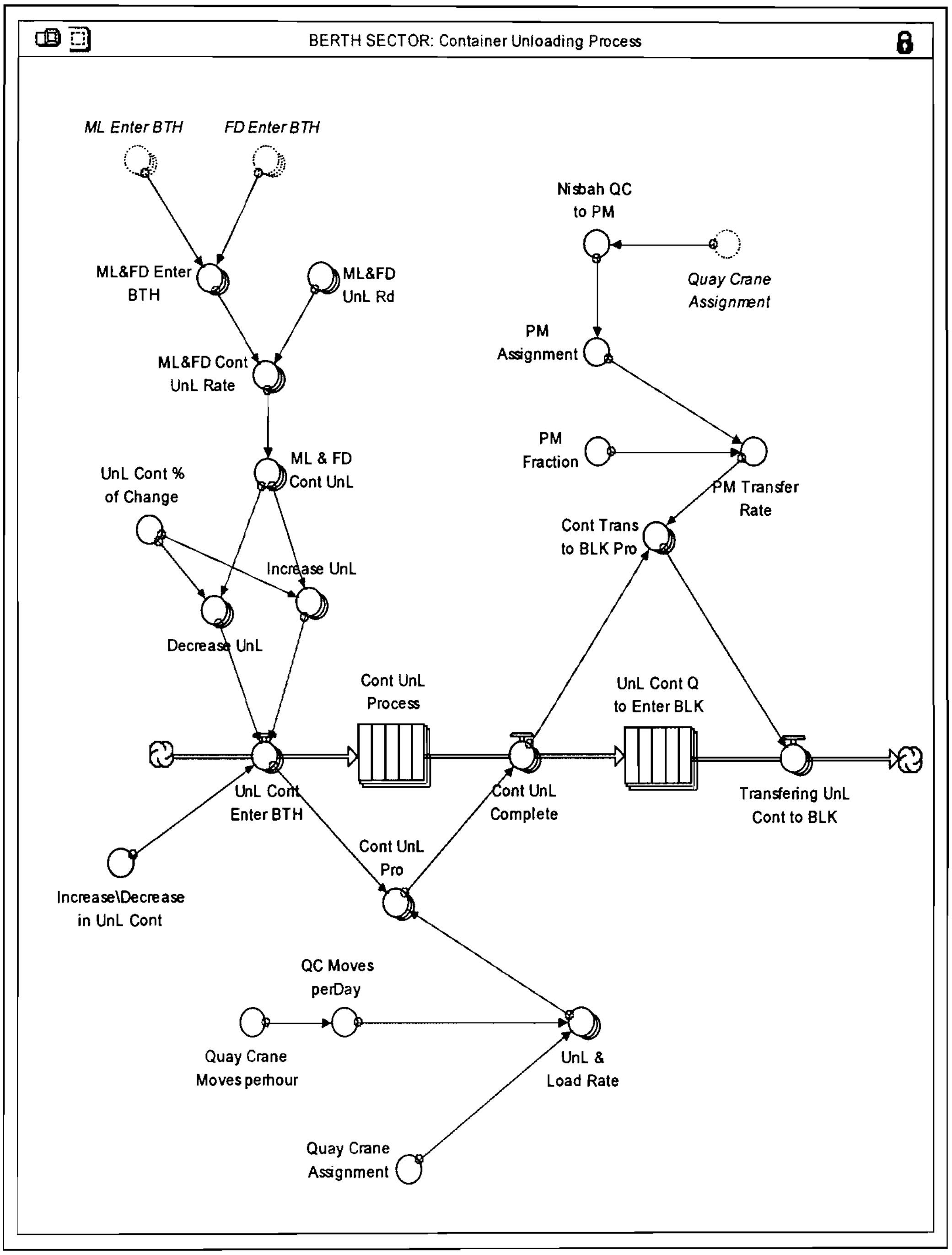


Figure 5.5: The container unloading process.

The number of containers entering the berth is represented by the *UnL Cont Enter BTH* flow where this flow is dependent on the value of *ML & FD Cont UnL* converter. *ML & FD Cont UnL* converter and *ML&FD Cont UnL Rate* converter represent the total containers unloaded from the arrived mainlines and feeders. *ML & FD Cont UnL* converter and *ML&FD Cont UnL Rate* converter hold the following formulas:

ML & FD Cont UnL converter = *ML&FD Cont UnL Rate* (5.12)
ML&FD Cont UnL Rate converter = *ML&FD Enter BTH* * *ML&FD UnL Rd* (5.13)

The *ML&FD Enter BTH* converter represents the total number of mainlines and feeders entering the berth. *ML&FD Enter BTH* converter holds the summation of *ML Enter BTH* converter which represents the number of mainlines entering the berth and *FD Enter BTH* converter which represents the number of feeders entering the berth. The formula of *ML&FD Enter BTH* converter is as follows:

ML&FD Enter BTH converter = *ML Enter BTH* + FD *Enter BTH* (5.14)

ML Enter BTH dotted converter and *FD Enter BTH dotted* converter are discussed in Figure 5.4. *ML&FD UnL Rd* converter holds the randomization value of containers unloaded from mainlines and feeders that enter each berth. The daily number of containers unloaded from both mainlines and feeders at each berth at the case study container terminal varied significantly. Therefore, the value of containers unloaded from both mainlines and feeders are generated through randomization to represent the real operation at the case study container terminal.

Table 5.3 shows the function of *ML&FD UnL Rd* converter for the eight berths. The function inputted in this converter is defined in the format of RANDOM (x, y) where x represents the minimum value while y represents the maximum value. The model therefore generates values between x and y randomly. For example, the function of ML*&FD UnL Rd [BTH 1]* converter is defined as RANDOM (156, 1974). Therefore, the model l generates randomly the number of containers that are unloaded from both mainlines and feeders in Berth 1 between the minimum of 156 containers and the maximum of 1974 containers.

Table 5.3: The function of arrayed *ML&FD UnL Rd* converter for eight berths.

***ML&FD UnL Rd* converter**	**Function**
ML&FD UnL Rd [BTH 1]	RANDOM (156, 1974)
ML&FD UnL Rd [BTH 2]	RANDOM (10, 1262)
ML&FD UnL Rd [BTH 3]	RANDOM (25, 1882)
ML&FD UnL Rd [BTH 4]	RANDOM (24, 1700)
ML&FD UnL Rd [BTH 5]	RANDOM (43, 1553)
ML&FD UnL Rd [BTH 6]	RANDOM (50, 1292)
ML&FD UnL Rd [BTH 7]	RANDOM (54, 1016)
ML&FD UnL Rd [BTH 8]	RANDOM (4, 2173)

The minimum and maximum value of containers unloaded from mainlines and feeders at berths 1 to 8 as seen in Table 5.3 were calculated from the data supplied by the case study container terminal. Due to the unavailability of data for berths 9 to 15 as these additional berths are for future capacity planning, the default value of *ML*

UnL Rd converter for berths 9 until 15 is set as the value of 1. This value however can be modified by users.

Continuing the discussion on the container unloading process in Figure 5.5, the *Increase\Decrease in UnL Cont* converter, *Increase UnL* converter and *Decrease UnL* converter allow users to increase or decrease the number of containers unloaded from vessels at each berth. The percentage of change can be set in the *UnL Cont % Change* converter. By default, the value of *UnL Cont % Change* converter is set to zero. *Increase UnL* converter and *Decrease UnL* converter hold the following formulas:

$$\textit{Increase UnL}\text{ converter} = (\textit{ML \& FD Cont UnL} * (\textit{UnL Cont \% of Change} / 100)) + \textit{ML \& FD Cont UnL} \quad (5.15)$$

$$\textit{Decrease UnL}\text{ converter} = \textit{ML \& FD Cont UnL} - (\textit{ML \& FD Cont UnL} * (\textit{UnL Cont \% of Change} / 100)) \quad (5.16)$$

As containers enter berths via *UnL Cont Enter BTH* flow, the unloading process is performed in the *Cont UnL Process* conveyor. The *Cont UnL Complete* flow represents the containers being unloaded from the vessel to the berth. The rate of *Cont UnL Complete* flow depends on the *Cont UnL Pro* converter. The value of *Cont UnL Pro* converter on the other hand is dependent on the *UnL & Load Rate* converter. *UnL & Load Rate* converter represents the unloading and loading rate or speed performed by the quay cranes. *Cont UnL Pro* converter and *UnL & Load Rate* converter hold the formulas:

Cont UnL Pro converter = *IF (UnL Cont Enter BTH = 0) THEN (UnL Cont Enter BTH / 1) ELSE (UnL Cont Enter BTH / UnL & Load Rate)* (5.17)

UnL & Load Rate converter = *Quay Crane Assignment * QC Moves perDay* (5.18)

Quay Crane Assignment converter holds the value of 2. This value represents the average number of quay cranes assigned to each vessel. *QC moves perday* converter converts the value in *Quay Cranes Moves perhour* converter into unit days. Quay cranes at the case study container terminal performed an average of 33 moves perhour. However, majority of the quay cranes at the case study container terminal were of the twin-lift type, meaning that these quay cranes were able to lift two containers in one move. Therefore, the value of *Quay Cranes Moves perhour* converter is set to 66 (33 x 2) while the formula of *QC moves perday* converter is set as *Quay Cranes Moves perhour* converter multiply by 24 (24 hours perday).

Upon completing the unloading process, the unloaded containers queue to enter the yard block for temporary storage. The *UnL Cont Q to Enter BLK* conveyor represents this process. The rate of the unloaded containers being transferred to yard block by prime movers is represented by the *Transferring UnL Cont to BLK* flow which is dependent on the *Cont Trans to BLK Pro* converter. *Cont Trans to BLK Pro* converter on the other hand depends on the *PM Transfer Rate* converter. *PM Transfer Rate* converter represents the total rate or speed of prime movers transferring the unloaded containers from the berth side to the yard block. *Cont Trans to BLK Pro* converter and *PM Transfer Rate* converter hold the formulas:

Cont Trans to BLK Pro converter = *IF (Cont UnL Complete = 0) THEN (Cont UnL Complete /1) ELSE (Cont UnL Complete / PM Transfer Rate)* (5.19)

PM Transfer Rate converter = *PM Assignment * PM Fraction* (5.20)

PM Assignment converter represents the number of prime movers assigned under each quay cranes while *PM Fraction* converter represents the rate of prime movers transferring unloaded containers from berth to yard block per day. There are seven prime movers assigned under each quay cranes and as *Quay Cranes Assignment* converter value represents the average number of quay cranes used to serve a vessel, *Nisbah QC to PM* converter which influences the value of *PM Assignment* converter holds the formula of:

Nisbah QC to PM converter = *Quay Crane Assignment * 7* (5.21)

Next, the discussion on the container loading process is presented. Typically, container loading process starts after the completion of container unloading process. While container unloading process represents the movement of containers into the container terminal, container loading process, on the other hand, represents the movement of containers leaving the container terminal. Figure 5.6 exhibits the container loading process. Similar to the container unloading process in Figure 5.5, the container loading process in Figure 5.6 also comprises 15 arrays, representing 15 berths.

The model in Figure 5.6 starts off with *Cont Entering BTH for Loading* flow. This flow represents the movement of containers leaving the yard block to the berth side, to be loaded onto the vessels. *Cont Entering BTH for Loading* flow depends on the

Cont Leaving BLK Rate converter. *Cont Leaving BLK Rate* converter controls the rate of containers leaving the yard block. The value of this converter is dependent on the *ML&FD Load* converter. Both *ML&FD Load* converter and *ML&FD Load Rate* converter represent the total containers that are loaded onto mainlines and feeders at each berth. The formulas for *ML&FD Load* converter and *ML&FD Load Rate* converter are as follows:

ML&FD Load converter = *ML&FD Load Rate* (5.22)
ML&FD Load Rate converter = *ML&FD Enter BTH* * *ML&FD Load Rd* (5.23)

As in Figure 5.5, *ML&FD Enter BTH* converter holds the total of number of mainlines and feeders that arrived at the berth. *ML&FD Load Rd* converter on the other hand generates the randomization number of containers that are loaded onto mainlines and feeders at each berth. As with the number of containers unloaded from both mainlines and feeders at each berth, the daily number of containers loaded to mainlines and feeders at each berth also varies significantly. Therefore, the value of containers loaded to both mainlines and feeders are generated through randomization to represent the real operation at the case study container terminal.

Table 5.4 shows the functions of *ML&FD Load Rd* converter for the eight berths. *ML&FD Load Rd* converter is defined as RANDOM (x, y) where x represents the minimum number of containers while y represents the maximum containers that are loaded onto a mainline for a given berth. The function RANDOM (x, y) allows the model to generate random value between x and y. By default, the function of *ML&FD Load Rd* converter for berth 9 until 15 is given the value of 1.

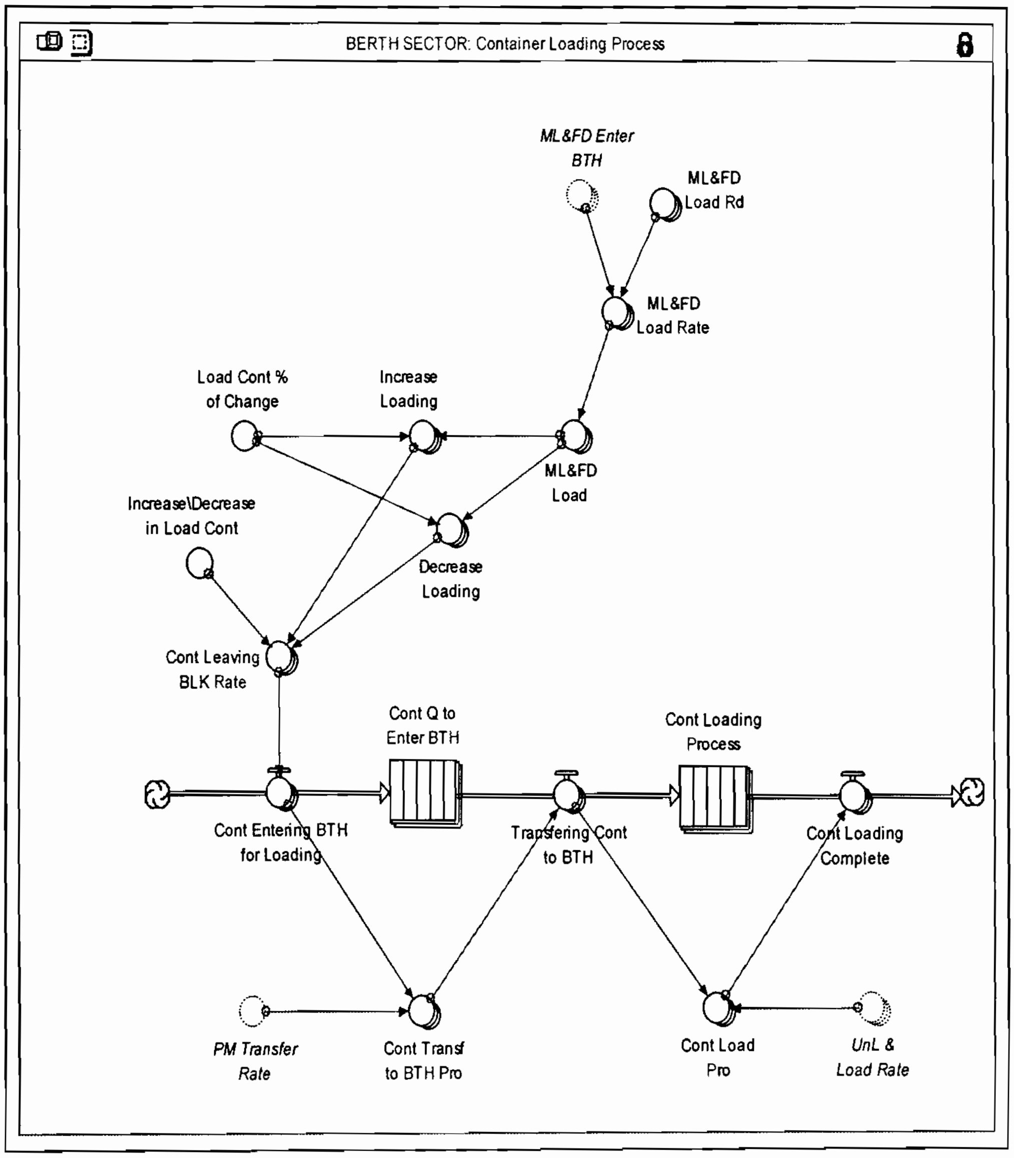


Figure 5.6: The container loading process.

Table 5.4: The function of arrayed *ML&FD Load Rd* converter for eight berths.

***ML Load Rd* converter**	**Function**
ML&FD Load Rd [BTH 1]	RANDOM (80, 1303)
ML&FD Load Rd [BTH 2]	RANDOM (55, 1998)
ML&FD Load Rd [BTH 3]	RANDOM (85, 1764)
ML&FD Load Rd [BTH 4]	RANDOM (10, 1572)
ML&FD Load Rd [BTH 5]	RANDOM (35, 2327)
ML&FD Load Rd [BTH 6]	RANDOM (58, 1267)
ML&FD Load Rd [BTH 7]	RANDOM (55, 2288)
ML&FD Load Rd [BTH 8]	RANDOM (280, 1172)

The minimum and maximum values of containers that are loaded onto mainlines and feeders at berths 1 to 8 as seen in Table 5.4 are calculated from the data supplied by the case study container terminal.

Going back to the discussion on the container loading model in Figure 5.6, the *Cont Entering BTH for Loading* flow represents the number of containers to be extracted out from the yard block as these containers will be transported to the berth, finally to be loaded onto the vessels. *Increase Loading* converter, *Decrease Loading* converter, *Load Cont % of Change* converter and *Increase\Decrease in Load Cont* converter allow users to modify the amount of containers to be extracted from the yard block by increasing or decreasing the default value. The formulas for *Increase Loading* converter and *Decrease Loading* converter are:

Increase Loading converter = (ML&FD Load * (Load Cont % of Change / 100)) + ML&FD Load (5.24)

Decrease Loading converter = ML&FD Load - ML&FD Load * (Load Cont % of Change / 100) (5.25)

Upon entering *Cont Entering BTH for Loading* flow, these containers queue to enter their destined berth in *Cont Q to Enter BTH* conveyor. *Transferring Cont to BTH* flow controls the rate of these containers being transferred to the berth side from the yard side using prime movers. *Transferring Cont to BTH* flow depends on the value of *Cont Transf to BTH Pro* converter. *Cont Transf to BTH Pro* converter holds the formula:

Cont Transf to BTH Pro converter = *IF (Cont Entering BTH for Loading = 0) THEN (Cont Entering BTH for Loading / 1) ELSE (Cont Entering BTH for Loading / PM Transfer Rate)* (5.26)

Upon reaching the berth side, these containers are loaded onto the vessels. Here, the container loading process begins. *Cont Loading Process* conveyor represents the container loading process while *Cont Loading Complete* flow represents the completion of the loading process, where the containers are successfully loaded onto the vessels. *Cont Loading Complete* flow depends on the value of *Cont Load Pro* converter. This converter represents the rate of the loading process carried out. The formula assigned into *Cont Load Pro converter* is:

Cont Load Pro converter = *IF (Transferring Cont to BTH = 0) THEN (Transferring Cont to BTH / 1) ELSE (Transferring Cont to BTH / UnL &Load Rate)* (5.27)

A detailed discussion on the berth sector model development has been presented. The discussion comprises the construction of berth allocation model, the physical process in the berth and finally the development of container unloading and loading model. With this, the discussion on the berth sector is complete. The next section emphasizes the discussion on yard sector model development.

5.5.1.2 Yard Sector at Operation Level

The yard sector captures the operations of the yard block. Figure 5.7 shows operations of the yard sector. Similar to Figures 5.5 and 5.6, the model in Figure 5.7 also consists of 15 arrays, which represent 15 berths. Figure 5.7 is the continuation of the container unloading model in Figure 5.5. As the unloaded containers are transferred to the block via *Transferring UnL Cont to BLK* flow in Figure 5.5, the containers in this same flow in Figure 5.7 are directed to enter the yard block.

The *Sorting UnL Cont to BLK* stock prepares to sort the unloaded containers from the berth to 12 different yard blocks. The reason for constructing 12 blocks is to mirror the actual number of yard blocks at the case study container terminal. Emerging from the *Sorting UnL Cont to BLK* stocks are 12 flows which represent the number of unloaded containers entering each of the 12 blocks. The 12 flows are *Cont Enter BLK 1* flow, *Cont Enter BLK 2* flow, *Cont Enter BLK 3* flow, *Cont Enter BLK 4* flow, *Cont Enter BLK 5* flow, *Cont Enter BLK 6* flow, *Cont Enter BLK 7* flow, *Cont Enter BLK 8* flow, *Cont Enter BLK 9* flow, *Cont Enter BLK 10* flow, *Cont Enter BLK 11* flow and *Cont Enter BLK 12* flow.

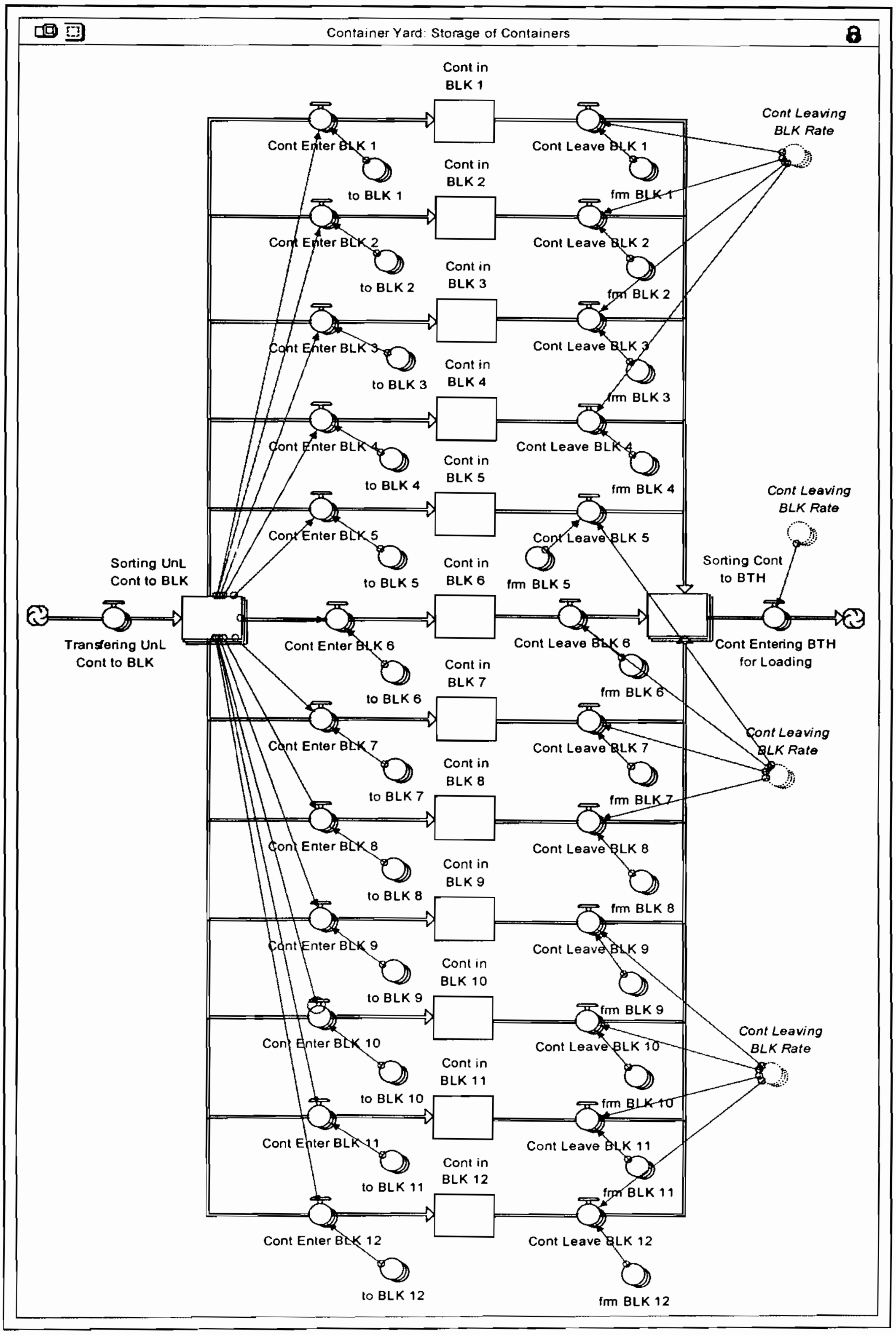


Figure 5.7: Storage of containers at the yard block.

Each of these flows represents the number of containers entering the respective block. The formula for these 12 blocks is illustrated in Equation 5.28 where i represents the block while j represents the berth:

$$\textit{Cont Enter BLK i } \text{flow} = \textit{Sorting UnL Cont to BLK [BTHj]} * \textit{to BLK i [BTHj]} \qquad \forall i = 1, \ldots, 12 \quad \forall j = 1, \ldots, 15 \tag{5.28}$$

The *to BLK i* converter holds the percentage of unloaded containers that enters from the berth to Block i. Table 5.5 exhibits the percentage (in fraction) of unloaded containers that enter from Berths 1 to Berth 8 into Block 1. The values given in Table 5.5 were calculated from the data supplied by the case study container terminal. By default, the additional seven berths (to *BLK i [BTH 9]* converter until *to BLK i* [BTH 15] converter) is given as 2.384649308 / 100. This value can be modified by the users accordingly. The functions of arrays *to BLK 2* converter, *to BLK 3* converter, *to BLK 4* converter, *to BLK 5* converter, *to BLK 6* converter, *to BLK 7* converter, *to BLK 8* converter, *to BLK 9* converter, *to BLK 10* converter, *to BLK 11* converter and *to BLK 12* converter are included in Appendix 3.

Table 5.5: The function of arrayed *to BLK 1* converter for eight berths.

***to BLK 1* converter**	**Function**
to BLK 1 [BTH 1]	8.220390928 / 100
to BLK 1 [BTH 2]	6.212221954 / 100
to BLK 1 [BTH 3]	10.74345106 / 100
to BLK 1 [BTH 4]	9.805298436 / 100
to BLK 1 [BTH 5]	6.293462773 / 100
to BLK 1 [BTH 6]	6.242540582 / 100

to BLK 1 [BTH 7]	3.883931741 / 100
to BLK 1 [BTH 8]	2.384649308 / 100

As shown in Figure 5.7, containers that enter from *Cont Enter BLK 1* flow until *Cont Enter BLK 12* flow are stored temporarily at its representative block. For example, *Cont Enter BLK i* flow directs containers into *Cont in BLK i* stock. Here, containers are stored temporarily while waiting for incoming vessels to collect them.

During the visit to the case study container terminal in January 2008, the yard strategist revealed that the current capacity of the yard blocks at that time was 30%. A sensitivity analysis was performed on the yard capacity to see if different initial value of yard capacity influences the yard occupancy rate. Table 5.6 presents the sensitivity analysis for the initial yard capacity value. From Table 1, if the initial yard capacity value is assigned to 30%, the yard occupancy rate is 24.9%. If the initial yard capacity value is assigned to 20% or 40%, the yard occupancy rate is 25.3% and 24.5% respectively. The sensitivity analysis shows that different initial value of yard capacity does not impact the yard occupancy rate.

Table 5.6: Sensitivity analysis for the initial value of yard capacity.

	30% Full	**20% Full**	**40% Full**
Yard Occupancy Rate	24.9%	25.3%	24.5%

Therefore, the initial value of *Cont in BLK i* stock is set to 3780 which assumes that at the starting time of this simulation, Block i is 30% full with containers (maximum capacity per block is 12600 TEUs).

The rate of containers exiting each of the 12 blocks is dependent on its attached outflow. There are 12 outflows (*Cont Leave BLK 1* flow, *Cont Leave BLK 2* flow, *Cont Leave BLK 3* flow, *Cont Leave BLK 4* flow, *Cont Leave BLK 5* flow, *Cont Leave BLK 6* flow, *Cont Leave BLK 7* flow, *Cont Leave BLK 8* flow, *Cont Leave BLK 9* flow, *Cont Leave BLK 10* flow, *Cont Leave BLK 11* flow and *Cont Leave BLK 12* flow) where each of these outflows bears the same logic.

The rate of containers leaving *Cont in BLK i* stock is controlled by *Cont Leave BLK i* flow. The same goes for the other 11 blocks where the rate of containers exiting the block is being influenced by the subsequent flow. *Cont Leave BLK i* flow holds the formula:

Cont Leave BLK i flow = IF (Cont Leaving BLK Rate [BTHj] > 1) THEN (Cont Leaving BLK Rate [BTHj] * frm BLK i[BTHj]) ELSE (0) (5.29)

□ i = 1, …, 12
□ j = 1, …, 15

Cont Leaving BLK rate dotted converter has already been discussed in Figure 5.6; *from BLK i* converter, on the other hand, holds the percentage of containers that leaves Block i to be transferred to the berth side. Table 5.7 presents the percentage (in fraction) of containers that leaves Block 1 to enter Berths 1 to Berth 8.

Table 5.7: The function of arrayed *from BLK 1* converter for eight berths.

***from BLK 1* converter**	**Function**
from BLK 1 [BTH 1]	20.49551224 / 100
from BLK 1 [BTH 2]	19.40455368 / 100
from BLK 1 [BTH 3]	1.341240187 / 100
from BLK 1 [BTH 4]	6.684220501 / 100
from BLK 1 [BTH 5]	2.659569304 / 100
from BLK 1 [BTH 6]	0.368416598 / 100
from BLK 1 [BTH 7]	0.172181552 / 100
from BLK 1 [BTH 8]	4.951656992 / 100

The values given in Table 5.7 were calculated from the data supplied by the case study container terminal. Similarly, by default, the additional seven berths (from *BLK i [BTH 9]* converter until *from BLK i* [BTH 15] converter) is given as 4.951656992 / 100. This value can be modified by the users accordingly. The functions of arrays *from BLK 2* converter, *from BLK 3* converter, *from BLK 4* converter, *from BLK 5* converter, *from BLK 6* converter, *from BLK 7* converter, *from BLK 8* converter, *from BLK 9* converter, *from BLK 10* converter, *from BLK 11* converter and *from BLK 12* converter are included in Appendix 3.

The *Sorting Cont to BTH* stock prepares the containers from *Cont Leave BLK 1* flow to *Cont Leave BLK 12* flow to be transferred to the berth side. *Container Entering BTH for Loading* flow represents the rate of prime movers transferring containers from the yard block to the berth side. The subsequent processes after the containers enter *Container Entering BTH for Loading* flow have been discussed in the container loading model in Figure 5.6.

So far, the description on the development of the operation level model for both berth sector and yard sector has been presented. The construction of this operational level model is to achieve the first and second objectives of this research. The next section encompasses the development of the strategic level model in order to achieve the third objective of this research.

5.5.2 Strategic Level Model Formulation

The third objective of this research is to determine a capacity planning by identifying when new terminal facilities will be needed to ensure the terminal has sufficient capacity to sustain demand and at the same time, avoid overcapacity. In order to achieve this objective a strategic level model was developed. While the operational level model, discussed earlier aids users to make day to day decisions, the strategic level model helps users to understand the impact of their decision on the performance of the container terminal in the long run.

In order to run the strategic level model, the total vessels arrived at the case study container terminal annually and its total container throughput handled annually is required. This information was obtained from the website of the Ministry of Transportation, Malaysia (MOT, retrieved 6 October 2009). The unit for time measurement is in years and the strategic level model is run for 10 years (throughout the year 2002 to 2012). The data retrieved from the MOT website only consists of the occurring year of 2002 to 2009. Therefore, the data for the subsequent year of 2010 to 2012 are forecasted values.

The data patterns of the available data were studied to determine which forecasting technique is suitable to forecast the total vessels arrival and total container throughput from the year 2010 to 2012. According to Hanke, Wichern and Reitsch (2001), time series data patterns can be identified by studying the autocorrelation coefficients for different time lags of a variable. Therefore the autocorrelation analysis was performed on the available data to identify the data patterns.

The autocorrelation coefficients at lag 1, lag 2 and lag 3 for the total vessels arrival from year 2002 to 2009 are 0.5828, 0.1252 and -0.3381 respectively. Meanwhile, the autocorrelation coefficients at lag 1, lag 2 and lag 3 for the total container throughput handled from year 2002 to 2009 are 0.6116, 0.2819 and -0.0319 respectively. The calculation of the autocorrelation coefficients at three different lags for both total vessels arrival and total container throughput are included in Appendix 4.

It is noted that the autocorrelation coefficients for both total vessels arrival and total container throughput decline rapidly to zero after the lag 1. Hanke, Wichern and Reitsch (2001) stated that if the autocorrelation coefficients decline to zero fairly rapidly, generally, after the second or the third lag, then the data consists of stationary series. The authors explained that a stationary series is one whose basic statistical properties, such as the mean and variance, remain constant over time.

Hanke, Wichern and Reitsch (2001) also explained that one of the techniques that can be considered when forecasting stationary series is moving average forecasting technique. The authors further explained that moving average is a method that uses the mean of all the data to forecast. Therefore, the double moving average forecasting technique was used to forecast the total vessels arrival and total container throughput from the year 2010 to 2012. The calculations for the forecasted values are included in Appendix 5. Table 5.8 exhibits the total vessels arrived at the case study container terminal from the years 2002 to 2012 while Table 5.9 exhibits the total container throughput at the case study container terminal for the years 2002 to 2012.

Table 5.8: Total number of vessels calling at case study container terminal.

Year	Total Vessels
2002	3483
2003	3148
2004	3193
2005	3128
2006	3261
2007	3747
2008	3760
2009	3776
2010	*3790*
2011	*3804*
2012	*3833*

Table 5.9: Total container throughput at case study container terminal.

Year	Total Container Throughput
2002	2668512 TEUs
2003	3316954 TEUs
2004	3835970 TEUs
2005	4177123 TEUs
2006	4637418 TEUs
2007	5297631 TEUs
2008	5466191 TEUs
2009	5835085 TEUs
2010	*6351725 TEUs*
2011	*6761103 TEUs*
2012	*7170481 TEUs*

The following discussion is focused on the formulation of the strategic level model. Presentation kicks off with the discussion on how strategic level berth allocation model is constructed. Figure 5.8 exhibits the system dynamics model of berth allocation model at the strategic level. At one glance, this strategic level berth allocation model is no different compared with the one in Figure 5.3 which exhibits the berth allocation for the operation model.

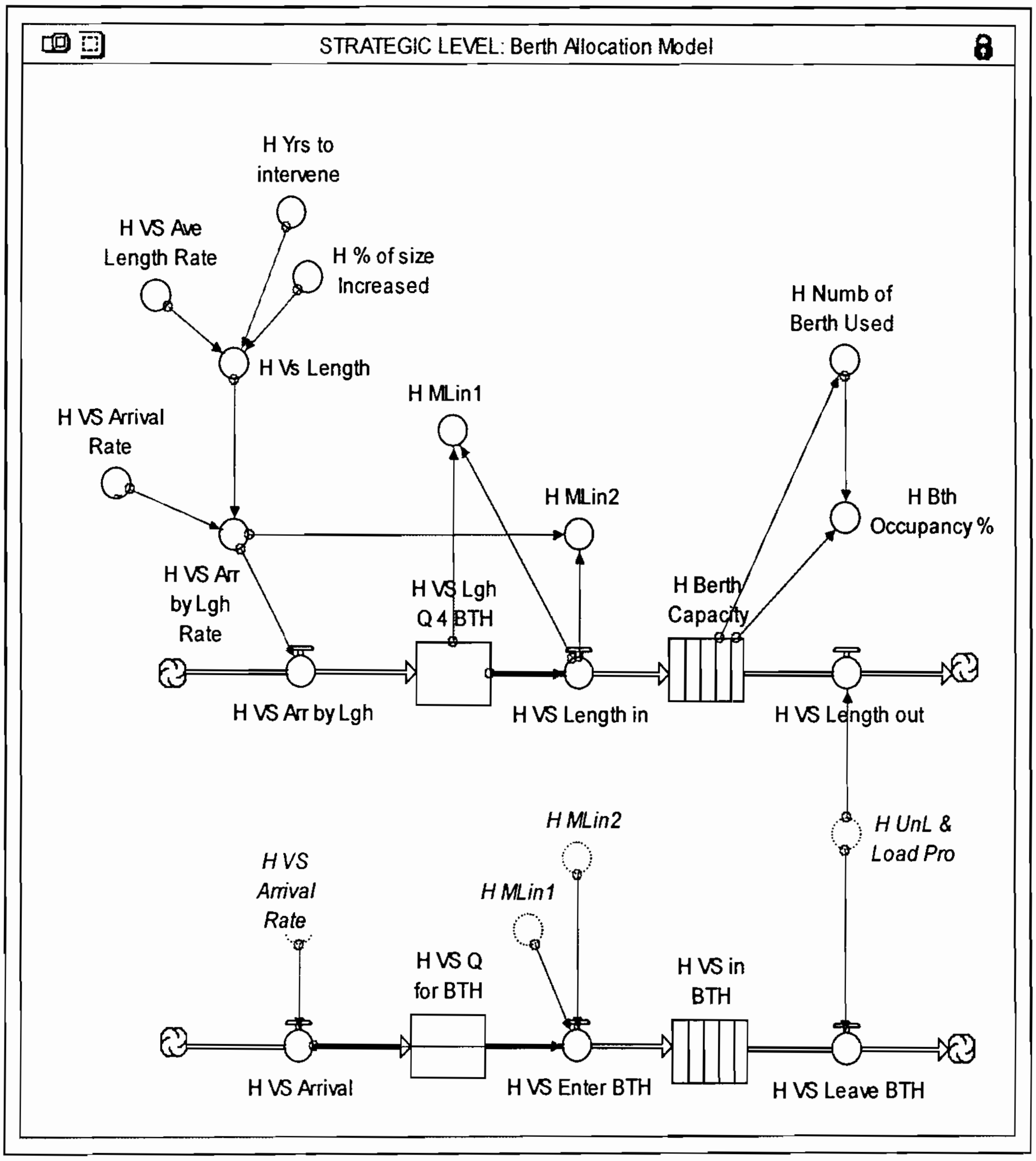


Figure 5.8: Strategic level of berth allocation model.

There are two main features that distinguished the berth allocation model of the strategic level from the operational level. The first is that mainlines and feeder ships are combined in the strategic level model as compared to the operational level model (see Figure 5.3 on page 90) where both of these ship types are separated. Secondly, while the model in Figure 5.8 runs by years, the model in Figure 5.3 runs in days.

Apart from these two major differences, the strategic level model runs with the same logic as the operational level model.

The process of berth allocation starts with vessel arrival. The length of vessels that enter the container terminal is compared against the available berth length or capacity. During the visit to the case study container terminal in early year 2008, only the data of the average vessels length for the year 2007 was available. Therefore, the *H VS Ave Length Rate* converter holds the value 251.662m which represents the average length of vessels that enter the case study container terminal. This value was calculated from the total length of vessels that entered the case study container terminal in the year of 2007 divided by the total number of vessels that entered the case study container terminal during the same year.

H% of size increased converter allows users to increase the length of the arrived vessel while *H Yrs to intervene* converter permits users to decide on which year to intervene the increase of the average vessel length. *H VS Length* converter carries out the function defined by *H% of size increased* converter, *H Yrs to intervene* converter and *H VS Ave Length Rate* converter. *H VS Length* converter holds the formula:

H VS Length converter = *IF (TIME >= H Yrs to intervene THEN ((H VS Ave Length Rate * (H % of size Increased / 100)) + H VS Ave Length Rate) ELSE (H VS Ave Length Rate)* (5.30)

H VS Arrival Rate converter represents the total number of vessels arrived each year by representing the value presented in Table 5.8. *H VS Arr byLgh Rate* converter on the other hand, holds the total length of vessels that enter the container terminal annually.

This value can be obtained by multiplying the average length of arrived vessel with the total number of vessels arrived each year. The equation for this computation is:

H VS Arr byLgh Rate converter = *H Vs Length* * *H VS Arrival Rate* (5.31)

The *H VS Arr byLgh* flow basically takes on the value of *H VS Arr byLgh Rate.* If the berth capacity has sufficient length which permits the vessels to enter the berth, the vessels enter the berth via *H VS Length in* flow. On the other hand, if the value of *H VS Arr byLgh* flow is greater than the berth capacity, the vessels queue at the *H VS Lgh Q 4 BTH* stock. The berth capacity is represented by the *H Berth Capacity* conveyor. The capacity of this conveyor which represents the total length of the current working berth is the main constraint that controls the access of vessels into the berth.

As the daily available capacity of the initial eight berths are 2880m (360m each * 8), in order to calculate the available berth capacity for one year, the initial daily berth capacity was multiplied by the number of days in one year. Therefore, by default, the capacity of the *H Berth Capacity* conveyor is given as 1,054,080 (2880m * 366 days) which represents eight working berths. The total berth length of 2880m is multiplied by 366 days instead of 365 days because there are 366 days in the year 2008. However, users may modify the value of *H Berth Capacity* conveyor to see how increasing berth length influences the terminal operations. Table 5.10 shows the length of each of the15 berths that users can experiment with.

Table 5.10: Number of berths and its length for strategic level model.

Number of Berths	Length
8	1054080 m
9	1185840 m
10	1317600 m
11	1449360 m
12	1581120 m
13	1712880 m
14	1844640 m
15	1976400 m

Upon completion of the container unloading and loading process, vessels are released from the berth via *H VS Length Out* flow. *H VS Length Out* flow is dependent on the *H UnL & Load Pro* converter which represents the speed of container unloading and loading process. The *H Numb of Berth Used* converter calculates the number of berths operating in *H Berth Capacity* conveyor. *H Bth Occupancy%* converter represents the capacity of *H Berth Capacity* conveyor at a given year. *H Numb of Berth Used* converter and *H Bth Occupancy%* converter hold the formulae:

H Numb of Berth Used converter = *(CAP (H Berth Capacity) / 366) / (360)* (5.32)

H Bth Occupancy% converter = *(H Berth Capacity / ((H Numb of Berth Used * 360) * 366)) * 100* (5.33)

As all the variables in the first model of Figure 5.8 (currently being discussed) run with the unit of *vessel length*, the *H MLin1* converter and *H MLin2* converter are used to convert the unit of total vessel length represented in *H VS Lgh Q 4 BTH* stock and *H VS Length in* flow into the number of actual vessels representing the total length. *H MLin1* converter and *H MLin2* converter are used in the second model of Figure 5.8. *H MLin1* converter and *H MLin2* converter hold the formulae:

H MLin1 converter = *IF (H VS Lgh Q 4_BTH = H VS Length in) THEN (1) ELSE (0)* (5.34)

H MLin2 converter = *H VS Length in / H VS Arr by Lgh Rate* (5.35)

The purpose of the second model in Figure 5.8 is to convert the unit *vessel length* of the first model to unit *vessel* in order to mimic the actual terminal operation as well as to facilitate the calculation of performance indicators at the later part of this chapter. The second model starts with the *H VS Arrival* flow which represents the number of vessels that enter the case study container terminal on annual basis. This flow is dependent on the *H VS Arrival rate* converter which captured the values in Table 5.8.

Vessels that arrived at the container terminal enter the berth (represented by *H VS in BTH* conveyor) via *H VS Enter BTH* flow. The value of *H VS Enter BTH* flow is dependent on the value of *H MLin1* converter and *H MLin2* converter. Vessels that fail to enter the berth due to insufficient berth space queued at the *H VS Q for BTH* stock. Finally, vessels are released from *H VS in BTH* conveyor upon completion of container unloading and loading process via *H VS Leave BTH* flow. Again, this *H VS Leave BTH* flow is dependent on the value of *H UnL & Load Pro* converter.

Having discussed the construction of the decision process for strategic level berth allocation model, the subsequent presentation focuses on the formulation of container unloading and loading process for the strategic level model. Figure 5.9 exhibits the strategic level model for container unloading and loading process. As seen in Figure 5.9, the strategic level model is less complex when compared to the operation level model in Figure 5.5 and Figure 5.6.

One major difference between the strategic level model and the operation level model is that at the strategic level, the container unloading and loading process is combined as one. This is because the main aim of the strategic level model is to calculate the total time used during the container unloading and loading process. The second difference is that the strategic level model is not assigned to the 15 arrays as the model itself represents the operation of the whole berth in an accumulated manner.

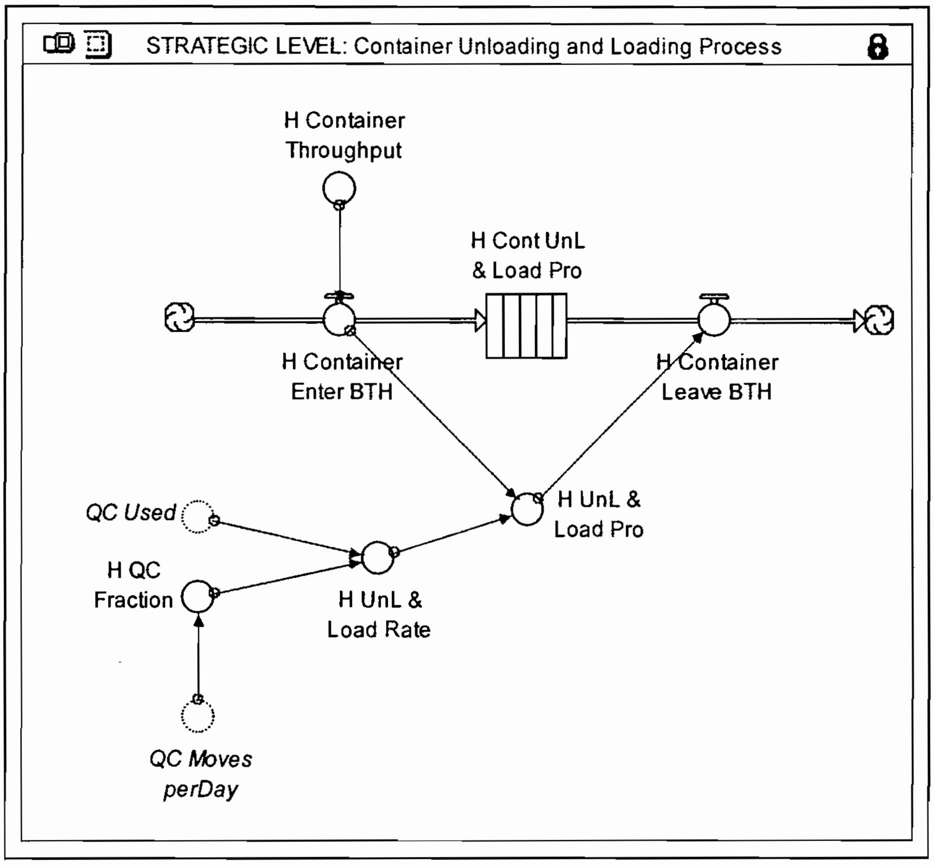


Figure 5.9: Strategic level of container unloading and loading process.

The *H Container Enter BTH* flow represents the number of containers that enter the container terminal on annual basis. This H *Container Enter BTH* flow is dependent on the *H Container Throughput* converter. *H Container Throughput* converter is assigned with the values in Table 5.9 which represents the total container throughput for the years 2002 to 2012. The container unloading and loading process is represented by the *H ContUnL & Load Pro* conveyor. Upon completing the unloading process, container loading process starts where containers are loaded onto the vessels (containers leave berth) via *H Container Leave BTH* flow. The rate of containers leaving the berth via *H Container Leave BTH* flow is very much dependent on the rate or speed of the container unloading and loading process represented by the *H UnL & Load Pro* converter. *H UnL & Load Pro* converter and *H UnL & Load Rate* converter hold the formulas:

H UnL & Load Pro converter = *IF (H Container Enter BTH = 0) THEN (H Container Enter BTH / 1) ELSE (H Container Enter BTH / H UnL & Load Rate)* (3.36)

H UnL & Load Rate converter = *H QC Fraction * (QC Used)* (3.37)

H QC Fraction converter represents the value of quay crane moves per year by multiplying *QC Moves per day* converter with 366 days. *QC Moves per day* converter represents the average moves performed by one quay crane per day while *QC Used* converter represents the total number of quay cranes used at the case study container terminal per day. Discussion on *QC Moves per day* converter can be viewed in Figure 5.5.

So far, detailed explanation was presented on the construction of the strategic level model for both berth allocation decision making and container unloading and loading process. Earlier on, illustrative explanations were presented to describe the formulation of the operation level model. Thus, this wraps up the discussion on the model formulation section. The next section encompasses the model verification and validation for both operation and strategic level models.

5.6 Model Verification and Validation

This section focuses on verifying and validating the developed system dynamics model. Balci (1998) provided the definition of model verification and model validation where model verification is substantiating that the model is transformed from one form into another, as intended, with sufficient accuracy while model validation is substantiating that the model, within its domain of applicability, behaves with satisfactory accuracy consistent with the study objectives. The author also concluded that model verification deals with building the model right while model validation deals with building the right model.

As models are used to predict or compare the future performance of a new system, a modified system or an existing system under new conditions, the model needs to exhibit sufficient accuracy (Balci and Sargent, 1981), or any conclusions derived from the model are likely to be erroneous and may result in costly decisions being made (Law, 2009).

This research implemented the validation process discussed in Sargent (2009). There are four validation processes, namely data validity, conceptual model validation, computerized model verification and operational validity. Only three processes are performed on the data as well as on the developed model. Computerized model verification is excluded as this process is only used if the model is developed using programming language.

The next section gives a brief explanation on data validity, conceptual model validation and operational validity as well as how these validation processes are performed in this research.

5.6.1 Data Validity

According to Sargent (2009), data are needed for three purposes, i.e. for building the conceptual model, for validating the model and performing experiments with the validated model. Therefore, if data are not available sufficiently or accurately, the accuracy of the developed model is questionable.

The five-month data supplied by the case study container terminal only consisted of data ranging from January to May 2008. As the developed operational model is run for one year, it is questionable whether the five months worth of data can be used to represent the terminal activity for one year. The five-month data include:

1. The number of mainlines and feeders that arrived at the container terminal.
2. The number of containers unloaded from mainlines and feeders at each berth.
3. The number of containers loaded onto mainlines and feeders at each berth.
4. The number of unloaded containers that entered the yard blocks.
5. The number of loaded containers that left or exited the yard blocks.

If these data show huge variation between each month, these data are not suitable to represent the terminal activity for one year. However, if these data show no variation in each month during the course of January to May 2008, then these five months data can be used to represent the terminal operations for one year. Therefore the five data-sets were tested using the analysis of variance or ANOVA procedure to show that there are no differences on the data in January 2008, February 2008, March 2008, April 2008 and May 2008. ANOVA procedure is used because this procedure is useful in testing the difference between the mean score of more than two data sets (John & Lee-Ross, 1998).

The first data-set that was tested was the number of mainlines and feeders that arrived at the container terminal. There is not enough evidence to reject that the mean arrival of mainlines and feeders for the month of January 2008, February 2008, March 2008, April 2008 and May 2008 are the same. [($F = 1.189$, $df = 4$, $P \geq 0.05$) and ($F = 0.203$, $df = 4$, $P \geq 0.05$)]. Therefore, the five months data for the number of mainlines and feeders that arrived at the container terminal can be used to represent the one year operation of the terminal.

Next, the second data-set that was tested was the number of containers unloaded from mainlines and feeders at each berth. Again, there is not enough evidence to reject that the mean of containers unloaded from mainlines and feeders in the month of January 2008, February 2008, March 2008, April 2008 and May 2008 at each berth are the same. [($F = 0.922$, $df = 4$, $P \geq 0.05$), ($F = 1.450$, $df = 4$, $P \geq 0.05$), ($F = 0.248$, $df = 4$, $P \geq 0.05$), ($F = 0.398$, $df = 4$, $P \geq 0.05$), ($F = 0.095$, $df = 4$, $P \geq 0.05$), ($F = 0.152$, $df = 4$, $P \geq 0.05$), ($F = 0.592$, $df = 4$, $P \geq 0.05$), ($F = 0.128$, $df = 4$, $P \geq 0.05$), ($F = 0.070$, $df = 4$, $P \geq 0.05$), ($F = 0.525$, $df = 4$, $P \geq 0.05$), ($F = 1.872$, $df = 4$, $P \geq 0.05$), ($F = 0.066$, $df = 4$, $P \geq 0.05$), ($F = 0.138$, $df = 4$, $P \geq 0.05$), ($F = 0.109$, $df = 4$, $P \geq 0.05$), ($F = 0.249$, $df = 4$, $P \geq 0.05$) and ($F = 0.686$, $df = 4$, $P \geq 0.05$)]. Therefore, the five months data for the number of containers unloaded from mainlines and feeders at each berth can be used to represent the one year operation of the terminal.

Similarly, the third data-set, i.e. the number of containers loaded onto mainlines and feeders at each berth was tested. There is not enough evidence to reject that the mean of containers loaded onto mainlines and feeders in the month of January 2008, February 2008, March 2008, April 2008 and May 2008 are the same. [($F = 1.750$, df $= 4$, $P \geq 0.05$), ($F = 0.886$, df $= 4$, $P \geq 0.05$), ($F = 0.614$, df $= 4$, $P \geq 0.05$), ($F = 0.051$, df $= 4$, $P \geq 0.05$), ($F = 0.680$, df $= 4$, $P \geq 0.05$), ($F = 0.268$, df $= 4$, $P \geq 0.05$), ($F = 0.237$, df $= 4$, $P \geq 0.05$), ($F = 1.418$, df $= 4$, $P \geq 0.05$), ($F = 0.351$, df $= 4$, $P \geq 0.05$), ($F = 0.086$, df $= 4$, $P \geq 0.05$), ($F = 1.786$, df $= 4$, $P \geq 0.05$), ($F = 0.689$, df $= 4$, $P \geq 0.05$), ($F = 0.572$, df $= 4$, $P \geq 0.05$), ($F = 0.337$, df $= 4$, $P \geq 0.05$), ($F = 0.300$, df $= 4$, $P \geq 0.05$) and ($F = 1.681$, df $= 4$, $P \geq 0.05$)]. Therefore, the five months data for the number of containers loaded onto mainlines and feeders at each berth can be used to represent the one year operation of the terminal.

Moving on, the fourth and fifth data-sets which were the number of unloaded containers entering the yard blocks from each berth and the number of loaded containers that left the yard blocks were tested. Results showed that there is not enough evidence to reject that the mean of unloaded containers entering the yard blocks from each berth in the month of January 2008, February 2008, March 2008, April 2008 and May 2008 are the same. [($F = 0.577$, df $= 4$, $P \geq 0.05$), ($F = 0.956$, df $= 4$, $P \geq 0.05$), ($F = 0.965$, df $= 4$, $P \geq 0.05$), ($F = 0.913$, df $= 4$, $P \geq 0.05$), ($F = 0.172$, df $= 4$, $P \geq 0.05$), ($F = 1.144$, df $= 4$, $P \geq 0.05$), ($F = 2.347$, df $= 4$, $P \geq 0.05$) and ($F = 0.613$, df $= 4$, $P \geq 0.05$)]

Similarly, there is also not enough evidence to reject that the mean of percentage of loaded containers leaving the yard blocks in the month of January 2008, February 2008, March 2008, April 2008 and May 2008 are different. [($F = 0.445$, $df = 4$, $P \geq 0.05$), ($F = 0.210$, $df = 4$, $P \geq 0.05$), ($F = 0.510$, $df = 4$, $P \geq 0.05$), ($F = 0.021$, $df = 4$, $P \geq 0.05$), ($F = 0.497$, $df = 4$, $P \geq 0.05$), ($F = 0.105$, $df = 4$, $P \geq 0.05$), ($F = 0.065$, $df = 4$, $P \geq 0.05$) and ($F = 0.206$, $df = 4$, $P \geq 0.05$)]. Therefore, the five months data for the number of unloaded containers that entered the yard blocks from berth 1 to berth 8 and the number of containers that exited the yard blocks to be loaded at berth 1 to berth 8 can be used to represent the one year operation of the terminal.

Analysis of variance or ANOVA procedure was performed on the five-month data of the number of mainlines and feeders that arrived at the container terminal, the number of containers unloaded from mainlines and feeders at each berth, the number of containers loaded onto mainlines and feeders at each berth, the number of unloaded containers that entered the yard blocks from berth 1 to berth 8 and the number of containers that exited the yard blocks, to be loaded at berth 1 to berth 8.

Results from the ANOVA justified that the five data-sets were not significantly different in each month (January to May 2008) and that it can be generalized to represent the subsequent seven months (June to December 2008).

5.6.2 Conceptual Model Validation

Sargent (2009) explained that conceptual model validity is to determine that (1) the theories and assumptions underlying the conceptual model are correct, and (2) the model representation of the problem entity and the model's structure, logic and mathematical and causal relationships are "reasonable" for the intended purpose of the model.

In the developed system dynamics model, the arrival of vessels is generated using the Poisson distribution. This section aims at validating whether the assumption of choosing Poisson distribution fits the actual pattern of vessels arrival. In Section 5.6.1, it was concluded that the mean arrival of mainlines and feeders in the month of January 2008, February 2008, March 2008, April 2008 and May 2008 are statistically not different. Therefore, this data-set can be generalized to represent the arrival of vessels in one year by fitting the data into a distribution.

In order to confirm that vessels arrival follow the Poisson distribution, the Kolmogorov-Smirnov test was performed on the data of the number of vessels (mainlines + feeders) that arrived at the container terminal in January to May 2008. Levin and Rubin (1998) explained that the Kolmogorov-Smirnov test can be used for testing whether there is a significant difference between an observed frequency distribution and a theoretical frequency distribution. The Kolmogorov-Smirnov test has several advantages over the chi-square test (another goodness-of-fit test), where Kolmogorov-Smirnov test is a more powerful test, and is easier to use because it does not require the data to be grouped in any way (Levin and Rubin, 1998). Hence, Kolmogorov-Smirnov test was chosen in this research.

From the five months of raw data supplied by the statistics officers at the case study container terminal, the mean of vessel arrival was calculated as 9.0855. The Kolmogorov-Smirnov test showed that the Poisson distribution with vessels arrival mean equals to 9.0855 is a good description of pattern of usage ($P \geq 0.05$). Therefore, it can be concluded that Poisson distribution is a good fit for the number of vessels arrival.

After the assumption of using Poisson distribution to generate the vessel arrival was validated, the second part of this section continued with the process of evaluating the conceptual model to determine if the conceptual model is reasonable and correctly represents the actual operation of the container terminal.

Sargent (2009) suggested that one way to validate the conceptual model is through face validation. This author explained that in face validation, the conceptual model is evaluated by the experts of the problem entity to determine if the conceptual model is correct and reasonable for its purpose.

In this research, the conceptual model as seen in Figure 5.1 is validated by the case study container terminal's yard strategist. Therefore, with the vessel arrival assumption and conceptual model validated, the next section moves on to operational validity.

5.6.3 Operational Validity

Sargent (2009) elaborated that operational validity is concerned with determining that the model's output behavior has the accuracy required for the model's intended purpose over the domain of its intended applicability. Sargent (2009) also stated that in order to obtain a high degree of confidence in a model and its results, comparison of the model and actual input-output behaviors for at least two different sets of experimental conditions is usually required.

Both Law (2009) and Sargent (2009) agreed that the comparison of the model and actual output data can be done using numerical statistics and graphical plots. Hence, this research employed both techniques to compare the model and actual output data. The outputs validated are:

1. The number of vessels that arrived.
2. The number of containers unloaded at each berth.
3. The number of containers loaded at each berth.
4. The berth occupancy rate.
5. The yard occupancy rate.

As there are only five months of actual data, the developed system dynamics model was run for five months to simulate the five data-sets. These outputs generated by the developed system dynamics model were compared with the actual data-sets available. Figures 5.10 to 5.14 show that the five pairs of data sets are comparable. Their respective pair-sample t-test also shows that they are not statistically different at 95% significance level (see Appendix 6).

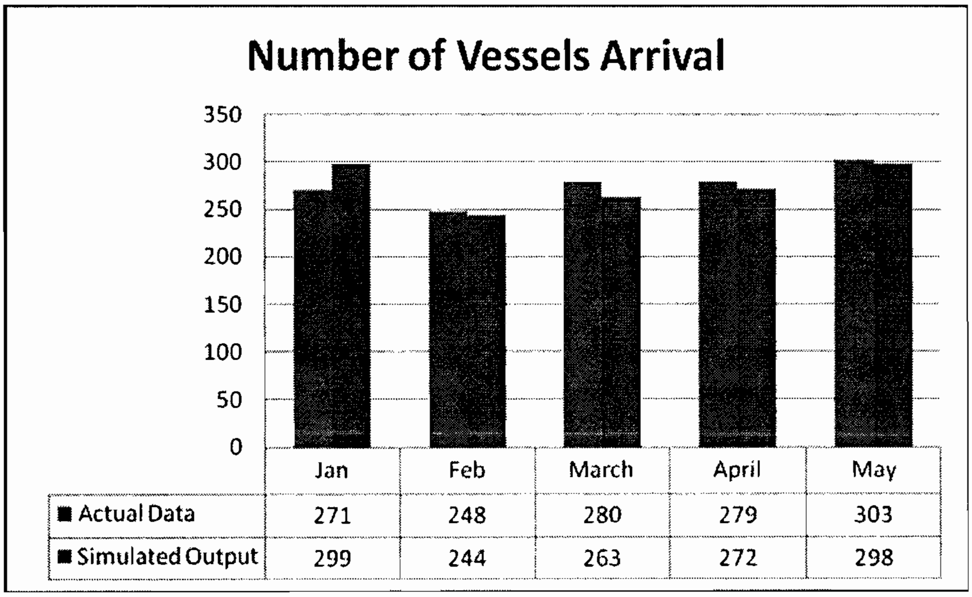


Figure 5.10: The bar chart of the actual data and the simulated output of the number of vessels arrival in January to May 2008.

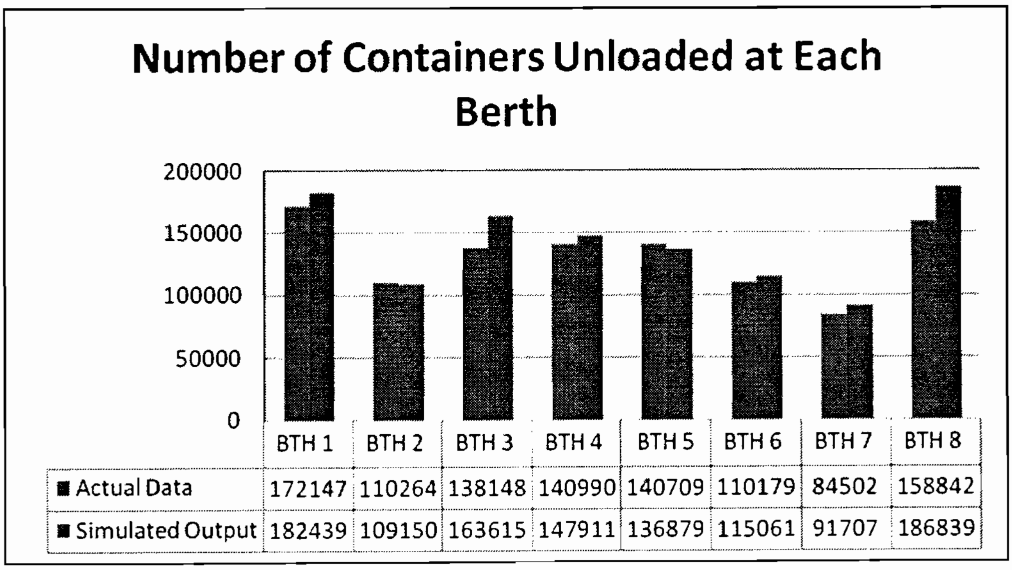
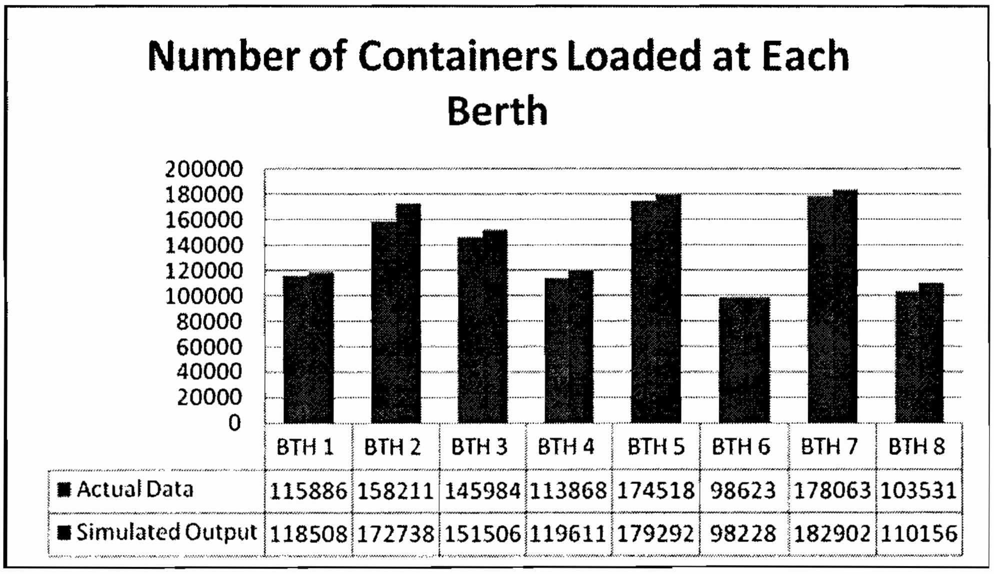
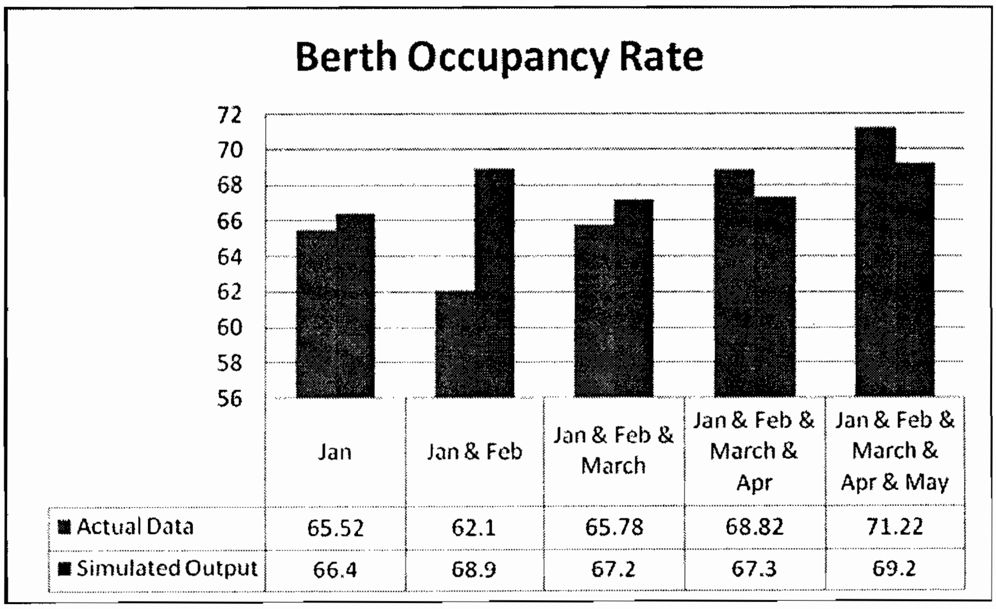


Figure 5.11: The bar chart of the actual data and the simulated output of the number of containers unloaded at each berth in January to May 2008.



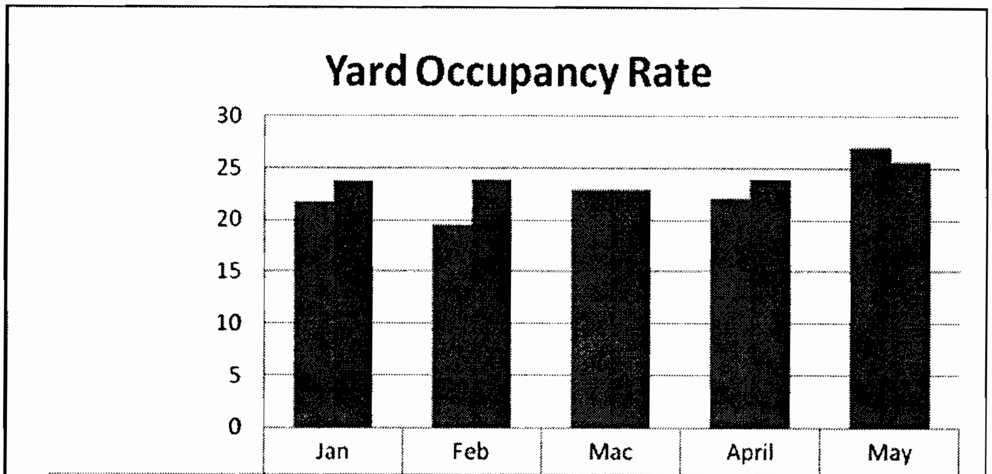
	BTH 1	BTH 2	BTH 3	BTH 4	BTH 5	BTH 6	BTH 7	BTH 8
■ Actual Data	115886	158211	145984	113868	174518	98623	178063	103531
■ Simulated Output	118508	172738	151506	119611	179292	98228	182902	110156

Figure 5.12: The bar chart of the actual data and the simulated output for the number of containers loaded at each berth in January to May 2008.



	Jan	Jan & Feb	Jan & Feb & March	Jan & Feb & March & Apr	Jan & Feb & March & Apr & May
■ Actual Data	65.52	62.1	65.78	68.82	71.22
■ Simulated Output	66.4	68.9	67.2	67.3	69.2

Figure 5.13: The comparison of berth occupancy rate for both actual data and simulated output.



	Jan	Feb	Mac	April	May
■ Actual Data	21.74	19.54	22.95	22.08	27.09
■ Simulated Output	23.8	23.89	22.92	23.99	25.66

Figure 5.14: The comparison of yard occupancy rate for both actual data and simulated output.

This research involved the development of both operational and strategic level models of a container terminal. However, the discussion prior to this emphasized on validating output generated from the operational level model (number of vessels that arrived, the number of containers unloaded at each berth, the number of containers loaded at each berth, the berth occupancy rate and the yard occupancy rate). The following focuses on comparing the output of the strategic level model with the output of the operational level model of the developed system dynamics model.

As the data entered into the operational level model are of January until May 2008, with all the validation procedure performed above, these five months data can be used to run the operational level model for one year, which represents the whole year of 2008. The output generated from the operational level model which was run for one year is compared with the data input of the strategic level model of the year 2008 which was retrieved from the website of the Ministry of Transportation (MOT), Malaysia. Figure 5.15 shows the comparison between the total container throughputs in year 2008 retrieved from MOT's website and the simulation output generated from the operational level model for the same year.

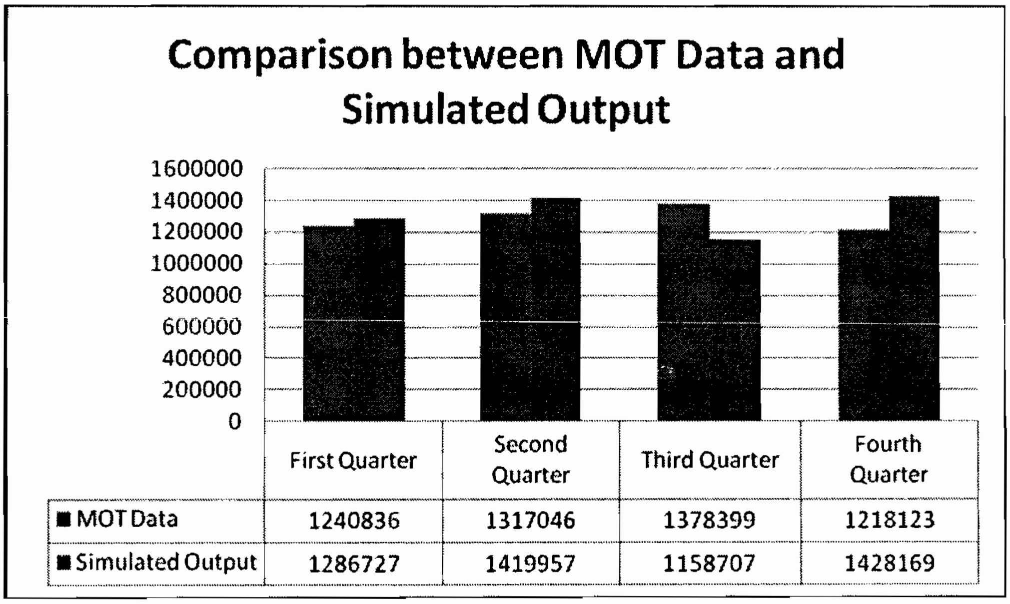


Figure 5.15: The comparison between container throughput in year 2008 for both the data from MOT and simulated output.

Figure 5.15 shows that the container throughput for all four quarters in year 2008, for both MOT data and simulated output, are comparable. Their respective pair-sample t-test also shows that they are not statistically different at 95% significance level (see Appendix 6).

Having shown that the developed system dynamics model is capable of capturing the actual system, in terms of data validity, conceptual model validity and operational validity, the model is deemed fit for experimentation.

5.7 Data Analysis

This section presents the discussion on the analysis of the developed system dynamics model as well as results generated from both the operational and strategic level models. The presentation of this section is divided into two parts. The first is the discussion of the output generated from the operational level model followed by the discussion of the output generated from the strategic level model.

5.7.1 Data Analysis of Operational Level Model

This section presents the discussion on the results generated from the operational level model using system dynamics modeling approach. As the purpose of constructing the operational level model is to achieve the first and second objectives of this research, the presentation of this section encompasses the analysis of the results gathered from the developed operational level model parallel with the process of achieving both the first and second objectives of this research.

The first objective of this research is to model the flow of containers across the berth side and yard side to evaluate the system and to measure performance indicators. Performance indicators are indicators used to describe the behavior of a container terminal (Ragheb Mohamed, 2005). Ragheb Mohamed (2005) and De Monie (1987) divided the performance indicators into six main categories, i.e. the vessel performance indicators, berth performance indicators, cargo handling performance indicators, container yard performance indicators, inland transportation performance indicators and economic impact performance indicators. However, only the first four categories of performance indicators are relevant to this study. The inland transportation performance indicators and the economy impact performance indicators are not within the scope of this research and therefore were not considered.

The four categories of performance indicators used to understand the behavior of the container terminal operations are:

1. Vessel Performance Indicators:
    - Average waiting time
    - Average service time
    - Average turnaround time
2. Berth Performance Indicators:
    - Berth productivity
    - Berth occupancy
    - Loading productivity
    - Unloading productivity
3. Container Handling Performance Indicators:
    - Total container handled
4. Container Yard Performance Indicators:
    - Container yard occupancy

The operational level model was run for one year to generate the four categories of performance indicators.

5.7.1.1 Vessel Performance Indicators

In the category of vessel performance indicators, the average waiting time, average service time and average turnaround time were generated. The average waiting time represents the average time a vessel has to wait before it is served. The average vessel waiting time is calculated by dividing the total waiting time for all vessels with the total vessels served at the container terminal. The total waiting time is the summation of total delay time in departure for all vessels and the total time spent queuing for berth, for all vessels. Formulas for the vessel average waiting time as well as the value generated from the developed system dynamics model are as follows:

$$\text{Total Waiting Time} = \text{Total Delay in Depart} + \text{Total Queue Time for Vessels} \quad (5.38)$$

$$\text{Total Waiting Time} = 182.5 \text{ days} + 118.8 \text{ days} = 301.3 \text{ days} \quad (5.39)$$

$$\text{Average Vessel Waiting Time} = \text{Total Waiting Time} / \text{Total Vessel Served} \quad (5.40)$$

$$\text{Average Vessel Waiting Time} = 301.3 \text{ days} / 3285 \text{ vessels} = 0.0917 \text{ days per vessel} = 2.201 \text{ hours per vessel} \quad (5.41)$$

Next, the average service time represents the average time spent to serve a vessel. The average vessel service time is calculated by dividing the total service time for all vessels with the total of vessels served at the container terminal. The total service time is the summation of the total container unloading time, the total transfer time of containers from berth to block, the total transfer time of containers from block to berth and the total container loading time. Formulas for the average vessel service time as well as the value generated from the developed system dynamics model are as follows:

Total Service Time = Total Container Unloading Time + (5.42)
Total Transfer Time from Berth to Block +
Total Transfer Time from Block to Berth +
Total Container Loading Time

Total Service Time = 842.2 days + 34.3 days + (5.43)
34.2 days + 837.2 days
=1747.9 days

Average Vessel Service Time = Total Service Time / (5.44)
Total Vessel Served

Average Vessel Service Time = 1747.9 days / 3285 vessels (5.45)
= 0.5321 days
= 12.77 hours per vessel

Thirdly, the average vessel turnaround time represents the average time spent by a vessel in the container terminal. The average vessel turnaround time is calculated by dividing the total vessel turnaround time for all vessels with the total vessels served at the container terminal. The total vessel turnaround time is the summation of total vessel waiting time and total vessel service time. Formulas for the average vessel turnaround time as well as the value generated from the model are as follows:

Total Turnaround Time = Total Waiting Time + (5.46)
Total Service Time

Total Turnaround Time = 301.3 days + 1747.9 days (5.47)
= 2049.2 days

Average Vessel Turnaround Time = Total Turnaround Time / (5.48)
Total Vessel Served

Average Vessel Turnaround Time = 2049.2 days / (5.49)
3285 vessels
= 0.6238 days per vessel
= 14.97 hours per vessel

5.7.1.2 Berth Performance Indicators

In this category of performance indicators, the berth productivity per berth, average berth occupancy, loading productivity and unloading productivity were generated. The berth productivity represents the number of containers transferred per berth. Berth productivity is calculated by dividing the total number of containers transferred at all berths with the total number of berths operating. Formulas for the berth productivity as well as the value generated from the developed system dynamics model are as follows:

$$\text{Berth Productivity} = \text{Total Containers Transferred} / \text{Number of Berth Used} \quad (5.50)$$

$$\text{Berth Productivity} = 5,293,560 \text{ TEUs} / 8 \text{ berths} = 661,695 \text{ TEUs per berth} \quad (5.51)$$

Next, the berth occupancy rate indicates the rate berths are occupied with vessels at a given duration. The average berth occupancy rate is defined as follows:

$$\text{Average Berth Occupancy Rate} = (\text{Total Time of Vessels at Berth} * 100) / (\text{Total Number of Berth Used} * \text{Time}) \quad (5.52)$$

$$\text{Average Berth Occupancy Rate} = (2049 \text{ days} * 100) / (8 * 366) = 204900 / 2928 = 69.8\ \% \quad (5.53)$$

Moving on, the loading productivity and unloading productivity represent the average containers loaded per berth and the average containers unloaded per berth respectively. The formula and value for both of these performance indicators are given as follows:

Loading Productivity = Total Container Loaded at Berth / Number of Berth Used (5.54)
= 2, 645, 391 TEUs / 8 berths
= 330, 674 TEUs per berth

Unloading Productivity = Total Container Unloaded at Berth / Number of Berth Used (5.55)
= 2, 648, 170 TEUs / 8 berths
= 331, 021 TEUs per berth

5.7.1.3 Container Handling Performance Indicators

In the category of container handling performance indicators, the total containers handled at the container terminal was generated. The total containers handled is the summation of the total containers unloaded and the total containers loaded at the container terminal. Formulas for the total container handled as well as the value generated from the developed system dynamics model are as follows:

Total Container Handled = Total Container Unloaded + Total Container Loaded (5.56)
= 2, 648, 170 TEUs + 2, 645, 391 TEUs
= 5, 293, 560 TEUs

5.7.1.4 Container Yard Performance Indicators

Finally, for the last category of performance indicators, which is the container yard performance indicators, the yard occupancy rate was generated. The yard occupancy rate indicates the level of demand for yard services. Average yard occupancy rate is defined as follows:

Yard Occupancy Rate = (Total Containers Occupied / Total Yard Capacity) * 100 (5.57)

Yard Occupancy Rate = (37652 TEUs / 151200 TEUs) * 100 (5.58)
= 24.9 %

To summarize, the output generated from the developed model indicated that the average vessel turnaround time is 14.97 hours while both the average vessel waiting time and average vessel service time are 2.20 hours and 12.77 hours respectively. On the other hand, the berth productivity is 661, 695 TEUs per berth while the berth occupancy rate is 69.8%. As for loading and unloading productivity, the values are 330, 674 TEUs per berth and 331, 021 TEUs per berth respectively. Finally, the total containers handled in one year are 5, 293, 560 TEUs and the yard occupancy rate is 24.9% (Table 5.11).

Table 5.11: Summary of the performance indicators generated from the developed system dynamics model.

Performance Indicators	Output
Vessels Performance Indicators	
1. Average Turnaround Time	14.97 hours per vessel
2. Average Service Time	12.77 hours per vessel
3. Average Waiting Time	2.20 hours per vessel
Berth Performance Indicators	
1. Berth Productivity	661, 695 TEUs per berth
2. Berth Occupancy Rate	69.8 %
3. Loading Productivity	330, 674 TEUs per berth
4. Unloading Productivity	331, 021 TEUs per berth
Container Handling Performance Indicators	
1. Total Container Handled	5, 293, 560 TEUs
Container Performance Indicator	
1. Container Yard Occupancy	24.9 %

The subsequent presentation emphasizes on the analysis of the generated performance indicators and a detailed discussion on how these performance indicators are interrelated with each other. This is parallel with the fulfillment of the second objective of this research which is to evaluate the relationship and interdependency between the berth and yard operations and to understand how it affects the terminal overall efficiency and determine which area should be focused on to create improvement.

As the essence of the second objective is to evaluate the relationship and interdependency between the operation of the berth and yard, the berth occupancy rate and yard occupancy rate give an overview of how these two subsystems are operating. The berth occupancy rate is 69.8%. The maximum internationally acceptable standard for berth occupancy rate according to Ray and Blankfeld (2002) is 40%. Therefore, it can be concluded that the terminal has a high berth occupancy rate.

Ramani (1996) explained that a high value of berth occupancy shows that either the terminal is congested with long vessel waiting time to berth, or inefficient utilization of the terminal equipment, while a low value of berth occupancy rate indicates few vessels arrival and underutilization of terminal resources. Therefore, the berth occupancy rate of 69.8% can be translated as the case study container terminal facing congestion to a certain degree. In order to determine how to reduce the berth occupancy rate, the relationship of berth occupancy rate with the other performance indicators such as vessel turnaround time, vessel service time and vessel waiting time was examined.

The generated yard occupancy rate is 24.9%. Up to date, there is no published report on the benchmark of yard occupancy rate. According to the yard strategist at the case study container terminal, it is the strategy of the terminal to maintain a large yard capacity for future container growth. Therefore, the low value of yard occupancy rate (24.9%) generated from the model is parallel with the strategy adopted by the case study container terminal.

5.7.1.5 Analyzing Berth Occupancy Rate

The berth occupancy rate represents the percentage of time vessels are berthed at the container terminal. According to Ramani (1996), there are two possibilities for a high berth occupancy rate. First, the terminal is congested and vessels have to spend long time waiting for available berth. The second possibility is inefficient utilization of the terminal equipments.

The berth occupancy rate of 69.8%, which represents the average berth occupancy rate for the year 2008, is higher than the maximum international standard of 40%. In order to gain insight to the berth operations, the daily berth occupancy rate throughout the year of 2008 was analyzed. Figure 5.16 shows the daily berth occupancy rate of the case study container terminal throughout the year 2008. It can be seen that there are frequent days where the berth occupancy rate reaches the maximum of 100%, although the average is 69.8%.

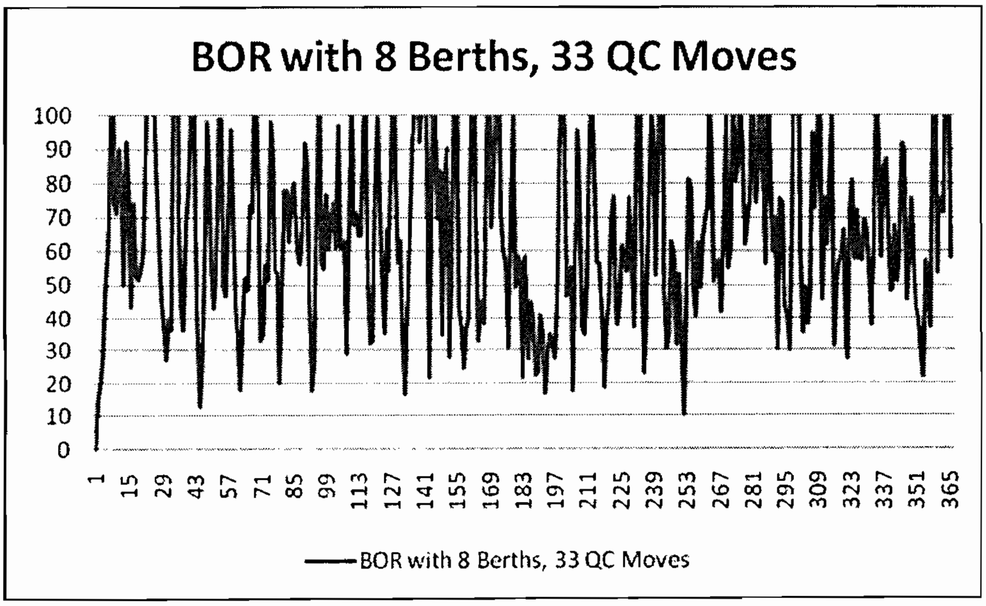


Figure 5.16: Daily berth occupancy rate for the year 2008.

Therefore, in order to reduce the berth occupancy rate, there is a need to understand which factors play a major role in affecting the berth occupancy rate. The berth occupancy rate is influenced by the vessel turnaround time, which in turn is influenced by the vessel service time. The berth occupancy rate directly impacts vessel waiting time. The higher the berth occupancy rate, the longer vessels have to wait to be berthed. The vessel service time, on the other hand, is dependent on the speed of the container unloading and loading process. The whole process is influenced by quay cranes moves and prime movers travelling distances. Figure 5.17 exhibits the discussed relationships.

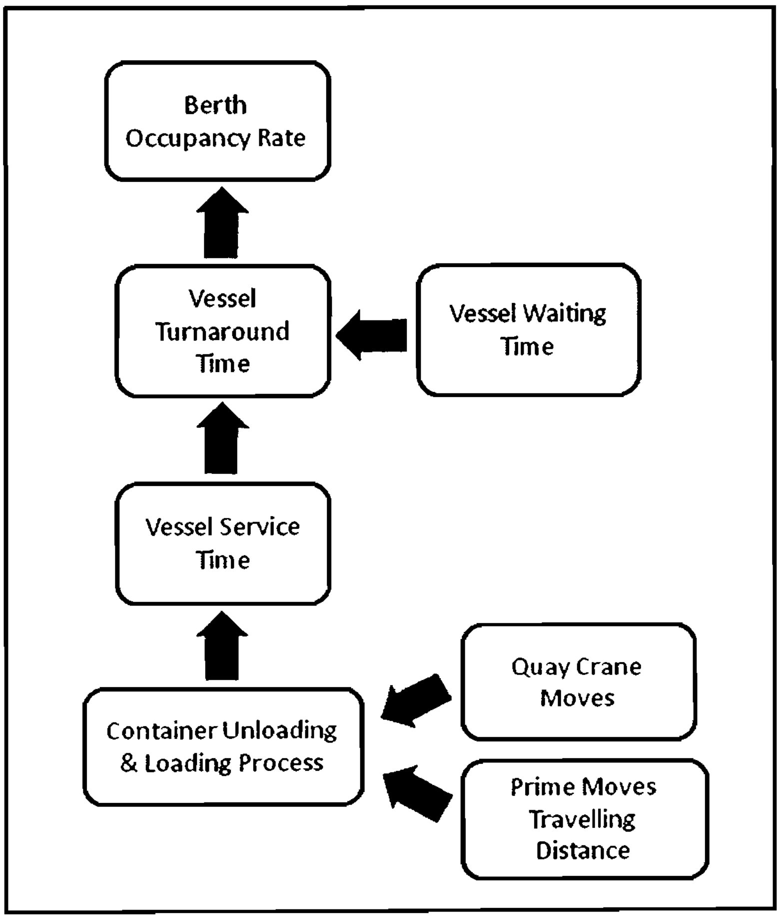


Figure 5.17: Relationship and interdependency of berth occupancy rate with other elements in the case study container terminal.

5.7.1.6 Analyzing Quay Crane Moves

Quay cranes are used to unload containers from vessels and load containers onto vessels. The quay cranes at the case study container terminal performed an average of 33 container moves per hour, which means in one hour, the quay cranes transferred an average of 33 containers. Thirty-three container moves per hour are relatively low compared to Hong Kong Port and Kaohsiung Port which ranked top ten in the world leading ports. Both ports managed 40 container moves per hour and 35 container moves per hour respectively (Chou et. al., 2003).

Given the available capacity and infrastructures, the case study container terminal is capable of delivering up to 40 container moves per hour. In order for the case study container terminal to provide a more efficient service in terms of servicing vessels within the shortest time period, the developed system dynamics model was used to experiment the impact of increasing the quay cranes moves gradually from 33 moves to 40 moves. Table 5.12 shows the adjustment of quay cranes moves and its impact on the average berth occupancy rate and average vessel turnaround time. With every increase of one quay crane move per hour, the average berth occupancy rate is reduced by about 1% and the average vessel turnaround time reduced by about 25 minutes.

Table 5.12: Impact of the adjustment in quay crane moves on average vessel turnaround time and average berth occupancy rate.

Quay Cranes Moves Per Hour	Average Vessel Turnaround Time	Average Berth Occupancy Rate
33	14.98 hours	69.8 %
34	14.62 hours	68.1 %
35	14.28 hours	66.5 %
36	13.96 hours	65 %
37	13.66 hours	63.6 %
38	13.37 hours	62.3 %
39	13.09 hours	61 %
40	12.83 hours	59.8 %

Although the increase in quay crane productivity from 33 container moves per hour to 40 container moves per hour did decrease both average vessel turnaround time and average berth occupancy rate, the berth occupancy rates (60%) is still higher than the maximum internationally acceptable standard of 40% (Ray and Blankfeld, 2002).

While there is a need to improve the utilization of quay cranes, the analysis so far shows that the case study container terminal may need to expand its current berth capacity in order to alleviate the high berth occupancy problem as well as prepare the terminal for servicing a larger container volume in future. Before justifying that the case study container terminal may need additional capacity, the prime movers travelling distance was analyzed as it also has an impact on berth occupancy rate.

5.7.1.7 Analyzing Prime Mover Traveling Distance

The function of a prime mover is to transfer containers between the berth and blocks. Rationally, it is better to store containers at blocks which are situated near to its loading point at the berth. This is to minimize the travelling distance of the prime movers between the yard blocks and the berth. If the travelling distance is minimized, automatically, the time spent to transfer these containers between yard blocks and berths can be minimized as well. Eventually containers unloading and loading time will be minimized, thus contributing to a lower vessel service time, vessel turnaround time and also the berth occupancy rate.

There are a total of 12 yard blocks at the case study container terminal. The travelling distance of prime movers was analyzed using the system dynamics model developed. Basically, from the analysis, there are two types of position determined. The first type is the favorable position where containers that are stored in these blocks are situated near to their loading position at berth, therefore the travelling distance of prime movers is minimized. The second type is the unfavorable position where containers stored in these blocks have to travel more to their loading position at the berth due to the greater distance between the yard block and the loading point at berths.

Figure 5.18 demonstrates an example of favorable position (FP) and unfavorable position (UnFP) of container storing location in yard blocks, given the containers are loaded at berth 1. The location of yard blocks 1, 2, and 3 is situated near berth 1, therefore, if the containers that are loaded at berth 1 are stored in these three blocks, the traveling distance of the prime mover as well as time taken to transfer these containers from the yard block to the berth can be minimized. Thus, yard blocks 1, 2 and 3, for example, can be classified as favorable storing positions for containers to be loaded at berth 1. On the other hand, if the containers are stored in yard block 12, but to be loaded at berth 1, prime movers need to travel a further distance to transport the containers from block 12 to berth 1, thus increasing the prime mover traveling distance. Therefore, yard block 12 is classified as an unfavorable position for storing containers to be loaded at berth 1.

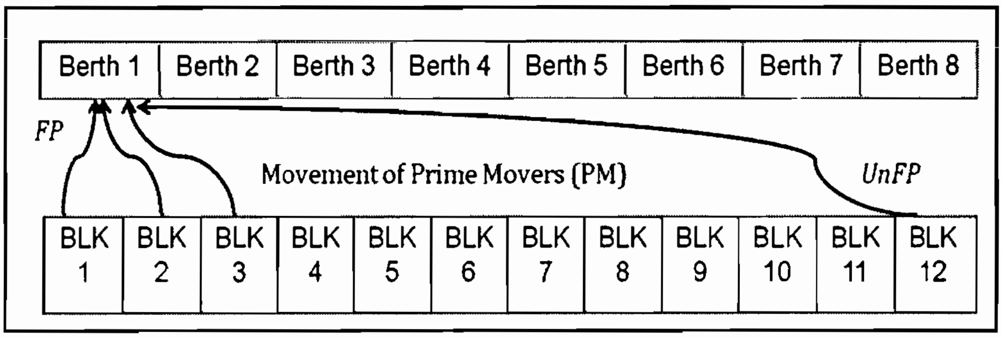


Figure 5.18: Example of favorable positions and unfavorable positions of containers storing locations.

The decision rules that the yard strategist at the case study container terminal uses to allocate containers to yard blocks is that, the storage location of the containers must not be more than 800m from the berthing point at the berth. Table 5.13 shows the 12 yard blocks and their favorable loading points at berths. The listed favorable positions are all within the range of 800m between the yard blocks and the berths.

Table 5.13: The favorable loading point at berths for all twelve yard blocks.

Yard Block	Favorable Position for Loading Point at Berth
Yard Block 1	Berth 1, Berth 2 and Berth 3
Yard Block 2	Berth 1, Berth 2, Berth 3 and Berth 4
Yard Block 3	Berth 1, Berth 2, Berth 3 and Berth 4
Yard Block 4	Berth 1, Berth 2, Berth 3, Berth 4 and Berth 5
Yard Block 5	Berth 1, Berth 2, Berth 3, Berth 4, Berth 5 and Berth 6
Yard Block 6	Berth 2, Berth 3, Berth 4, Berth 5 and Berth 6
Yard Block 7	Berth 3, Berth 4, Berth 5, Berth 6 and Berth 7
Yard Block 8	Berth 3, Berth 4, Berth 5, Berth 6, Berth 7 and Berth 8
Yard Block 9	Berth 4, Berth 5, Berth 6, Berth 7 and Berth 8
Yard Block 10	Berth 5, Berth 6, Berth 7 and Berth 8
Yard Block 11	Berth 5, Berth 6, Berth 7 and Berth 8
Yard Block 12	Berth 6, Berth 7 and Berth 8

The analysis of the 12 yard blocks was performed using the operational level system dynamics model to investigate the composition of containers stored at the 12 yard blocks to its loading point at berth. Figure 5.19 exhibits the percentage of containers that exited from yard block 1 to all eight berths. The percentage of containers that exited from yard block 2 to yard block 12 to the loading point at each of the eight berths are shown in Appendix 7.

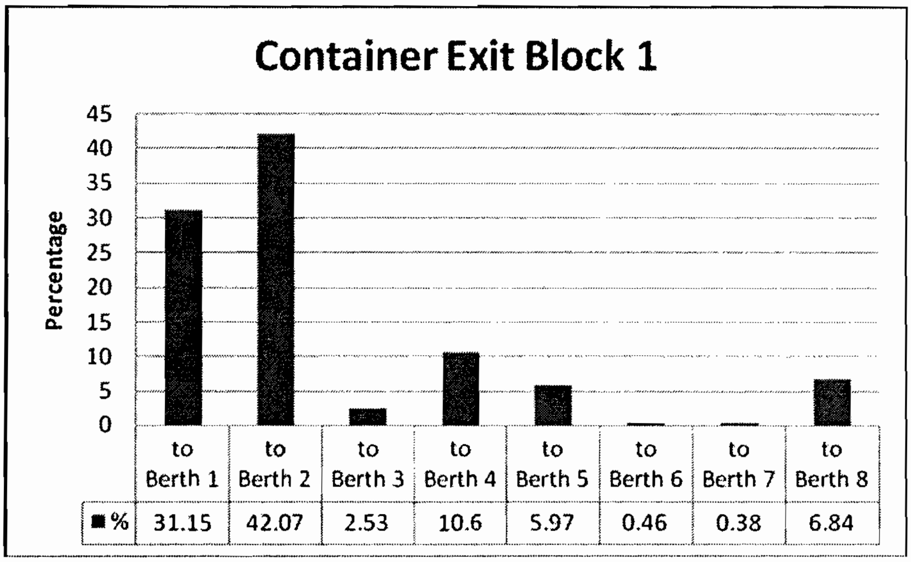


Figure 5.19: Percentage of containers exiting Yard Block 1 to the all eight berths.

Figure 5.19 shows the percentage of containers exiting yard block 1 to all eight berths. From Table 5.12, the favorable loading point of containers stored in yard block 1 is berth 1, berth 2 or berth 3 in order to be within the range of 800m distance between yard block and berth. Although in real operations, as exhibited in Figure 5.19, the majority of containers stored in yard block 1 are transferred to berth 1 (31.15%) and berth 2 (42.07%) which are the favorable positions, there are containers being loaded in berth 4 (10.6%), 5 (5.97%) and 8 (6.84%) where these percentages are even higher than that of berth 3 (2.53%). Therefore, as there are containers stored in yard block 1 being loaded in berths 4, 5 and 8 (unfavorable positions), the prime movers need to travel a further distance to transport these containers to their loading point, thus increasing the vessel service time and berth occupancy rate.

Table 5.14 summarizes the breakdown percentage of containers exiting all 12 yard blocks to favorable and unfavorable loading point at berth. The percentage values for both favorable position and unfavorable position exhibited in Table 5.14 are actually the summation of all percentage for favorable and unfavorable berth loading points with Table 5.13 as the guideline. For example, the favorable loading point for containers stored in yard block 1 are to berths 1, 2 and 3 as given in Table 5.13. In Figure 5.19, the total percentage of containers exiting to these favorable loading positions are 75.75% (31.15% to berth 1, 42.07% to berth 2 and 2.53% to berth 3) while the total percentage of containers exiting to unfavorable loading point at berths 4, 5, 6, 7 and 8 are 24.25%.

Although majority of the containers stored at each of the 12 yard blocks are loaded at its favorable loading points where the distance between these blocks and the berths are within the range of 800m, there are containers stored in these blocks being loaded at unfavorable loading point (Table 5.14). This is especially for yard block 1 and yard block 6 where the percentage of containers being loaded to further berths are the highest among all 12 yard blocks with 24.25% and 29.34% respectively.

Table 5.14: Percentage of containers at favorable position and unfavorable position for loading point at berth.

Yard Block	Percentage of containers exiting to favorable loading point at berth	Percentage of containers exiting to unfavorable loading point at berth
Yard Block 1	75.75 %	24.25 %
Yard Block 2	83.53 %	16.47 %
Yard Block 3	84.2 %	15.8 %
Yard Block 4	94.21 %	5.79 %
Yard Block 5	91.03 %	8.97 %
Yard Block 6	70.66 %	29.34 %
Yard Block 7	83.38 %	16.62 %
Yard Block 8	92.32 %	7.68 %
Yard Block 9	87.05 %	12.95 %
Yard Block 10	90.67 %	9.32 %
Yard Block 11	95.03 %	4.98 %
Yard Block 12	88.45 %	11.55 %

In order to minimize the prime mover traveling distance between the yard blocks and berths, it is best if the containers stored in each yard blocks are loaded at their favorable loading points at berth. By minimizing the prime mover traveling distance, the time spent by these prime movers to transport these containers can also be minimized. The minimization of prime mover traveling time can help to lower the total time spent to service a vessel, thus reducing the vessel turnaround time and the berth occupancy rate.

5.7.2 Data Analysis of Strategic Level Model

The third objective of this research is to develop a capacity plan to ensure the terminal has sufficient capacity to sustain demand and at the same time, avoid overcapacity. In order to achieve this objective a strategic level model was developed.

Before continuing the discussion on the results generated from the developed strategic level model, this section starts off with the continuation of discussion on the high berth occupancy rate in Section 5.7.1.6 as the theme of this section is dedicated to capacity planning discussion.

In Section 5.7.1.6, the berth occupancy rate appears to be higher than 40% which is the maximum international standard according to Ray and Blankfeld (2002), even when the quay crane moves is increased to 40 moves per hour. One alternative to lower the berth occupancy rate is by introducing additional capacity or new berths. There were a total of eight berths operating during the conduct of this research at the case study container terminal. Analysis was performed using the operational level model to investigate the impact of additional new berths to the berth occupancy rate.

Table 5.15 exhibits the impact on average berth occupancy rate by increasing quay crane productivity and introducing one additional berth while Table 5.16 presents the impact of both increasing quay crane productivity and the introduction of two new berths to the average berth occupancy rate.

Table 5.15: Impact of increasing quay crane productivity and adding one additional berth on average berth occupancy rate.

Quay Cranes Moves Per Hour	Average Vessel Turnaround Time	Average Berth Occupancy Rate
33	13.33 hours	55.2 %
34	13.01 hours	53.9 %
35	12.71 hours	52.7 %
36	12.42 hours	51.5 %
37	12.15 hours	50.3 %
38	11.89 hours	49.3 %
39	11.65 hours	48.3 %
40	11.42 hours	47.3 %

Table 5.16: Impact of increasing quay crane productivity and adding two additional berths on average berth occupancy rate.

Quay Cranes Moves Per Hour	Average Vessel Turnaround Time	Average Berth Occupancy Rate
33	12.31 hours	45.9 %
34	12.02 hours	44.8 %
35	11.75 hours	43.8 %
36	11.49 hours	42.8 %
37	11.25 hours	41.9 %
38	11.02 hours	41 %
39	10.80 hours	40.2 %
40	10.59 hours	39.5 %

As illustrated in Table 5.15, even with the introduction of one additional new berth and increase of quay crane productivity to 40 moves per hour, the average berth occupancy rate is still higher than the maximum international standard of 40%. On the other hand, with composition of two additional berths and quay crane productivity of 39 movers and 40 moves per hour, the average berth occupancy rate can be lowered to 40.2% and 39.5% respectively (Table 5.16).

This shows that not only introducing extra capacity is important but also the efficient use of terminal capacity in alleviating the high berth occupancy rate. This is because, even with the extra new berths, if the case study container terminal still maintains the average quay crane moves of 33 moves, the average berth occupancy rate of 45.9% is still higher than the maximum international standard.

The suggestion of introducing two additional berths to the case study container terminal as an alternative to lower the berth occupancy rate as well as preparing the terminal for the increase in future container throughput is actually parallel with the decision made by the case study container terminal. As of early 2009, the case study container terminal introduced two new berths to the terminal operations, thus having a total of ten berths operating at the container terminal.

While the operational level model successfully predicted the necessity of introducing two new berths, the strategic level model was used to identify when new terminal facilities are needed to ensure the case study container terminal has sufficient capacity to sustain demand and minimize the risk of overcapacity for the next three years.

The strategic level model was run for 10 years, starting from the year 2002 to 2012. Two important data that are inputted into the strategic level model are the annual container throughput of the case study container terminal and the total number of vessels that called at the case study container terminal.

Figure 5.20 exhibits the annual container throughput for years 2002 to 2012 while Figure 5.21 presents the total number of vessels that called at the case study container terminal for years 2002 to 2012. Both Figures 5.20 and 5.21 are illustrated based on the values from Tables 5.8 and 5.9 respectively. Similarly, the values beyond the year 2009 are forecasted values.

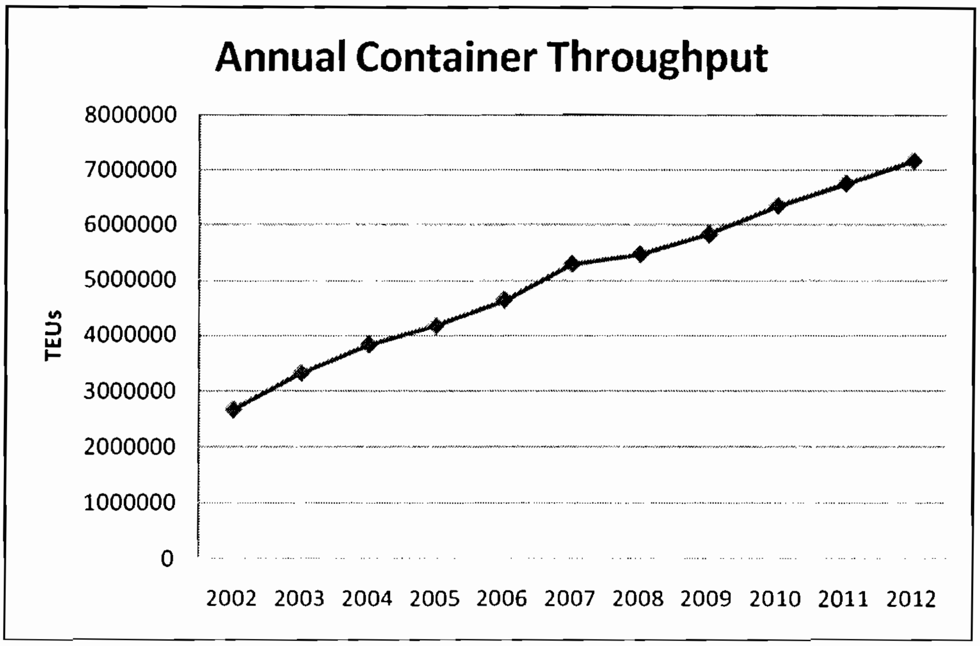


Figure 5.20: Annual container throughput from the year 2002 to 2012.

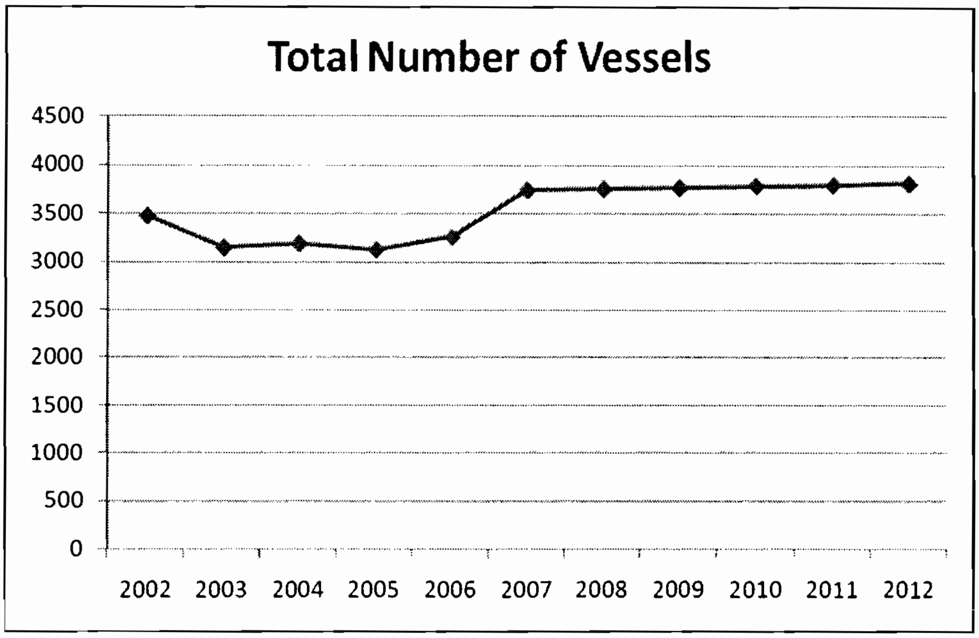


Figure 5.21: Total number of vessels arrived from the year 2002 to 2012.

From both Figures 5.20 and 5.21, it is clear that the total number of vessels calling at case study container terminal as well as the total container throughput of the case study container terminal are likely to increase each year due to the growing importance of international seaborne trade.

> Despite the rising energy prices and their potential implications for transport costs and trade, and despite growing global risks and uncertainties from factors such as soaring non-oil commodity prices, the global credit crunch, a depreciation of the United States dollar and unfolding food crisis, the world economy and trade have, so far, shown resilience (UNCTAD, 2008).

The continuous growth in world trade and economy has also increased the seaborne trade volume. In order to take advantage of the positive growth of seaborne trade, global liner operators are introducing larger vessels that are capable of carrying more containers and at the same time, achieving economies of scale. Therefore, the case study container terminal not only needs to anticipate both growing container throughput and the increasing number of vessels calling at the port but also larger vessels.

Without adequate capacity, the case study container terminal will not be able to gain competitive advantage from this positive growth. However, adding capacity without detailed planning can be costly as the terminal may face the problem of overcapacity. Therefore, the strategic level model can be used to provide insight and analysis on when extra capacity should be introduced parallel with the growth on container throughput, number of incoming vessels and vessel size.

The berth occupancy rates generated from the strategic level model for the year 2009 and beyond were analyzed for the future capacity. Table 5.17 demonstrates the value of berth occupancy rate for the year 2002 to 2012. As seen in Table 5.17, with the initial capacity of eight berths, the berth occupancy rate increases steadily parallel with the increase of the annual container throughput and vessels served at the case study container terminal. The berth occupancy rates for the year beyond 2004 are all higher than 40%, the maximum international standard for berth occupancy rate. At the year 2008, the berth occupancy rate is 58.93% and this value reaches 77.30% by the year 2012. This indicates that there is a need for new berths to alleviate the high value of berth occupancy rate.

Table 5.17: The berth occupancy rate for the year 2002 to 2012.

Year	Berth Occupancy Rate
2002	28.77%
2003	35.76%
2004	41.35%
2005	45.03%
2006	49.99%
2007	57.11%
2008	58.93%
2009	62.91%
2010	68.48%
2011	72.89%
2012	77.30%

It can be noted that the berth occupancy rate generated from the strategic level model for the year 2008 (Table 5.17) is 58.93% while the berth occupancy rate for the same year generated from the operational level model (Table 5.11 in page 147) is 69.98%. This variation occurred because the berth occupancy rate for the operational level model is calculated using the daily data supplied by the case study container terminal while the berth occupancy rate for the strategic level model is calculated using the yearly data retrieved from the Ministry of Transportation (MOT), Malaysia. The berth occupancy rate is dependent on the total container throughput handled by the case study container terminal. The comparison between container throughput in the year 2008 for both the data from MOT and simulated output has been validated (Figure 5.15 in page 139) using pair-sample t-test. Thus, the variation of berth occupancy rate for both operational and strategic level models is acceptable.

The maximum international standard of berth occupancy rate is 40% where the values beyond this signify that the container terminal faces congestion problem. However, there is no published report which discusses at what percentage of berth occupancy rate, additional berths are required and how many berths should be introduced at the container terminal. From Table 5.17, the berth occupancy rate for year 2009 is 62.91%. As was mentioned previously, in early 2009, the case study container terminal introduced two new berths, making a total of ten berths operating at the terminal. Therefore, the benchmark of 60% is used in this study as the indicator of when the case study container terminal will need new berths. In short, when the berth occupancy rate reaches 60%, two new berths should be added.

Table 5.18 presents the comparison of berth occupancy rate with the initial eight, 10 and 12 berths. With the two additional berths, the berth occupancy rate for the year 2009 is 50.32%, as compared with the higher 62.91% (initial eight berths). The addition of berth 9 and berth 10 reduces the berth occupancy rate significantly until the year 2012 where the berth occupancy rate reaches 61.84%. Therefore, there will be a need to introduce another two new berths at the year 2012 to alleviate the berth occupancy rate. With twelve berths, the berth occupancy rate for the year 2012 will be reduced to 51.35%. Next, the impact of the introduction of larger vessel size on the current berth space is discussed.

Table 5.18: The comparison of berth occupancy rate for eight, ten and twelve berths.

Year	Berth Occupancy Rate with 8 Berths	Berth Occupancy Rate with 10 Berths	Berth Occupancy Rate with 12 Berths
2007	57.11%	45.69%	38.07%
2008	58.93%	47.14%	38.29%
2009	**62.91%**	50.32%	41.94%
2010	**68.48%**	54.78%	45.65%
2011	**72.89%**	58.31%	48.59%
2012	**77.30%**	**61.84%**	51.53%

5.7.2.1 The Impact of Large Vessels Size on Berth Space

There were eight berths operating at the case study container terminal during this research in early 2008. The length of each berth is 360m, making the total capacity of all eight berths 2880m. The average length of vessels that arrived at the case study container terminal in year 2007 was given as 251.662m. The strategic level model is used to gain insight on the impact of vessel size on berth capacity.

Table 5.19 shows the berth capacity rate for the year 2002 to 2012. The berth capacity rate appears to be all above 80% starting from the year 2007 to the year 2012.

Table 5.19: The berth capacity rate for the year 2002 to 2012.

Year	Berth Capacity Rate
2002	83.16%
2003	75.16%
2004	76.23%
2005	74.68%
2006	77.86%
2007	89.46%
2008	89.77%
2009	90.15%
2010	90.49%
2011	90.82%
2012	91.18%

Similar to the berth occupancy rate, there is no published report of the standard maximum of berth capacity rate, where exceeding this maximum value indicates that the container terminal requires additional berths. Nevertheless, the case study container terminal introduced additional two berths in early 2009 when the berth capacity rate for the year 2008 reached 89.77% with the initial eight berths (Table 5.19). Therefore, in this research, the benchmark of 90% is used to indicate the maximum berth capacity rate when new berths need to be introduced in order for the container terminal to service vessels of larger sizes.

Table 5.20 exhibits the comparison of berth capacity rate for eight and 10 berths. With the introduction of two new berths of 360m each in 2009, the berth capacity rate was reduced to 72.12%, which is 18.03% lower than the initial capacity of eight berths. With berth 9 and berth 10, the berth capacity rate for the year 2010 to 2012 can be reduced to 72.39%, 72.66% and 72.94% respectively.

It should be noted that the discussion on the berth capacity rate so far is based on the average vessel length of 251.662m. As the average vessel length varies each year due to the quick growth in vessel length, the discussion so far is an estimation based on the average length of the vessels that arrived at the case study container terminal in the year 2007. The strategic level model however allows users to increase the average vessel size at any chosen year to see its effect on the overall berth capacity rate. This is discussed in the next chapter.

Table 5.20: The comparison of berth capacity rate for eight and ten berths.

Year	Berth Capacity Rate with 8 Berths	Berth Capacity Rate with 10 Berths
2007	89.46%	71.57%
2008	89.77%	71.82%
2009	**90.15%**	72.12%
2010	**90.49%**	72.39%
2011	**90.82%**	72.66%
2012	**91.18%**	72.94%

5.8 Conclusion

This chapter starts off with the discussion on scope of the system dynamics model to be developed. With a clear model boundary, a conceptual model that represents the operation of the case study container terminal was developed. The conceptual model was then translated to the causal loop diagram to understand the relationship and interdependency of each element contained in the conceptual model. The developed causal loop diagram was then translated into the stock and flow diagram.

Two models were developed in this study using *iThink* software; the first represents the operational level processes while the second represents the strategic level processes of the case study container terminal. These two models were validated and verified on both the input data as well as the output generated.

After going through the model verification and validation process, discussion on the data analysis or results generated from the both operational level model and strategic level model were presented. This covered the first, second and the third objectives of this research. The next chapter focuses on the fourth objective of this research, which is the development of management flight simulator or Microworlds.

CHAPTER 6

THE DEVELOPMENT OF MICROWORLDS

6.1 Introduction

This chapter focuses on achieving the fourth objective of this research, which is to develop a management flight simulator or Microworlds. The purpose of developing the Microworlds is to allow container terminal managers to understand the relationship and interdependency between subsystems at the container terminal as well as to test new strategies before implementing it into the real world. The developed Microworlds is a virtual world representing the container terminal operations. It allows container terminal managers to experiment and analyze the consequence of their actions and decision making in a safe environment.

This chapter begins with an overview of the developed Microworlds. Microworlds for the operational level model is discussed next, followed by the presentation on the Microworlds for strategic level model. This chapter wraps up with a conclusion.

6.2 Overview of the Developed Microworlds

As was discussed in Chapter 5, an operational level model and a strategic level model were developed to achieve the first three objectives of this research. Microworlds is constructed in order to aid port managers or general users throughout the developed model, without the need for dealing with the complexity of the model and its mathematical equations.

The Microworlds is constructed at the interface layer of the *iThink* software where user-friendly graphical interfaces can be developed. Microworlds developed in this research consists of two models: the operational level model and the strategic level model.

Figure 6.1 demonstrates the front page of the developed Microworlds. The front page consists of the title of the developed Microworlds, which is the *Interactive Microworlds for Container Terminal* and also a short introduction on the benefits and scope of the developed Microworlds. On the left side of the front page, a main menu panel with five buttons is embedded. The *Home* button navigates users to the front page of the developed Microworlds; the *Tour Model* button navigates users to the *Tour Model* page where the operations of the container terminal is discussed. Next, the *Operational Planning* button navigates users to the *Operational Planning* page. Similarly, the *Strategic Planning* button navigates users to the *Strategic Planning* page. The last button is the *Exit* button, where the Microworlds can be terminated with a single click.

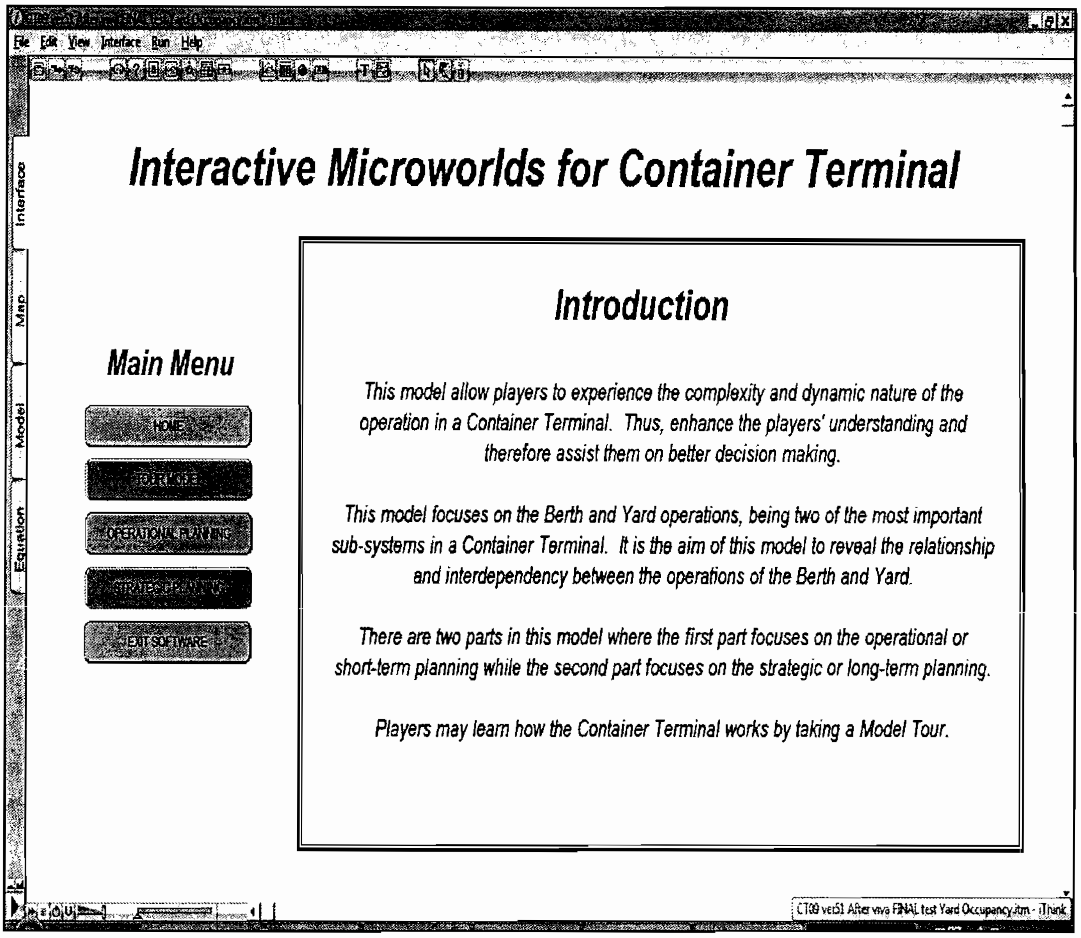


Figure 6.1: The front page of the developed Microworlds.

Figure 6.2 exhibits the *Tour Model* page. The purpose of this page is to provide users some insights on how the case study container terminal operates. A brief explanation on the berth operation and the yard operations are provided. This page also allows users to navigate into the system dynamics model for a graphical view on how each process is carried out at the berth and yard. The *Click to Understand Berth Operation* button and *Click to Understand Yard Operation* button support the navigations to the system dynamics model.

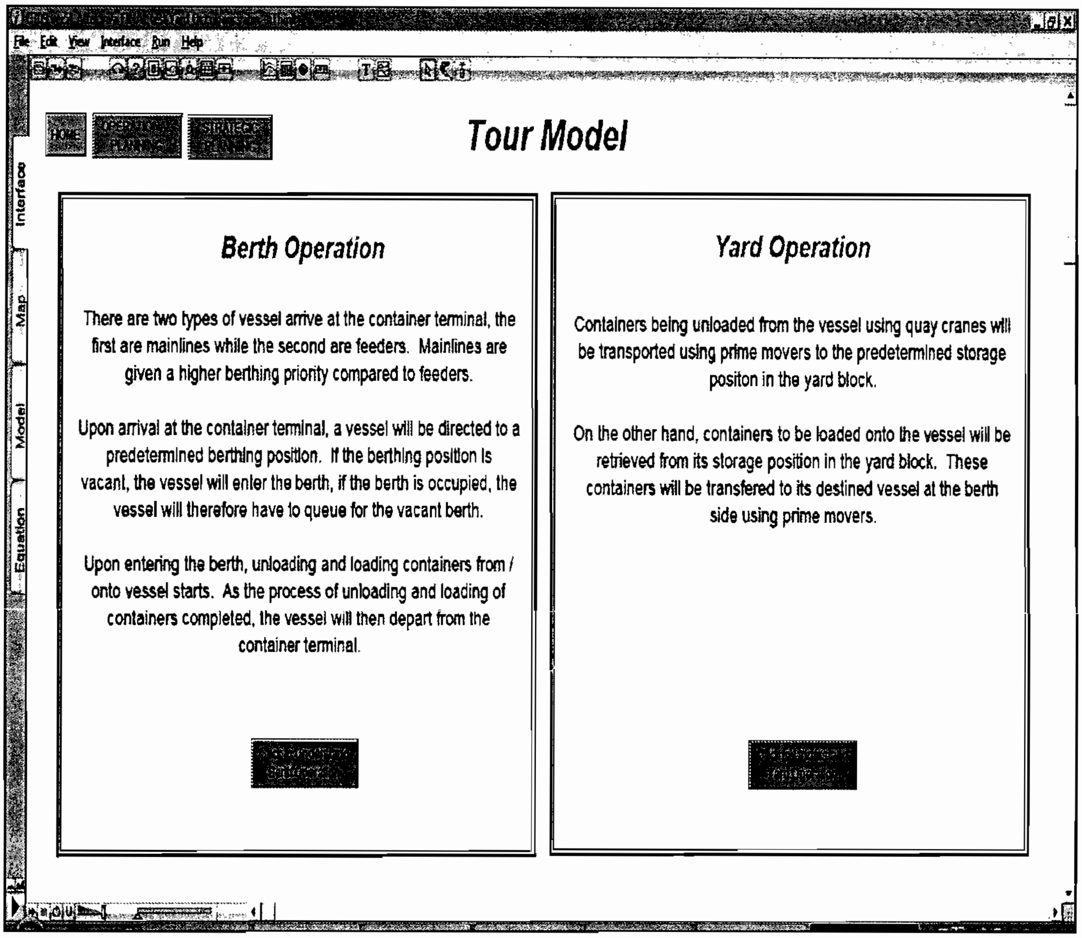


Figure 6.2: The *Tour Model* page of the developed Microworlds.

By clicking the *Click to Understand Berth Operation* button, users will be transported to a storytelling-like page which describes the details of berth operations starting from the arrival of vessels, containers being unloaded and loaded from/onto vessels and finally the departure of vessels (Figure 6.3).

Similarly, by clicking the *Click to Understand Yard Operation* button, a storytelling-like page that describes the yard operations pops up. Figure 6.4 exhibits the page that gives the detailed processes of the yard, starting from the process of prime movers transporting the unloaded containers from the berth side to the yard side, storing the unloaded containers at the yard blocks, to retrieving and transferring the containers to the berth side for loading.

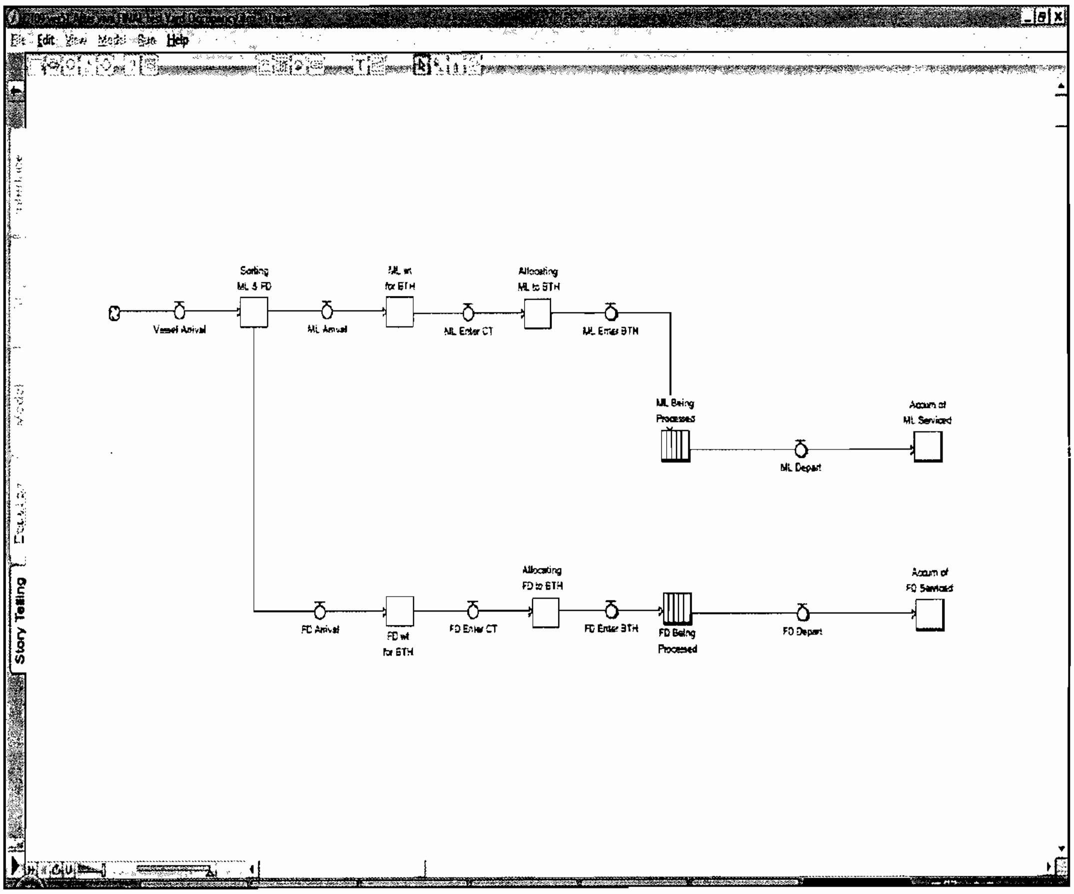


Figure 6.3: The detail processes of the berth operations.

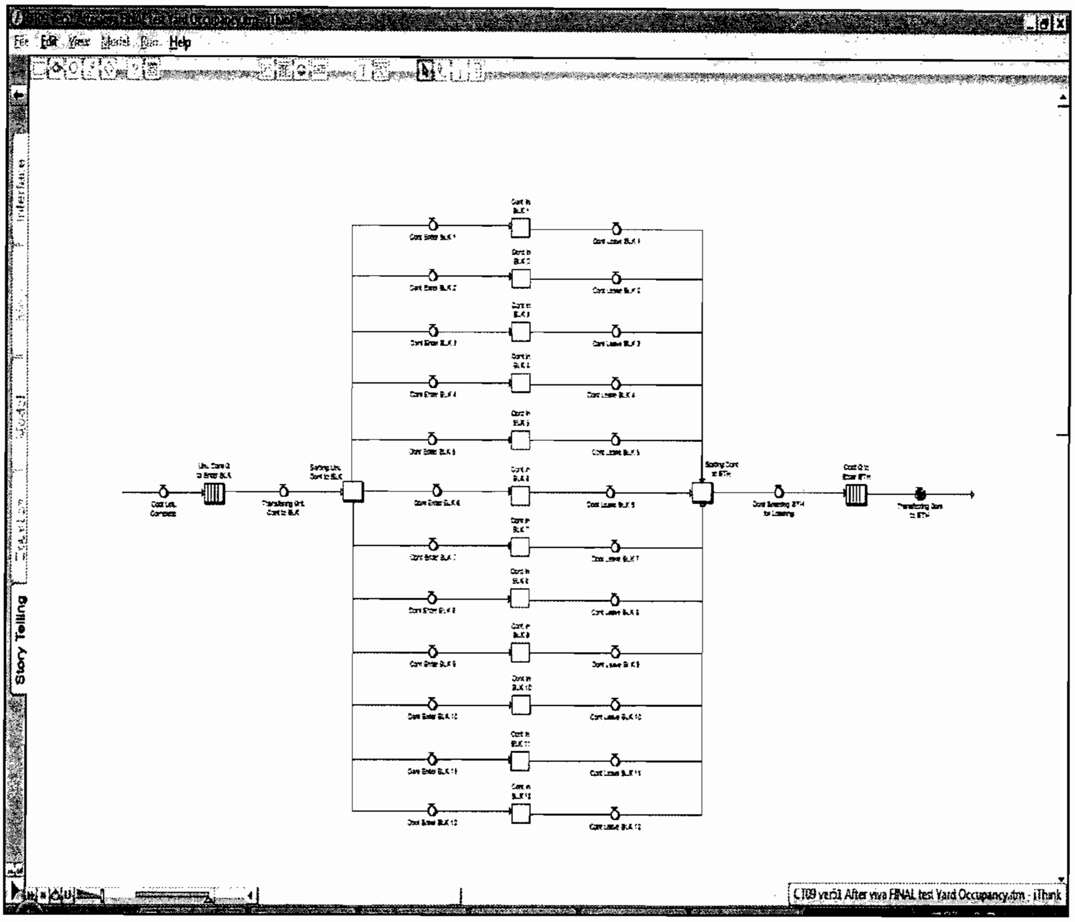


Figure 6.4: The detail processes of the yard operations.

6.3 Microworlds for Operational Level Model

The developed Microworlds for the operational level model allows users to have a glimpse at the current and daily performance of the case study container terminal. From the Microworlds, users are able to experiment the impact of numerous conditions such as the increase in the percentage of vessels arrival, number of containers serviced and the impact of additional facilities on the performance of the container terminal.

A click on the *Operational Planning* button navigates users to the *Operational Planning* page. Figure 6.5 presents the *Operational Planning* page of the developed Microworlds. The right side of the *Operational Planning* page demonstrates the performance of the container terminal through the illustration of graphs and key performance outputs. Two types of graphs are exhibited here where the first type of graph represents the berth occupancy rate and berth capacity rate of the berth operations while the second graph represents the yard occupancy rate of the yard operations.

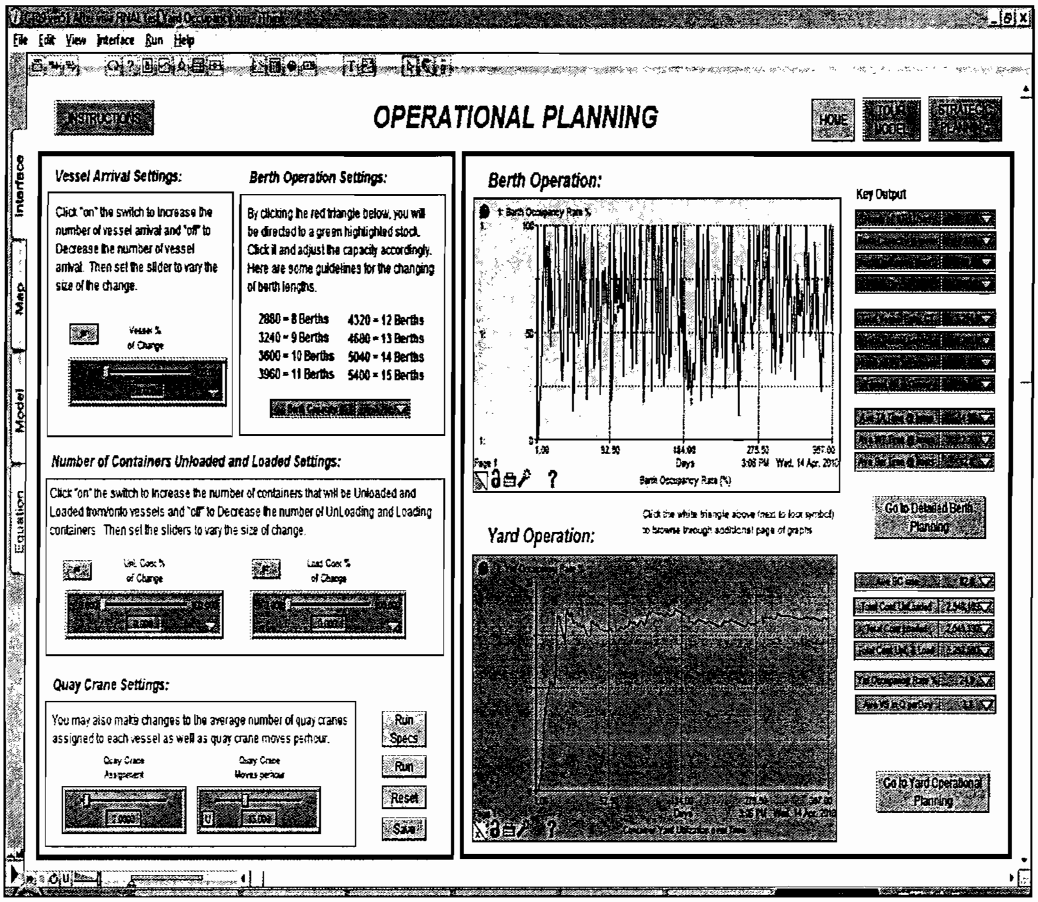


Figure 6.5: The *Operational Planning* page of the developed Microworlds.

The performance outputs that are presented at the *Operational Planning* page include the number of berths used, the total length of operating berths, the berth occupancy rate, the total vessels entering and being served at the container terminal, the number of vessels that are still in queue as well as the number of vessels that are still being processed. Besides that, outputs representing the average vessel turnaround time, average vessel waiting time, average vessel service time, the total containers unloaded and loaded at the container terminal and lastly the yard occupancy rate are also exhibited.

Users may gain insight on the performance of the case study container terminal through the value portrayed by each of the performance outputs. By default, the graphs and performance outputs demonstrated at the *Operational Planning* page represent the initial or current performance of the container terminal before any experiments or changes are made by the users.

The left side of the *Operational Planning* page allows users to experiment different model settings and see the impact of these changes on the case study container terminal performances. Model settings that are embedded in the developed Microworlds include the *vessel arrival* settings which allow users to either increase or decrease the percentage of vessels arrival, *number of containers unloaded and loaded* settings where users may either increase or decrease the percentage of containers that are being unloaded and loaded from/onto vessels.

Besides that, users can also experiment the impact of the increase in terminal capacity and facilities through the *berth operation* settings and *quay crane* settings.

The *berth operation* settings provide users with the flexibility to increase the number of berths to the maximum of 15 berths while the *quay crane* settings allow users to increase or decrease the number of quay cranes assigned to vessels as well as the flexibility to increase or decrease the quay crane moves per hour.

Any changes made on the discussed model settings are reflected on the graphs and performance outputs on the right side of the *Operation Planning* page. From this page, users may also view the detailed berth planning by clicking the *Go to Detailed Berth Planning* button and the analysis of the prime movers travelling distance by clicking the *Go to Yard Operation Planning* button.

Figure 6.6 exhibits the *Detailed Berth Operational Planning* page. The purpose of this page is to provide users with a detailed view on the terminal's berth occupancy rate and the berth capacity rate. There are two graphs representing the berth occupancy rate. The first graph demonstrates the berth occupancy rate after changes are made at the model settings while the second graph illustrates the initial berth occupancy rate before the model settings are manipulated. Users therefore may compare the impact of their experiments on the berth occupancy rate against the initial berth occupancy rate.

Similarly, two graphs are illustrated for the berth capacity rate where the first graph presents the berth capacity rate after the model settings are manipulated while the second graph demonstrates the initial berth capacity rate prior to any changes on the model settings.

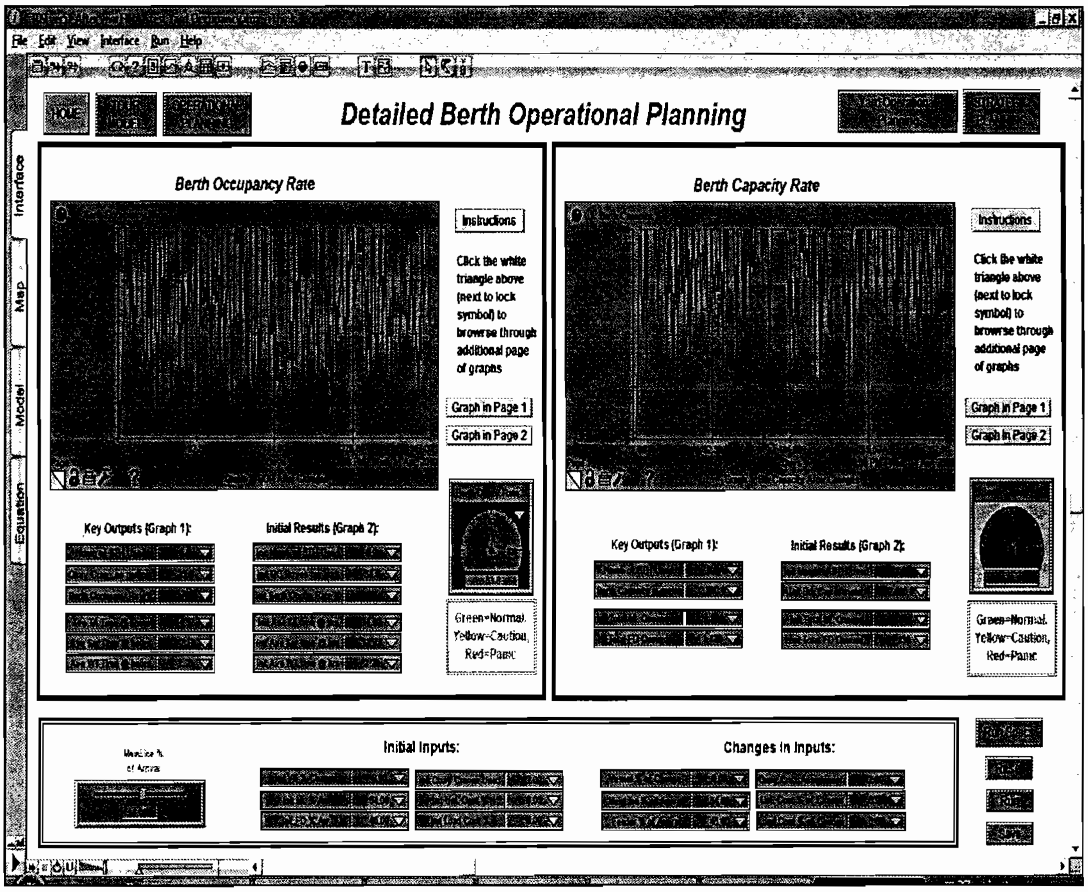


Figure 6.6: The *Detailed Berth Operational Planning* page of the developed Microworlds.

The performance outputs demonstrated on the *Detailed Berth Operational Planning* page also include both the performance outputs before and after changes for comparison. Status indicators for both berth occupancy rate and berth capacity rate are also included in this page.

There are three stages represented by the status indicator where the first is the normal stage, followed by the caution stage and the third is the panic stage. Each of these stages is represented by different colors, with color green representing the normal stage, yellow representing the caution stage and panic stage being represented by red. As the changes from each stage are represented by colors, users will be more alert on the berth occupancy rate and the berth capacity rate. By default, the settings of the status indicator for the berth occupancy rate and berth capacity rate are as shown in Table 6.1. Users however can modify the default percentage values. The changes from each stage are dependent on the current rate of the berth occupancy and berth capacity.

Table 6.1: The status indicator settings for berth occupancy rate and berth capacity rate in Microworlds for Operation Level Model.

Status Indicator	Berth Occupancy Rate	Berth Capacity Rate
Normal (Green)	0% to 30%	0% to 60%
Caution (Yellow)	30% to 40%	60% to 80%
Panic (Red)	40% to 100%	80% to 100%

Moving on, the discussion on the *Yard Operational Planning* page is presented. By clicking the Go *to Yard Operational Planning* button from the *Operational Planning* page, users are transported to the *Yard Operational Planning* page. The *Yard Operational Planning* page allows users to gain insight to the prime mover travelling distance by demonstrating the percentages of containers exiting from each of the 12 yard blocks to all eight berths.

The travelling distance of the prime movers are reflected through the analysis of the percentages of containers exiting from each of the 12 yard blocks to all eight berths. As there are 12 yard blocks to analyze, the *Yard Operational Planning* page is divided to three pages with each page discussing four yard blocks. The first page of the *Yard Operational Planning* page focuses on blocks 1, 2, 3 and 4 (Figure 6.7) while the second page is for blocks 5, 6, 7 and 8 (Figure 6.8) and the third page covers blocks 9, 10, 11 and 12 (Figure 6.9).

Each of the *Yard Operational Planning* pages displays the percentage of containers exiting the yard blocks to all berths and these values are also illustrated in the form of bar charts. The favorable loading points at berths for each of the 12 yard blocks are also displayed. Therefore, users may compare the percentage of containers exiting the 12 yard blocks to all eight berths generated from the developed system dynamics model with the favorable loading point at berths for each of the 12 yard blocks.

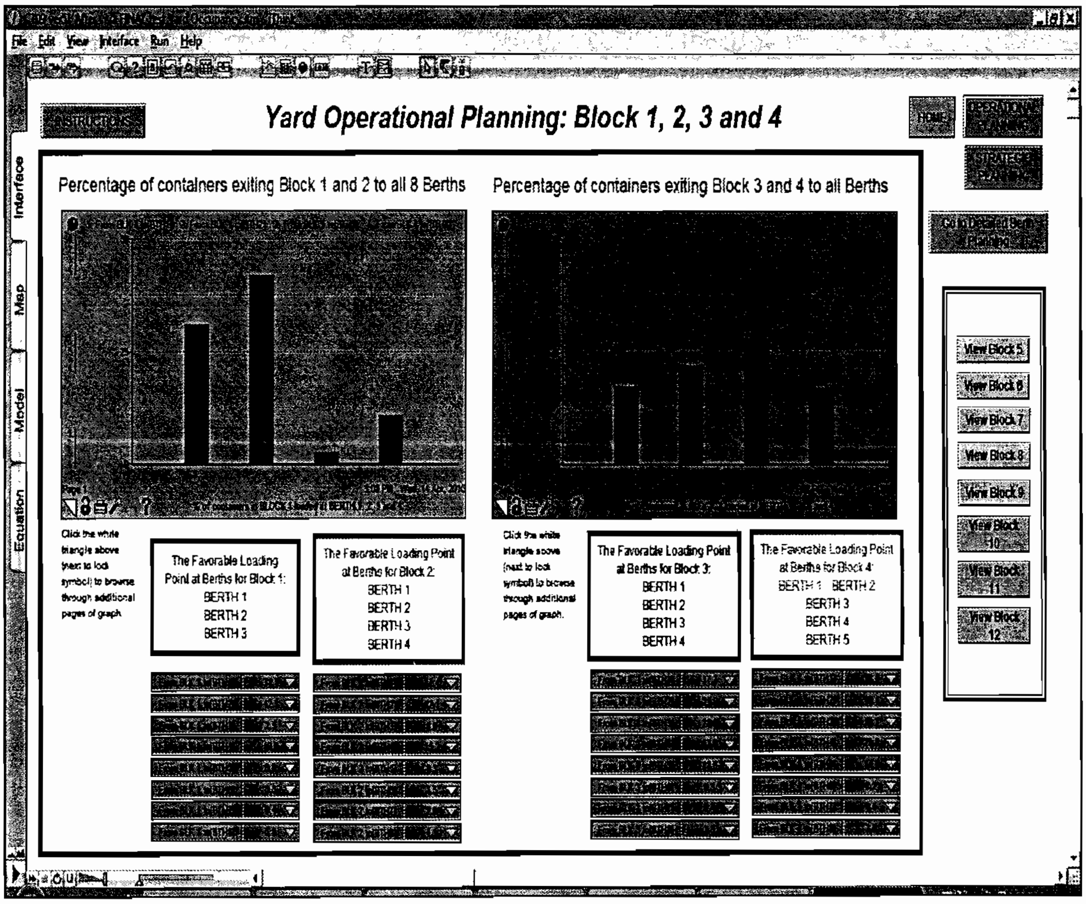


Figure 6.7: The *Yard Operational Planning* page for Block 1, 2, 3 and 4 of the developed Microworlds.

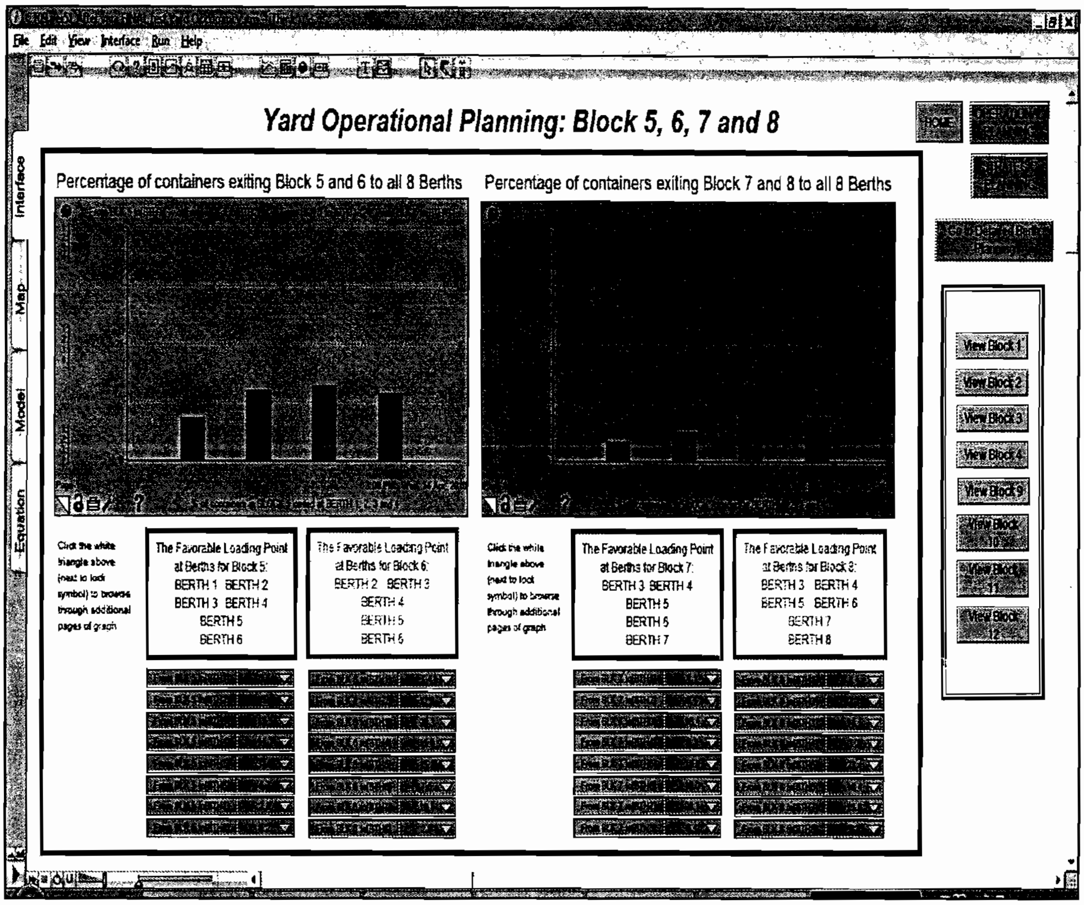


Figure 6.8: The *Yard Operational Planning* page for Block 5, 6, 7 and 8 of the developed Microworlds.

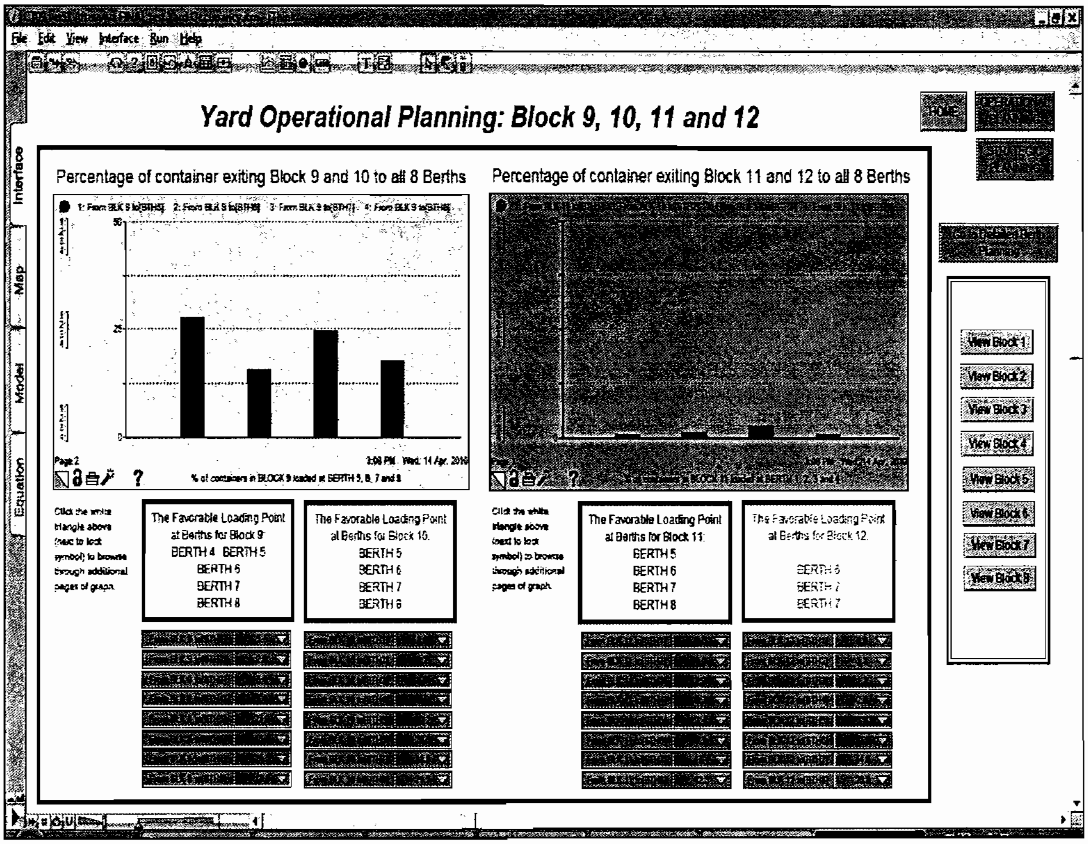


Figure 6.9: The *Yard Operational Planning* page for Block 9, 10, 11 and 12 of the developed Microworlds.

Through these comparisons, users can gain insight to the percentages of containers that are loaded at further berths. By understanding the flow of containers from the yard block to berths, the prime mover travelling distance can also be analyzed. The *Yard Operational Planning* pages provide users with an overview of the current container movement or rather the prime movers travelling distance between yard and block.

6.4 Microworlds for Strategic Level Model

While the Microworlds for the operation level model provides insight to the daily operations of the case study container terminal, the Microworlds developed for the strategic level model enables users to experiment on the future trend of the vessel arrival and container throughput. Users may observe the effect of these future trends on the berth occupancy rate and berth capacity rate.

A click at the *Strategic Planning* button navigates users to the *Strategic Planning* page. The *Strategic Planning* page (Figure 6.10) presents the analysis of the berth capacity and occupancy rate for the year 2002 to year 2012 parallel with the trend demonstrated by both total number of vessel arrivals and total container throughput for the same time period. The total vessel arrivals and container throughput for the year 2002 to 2012 are illustrated in the form of graph and tables. Similarly, the impacts of these two elements on the berth occupancy and capacity rate are also demonstrated in both graphs and tables.

A pair of status indicators representing both berth capacity rate and berth occupancy rate are also provided at the *Strategic Planning* page. The status indicators function to alert users on the current berth capacity rate and berth occupancy rate through changes of colors. The status indicators also allow users to manipulate the minimum and maximum value of each of the three stages. Table 6.2 presents the initial value for all three stages that represent both berth capacity rate and berth occupancy rate.

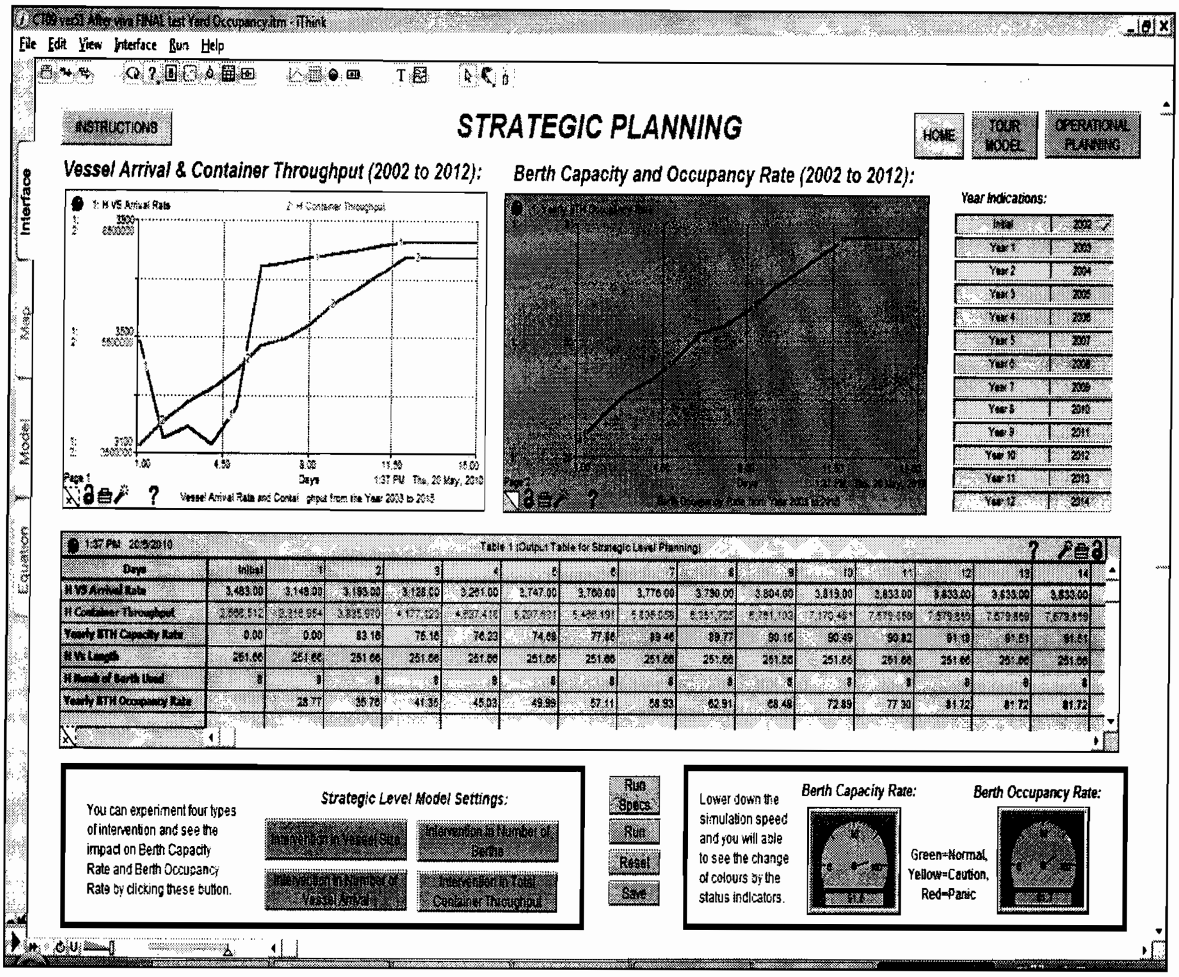


Figure 6.10: The *Strategic Level* page for the developed Microworlds.

Table 6.2: The status indicator settings for berth occupancy rate and berth capacity rate in Microworlds for Strategic Level Model.

Status Indicator	Berth Occupancy Rate	Berth Capacity Rate
Normal (Green)	0% to 30%	0% to 60%
Caution (Yellow)	30% to 40%	60% to 80%
Panic (Red)	40% to 100%	80% to 100%

Apart from that, the strategic planning Microworlds also provides flexibility where users can experiment four different interventions and observe the impact on the berth occupancy rate and berth capacity rate. The four types of intervention are intervention in vessel size, the number of vessels arrival, the total container throughput and lastly, the number of berths.

Four different buttons representing these interventions are embedded in the *Strategic Planning* page. These buttons are the *Intervention in Vessel Size* button, *Intervention in Number of Vessel Arrival* button, *Intervention in Number of Berths* button and the *Intervention in Total Container Throughout* button. By clicking any of these four buttons, users are navigated to the *Strategic Level Model Settings* page (Figure 6.11).

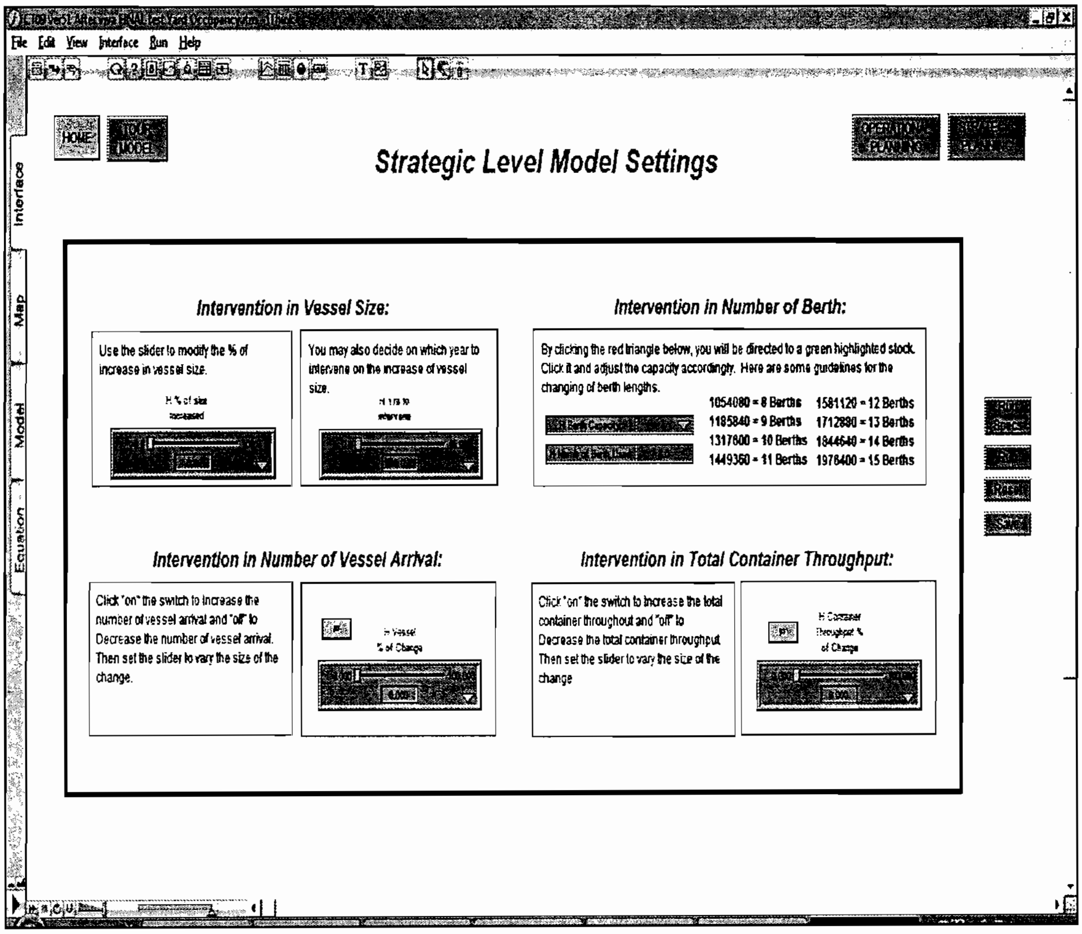


Figure 6.11: The *Strategic Level Model Settings* page of the developed Microworlds.

The *Strategic Level Model Setting* page allows users to perform numerous combinations of intervention and observe the impact of these interventions on the berth capacity rate and berth occupancy rate at the *Strategic Planning* page. The first type of intervention is the intervention in vessel size. Users can modify the average size of vessels arriving at the case study container terminal by increasing the percentage of vessel size. Furthermore, users can also decide on which year to intervene on the increase of vessel size.

Besides the intervention in vessel size, the Microworlds also allow users to intervene with the number of vessels arrival at the case study container terminal. Users may increase or decrease the percentage of vessels arrival and see the impact on the container terminal. Next, users can also increase the number of berths operating at the case study container terminal. By default, there are eight berths operating at the terminal. However, the Microworlds provide users with the flexibility to increase the number of berths to the maximum of 15 berths. The last type of intervention provided by the Microworlds is the intervention in the total container throughput. Users may increase or decrease the percentage of total container throughput handled by the case study container terminal.

Users can also execute numerous interventions simultaneously. The effect of these manipulations of the model settings on the berth capacity rate and berth occupancy rate can be observed at the *Strategic Planning* page. A click at the *Strategic Planning* button on the *Strategic Level Model Settings* page will transport users to the *Strategic Planning* page. Therefore, this completes the discussion on the Microworlds developed for strategic level model.

6.5 Conclusion

The fourth objective of this research was achieved through the development of Microworlds or management flight simulator. Microworlds developed in this research integrates both the operational and strategic level planning. The Microworlds developed for the operational level model aid users on daily planning and decision making while the strategic level model provides users with the ability to understand the future performance of the case study container terminal.

At the *Operational Planning* page, the performance of the berth operations and yard operations as well as the container terminal performance outputs are demonstrated. Users can experiment with numerous model settings to see the impact of these interventions on the berth and yard operations. The detailed berth planning and the analysis of prime mover travelling distance are also displayed on the *Detailed Berth Planning* and *Yard Operation Planning* pages.

On the other hand, the *Strategic Planning* page provides users with the overview of the total number of vessel arrival and total container throughput from the year 2002 to year 2012. The influence of the trend of vessel arrival and container throughput on berth capacity and berth occupancy can also be observed. The *Strategic Planning* page also allows four types of intervention where users can experiment these interventions separately or a combination of interventions and observe their impact on the annual container terminal performance.

As the developed Microworlds supports both the operational level model and strategic level model, it serves as a 'one stop' information centre where users can gain insight to both daily and future performance of the container terminal. The developed Microworlds is therefore beneficial to the case study container terminal managers as it not only aids them on planning and decision making, but also serves as a learning tool where users can gain insight to the complexity of the terminal operations by understanding the interdependency and relationship of different subsystems at the case study container terminal.

The developed Microworlds also provides users with a safe virtual environment where they can test numerous interventions and modify different model settings to see the impact of their experiments on the terminal performance before implementing it to the real container terminal operations.

CHAPTER 7

CONCLUSIONS AND RECOMMENDATIONS

7.1 Introduction

This chapter concludes this research report by presenting a summary of this work and how all four objectives of this research have been achieved. Besides that, this chapter also puts forward the contribution of this research to the body of knowledge and also its benefits to the container terminal operations. The suggestions on how the developed system dynamics model can be generalized to other container terminal operations for future research are also highlighted in this chapter.

7.2 Research Summary

This research was directed towards the development of a system dynamics simulation model for a container terminal. With the introduction of larger vessels, a container terminal not only faces the pressure of servicing these large vessels within the shortest time period but also of providing adequate facilities to service them. Without adequate facilities and efficient services, the container terminal faces the risk of extinction as these vessels will turn to another container terminal that is capable of providing better facilities and services.

System dynamics was applied to understand, evaluate and model the operation of a container terminal. Two models were developed in this study; the first captured the operational level processes while the second was modeled after the strategic level processes of the case study container terminal. The operational level model was used to evaluate the relationship and interdependency between the berth and yard operations and to understand how these interactions affect the terminal efficiency. On the other hand, the strategic level model was used to aid container terminal managers in capacity planning.

The flow of containers from the berth subsystem to the yard subsystem and vice versa was captured using *iThink* software, a system dynamics simulation tool. From the system dynamics model, four categories of performance indicators, namely the vessel performance indicators, berth performance indicators, container handling performance indicators and container yard performance indicators were generated.

The berth occupancy rate of 69.8% generated from the system dynamics model was found to be higher than the maximum international standard of 40%. A high value of berth occupancy rate indicated that the terminal is congested or the facilities at the terminal are not being utilized efficiently. Therefore, if no appropriate measures are taken to alleviate the congestion problem, the case study container terminal will face the risk of losing its potential customers or worse, fail to retain its existing customers as vessels need to spend longer time waiting to be berthed.

In order to lower the berth occupancy rate, there was a need to determine which factor played a major role in affecting the berth occupancy rate. The analysis on the relationship between berth occupancy rate and other processes and performance indicators showed that both quay crane moves and prime mover travelling distances had an impact on the berth occupancy rate.

The quay cranes at the case study container terminal performed an average of 33 container moves per hour. The developed system dynamics model was used to experiment the impact of increasing the quay crane moves gradually from 33 moves to 40 moves per hour where 40 moves per hour are the maximum moves performed by current world leading container terminals. With the increase in quay crane moves to the maximum of 40 moves, the berth occupancy rate dropped to 59.8%, a value still higher than the maximum international standard.

The experiment revealed that the case study container terminal needed to expand its current berth capacity in order to alleviate the high berth occupancy problem as well as to prepare the terminal for servicing a larger container volume in future. The strategic level model was then used to experiment on how much capacity to introduce in order to lower the berth occupancy rate and yet avoid the risk of over capacity.

In addition, the analysis on the prime mover travelling distance revealed that there were containers stored in each of the 12 yard blocks being loaded at the unfavorable loading point or 'distant' berth. This increased prime mover traveling distance and traveling time as the prime movers needed to transfer containers from the yard block to those 'distant' berths. With the increase in the prime mover traveling time, the vessel service time and vessel turnaround time became longer, thus increasing the berth occupancy rate as well. Therefore, to minimize the prime mover traveling distance between the yard blocks and berth, it is best to store containers at the yard blocks located nearer to its loading point at the berth.

The strategic level model also generated the forecasted value of container throughput and vessel arrival until year 2012. This model provides the flexibility to experiment the impact of the increase in container throughput, vessel arrival and vessel size on the operations of the case study container terminal. The number of berths operating and quay crane moves can also be manipulated to see their impact on the case study container terminal operations. Therefore, the developed strategic level model provides a guideline on when to introduce new capacity to avoid the problem of inadequate capacity as well as the problem of over capacity.

Finally, Microworlds or management flight simulator was developed in this research. The Microworlds developed in this research integrates both operational level and strategic level planning. The operational level model settings allow users to experiment the impact of the increase or decrease in vessel arrival, number of containers processed, number of berths operating, number of quay cranes used and quay crane moves per hour on the daily terminal operations. On the other hand, the Microworlds for strategic level model aids users on the understanding of the future performance of the container terminal. Here, users can experiment with four types of intervention namely, intervention in vessel size, number of vessel arrival, total container throughput and number of berth operating. The Microworlds developed in this research allows users to experiment the consequences of their actions and decision making before implementation into the actual container terminal operations.

7.3 Contribution of Study

There have been many studies conducted at container terminals, but many of these studies either only focused on the berth side (e.g. Razman and Khalid, 2000; Guan and Cheung, 2004; Dragovic et al., 2005 and Moorthy and Teo, 2006) or only on the yard side (e.g. Kim and Kim, 2002; Kim and Park, 2003; Zhang et al., 2003 and Chen et al., 2004).

The complexity and dynamic nature of the container terminal operations resulted in difficulties in using analytical tools as the method of investigation. As such, discrete event simulation is widely used to model the whole container terminal. Examples of studies that integrated both berth and yard operations using discrete event simulation include Ramani (1996), Yun and Choi (1999), Shabayek and Yeung (2002) and Duinkerken et al. (2006).

Discrete event simulation however is not capable of capturing the relationship and interdependency between the berth and yard operations to investigate how these subsystems affect each other as well as the overall container terminal performance. System dynamics, on the other hand, is capable of capturing the relationship and feedbacks between variables as well as providing platform for testing the impact of policy and strategy decision making.

Nevertheless, the application of system dynamics simulation modeling is uncommon in container terminal study as opposed to other similarly complex industries such as health care and manufacturing. Up to date, there is only one related work by Choi et al., 2007 that reported on the application of system dynamics in solving issues of the container terminal. Choi et al. (2007) focused on solving issues at the operational level and this study also did not incorporate the interdependency and relationship between various subsystems at the container terminal.

This research contributes by bridging the gap between the literatures in two ways. The first one is by developing a system dynamics model that is capable of capturing the relationship and interdependency between the berth and yard operations as well as how the operations of these two subsystems impact the overall container terminal. The second contribution is that the model of this research focused on solving issues at both the operational and strategic levels.

The operational level model or the micro level model provides the understanding on the current terminal operations through the analysis of the relationship between the berth and yard operations while the strategic level model or the macro level model provides insight into the dynamic behavior of the container terminal in the long run.

Besides the contribution of bridging the gap between literatures, this research also benefits the container terminal through the development of Microworlds or management flight simulator. The developed Microworlds is beneficial to the container terminal managers as it is not only capable of aiding them on planning and decision making but also serves as a learning tool where the container terminal managers can gain insight into the complexity of the terminal operations by understanding the interdependency and relationship among different subsystems at the container terminal.

The developed Microworlds also provides the container terminal managers with a safe virtual environment where they can test numerous interventions or modify different model settings to see the impact of their experiments on the terminal performance before implementing it to the real container terminal operations.

7.4 Future Research and Recommendations

The model developed in this research has demonstrated the usefulness of system dynamics. Further investigation and model development can be conducted to make the system dynamics model more realistic.

In this research, each vessel is assigned with two quay cranes as majority of the vessels at the case study container terminal were assigned with two quay cranes each. However, in real life operations, vessels can be assigned with different number of quay cranes according to the size of vessels or the number of containers to be transferred from the vessel to the container terminal. Therefore, future research can take into consideration the assigning of different number of quay cranes to the incoming vessels to capture more realistic and accurate berth operations.

Apart from that, the speed of quay crane moves in this research is represented by an average value. However, in reality, the speed of the quay crane moves is actually dependent on the skill of the quay crane drivers. If the quay cranes are handled by experienced drivers, the quay cranes are most likely to perform more moves as opposed to the quay cranes handled by inexperienced drivers. Therefore, further research can incorporate this qualitative aspect when modeling the container terminal operations using system dynamics. By capturing both the qualitative and quantitative aspects, a more realistic container terminal model can be developed.

The system dynamics model developed for the container terminal in this research can also serve as a framework to generalize the system dynamics approach to further container terminal studies. The scope of this research is to model the internal operations of the container terminal in order to understand the interdependency and relationship between the berth and yard subsystems. External factors such as government intervention and regulation as well as the economic impact do have their share of influence on the container terminal operations as well. Therefore, further research can extend this system dynamics model to include the interaction and relationship of these external factors on the terminal operations and decision making.

7.5 Conclusion

The container terminal faces many challenges in order to remain competitive in the shipping industry. One of the challenges faced by modern container terminal includes the growing size of incoming vessels. Not only does the terminal need to provide adequate facilities to service these mega vessels but also needs to provide efficient and timely services as well. Failing to do so is a loss to the container terminal. Besides that, the internal operations of a container terminal itself is highly complex as the terminal consists of many subsystems such as the berth and yard. The interaction and interdependency between the berth and yard is important to ensure the whole container terminal operates in an efficient manner.

Therefore, a system dynamics model was developed in this research to provide container terminal managers with a clearer understanding on how the berth and yard operations influence each other and the container terminal as a whole. Besides that, the system dynamics model is capable of guiding the container terminal managers through the process of capacity planning. The model also serves as an experimental tool where container terminal managers can test different interventions and observe the impact on the terminal operations.

REFERENCES

Ackere, A. V., & Smith, P. C. (1999). Towards a macro model of national health service waiting lists. *System Dynamics Review, 15*(3), 225-252.

Adaikappan, N. G. (2005). Modeling neonatal patient flow using system dynamics. Retrieved from Dissertation and Theses database. (AAT MR07401)

Ambrosino, D., & Sciomachen, A. (2003). Impact of yard organisation on the master bay planning problem. *Maritime Economics & Logistics,* 5, 285-300.

Balci, O. & Sargent, R. G. (1981). A methodology for cost-risk analysis in the statistical validation of simulation models. *Special Issues on Simulation Modeling and Statistical Computing, 24* (4), 190-197.

Balci, O. (1998). Verification, validation and testing. Handbook of simulation: Principles, methodology, advances, application and practice. USA: John Wiley & Sons.

Barlas, Y., & Diker, V. G. (1996). An interactive dynamic simulation model of a university management system.

Barlas, Y., Cirak, K., & Duman, E. (2000). Dynamic simulation for strategic insurance management. *System Dynamics Review, 16*(1), 43-58.

Beresford, A. K. C., Gardner, B. M., Pettit, S. J., Naniopuolos, A., & Wooldridge, C. F. (2004). The UNCTAD and WORKPORT models of port development: Evolution or revolution? *Maritime Policy Management.* 31, (2), 93-107.

Brailsford, S. C., Lattimer, V. A., Tarnaras, P., & Turnbull, J. C. (2004). Emergency and on-demand health care: Modelling a large complex system. *Journal of the Operational Research Society, 55*, 34-42.

Carpenter, B. C., & Ward, T. (1990). *The use of computer simulation for marine terminal planing.* Paper presented at the 1994 Winter Simulation Conference.

Casaca, A. C. P. (2005). Simulation and the lean port environment. Maritime Economics & Logistics. *Maritime Economics & Logistics* (7), 262-280.

Chan, C. (2005). *Allocating resources by modelling cardiac patient flow using a system dynamics approach.* Retrieved from Dissertations and Theses database. (AAT MR07403)

Chen, P., Fu, Z., Lim, A., & Rodrigues, B. (2004). Port yard storage optimization. *IEEE Transactions on Automation Science and Engineering 1*(1), 26-36.

Chen, T. (1999). Yard operations in the container terminal: A study in the 'unproductive moves'. *Maritime Policy Management 26*(1), 27-38.

Choi, H. R., Park, B. J., Yoo, D. H., & Kang, M. H. (2007). Development of a model based on system dynamics to strengthen the competitiveness of a container terminal. *WSEAE Transactions on Information Science and Application. 4*(5), 988-996.

Chou, C. C., Liang, G. S. & Chu, C. W. (2003). Competitiveness analysis of major port in Eastern Asia. *Journal of the Eastern Asia Society for Transportation Studies*. 5.

Dai, J., Lin, W., Moorthy, R., & Teo, C. P. (2004). Berth allocation planning optimization in container terminal [Electronic Version]. Retrieved 17 August 2006 from http://www.bschool.nus.edu.sg/staff/bizteocp/berthplanningjuly2004.pdf.

Dangerfield, B. C. (1999). System dynamics applications to European health care issues. *Journal of the Operational Research Society, 50*, 345-353.

De Monie, G. (1987). Measuring and evaluating port performance and productivity. *UNCTAD Monographs on Port Management*. Monograph No.6.

Deaton, M., & Winebrake, J. J. (2000). *Dynamic modeling of environmental system* Springer.

Donovan, A. (2004). The impact of containerization: From Adam Smith to the 21th century. Review of Business. *25*(3), 10-15.

Dragovic, B., Park, N. K., Radmilovic, K., & Maras, V. (2005). Simulation modeling of ship-berth link with priority service. *Maritime Economics & Logistics*(7), 316-335.

Duinkerken, M. B., Dekker, R., Kurstjens, S.T.G.L., Otjess, J. A. & Dellaert, N. P. (2006). Comparing transportation systems for inter-terminal transport at the Maasvlakte container terminals. *OR Spectrum. 28*, 469-493.

Ma, N. L. & Hadjiconstantinou, E. (2008). Evaluation of operational plans in container terminal yards using discrete-event simulation. *OR Insight. 21*(4).

Economist, T. (2006). Books and arts: The world in a box; the container industry [Electronic Version], 378. Retrieved 3rd March 2007 from http://proquest.umi.com/pqdweb?did=1010601531&sid=4&Fmt=3&clientId =28929&RQT=309&VName=PQD.

Fung, K. F. (1998). Essays on Hong Kong's container handling industry. Retrieved from Dissertation and Theses database. (ATT 9914335)

Gambardella, L. M., Rizzoli, A. E., & Zaffalon, M. (1998). Simulation and planning of a container terminal. *Habour and Maritime Simulation*

Georgantzas, N. C. (2003). Tourism dynamics: Cyprus' hotel value chain and profitability. *System Dynamics Review, 19*(3), 175.

Gonzalez-Busto, B., & Garcia, R. (1999). Waiting lists in Spanish public hospitals: A system dynamics approach. *System Dynamics Review, 15*(3), 201-224.

Goodchild, A. V., & Daganzo, C. F. (2004). *Reducing ship turn-around time using double-cycling.* Berkeley: Institute of Transportation Studies, University of California.

Goodchild, A. V., & Daganzo, C. F. (2005). *Crane double cycling in container ports: Affect on ship dwell time.* Berkeley: University of California.

Guan, Y., & Cheung, R. K. (2004). The berth allocation problem: Models and solution methods. *OR Spectrum, 26*(1), 75-92.

Guan, Y., Xiao, W. Q., Cheung, R. K., & Li, C. L. (2002). A multiprocessor task scheduling model for berth allocation: Heuristic and worst-case analysis. *Operational Research Letters, 30*, 343-350.

Hanke, J. E. Wichern, D. W. & Reitsch, A. G. (2001*). Business Forecasting (7th Edition).* New Jersey: Prentice hall.

Haines, S. G. (2000). *The system thinking approach to strategic planning and mangement* America: St. Lucie Press.

Hartmann, S. (2004). Generating scenerios for simulation and optimization of container terminal logistics. *OR Spectrum, 26*, 171-192.

Hirsch, G., & Immedianto, S. C. (1999). Microworlds and generic structures as resources for integrating care and improving health. *System Dynamics Review, 15*(3), 315-330.

Homer, J. B., & Hirsch, G. (2006). System dynamics modeling for public health: Background and opportunities. *American Journal of Public Health, 96*(3), 452.

Imai, A., Nishimura, E., & Papadimitriou, S. (2001). The dynamic berth allocation problem for a container port. *Transportation Research Part B, 35*, 401-417.

Imai, A., Nishimura, E., & Papadimitriou, S. (2003). Berth allocation with service priority. *Transportation Research Part B, 37*, 437-457.

Imai, A., Sasaki, K., Nishimura, E., & Papadimitriou, S. (2004). Multi-objective simultaneous stowage and load planning for a container ship with container rehandle in yard stacks. *European Journal of Operational Research.*

Intrapairot, A. (2000). *A study on the adoption and diffusion of information and communication technologies in the banking industry in Thailand using multiple-criteria decision making and system dynamics approach.* (Doctoral dissertation, Curtain University of Technology, 2000). Retrieved 12th July 2007 from http://adt.curtin.edu.au/theses/available/adt-WCU20020606.153708/

Johns, N., & Lee-Ross, D. (1998). *Research methods in service industry management*. London: Cassell.

Jutla, A. S. (2006). *Hydrologic modelling of reconstructed watersheds using a system dynamics approach.* (Master dissertation, University of Saskatchewan, 2006). Retrieved 28th October 2009 from http://library2.usask.ca/theses/available/etd-01122006-160841/

Kelton, W. D. (2002). *Simulation with arena* (2nd Edition). New York: McGraw-Hill.

Kendall, L. C. (1986). *The business of shipping*. London: Chapman and Hall Ltd.

Kia, M., Shayan, E., & Ghotb, F. (2002). Investigation of port capacity under a new approach by computer simulation. *Computers & Industrial Engineering, 42*, 533-540.

Kim, K. H., & Kim, H. B. (2002). The optimal sizing of the storage space and handling facilities for import containers. *Transportation Research Part B, 36*, 821-835.

Kim, K. H., & Moon, K. C. (2003). Berth scheduling by simulated annealing. *Transportation Research Part B, 37*, 541-560.

Kim, K. H., & Park, K. T. (2003). A note on a dynamic space-allocation method for outbound containers. *European Journal of Operational Research, 148*, 92-101.

Koelling, P., & Schwandt, M. (2005). *Health system: A dynamic system-Benefits from system dynamics.* Paper presented at the 2005 Winter Simulation Conference.

Kumar, S., & Yamaoka, T. (2007). System dynamics study of the Japanese automotive industry closed loop supply chain. *Journal of Manufacturing Technology Management, 18*(2), 115-138.

Lam, J. S. L., & Yap, W. Y. (2007). *Competition for transshipment containers by major ports in Southeast Asia: Slot capacity analysis*. Paper presented at the 2007 International Conference on Logistics, Shipping and Port Management, Taoyuan, Taiwan.

Lane, D. C., Monefeldt, C., & Rosenhead, J. V. (2000). Looking in the wrong place for healthcare improvements: A system dynamics study of an accident and emergency department. *Journal of the Operational Research Society, 51*, 518-531.

Law, A. M. (2009). How to build valid and credible simulation models. *IEEE Engineering Management Review, 37* (2).

Lee, L. H., Chew, E. P., Tan, K. C., & Han, Y. (2006). An optimization model for storage yard management in transshipment hubs. *OR Spectrum. 28*, 539-561.

Legato, P., & Mazza, R. N. (2001). Berth planning and resources optimisation at a container terminal via discrete event simulation. *European Journal of Operational Research, 133*, 537-547.

Levin, R. I. & Rubin, D. S. (1998). *Statistics for management* (7th Ed.). New Jersey: Prentice Hall.

Liehr, M., Grosler, A., Klein, M., & Milling, P. M. (2001). Cycles in the sky: Understanding and managing business cycles in the airline market. *System Dynamics Review, 17*(4), 311-332.

Liu, C.I., Jula, H. & Ioannou, P. A. (2002). Design, simulation, and evaluation of automated container terminals. *IEEE Transactions on Intelligent Transportation Systems*, 3(1), 12-26.

Liu, F. (2001). *A system dynamics model for hydropower generation planning.* (Master dissertation, University of Manitoba, 2001). Retrieved 28th October 2009 from http://mspace.lib.umanitoba.ca/dspace/handle/1993/2643

Maani, K. E., & Cavana, R. Y. (2000). *System thinking and modelling: Understanding change and complexity.* New Zealand: Pearson Education New Zealand Limited.

Martin, J., & Thomas, B. J. (2001). The container terminal community. *Maritime Policy Management.* 28, (3), 279-292.

Mayo, D. A., Callaghan, M. J., & Dalton, W. J. (2001). Aiming for restructuring success at London Underground. *System Dynamics Review, 17*(3), 261-289.

Meersmans, P. J. M., & Dekker, R. (2001). *Operations research supports container handling.* Rotterdam: Econometric Institute, Erasmus University

Moglia, F., & Sanguineri, M. (2003). Port planning: The need for a new approach? *Maritime Economics & Logistics, 5*, 413-425.

Moorthy, R., & Teo, C. P. (2006). Berth management in container terminal: The template design problem. *OR Spectrum.*

MOT. (2008). *Total container throughput by ports, Malaysia 1999-2008.* Retrieved 16th June 2009 from http://www.mot.gov.my/stat/maritim.htm

MOT. (2008). Type *of ships calling by ports, Malaysia 2003-2008.* Retrieved 16th June 2009 from http://www.mot.gov.my/stat/maritim.htm

Mukherjee, A., & Roy, R. (2006). A system dynamics model of management of a television game show. *System Dynamics Review, 1*(2), 95-115.

Musaphir, H. (1997). *A system dynamics approach to studying manufacturing strategy*. (Master dissertation, University of Manitoba, 1997). Retrieved 28th October 2009 from http://mspace.lib.umanitoba.ca/dspace/handle/1993/851

Nagorski, B. (1972). *Port problems in developing countries: Principles of port planning and organization*. Tokyo: The International Association of Ports and Harbors.

Nazery, K. (2005). The Development of ports and shipping sectors in Malaysia. Retrieved 5th April 2007 from http://www.mima.gov.my/mima/htmls/papers/pdf/nazery/nazery-LPJ%20article.pdf

Nazery, K., Armi Suzana, Z., & Farida, F. (2007). The Asian experience in developing the maritime sector: Some case study and lessons for Malaysia. Retrieved 28th January 2008 from http://www.mima.gov.my/mima/htmls/papers/pdf/nazery/Matime%20Nations%20Paper%20_26Dec07_.pdf

Nishimura, E., Imai, A., & Papadimitriou, S. (2001). Berth allocation planning in the public berth system by generic algorithms. *European Journal of Operational Research, 131*, 282-292.

Notteboom, T., & Rodrigue, J-P. (2005). Port regionalization: Towards a new phase in port development. *Maritime Policy & Management*. 32, (3), 297-313.

Notteboom, T.E. (2006). The time factor in liner shipping services. *Maritime Economics & Logistics. 8*, 19-39.

Ottjes, J. A., Veeke, H. P. M, Duinkerken, M. B., Rijsenbrij, J. C. & Lodewijks, G. (2006). Simulation of a multiterminal system for container handling. *OR Spectrum. 28*, 447-468.

Paixao, A. C., & Marlow, P. B. (2003). Fourth generation ports: A question of agility? *International Journal of Physical.* 33, (4), 355-376.

Panayides, P. M. (2006). Maritime Logistics and global supply chains: Towards a research agenda. *Maritime Economics & Logistics*. (8), 2-18.

Park, Y.M., & Kim K.H. (2003). *A scheduling method for berth and quay cranes. OR Spectrum.* 25, (1), 1-23.

Preston, P., & Kozan, E. (2001). An approach to determine storage locations of containers at seaport terminals. *Computers & Operations Research 28*, 983-995.

Ragheb Mohamed, M. A. S. (2005). *Decision support model for a seaport*. Retrieved from Dissertations and Theses database. (AAT 3166182)

Raine, G. (2006). A sea change in shipping: 50 years ago, container ships altered the world. *San Francisco Chronicle.* Retrieved 4th June 2008 from http://www.sfgate.com/cgi-bin/article.cgi?f=/c/a/2006/02/05/BUGG4H2FIB1.DTL&hw=container+shipping&sn=004&sc=781

Ramani, K. V. (1996). An interactive simulation model for the logistics planning of container operations in seaports. *Simulation 66*(5), 291-300.

Ray, D. & Blankfeld, R. (2002). *Reforming Indonesia's ports.* Technical Report. Robert R. Nathan Associates, Inc and Checchi and Co. Retrieved 28th October 2009 from http://dec.usaid.gov/index.cfm?p=search.getSQLResults&CFID=21302394&CFTOKEN=98804034

Razman, M. T., & Khalid, H. (2000). Simulation and analysis for the Kelang container terminal operations. *Logistics Information Management, 13*(1), 14-20. Retrieved 6th October 2009 from http://www.wto.org/english/res_e/statis_e/statis_e.htm

Richardson, G. P., & Pugh III, A. L. (1981). *Introduction to system dynamics modeling.* Waltham: Pegasus Communications.

Robert, D. S., Song, Z., & Liu, C. (2000). Supply chain dynamics and optimization. *Integrated Manufacturing Systems, 11*(5), 348-364.

Robinson, Ross. (2002). Ports as elements in value-driven chain systems: The new paradigm. *Maritime Policy & Management.* 29, (3), 241-255.

Royston, G., Dost, A., Townshend, J., & Turner, H. (1999). Using system dynamics to help develop and implement policies and programmes in health care in England. *System Dynamics Review, 15*(3), 293-313.

Ruth, M., & Hannon, B. (2004). *Modeling dynamic system.* USA: Springer-Verlag.

Sargent, R. (2009). Verification and validation of simulation models. *IEEE Engineering Management Review, 37* (2).

Schinas, O., & Papadimitriuo, S. (2003). The Mediterranean ports in the era of mega-carriers: A strategic approach. Retrieved 21th October 2008 from http://www.maritime.deslab.naval.ntua.gr/documents/2003/ports/OSpaper_megacarriers.pdf

Seel, K. C. (2004). *Boom and bust cycles in power plant construction: A simulation study of the temporal and geographical aspects of the Alberta competitive electrical industry.* Retrieved from Dissertations and Theses database. (AAT NQ 93467)

Shabayek, A.A. & Yeung, W. W. (2002). A simulation model for the Kwai Chung container terminals in Hong Kong. *European Journal of Operational Research. 140*, 11-11.

Slack, B. (2001). Globalisation in maritime transportation: Competition, uncertainty and implication for port development strategy. FEEM Working Paper No. 8. Concordia University.

Sobol, M. G. & Starr, M. K. (1993). *Introduction to statistics for executive*. United States: McGraw-Hill.

Stacy, R. D., Griffin, D., & Shaw, P. (2000). *Complexity and management: Fad or radical challenge to systems thinking?* London: Routledge.

Steenken, D., Vos, S., & Stahlbock, R. (2004). Container terminal operation and operations research: A classification and literature review. *OR Spectrum, 26*, 3-49.

Stahlbock, R., & Vos, S. (2008). Operations research at container terminals: A literature update. OR Spectrum, *30*, 1-52.

STELLA for educational and research & *iThink* for business. *issesystems*. Retrieved 1st November 2009 from http://www.iseesystems.com/

Sterman, J. D. (2000). *Business dynamics: System thinking and modeling for a complex world*. McGraw-Hill Companies.

Taylor, K., & Dangerfield, B. (2005). Modeling the feedback effects of reconfiguring health services. *Journal of the Operational Research Society, 56*, 659-675.

UNCTAD. (2008). *Review of maritime transport 2008*. United Nations, New York and Geneva. Retrieved 6th October 2009 from http://www.unctad.org/en/docs/rmt2008_en.pdf

UNESCAP. (2007). *Regional shipping and port development: Container traffic forecast 2007 update.* United Nations, New York and Geneva. Retrieved 4th June 2008 from http://www.unescap.org/ttdw/Publications/TIS_pubs/pub_2484/pub_2484_fulltext.pdf

Vacca, I., Bierlaire, M., & Salani, M. (2007). *Optimization at container terminals: Status, trends and perspectives*. Paper presented at the 7th Swiss Transport Research Conference, Monte Verita, Ascona.

Valencia, G. (2006). Computer *modelling of container traffic in shipping terminal*. Retrieved from Dissertations and Theses database. (AAT 1431829)

Vis, I. F. A., & Koster, R. D. (2003). Transshipment of containers at a container terminal: An overview. *European Journal of Operational Research, 147*, 1-16.

Warren, K., & Langley, P. (1999). The effective communcation of system dynamics to improve insight and learning in management education. *Journal of the Operational Research Society, 50*, 396-404.

Warren, K. (2002). *Competitive strategy dynamics*. England: John Willey & Sons.

Warwick, J. (2009). On 40 years of queuing in libraries. *Literature Review. 58*(1).

Wolstenholme, E. (1999). A patient flow perspective of U.K. health service: Exploring the case for new 'intermediate care' initiatives. *System Dynamics Review, 15*(3), 253-271.

WTO. (2008). Chapter 1: World trade development. *International Trade Statistics 2008*. Retrieved 1st November 2009 from http://www.wto.org/english/res_e/statis_e/its2008_e/its08_toc_e.htm

Yap, W. K., Lam, J. S. L, & Notteboom, T. (2006). Developments in container port competition in East Asia. *Transport Review*. 26, (2), 167-168.

Yi, D. W., Kim, S. H., & Kim, N. K. (2002). *Combined modeling with multi-agent systems and simulation: Its application to harbor supply chain management.* Paper presented at the The 35th Hawaii International Conference on System Sciences, Hawaii.

Yun, W. Y., & Choi, Y. S. (1999). A simulation model for container-terminal operation analysis using an object-oriented approach. *International Journal of Production Economics 59*, 221-230.

Zhang, C., Wan, Y. W., Liu, J., & Linn, R. J. (2002). Dynamic crane deployment in container storage yards. *Transportation Research Part B, 36*, 537-555.

Zhang, C., Liu, J., Wan, Y. W., Murty, K. G., & Linn, R. J. (2003). Storage space allocation in container terminals. *Transportation Research Part B, 37*, 883-903.

APPENDIX 1: Aggregated data, calculated from the data provided by the case study container terminal.

Day	Total arrival of mainline and feeder at all berths	Total container unloaded from mainline and feeder at all berths	Total container loaded unto mainline and feeder at all berths
1	6	3680	1265
2	11	8463	1355
3	7	6701	3054
4	7	7671	2758
5	10	4441	1610
6	9	6683	3967
7	8	4093	3079
8	8	5553	2952
9	11	6500	5810
10	4	2577	5462
11	11	7177	7982
12	12	5530	3429
13	8	3173	5412
14	10	4245	3992
15	8	4674	2819
16	9	5656	5394
17	4	3499	2955
18	10	7771	6132
19	13	5615	6840
20	8	3484	3044
21	7	4179	3149
22	8	5305	2231
23	7	7882	4735
24	11	3671	7247
25	11	8718	7724
26	9	5007	4913
27	11	5814	6844
28	6	2682	1877
29	8	6540	3546
30	13	9301	5912
31	6	3241	5224
32	9	6423	2747
33	8	7085	6813
34	12	5671	12291
35	5	2940	2194
36	11	7414	4526
37	9	7239	6273
38	7	4921	7111
39	10	10224	6599
40	8	5654	6067
41	11	4736	7122

42	7	2809	3009
43	7	4038	3057
44	10	7213	5520
45	8	4866	5081
46	10	6516	8114
47	11	4714	6642
48	13	5387	8834
49	3	1590	1218
50	9	6140	4444
51	7	6767	3794
52	11	4563	7981
53	8	4834	7668
54	7	1839	2041
55	12	5113	5295
56	5	2647	3085
57	6	5478	2824
58	8	7989	4189
59	6	5955	4594
60	10	8560	6502
61	6	1921	4081
62	14	9725	11734
63	6	1818	1717
64	7	6150	4215
65	11	6321	5333
66	5	3555	6258
67	9	7853	6458
68	12	5270	7040
69	10	6239	8957
70	9	4401	4035
71	7	6240	4896
72	7	3409	4511
73	7	5571	6784
74	10	5474	3453
75	9	6537	5742
76	8	4080	7547
77	11	7512	4695
78	10	9416	9023
79	11	3413	3513
80	7	3801	5528
81	10	10793	7806
82	7	4524	4378
83	12	5293	7878
84	8	6262	5274
85	8	5046	5125
86	11	6905	8667
87	11	6029	9144
88	10	8165	3422
89	8	4459	3790

90	10	4230	7255
91	9	7034	6607
92	7	7229	5540
93	7	3379	3844
94	10	5300	6231
95	11	11728	7790
96	9	5396	4604
97	10	5851	6970
98	7	4195	6381
99	11	7677	6051
100	11	6075	8202
101	7	7446	8973
102	11	7448	5548
103	7	3173	4980
104	7	5146	4196
105	11	6485	9510
106	6	5791	5446
107	11	5721	4435
108	7	5307	10587
109	10	5959	4995
110	12	5947	5211
111	9	4883	8520
112	9	6002	5079
113	9	6033	6681
114	9	5248	6355
115	10	7917	7736
116	10	7635	5124
117	13	5666	5093
118	8	3866	7262
119	9	4588	6719
120	12	9760	9244
121	9	8310	5211
122	7	4922	6144
123	11	9377	5158
124	10	3692	5456
125	9	7293	7792
126	8	5140	7669
127	9	7403	7008
128	10	6635	6093
129	9	6957	5902
130	7	3549	3227
131	12	8469	8599
132	10	4525	7874
133	10	5177	6862
134	7	3446	3070
135	13	9287	7467
136	11	5386	5454
137	11	5228	4981

138	11	6222	7191
139	13	5415	8405
140	10	6869	7409
141	5	5115	2949
142	11	5176	6378
143	9	8016	5307
144	7	4146	2323
145	14	7090	7704
146	8	5598	7262
147	10	6174	7202
148	12	6673	6446
149	7	6892	5647
150	11	6829	6185
151	12	6338	5931
152	9	5335	3334

APPENDIX 2: Arena Input Analyzer for Vessel Arrival Distribution

Fit All Summary

Function	Sq Error
Poisson	0.00163
Weibull	0.00166
Beta	0.00167
Normal	0.00231
Gamma	0.00405
Erlang	0.00409
Triangular	0.00543
Lognormal	0.00887
Uniform	0.0218
Exponential	0.0582

APPENDIX 3: Functions for the percentage of containers entering/exiting each yard blocks from/to all eight berths.

The function of arrayed *to BLK 2* converter for eight berths.

***to BLK 2* converter**	**Function**
to BLK 2 [BTH 1]	10.58858695 / 100
to BLK 2 [BTH 2]	9.807180851 / 100
to BLK 2 [BTH 3]	15.25612472 / 100
to BLK 2 [BTH 4]	12.82476859 / 100
to BLK 2 [BTH 5]	9.899118396 / 100
to BLK 2 [BTH 6]	8.632277329 / 100
to BLK 2 [BTH 7]	5.112305034 / 100
to BLK 2 [BTH 8]	3.575825509 / 100

The function of arrayed *to BLK 3* converter for eight berths.

***to BLK 3* converter**	**Function**
to BLK 3 [BTH 1]	12.67489992 / 100
to BLK 3 [BTH 2]	13.47165135 / 100
to BLK 3 [BTH 3]	10.48052109 / 100
to BLK 3 [BTH 4]	13.54984573 / 100
to BLK 3 [BTH 5]	10.45380731 / 100
to BLK 3 [BTH 6]	10.30318707 / 100
to BLK 3 [BTH 7]	8.531750728 / 100
to BLK 3 [BTH 8]	3.732015215 / 100

The function of arrayed *to BLK 4* converter for eight berths.

***to BLK 4* converter**	**Function**
to BLK 4 [BTH 1]	12.15351086 / 100
to BLK 4 [BTH 2]	12.13934558 / 100
to BLK 4 [BTH 3]	9.437639198 / 100
to BLK 4 [BTH 4]	12.25981487 / 100
to BLK 4 [BTH 5]	12.22092325 / 100
to BLK 4 [BTH 6]	9.540340989 / 100
to BLK 4 [BTH 7]	8.776123642 / 100
to BLK 4 [BTH 8]	4.388930744 / 100

The function of arrayed *to BLK 5* converter for eight berths.

to BLK 5 converter	Function
to BLK 5 [BTH 1]	10.75696005 / 100
to BLK 5 [BTH 2]	10.41364442 / 100
to BLK 5 [BTH 3]	7.708487998 / 100
to BLK 5 [BTH 4]	10.21119268 / 100
to BLK 5 [BTH 5]	10.16171616 / 100
to BLK 5 [BTH 6]	7.424702417 / 100
to BLK 5 [BTH 7]	9.679652553 / 100
to BLK 5 [BTH 8]	7.489296411 / 100

The function of arrayed *to BLK 6* converter for eight berths.

to BLK 6 converter	Function
to BLK 6 [BTH 1]	13.06083377 / 100
to BLK 6 [BTH 2]	14.54757012 / 100
to BLK 6 [BTH 3]	13.89154028 / 100
to BLK 6 [BTH 4]	10.84370678 / 100
to BLK 6 [BTH 5]	14.62196938 / 100
to BLK 6 [BTH 6]	11.32197913 / 100
to BLK 6 [BTH 7]	7.318169984 / 100
to BLK 6 [BTH 8]	5.292534132 / 100

The function of arrayed *to BLK 7* converter for eight berths.

to BLK 7 converter	Function
to BLK 7 [BTH 1]	10.46221254 / 100
to BLK 7 [BTH 2]	11.40494036 / 100
to BLK 7 [BTH 3]	11.53356666 / 100
to BLK 7 [BTH 4]	10.92935419 / 100
to BLK 7 [BTH 5]	11.97324985 / 100
to BLK 7 [BTH 6]	9.461832737 / 100
to BLK 7 [BTH 7]	8.543584767 / 100
to BLK 7 [BTH 8]	7.44887084 / 100

The function of arrayed *to BLK 8* converter for eight berths.

to BLK 8 converter	Function
to BLK 8 [BTH 1]	7.300960294 / 100
to BLK 8 [BTH 2]	8.338874113 / 100
to BLK 8 [BTH 3]	6.361579524 / 100
to BLK 8 [BTH 4]	8.329077561 / 100
to BLK 8 [BTH 5]	9.308184593 / 100
to BLK 8 [BTH 6]	8.212053966 / 100
to BLK 8 [BTH 7]	6.844216705 / 100
to BLK 8 [BTH 8]	7.984050275 / 100

The function of arrayed *to BLK 9* converter for eight berths.

to BLK 9 converter	Function
to BLK 9 [BTH 1]	6.470445791 / 100
to BLK 9 [BTH 2]	6.067154255 / 100
to BLK 9 [BTH 3]	5.058419062 / 100
to BLK 9 [BTH 4]	6.870411746 / 100
to BLK 9 [BTH 5]	7.706302702 / 100
to BLK 9 [BTH 6]	7.860808953 / 100
to BLK 9 [BTH 7]	9.396227308 / 100
to BLK 9 [BTH 8]	11.65313023 / 100

The function of arrayed *to BLK 10* converter for eight berths.

to BLK 10 converter	Function
to BLK 10 [BTH 1]	4.816984116 / 100
to BLK 10 [BTH 2]	4.186311251 / 100
to BLK 10 [BTH 3]	2.753031428 / 100
to BLK 10 [BTH 4]	2.94552612 / 100
to BLK 10 [BTH 5]	4.621223158 / 100
to BLK 10 [BTH 6]	7.696985374 / 100
to BLK 10 [BTH 7]	12.18314359 / 100
to BLK 10 [BTH 8]	16.3962441 / 100

The function of arrayed *to BLK 11* converter for eight berths.

***to BLK 11* converter**	**Function**
to BLK 11 [BTH 1]	2.076853807 / 100
to BLK 11 [BTH 2]	1.553936976 / 100
to BLK 11 [BTH 3]	1.567858027 / 100
to BLK 11 [BTH 4]	0.86604958 / 100
to BLK 11 [BTH 5]	1.676503719 / 100
to BLK 11 [BTH 6]	7.804083337 / 100
to BLK 11 [BTH 7]	11.74883435 / 100
to BLK 11 [BTH 8]	16.99987137 / 100

The function of arrayed *to BLK 12* converter for eight berths.

***to BLK 12* converter**	**Function**
to BLK 12 [BTH 1]	1.417360969 / 100
to BLK 12 [BTH 2]	1.857168762 / 100
to BLK 12 [BTH 3]	5.207780959 / 100
to BLK 12 [BTH 4]	0.564953718 / 100
to BLK 12 [BTH 5]	1.063538709 / 100
to BLK 12 [BTH 6]	5.49920811 / 100
to BLK 12 [BTH 7]	7.982059596 / 100
to BLK 12 [BTH 8]	12.65458187 / 100

The function of arrayed *from BLK 2* converter for eight berths.

***from BLK 2* converter**	**Function**
from BLK 2 [BTH 1]	16.18220656 / 100
from BLK 2 [BTH 2]	27.54352118 / 100
from BLK 2 [BTH 3]	6.512303744 / 100
from BLK 2 [BTH 4]	15.34764715 / 100
from BLK 2 [BTH 5]	5.556975747 / 100
from BLK 2 [BTH 6]	2.136590595 / 100
from BLK 2 [BTH 7]	1.806539776 / 100
from BLK 2 [BTH 8]	5.29937893 / 100

The function of arrayed *from BLK 3* converter for eight berths.

***from BLK 3* converter**	**Function**
from BLK 3 [BTH 1]	16.7640543 / 100
from BLK 3 [BTH 2]	15.39099624 / 100
from BLK 3 [BTH 3]	22.91430782 / 100
from BLK 3 [BTH 4]	17.84940705 / 100
from BLK 3 [BTH 5]	10.59556361 / 100
from BLK 3 [BTH 6]	0.448531693 / 100
from BLK 3 [BTH 7]	0.377550024 / 100
from BLK 3 [BTH 8]	2.378031701 / 100

The function of arrayed *from BLK 4* converter for eight berths.

***from BLK 4* converter**	**Function**
from BLK 4 [BTH 1]	17.61254095 / 100
from BLK 4 [BTH 2]	12.92836177 / 100
from BLK 4 [BTH 3]	14.75741621 / 100
from BLK 4 [BTH 4]	19.67631451 / 100
from BLK 4 [BTH 5]	13.45058003 / 100
from BLK 4 [BTH 6]	3.299952044 / 100
from BLK 4 [BTH 7]	0.511078575 / 100
from BLK 4 [BTH 8]	3.921530749 / 100

The function of arrayed *from BLK 5* converter for eight berths.

***from BLK 5* converter**	**Function**
from BLK 5 [BTH 1]	7.672559486 / 100
from BLK 5 [BTH 2]	8.50567896 / 100
from BLK 5 [BTH 3]	10.89315368 / 100
from BLK 5 [BTH 4]	12.11688417 / 100
from BLK 5 [BTH 5]	18.71567945 / 100
from BLK 5 [BTH 6]	6.804705351 / 100
from BLK 5 [BTH 7]	2.417569546 / 100
from BLK 5 [BTH 8]	7.300711864 / 100

The function of arrayed *from BLK 6* converter for eight berths.

***from BLK 6* converter**	**Function**
from BLK 6 [BTH 1]	7.610112454 / 100
from BLK 6 [BTH 2]	8.786061211 / 100
from BLK 6 [BTH 3]	17.02332428 / 100
from BLK 6 [BTH 4]	16.05092868 / 100
from BLK 6 [BTH 5]	11.00813824 / 100
from BLK 6 [BTH 6]	9.832153234 / 100
from BLK 6 [BTH 7]	16.05661298 / 100
from BLK 6 [BTH 8]	4.071727309 / 100

The function of arrayed *from BLK 7* converter for eight berths.

***from BLK 7* converter**	**Function**
from BLK 7 [BTH 1]	4.064508775 / 100
from BLK 7 [BTH 2]	4.02481777 / 100
from BLK 7 [BTH 3]	12.33488074 / 100
from BLK 7 [BTH 4]	8.119889533 / 100
from BLK 7 [BTH 5]	10.58456162 / 100
from BLK 7 [BTH 6]	20.811870575 / 100
from BLK 7 [BTH 7]	13.2193265 / 100
from BLK 7 [BTH 8]	7.267871459 / 100

The function of arrayed *from BLK 8* converter for eight berths.

***from BLK 8* converter**	**Function**
from BLK 8 [BTH 1]	3.30176289 / 100
from BLK 8 [BTH 2]	1.642882345 / 100
from BLK 8 [BTH 3]	7.859205163 / 100
from BLK 8 [BTH 4]	2.088184329 / 100
from BLK 8 [BTH 5]	10.29300888 / 100
from BLK 8 [BTH 6]	14.76148834 / 100
from BLK 8 [BTH 7]	12.38613958 / 100
from BLK 8 [BTH 8]	5.750451556 / 100

The function of arrayed *from BLK 9* converter for eight berths.

from BLK 9 converter	Function
from BLK 9 [BTH 1]	3.652655733 / 100
from BLK 9 [BTH 2]	0.776587519 / 100
from BLK 9 [BTH 3]	2.673516757 / 100
from BLK 9 [BTH 4]	0.630855039 / 100
from BLK 9 [BTH 5]	12.07080104 / 100
from BLK 9 [BTH 6]	11.95576744 / 100
from BLK 9 [BTH 7]	11.21561737 / 100
from BLK 9 [BTH 8]	12.50253547 / 100

The function of arrayed *from BLK 10* converter for eight berths.

from BLK 10 converter	Function
from BLK 10 [BTH 1]	2.055795927 / 100
from BLK 10 [BTH 2]	0.103970393 / 100
from BLK 10 [BTH 3]	2.569727506 / 100
from BLK 10 [BTH 4]	0.075134023 / 100
from BLK 10 [BTH 5]	3.566586309 / 100
from BLK 10 [BTH 6]	14.18827047 / 100
from BLK 10 [BTH 7]	15.9363592 / 100
from BLK 10 [BTH 8]	15.18289208 / 100

The function of arrayed *from BLK 11* converter for eight berths.

from BLK 11 converter	Function
from BLK 11 [BTH 1]	0.303809765 / 100
from BLK 11 [BTH 2]	0.346067517 / 100
from BLK 11 [BTH 3]	0.68359375 / 100
from BLK 11 [BTH 4]	0.289029079 / 100
from BLK 11 [BTH 5]	0.553658971 / 100
from BLK 11 [BTH 6]	7.074388558 / 100
from BLK 11 [BTH 7]	13.9681796 / 100
from BLK 11 [BTH 8]	19.73998126 / 100

The function of arrayed *from BLK 12* converter for eight berths.

from BLK 12 **converter**	**Function**
from BLK 12 [BTH 1]	0.284480921 / 100
from BLK 12 [BTH 2]	0.546501415 / 100
from BLK 12 [BTH 3]	0.437330163 / 100
from BLK 12 [BTH 4]	1.071505929 / 100
from BLK 12 [BTH 5]	0.944876794 / 100
from BLK 12 [BTH 6]	8.317865102 / 100
from BLK 12 [BTH 7]	11.93284529 / 100
from BLK 12 [BTH 8]	11.63323063 / 100

APPENDIX 4: Computation of the autocorrelation coefficient for the total container throughput data and vessel arrival data.

Computation of the Autocorrelation Coefficient for the Total Container Throughput Data.

i. Lag 1 (r_1)

Year, t	**Container Throughput, Yt**	**Y_{t-1}**	**(Y_t - □)**	**(Y_{t-1} - □)**	**(Y_t - □)^2**	**(Y_t - □)(Y_{t-1}-□)**
2002	2668512		-1735848.5		3.013E+12	
2003	3316954	2668512	-1087406.5	-1735848.5	1.182E+12	1.888E+12
2004	3835970	3316954	-568390.5	-1087406.5	3.231E+11	6.181E+11
2005	4177123	3835970	-227237.5	-568390.5	5.164E+10	1.292E+11
2006	4637418	4177123	233057.5	-227237.5	5.432E+10	-5.296E+10
2007	5297631	4637418	893270.5	233057.5	7.979E+11	2.082E+11
2008	5466191	5297631	1061830.5	893270.5	1.127E+12	9.485E+11
2009	5835085	5466191	1430724.5	1061830.5	2.047E+12	1.519E+12
Total	**35234884**				**8.597E+12**	**5.258E+12**

□ = 35234884 / 8
= 4404360.5

Lag 1 (r_1)
= 5.258E+12 / 8.597E+12
= 0.61157

ii. Lag 2 (r_2) and Lag 3 (r_3)

Year, t	**Y_{t-2}**	**(Y_{t-2} - □)**	**(Y_{t-2}) (Y_{t-2} - □)**	**Y_{t-3}**	**(Y_{t-3} - □)**	**(Y_{t-3}) (Y_{t-3} - □)**
2002						
2003						
2004	2668512	-1735848.5	9.866E+11			
2005	3316954	-1087406.5	2.471E+11	2668512	-1735848.5	3.944E+11
2006	3835970	-568390.5	-1.325E+11	3316954	-1087406.5	-2.53E+11
2007	4177123	-227237.5	-2.03E+11	3835970	-568390.5	-5.08E+11
2008	4637418	233057.5	2.475E+11	4177123	-227237.5	-2.41E+11
2009	5297631	893270.5	1.278E+12	4637418	233057.5	3.334E+11
Total			**2.424E+12**			**-2.75E+11**

Lag 2 (r_2)
= 2.424E+12 / 8.597E+12
= 0.28193

Lag 3 (r_3)
= -2.75E+11 / 8.597E+12
= -0.03194

Computation of the Autocorrelation Coefficient for the Vessel Arrival Data.

i. Lag 1 (r_1)

Year, t	**Vessel Arrival, Yt**	Y_{t-1}	**(Y_t - □)**	**(Y_{t-1} - □)**	**(Y_t - □)^2**	**(Y_t - □)(Y_{t-1}-□)**
2002	3483		46		2116	
2003	3148	3483	-289	46	83521	-13294
2004	3193	3148	-244	-289	59536	70516
2005	3128	3193	-309	-244	95481	75396
2006	3261	3128	-176	-309	30976	54384
2007	3747	3261	310	-176	96100	-54560
2008	3760	3747	323	310	104329	100130
2009	3776	3760	339	323	114921	109497
Total	**27496**				**586980**	**342069**

□ = 27496 / 8
= 3437

Lag 1 (r_1)
= 342069 / 586980
= 0.58276

ii. Lag 2 (r_2) and Lag 3 (r_3)

Year, t	Y_{t-2}	**(Y_{t-2} - □)**	**(Y_{t-2}) (Y_{t-2} - □)**	Y_{t-3}	**(Y_{t-3} - □)**	**(Y_{t-3}) (Y_{t-3} - □)**
2002						
2003						
2004	3483	46	-11224			
2005	3148	-289	89301	3483	46	-14214
2006	3193	-244	42944	3148	-289	50864
2007	3128	-309	-95790	3193	-244	-75640
2008	3261	-176	-56848	3128	-309	-99807
2009	3747	310	105090	3261	-176	-59664
Total			**73473**			**-198461**

Lag 2 (r_2)
= 73473 / 586980
= 0.12517

Lag 3 (r_3)
= -198461 / 586980
= -0.33811

APPENDIX 5: Double moving average forecast for the total container throughput and total vessel arrival.

Double Moving Average Forecast for the Total Container Throughput.

Year, t	Container Throughput, Yt	3-Years Moving Average M_t	Double Moving Average M_t`	Value of a	Value of b	Forecast a + bp
2002	2668512					
2003	3316954					
2004	3835970	3273812				
2005	4177123	3776682.3				
2006	4637418	4216837	3755777	4677897	461059.9	
2007	5297631	4704057.3	4232526	5175589	471531.8	5138956.7
2008	5466191	5133746.7	4684880	5582613	448866.3	5647120.9
2009	5835085	5532969	5123591	5942347	409378	6031479.3
2010						**6351725**
2011						**6761103**
2012						**7170481**

M_t = $(Yt + Y_{t-1} + Y_{t-2} + ... + Y_{t-K+1}) / k$
M_t` = $(Mt + M_{t-1} + M_{t-2} + ... + M_{t-K+1}) / k$
a_t = Mt + (Mt - M_t`)
= 2Mt - M_t`
b_t = (2 / k-1) (Mt - M_t`)

Where
k = Number of periods in moving average
p = Number of periods ahead to be forecast

Double Moving Average Forecast for the Total Vessel Arrival.

Year, t	Vessel Arrival, Yt	2-Years Moving Average M_t	Double Moving Average M_t`	Value of a	Value of b	Forecast a + bp
2002	3483					
2003	3148	3315.5				
2004	3193	3170.5	3243	3098	-145	
2005	3128	3160.5	3165.5	3155.5	-10	2953
2006	3261	3194.5	3177.5	3211.5	34	3146
2007	3747	3504	3349.25	3658.75	309.5	3246
2008	3760	3753.5	3628.75	3878.25	249.5	3968
2009	3776	3768	3760.75	3775.25	14.5	4128
2010						**3790**
2011						**3804**
2012						**3818**

M_t = $(Yt + Y_{t-1} + Y_{t-2} + ... + Y_{t-K+1}) / k$

$M_t`$ = $(Mt + M_{t-1} + M_{t-2} + ... + M_{t-K+1}) / k$

a_t = $Mt + (Mt - M_t`)$

= $2Mt - M_t`$

b_t = $(2 / k-1) (Mt - M_t`)$

Where

k = Number of periods in moving average

p = Number of periods ahead to be forecast

APPENDIX 6: Paired Samples Test between actual data and simulated output.

Paired Samples Test for the actual data and the simulated output of the number of vessels arrival in January 2008 to May 2008.

Paired Samples Test

	Paired Differences					t	df	Sig. (2-tailed)
				95% Confidence Interval of the Difference				
	Mean	Std. Deviation	Std. Error Mean	Lower	Upper			
Pair 1 ACTARR - SIMUVARR	.0329	4.03038	.32691	-.6130	.6788	.101	151	.920

Paired Samples Test for the actual data and the simulated output of the number of containers unloaded at each berth in January 2008 to May 2008.

Paired Samples Test

	Paired Differences					t	df	Sig. (2-tailed)
				95% Confidence Interval of the Difference				
	Mean	Std. Deviation	Std. Error Mean	Lower	Upper			
Pair 1 ACTUNLB1 - SIMUNLB1	-2058.40	3722.32875	1664.676	-6680.28	2563.4816	-1.237	4	.284
Pair 2 ACTUNLB2 - SIMUNLB2	222.8000	3282.91230	1468.163	-3853.47	4299.0740	.152	4	.887
Pair 3 ACTUNLB3 - SIMUNLB3	-5093.40	5434.01052	2430.163	-11840.6	1653.8152	-2.096	4	.104
Pair 4 ACTUNLB4 - SIMUNLB4	-1384.20	3152.00590	1409.620	-5297.93	2529.5322	-.982	4	.382
Pair 5 ACTUNLB5 - SIMUNLB5	766.1000	3383.66762	1513.222	-3435.28	4967.4783	.506	4	.639
Pair 6 ACTUNLB6 - SIMUNLB6	-976.3000	2396.37598	1071.692	-3951.79	1999.1938	-.911	4	.414
Pair 7 ACTUNLB7 - SIMUNLB7	-1441.00	2917.57545	1304.779	-5063.65	2181.6484	-1.104	4	.331
Pair 8 ACTUNLB8 - SIMUNLB8	-5599.40	5568.79367	2490.440	-12514.0	1315.1706	-2.248	4	.088

Paired Samples Test for the actual data and the simulated output of the number of containers loaded at each berth in January 2008 to May 2008.

Paired Samples Test

	Paired Differences					t	df	Sig. (2-tailed)
				95% Confidence Interval of the Difference				
	Mean	Std. Deviation	Std. Error Mean	Lower	Upper			
Pair 1 ACTLDB1 - SIMLDB1	-524.5000	3984.06030	1781.726	-5471.36	4422.3642	-.294	4	.783
Pair 2 ACTLDB2 - SIMLDB2	-2905.40	5737.47674	2565.878	-10029.4	4218.6183	-1.132	4	.321
Pair 3 ACTLDB3 - SIMLDB3	-1104.40	2700.41771	1207.664	-4457.41	2248.6114	-.914	4	.412
Pair 4 ACTLDB4 - SIMLDB4	-1148.60	3571.74401	1597.332	-5583.51	3286.3060	-.719	4	.512
Pair 5 ACTLDB5 - SIMLDB5	-954.9000	6484.05258	2899.756	-9005.91	7096.1146	-.329	4	.758
Pair 6 ACTLDB6 - SIMLDB6	78.9000	3218.31793	1439.276	-3917.17	4074.9695	.055	4	.959
Pair 7 ACTLDB7 - SIMLDB7	-967.9000	3756.24201	1679.842	-5631.89	3696.0905	-.576	4	.595
Pair 8 ACTLDB8 - SIMLDB8	-1325.00	2446.07982	1093.920	-4362.21	1712.2093	-1.211	4	.292

Paired Samples Test for the actual data and the simulated output of the berth occupancy rate.

Paired Samples Test

		Paired Differences						
				95% Confidence Interval of the Difference				
	Mean	Std. Deviation	Std. Error Mean	Lower	Upper	t	df	Sig. (2-tailed)
Pair 1 VAR00001 - VAR00002	-1.1120	3.50852	1.56906	-5.4684	3.2444	-.709	4	.518

Paired Samples Test for the actual data and the simulated output of the yard occupancy rate.

Paired Samples Test

		Paired Differences						
				95% Confidence Interval of the Difference				
	Mean	Std. Deviation	Std. Error Mean	Lower	Upper	t	df	Sig. (2-tailed)
Pair 1 ACTDATA - SIMOUT	-1.3720	2.20500	.98611	-4.1099	1.3659	-1.391	4	.237

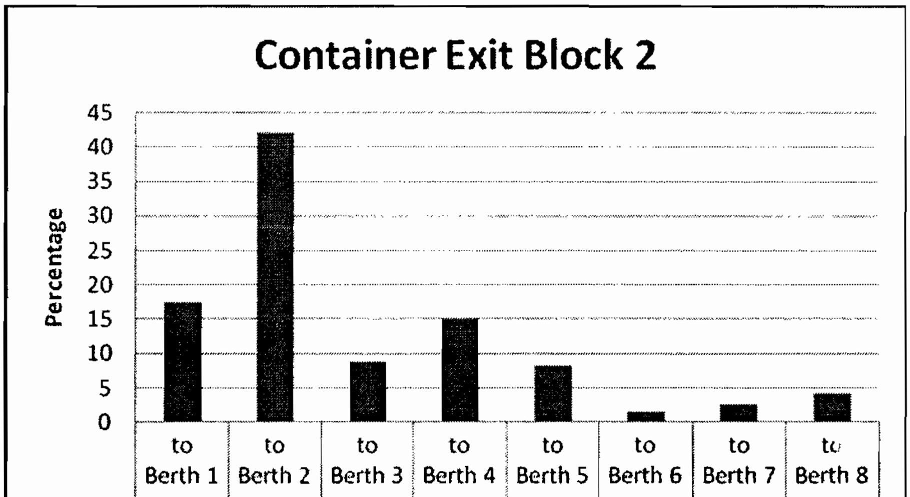
Paired Samples Test for the MOT data and the simulated output of the container throughput in year 2008.

Paired Samples Test

		Paired Differences						
				95% Confidence Interval of the Difference				
	Mean	Std. Deviation	Std. Error Mean	Lower	Upper	t	df	Sig. (2-tailed)
Pair 1 VAR00001 - VAR00002	-34789.0	182792.64819	91396.32	-325653	256074.9	-.381	3	.729

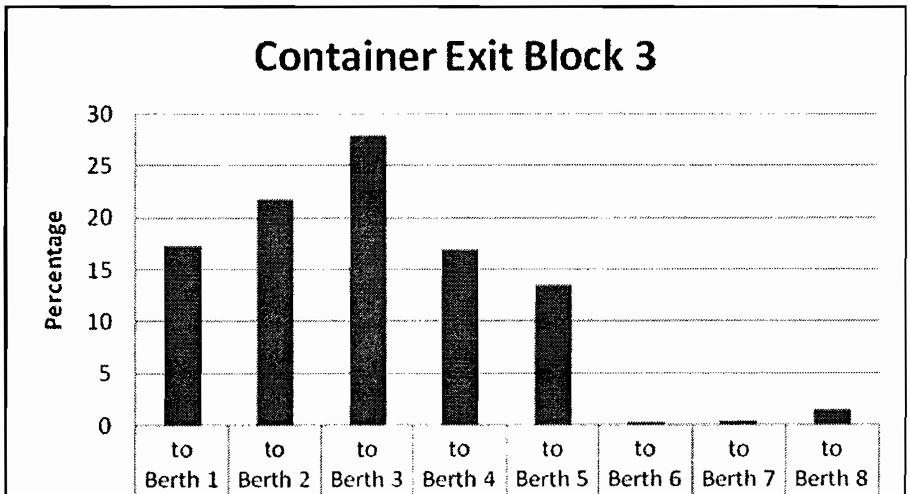
APPENDIX 7: Bar graphs showing the percentage of containers exiting each yard blocks to all eight berths.

Percentage of containers exiting Yard Block 2 to the all eight berths.



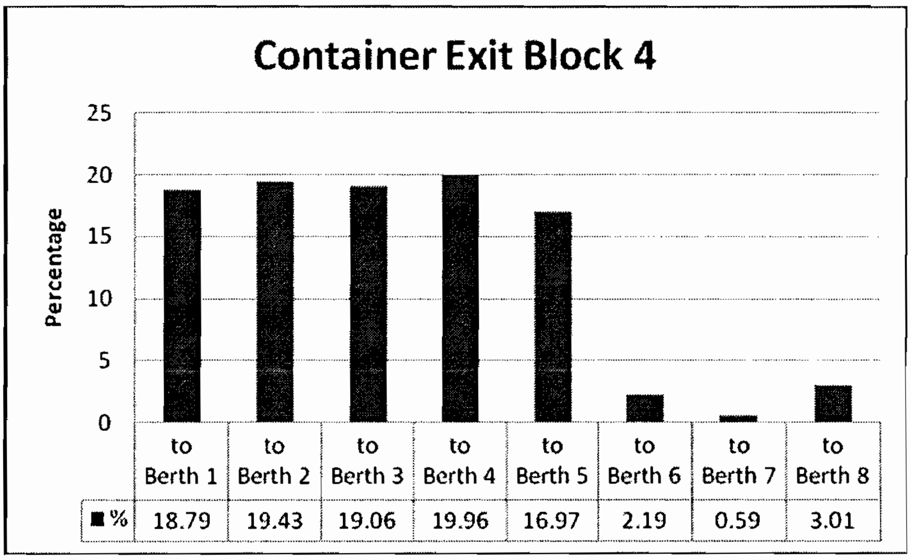
	to Berth 1	to Berth 2	to Berth 3	to Berth 4	to Berth 5	to Berth 6	to Berth 7	to Berth 8
■ %	17.39	42.15	8.77	15.22	8.28	1.5	2.58	4.11

Percentage of containers exiting Yard Block 3 to the all eight berths.

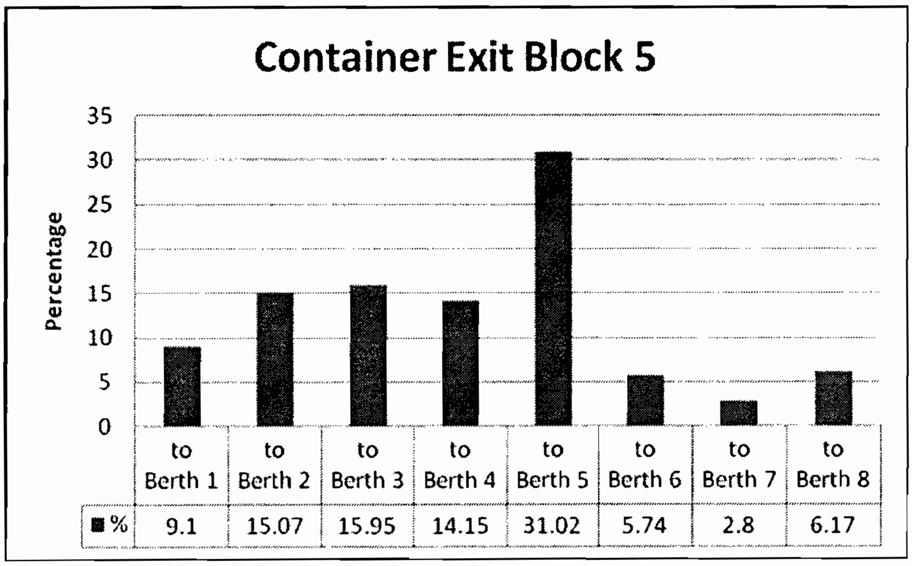


	to Berth 1	to Berth 2	to Berth 3	to Berth 4	to Berth 5	to Berth 6	to Berth 7	to Berth 8
■ %	17.34	21.86	27.99	17.01	13.57	0.29	0.43	1.5

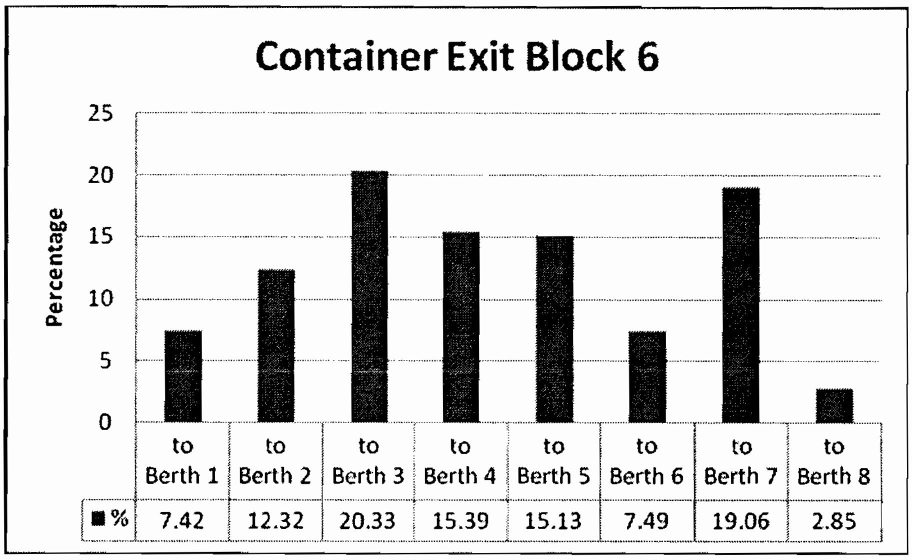
Percentage of containers exiting Yard Block 4 to the all eight berths.



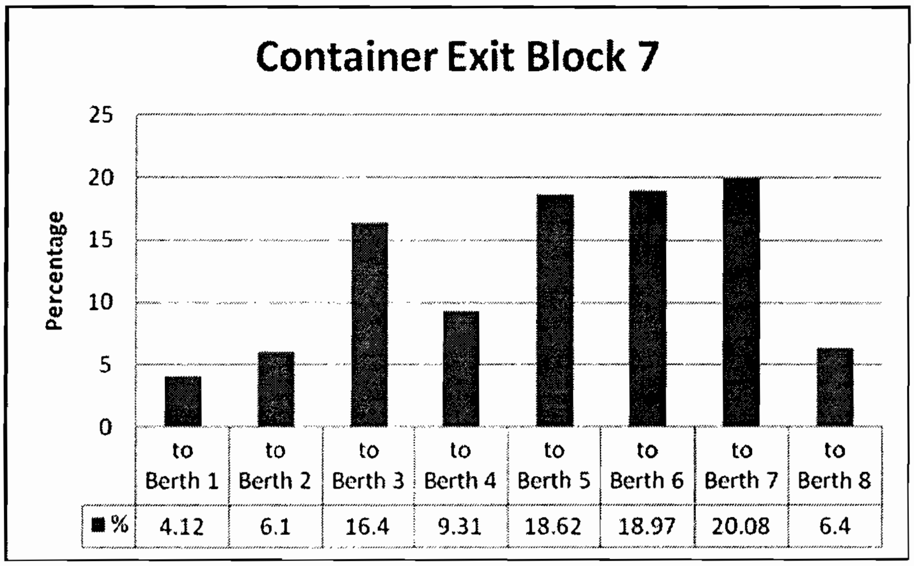
Percentage of containers exiting Yard Block 5 to the all eight Berths.



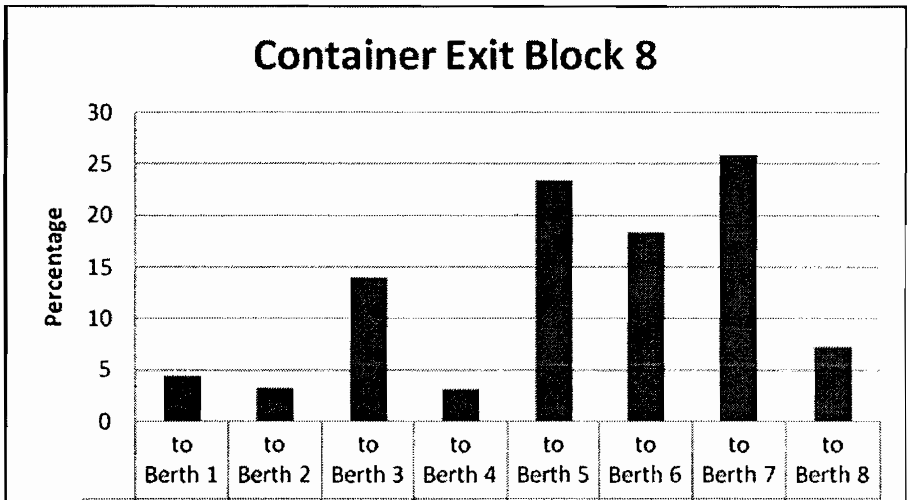
Percentage of containers exiting Yard Block 6 to the all eight berths.



Percentage of containers exiting Yard Block 7 to the all eight berths.

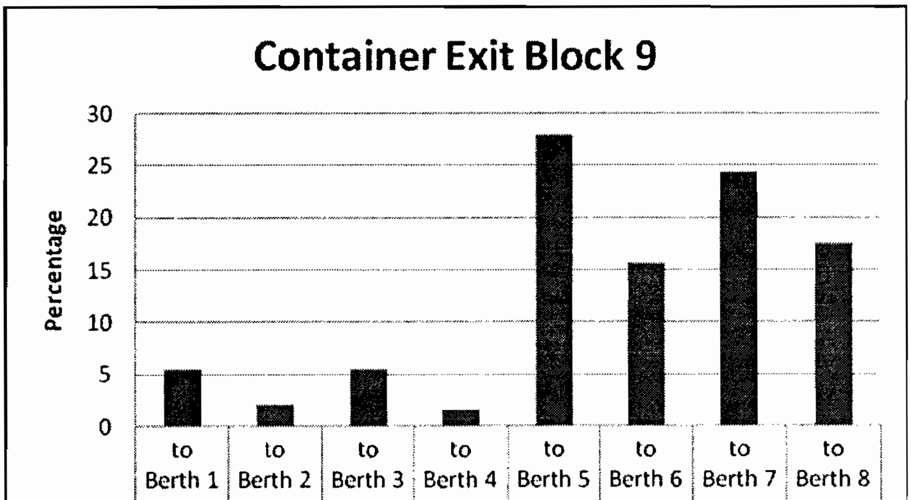


Percentage of containers exiting Yard Block 8 to the all eight berths.



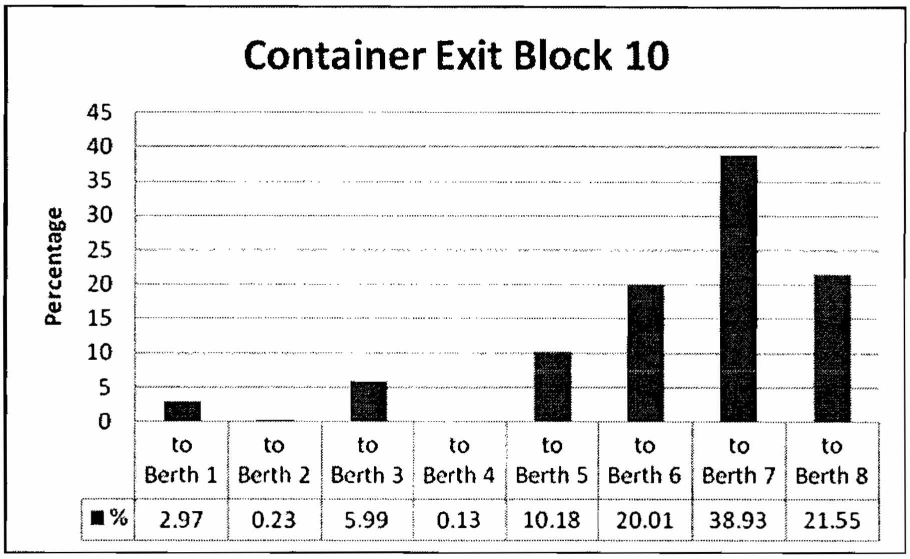
	to Berth 1	to Berth 2	to Berth 3	to Berth 4	to Berth 5	to Berth 6	to Berth 7	to Berth 8
%	4.42	3.26	14.06	3.19	23.44	18.43	25.95	7.26

Percentage of containers exiting Yard Block 9 to the all eight berths.

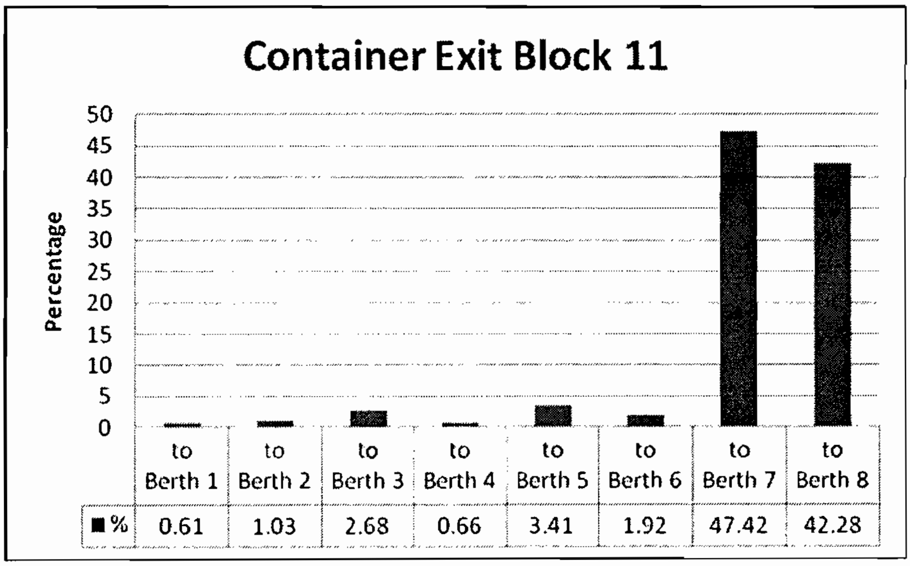


	to Berth 1	to Berth 2	to Berth 3	to Berth 4	to Berth 5	to Berth 6	to Berth 7	to Berth 8
%	5.46	2.02	5.47	1.56	27.9	15.63	24.45	17.51

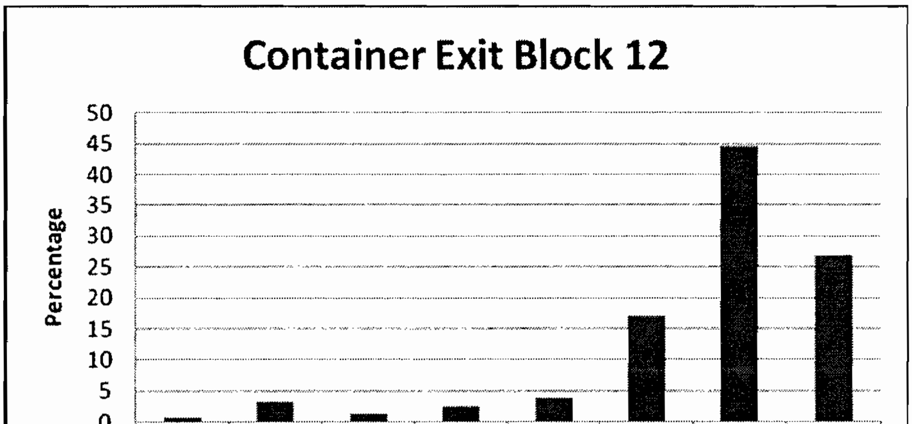
Percentage of containers exiting Yard Block 10 to the all eight berths.



Percentage of containers exiting Yard Block 11 to the all eight berths.



Percentage of containers exiting Yard Block 12 to the all eight berths.



	to BTH 1	to BTH 2	to Berth 3	to Berth 4	to Berth 5	to Berth 6	to Berth 7	to Berth 8
%	0.63	3.29	1.3	2.54	3.79	17.02	44.55	26.88

APPENDIX 8: Simulation code for *iThink*

BERTH: Berth Allocation Rules

```
Berth_Capacity(t) = Berth_Capacity(t - dt) + (ML_Length_Check +
FD_Length_Check - VS_Length_Out) * dt
INIT Berth_Capacity = 0
        TRANSIT TIME = varies
        INFLOW LIMIT = INF
        CAPACITY = 2880
INFLOWS:
ML_Length_Check = ML_Length_Q
FD_Length_Check = FD_Length_Q
OUTFLOWS:
VS_Length_Out = CONVEYOR OUTFLOW
        TRANSIT TIME = Ave_Service_Time

Check_Tt_VS_Served(t) = Check_Tt_VS_Served(t - dt) +(VS_Length_Leave_BTH)
* dt
INIT Check_Tt_VS_Served = 0
INFLOWS:
VS_Length_Leave_BTH = CONVEYOR OUTFLOW
        TRANSIT TIME = Ave_Service_Time
Check_VS_in_Process(t) = Check_VS_in_Process(t - dt) + (ML_Entering_Rate +
FD_Entering_Rate - VS_Length_Leave_BTH) * dt
INIT Check_VS_in_Process = 0
        TRANSIT TIME = varies
        INFLOW LIMIT = INF
        CAPACITY = INF
INFLOWS:
ML_Entering_Rate = IF ( ML_In2 = 1 ) THEN ( ML_arr ) ELSE (IF ( ML_In2 = 0 )
THEN ( ML_In1 ) ELSE ( 0 ) )
FD_Entering_Rate = IF ( FD_In2 = 1 ) THEN ( FD_arr ) ELSE (IF ( FD_In2 = 0 )
THEN ( FD_In1 ) ELSE ( 0 ) )
OUTFLOWS:
VS_Length_Leave_BTH = CONVEYOR OUTFLOW
        TRANSIT TIME = Ave_Service_Time

FD_Length_Q(t) = FD_Length_Q(t - dt) + (FD_Length_in - FD_Length_Check) * dt
INIT FD_Length_Q = 0
INFLOWS:
FD_Length_in = FD_Length_Rd*FD_Arrival
OUTFLOWS:
FD_Length_Check = FD_Length_Q

FD_Q(t) = FD_Q(t - dt) + (FD_arr - FD_Entering_Rate) * dt
INIT FD_Q = 0
INFLOWS:
FD_arr = FD_Arrival
OUTFLOWS:
```

FD_Entering_Rate = IF (FD_In2 = 1) THEN (FD_arr) ELSE (IF (FD_In2 = 0) THEN (FD_In1) ELSE (0))

ML_Length_Q(t) = ML_Length_Q(t - dt) + (ML_Length_in - ML_Length_Check) * dt
INIT ML_Length_Q = 0
INFLOWS:
ML_Length_in = ML_Length_Rd*ML_Arrival
OUTFLOWS:
ML_Length_Check = ML_Length_Q

ML_Q(t) = ML_Q(t - dt) + (ML_arr - ML_Entering_Rate) * dt
INIT ML_Q = 0
INFLOWS:
ML_arr = ML_Arrival
OUTFLOWS:
ML_Entering_Rate = IF (ML_In2 = 1) THEN (ML_arr) ELSE (IF (ML_In2 = 0) THEN (ML_In1) ELSE (0))

Noname_356(t) = Noname_356(t - dt) + (Noname_355) * dt
INIT Noname_356 = 0
INFLOWS:
Noname_355 = ML_Entering_Rate

Noname_358(t) = Noname_358(t - dt) + (Noname_357) * dt
INIT Noname_358 = 0
INFLOWS:
Noname_357 = FD_Entering_Rate

Total_FD_Enter_CT(t) = Total_FD_Enter_CT(t - dt) + (Noname_149) * dt
INIT Total_FD_Enter_CT = 0
INFLOWS:
Noname_149 = FD_Arrival

Total_FD_Queued(t) = Total_FD_Queued(t - dt) + (Noname_426) * dt
INIT Total_FD_Queued = 0
INFLOWS:
Noname_426 = Numb_of_FD_Queuing

Total_ML_Enter_CT(t) = Total_ML_Enter_CT(t - dt) + (Noname_145) * dt
INIT Total_ML_Enter_CT = 0
INFLOWS:
Noname_145 = ML_Arrival

Total_ML_Queued(t) = Total_ML_Queued(t - dt) + (Noname_425) * dt
INIT Total_ML_Queued = 0
INFLOWS:
Noname_425 = Num_of_ML_Queuing
Berth_Capacity_% = (Berth_Capacity / CAP (Berth_Capacity))*100
FD_&_ML_Check = FD_In1+ML_In1

FD_In1 = FD_Length_Check/FD_Length_Rd
FD_In2 = IF (FD_Length_Q = FD_Length_Check) THEN (1) ELSE (0)
FD_Length_Rd = ROUND (RANDOM (128 , 240 , 28))
ML_In1 = ML_Length_Check/ (ML_Length_Rd)
ML_In2 = IF (ML_Length_Q = ML_Length_Check) THEN (1) ELSE (0)
ML_Length_Rd = ROUND (RANDOM (230 , 430 , 28))
Num_of_ML_Queuing = ML_Q
Numb_of_BTH_Used = CAP (Berth_Capacity) /360
Numb_of_FD_Queuing = FD_Q

BERTH: Vesel Arrival and Departure

Accum_of_FD_Serviced[Berth](t) = Accum_of_FD_Serviced[Berth](t - dt) + (FD_Depart[Berth]) * dt
INIT Accum_of_FD_Serviced[Berth] = 0
INFLOWS:
FD_Depart[Berth] = CONVEYOR OUTFLOW
TRANSIT TIME = Ave_Service_Time

Accum_of_ML_Serviced[Berth](t) = Accum_of_ML_Serviced[Berth](t - dt) + (ML_Depart[Berth]) * dt
INIT Accum_of_ML_Serviced[Berth] = 0
INFLOWS:
ML_Depart[Berth] = CONVEYOR OUTFLOW
TRANSIT TIME = Ave_Service_Time

Allocating_FD_to_BTH(t) = Allocating_FD_to_BTH(t - dt) + (FD_Enter_CT - FD_Enter_BTH[BTH1] - FD_Enter_BTH[BTH2] - FD_Enter_BTH[BTH3] - FD_Enter_BTH[BTH4] - FD_Enter_BTH[BTH5] - FD_Enter_BTH[BTH6] - FD_Enter_BTH[BTH7] - FD_Enter_BTH[BTH8] - FD_Enter_BTH[BTH9] - FD_Enter_BTH[BTH10] - FD_Enter_BTH[BTH11] - FD_Enter_BTH[BTH12] - FD_Enter_BTH[BTH13] - FD_Enter_BTH[BTH14] - FD_Enter_BTH[BTH15] - FD_Enter_BTH[Berth]) * dt
INIT Allocating_FD_to_BTH = 0
INFLOWS:
FD_Enter_CT = FD_Entering_Rate
OUTFLOWS:
FD_Enter_BTH[BTH1] = IF (Dec_on_Numb_of_BTH[BTH1] = 1) THEN (FD_Enter_CT / Numb_of_BTH_Used)ELSE (0)
FD_Enter_BTH[BTH2] = IF (Dec_on_Numb_of_BTH[BTH2] = 1) THEN (FD_Enter_CT / Numb_of_BTH_Used) ELSE (0)
FD_Enter_BTH[BTH3] = IF (Dec_on_Numb_of_BTH[BTH3] = 1) THEN (FD_Enter_CT / Numb_of_BTH_Used)ELSE (0)
FD_Enter_BTH[BTH4] = IF (Dec_on_Numb_of_BTH[BTH4] = 1) THEN (FD_Enter_CT / Numb_of_BTH_Used)ELSE (0)
FD_Enter_BTH[BTH5] = IF (Dec_on_Numb_of_BTH[BTH5] = 1) THEN (FD_Enter_CT / Numb_of_BTH_Used)ELSE (0)
FD_Enter_BTH[BTH6] = IF (Dec_on_Numb_of_BTH[BTH6] = 1) THEN (FD_Enter_CT / Numb_of_BTH_Used)ELSE (0)
FD_Enter_BTH[BTH7] = IF (Dec_on_Numb_of_BTH[BTH7] = 1)

THEN (FD_Enter_CT / Numb_of_BTH_Used)ELSE (0)
FD_Enter_BTH[BTH8] = IF (Dec_on_Numb_of_BTH[BTH8] = 1)
THEN (FD_Enter_CT / Numb_of_BTH_Used)ELSE (0)
FD_Enter_BTH[BTH9] = IF (Dec_on_Numb_of_BTH[BTH9] = 1)
THEN (FD_Enter_CT / Numb_of_BTH_Used)ELSE (0)
FD_Enter_BTH[BTH10] = IF (Dec_on_Numb_of_BTH[BTH10] = 1)
THEN (FD_Enter_CT / Numb_of_BTH_Used)ELSE (0)
FD_Enter_BTH[BTH11] = IF (Dec_on_Numb_of_BTH[BTH11] = 1)
THEN (FD_Enter_CT / Numb_of_BTH_Used)ELSE (0)
FD_Enter_BTH[BTH12] = IF (Dec_on_Numb_of_BTH[BTH12] = 1)
THEN (FD_Enter_CT / Numb_of_BTH_Used)ELSE (0)
FD_Enter_BTH[BTH13] = IF (Dec_on_Numb_of_BTH[BTH13] = 1)
THEN (FD_Enter_CT / Numb_of_BTH_Used)ELSE (0)
FD_Enter_BTH[BTH14] = IF (Dec_on_Numb_of_BTH[BTH14] = 1)
THEN (FD_Enter_CT / Numb_of_BTH_Used)ELSE (0)
FD_Enter_BTH[BTH15] = IF (Dec_on_Numb_of_BTH[BTH15] = 1)
THEN (FD_Enter_CT / Numb_of_BTH_Used)ELSE (0)

Allocating_ML_to_BTH(t) = Allocating_ML_to_BTH(t - dt) + (ML_Enter_CT - ML_Enter_BTH[BTH1] - ML_Enter_BTH[BTH2] - ML_Enter_BTH[BTH3] - ML_Enter_BTH[BTH4] - ML_Enter_BTH[BTH5] - ML_Enter_BTH[BTH6] - ML_Enter_BTH[BTH7] - ML_Enter_BTH[BTH8] - ML_Enter_BTH[BTH9] - ML_Enter_BTH[BTH10] - ML_Enter_BTH[BTH11] - ML_Enter_BTH[BTH12] - ML_Enter_BTH[BTH13] - ML_Enter_BTH[BTH14] - ML_Enter_BTH[BTH15] - ML_Enter_BTH[Berth]) * dt
INIT Allocating_ML_to_BTH = 0
INFLOWS:
ML_Enter_CT = ML_Entering_Rate
OUTFLOWS:
ML_Enter_BTH[BTH1] = IF (Dec_on_Numb_of_BTH[BTH1] = 1)THEN (ML_Enter_CT / Numb_of_BTH_Used)ELSE (0)
ML_Enter_BTH[BTH2] = IF (Dec_on_Numb_of_BTH[BTH2] = 1)THEN (ML_Enter_CT / Numb_of_BTH_Used)ELSE (0)
ML_Enter_BTH[BTH3] = IF (Dec_on_Numb_of_BTH[BTH3] = 1)THEN (ML_Enter_CT / Numb_of_BTH_Used)ELSE (0)
ML_Enter_BTH[BTH4] = IF (Dec_on_Numb_of_BTH[BTH4] = 1)THEN (ML_Enter_CT / Numb_of_BTH_Used)ELSE (0)
ML_Enter_BTH[BTH5] = IF (Dec_on_Numb_of_BTH[BTH5] = 1)THEN (ML_Enter_CT / Numb_of_BTH_Used)ELSE (0)
ML_Enter_BTH[BTH6] = IF (Dec_on_Numb_of_BTH[BTH6] = 1)THEN (ML_Enter_CT / Numb_of_BTH_Used)ELSE (0)
ML_Enter_BTH[BTH7] = IF (Dec_on_Numb_of_BTH[BTH7] = 1)THEN (ML_Enter_CT / Numb_of_BTH_Used)ELSE (0)
ML_Enter_BTH[BTH8] = IF (Dec_on_Numb_of_BTH[BTH8] = 1)THEN (ML_Enter_CT / Numb_of_BTH_Used)ELSE (0)
ML_Enter_BTH[BTH9] = IF (Dec_on_Numb_of_BTH[BTH9] = 1)THEN (ML_Enter_CT / Numb_of_BTH_Used)ELSE (0)
ML_Enter_BTH[BTH10] = IF (Dec_on_Numb_of_BTH[BTH10] = 1)THEN (ML_Enter_CT / Numb_of_BTH_Used)ELSE (0)
ML_Enter_BTH[BTH11] = IF (Dec_on_Numb_of_BTH[BTH11] = 1)THEN (

ML_Enter_CT / Numb_of_BTH_Used)ELSE (0)
ML_Enter_BTH[BTH12] = IF (Dec_on_Numb_of_BTH[BTH12] = 1)THEN (
ML_Enter_CT / Numb_of_BTH_Used)ELSE (0)
ML_Enter_BTH[BTH13] = IF (Dec_on_Numb_of_BTH[BTH13] = 1)THEN (
ML_Enter_CT / Numb_of_BTH_Used)ELSE (0)
ML_Enter_BTH[BTH14] = IF (Dec_on_Numb_of_BTH[BTH14] = 1)THEN (
ML_Enter_CT / Numb_of_BTH_Used)ELSE (0)
ML_Enter_BTH[BTH15] = IF (Dec_on_Numb_of_BTH[BTH15] = 1)THEN (
ML_Enter_CT / Numb_of_BTH_Used)ELSE (0)

Base_Case_Arrival(t) = Base_Case_Arrival(t - dt) + (Noname_243) * dt
INIT Base_Case_Arrival = 0
INFLOWS:
Noname_243 = Dec_on_Daily_or_Annual

Change_in_Arrival_Rate(t) = Change_in_Arrival_Rate(t - dt) + (Noname_89) * dt
INIT Change_in_Arrival_Rate = 0
INFLOWS:
Noname_89 = Vessel_Arrival

FD_Being_Processed[Berth](t) = FD_Being_Processed[Berth](t - dt) +
(FD_Enter_BTH[Berth] - FD_Depart[Berth]) * dt
INIT FD_Being_Processed[Berth] = 0
INFLOWS:
FD_Enter_BTH[BTH1] = IF (Dec_on_Numb_of_BTH[BTH1] = 1)THEN (
FD_Enter_CT / Numb_of_BTH_Used)ELSE (0)
FD_Enter_BTH[BTH2] = IF (Dec_on_Numb_of_BTH[BTH2] = 1)THEN (
FD_Enter_CT / Numb_of_BTH_Used)ELSE (0)
FD_Enter_BTH[BTH3] = IF (Dec_on_Numb_of_BTH[BTH3] = 1)THEN (
FD_Enter_CT / Numb_of_BTH_Used)ELSE (0)
FD_Enter_BTH[BTH4] = IF (Dec_on_Numb_of_BTH[BTH4] = 1)THEN (
FD_Enter_CT / Numb_of_BTH_Used)ELSE (0)
FD_Enter_BTH[BTH5] = IF (Dec_on_Numb_of_BTH[BTH5] = 1)THEN (
FD_Enter_CT / Numb_of_BTH_Used)ELSE (0)
FD_Enter_BTH[BTH6] = IF (Dec_on_Numb_of_BTH[BTH6] = 1)THEN (
FD_Enter_CT / Numb_of_BTH_Used)ELSE (0)
FD_Enter_BTH[BTH7] = IF (Dec_on_Numb_of_BTH[BTH7] = 1)THEN (
FD_Enter_CT / Numb_of_BTH_Used)ELSE (0)
FD_Enter_BTH[BTH8] = IF (Dec_on_Numb_of_BTH[BTH8] = 1)THEN (
FD_Enter_CT / Numb_of_BTH_Used)ELSE (0)
FD_Enter_BTH[BTH9] = IF (Dec_on_Numb_of_BTH[BTH9] = 1)THEN (
FD_Enter_CT / Numb_of_BTH_Used)ELSE (0)
FD_Enter_BTH[BTH10] = IF (Dec_on_Numb_of_BTH[BTH10] = 1)THEN (
FD_Enter_CT / Numb_of_BTH_Used)ELSE (0)
FD_Enter_BTH[BTH11] = IF (Dec_on_Numb_of_BTH[BTH11] = 1)THEN (
FD_Enter_CT / Numb_of_BTH_Used)ELSE (0)
FD_Enter_BTH[BTH12] = IF (Dec_on_Numb_of_BTH[BTH12] = 1)THEN (
FD_Enter_CT / Numb_of_BTH_Used)ELSE (0)
FD_Enter_BTH[BTH13] = IF (Dec_on_Numb_of_BTH[BTH13] = 1)THEN (
FD_Enter_CT / Numb_of_BTH_Used)ELSE (0)

FD_Enter_BTH[BTH14] = IF (Dec_on_Numb_of_BTH[BTH14] = 1)THEN (
FD_Enter_CT / Numb_of_BTH_Used)ELSE (0)
FD_Enter_BTH[BTH15] = IF (Dec_on_Numb_of_BTH[BTH15] = 1)THEN (
FD_Enter_CT / Numb_of_BTH_Used)ELSE (0)
OUTFLOWS:
FD_Depart[Berth] = CONVEYOR OUTFLOW
TRANSIT TIME = Ave_Service_Time

FD_wt_for_BTH(t) = FD_wt_for_BTH(t - dt) + (FD_Arrival - FD_Enter_CT) * dt
INIT FD_wt_for_BTH = 0
INFLOWS:
FD_Arrival = ROUND (Sorting_ML_&_FD * (Feeder_%_of_Arrival / 100))
OUTFLOWS:
FD_Enter_CT = FD_Entering_Rate

ML_Being_Processed[Berth](t) = ML_Being_Processed[Berth](t - dt) +
(ML_Enter_BTH[Berth] - ML_Depart[Berth]) * dt
INIT ML_Being_Processed[Berth] = 0
INFLOWS:
ML_Enter_BTH[BTH1] = IF (Dec_on_Numb_of_BTH[BTH1] = 1)THEN (
ML_Enter_CT / Numb_of_BTH_Used)ELSE (0)
ML_Enter_BTH[BTH2] = IF (Dec_on_Numb_of_BTH[BTH2] = 1)THEN (
ML_Enter_CT / Numb_of_BTH_Used)ELSE (0)
ML_Enter_BTH[BTH3] = IF (Dec_on_Numb_of_BTH[BTH3] = 1)THEN (
ML_Enter_CT / Numb_of_BTH_Used)ELSE (0)
ML_Enter_BTH[BTH4] = IF (Dec_on_Numb_of_BTH[BTH4] = 1)THEN (
ML_Enter_CT / Numb_of_BTH_Used)ELSE (0)
ML_Enter_BTH[BTH5] = IF (Dec_on_Numb_of_BTH[BTH5] = 1)THEN (
ML_Enter_CT / Numb_of_BTH_Used)ELSE (0)
ML_Enter_BTH[BTH6] = IF (Dec_on_Numb_of_BTH[BTH6] = 1)THEN (
ML_Enter_CT / Numb_of_BTH_Used)ELSE (0)
ML_Enter_BTH[BTH7] = IF (Dec_on_Numb_of_BTH[BTH7] = 1)THEN (
ML_Enter_CT / Numb_of_BTH_Used)ELSE (0)
ML_Enter_BTH[BTH8] = IF (Dec_on_Numb_of_BTH[BTH8] = 1)THEN (
ML_Enter_CT / Numb_of_BTH_Used)ELSE (0)
ML_Enter_BTH[BTH9] = IF (Dec_on_Numb_of_BTH[BTH9] = 1)THEN (
ML_Enter_CT / Numb_of_BTH_Used)ELSE (0)
ML_Enter_BTH[BTH10] = IF (Dec_on_Numb_of_BTH[BTH10] = 1)THEN (
ML_Enter_CT / Numb_of_BTH_Used)ELSE (0)
ML_Enter_BTH[BTH11] = IF (Dec_on_Numb_of_BTH[BTH11] = 1)THEN (
ML_Enter_CT / Numb_of_BTH_Used)ELSE (0)
ML_Enter_BTH[BTH12] = IF (Dec_on_Numb_of_BTH[BTH12] = 1)THEN (
ML_Enter_CT / Numb_of_BTH_Used)ELSE (0)
ML_Enter_BTH[BTH13] = IF (Dec_on_Numb_of_BTH[BTH13] = 1)THEN (
ML_Enter_CT / Numb_of_BTH_Used)ELSE (0)
ML_Enter_BTH[BTH14] = IF (Dec_on_Numb_of_BTH[BTH14] = 1)THEN (
ML_Enter_CT / Numb_of_BTH_Used)ELSE (0)
ML_Enter_BTH[BTH15] = IF (Dec_on_Numb_of_BTH[BTH15] = 1)THEN (
ML_Enter_CT / Numb_of_BTH_Used)ELSE (0)
OUTFLOWS:

ML_Depart[Berth] = CONVEYOR OUTFLOW
TRANSIT TIME = Ave_Service_Time

ML_wt_for_BTH(t) = ML_wt_for_BTH(t - dt) + (ML_Arrival - ML_Enter_CT) * dt
INIT ML_wt_for_BTH = 0
INFLOWS:
ML_Arrival = ROUND (Sorting_ML_&_FD * (MainLine_%_of_Arrival / 100))
OUTFLOWS:
ML_Enter_CT = ML_Entering_Rate

Noname_384(t) = Noname_384(t - dt) + (Noname_383) * dt
INIT Noname_384 = 0
INFLOWS:
Noname_383 = ML_Q

Noname_386(t) = Noname_386(t - dt) + (Noname_385) * dt
INIT Noname_386 = 0
INFLOWS:
Noname_385 = FD_Q

Numb_of_FD_Enter_CT(t) = Numb_of_FD_Enter_CT(t - dt) + (Noname_147) * dt
INIT Numb_of_FD_Enter_CT = 0
INFLOWS:
Noname_147 = FD_Arrival

Numb_of_ML_Enter_CT(t) = Numb_of_ML_Enter_CT(t - dt) + (Noname_143) * dt
INIT Numb_of_ML_Enter_CT = 0
INFLOWS:
Noname_143 = ML_Arrival

Sorting_ML_&_FD(t) = Sorting_ML_&_FD(t - dt) + (Vessel_Arrival - ML_Arrival - FD_Arrival) * dt
INIT Sorting_ML_&_FD = 0
INFLOWS:
Vessel_Arrival = IF (Increase__Decrease_VS_Arrival_? = 1) THEN (Increase_in_Arrival)ELSE (IF (Increase__Decrease_VS_Arrival_? = 0) THEN (Decrease_in_Arrival)ELSE (0))
OUTFLOWS:
ML_Arrival = ROUND (Sorting_ML_&_FD * (MainLine_%_of_Arrival / 100))
FD_Arrival = ROUND (Sorting_ML_&_FD * (Feeder_%_of_Arrival / 100))
Ave_Arrival_Rate = ENDVAL (Base_Case_Arrival)/Days
Ave_VS_in_Q_at_TIME = Total_Vessel_Queuing /TIME
Ave_VS_Q_time_at_TIME = Total_VS_Queue_Time/TIME
b_2[BTH1] = Numb_of_BTH_Used
b_2[BTH2] = Numb_of_BTH_Used-1
b_2[BTH3] = Numb_of_BTH_Used-2
b_2[BTH4] = Numb_of_BTH_Used-3
b_2[BTH5] = Numb_of_BTH_Used-4
b_2[BTH6] = Numb_of_BTH_Used-5
b_2[BTH7] = Numb_of_BTH_Used-6

b_2[BTH8] = Numb_of_BTH_Used-7
b_2[BTH9] = Numb_of_BTH_Used-8
b_2[BTH10] = Numb_of_BTH_Used-9
b_2[BTH11] = Numb_of_BTH_Used-10
b_2[BTH12] = Numb_of_BTH_Used-11
b_2[BTH13] = Numb_of_BTH_Used-12
b_2[BTH14] = Numb_of_BTH_Used-13
b_2[BTH15] = Numb_of_BTH_Used-14

Dec_on_Daily_or_Annual = IF (Run_Daily__Annually = 1) THEN (Vessel_Arrival_PoisDist) ELSE (IF (Run_Daily__Annually = 2) THEN (Yr_Vessel_Arrival) ELSE (0))

Dec_on_Numb_of_BTH[BTH1] = IF (b_2[BTH1] > 0) THEN (1) ELSE (0)
Dec_on_Numb_of_BTH[BTH2] = IF (b_2[BTH2] > 0) THEN (1) ELSE (0)
Dec_on_Numb_of_BTH[BTH3] = IF (b_2[BTH3] > 0) THEN (1) ELSE (0)
Dec_on_Numb_of_BTH[BTH4] = IF (b_2[BTH4] > 0) THEN (1) ELSE (0)
Dec_on_Numb_of_BTH[BTH5] = IF (b_2[BTH5] > 0) THEN (1) ELSE (0)
Dec_on_Numb_of_BTH[BTH6] = IF (b_2[BTH6] > 0) THEN (1) ELSE (0)
Dec_on_Numb_of_BTH[BTH7] = IF (b_2[BTH7] > 0) THEN (1) ELSE (0)
Dec_on_Numb_of_BTH[BTH8] = IF (b_2[BTH8] > 0) THEN (1) ELSE (0)
Dec_on_Numb_of_BTH[BTH9] = IF (b_2[BTH9] > 0) THEN (1) ELSE (0)
Dec_on_Numb_of_BTH[BTH10] = IF (b_2[BTH10] > 0) THEN (1) ELSE (0)
Dec_on_Numb_of_BTH[BTH11] = IF (b_2[BTH11] > 0) THEN (1) ELSE (0)
Dec_on_Numb_of_BTH[BTH12] = IF (b_2[BTH12] > 0) THEN (1) ELSE (0)
Dec_on_Numb_of_BTH[BTH13] = IF (b_2[BTH13] > 0) THEN (1) ELSE (0)
Dec_on_Numb_of_BTH[BTH14] = IF (b_2[BTH14] > 0) THEN (1) ELSE (0)
Dec_on_Numb_of_BTH[BTH15] = IF (b_2[BTH15] > 0) THEN (1) ELSE (0)

Decrease_in_Arrival = Dec_on_Daily_or_Annual-(Dec_on_Daily_or_Annual * (Vessel_%_of_Change / 100))
Feeder_%_of_Arrival = 100 - MainLine_%_of_Arrival
Increase__Decrease_VS_Arrival_? = 1
Increase_in_Arrival = (Dec_on_Daily_or_Annual * (Vessel_%_of_Change / 100))+Dec_on_Daily_or_Annual
Init_FD_%_Arr = 46
Init_Load_Cont_% = 0
Init_ML_%_Arr = 54
Init_Numb_of_BTH_Used = 8
Init_UnL_Cont_% = 0
Init_VS_%_Change = 0
MainLine_%_of_Arrival = 54
Noname_277[BTH1] = ENDVAL (Accum_of_ML_Serviced[BTH1])
Noname_277[BTH2] = ENDVAL (Accum_of_ML_Serviced[BTH2])
Noname_277[BTH3] = ENDVAL (Accum_of_ML_Serviced[BTH3])
Noname_277[BTH4] = ENDVAL (Accum_of_ML_Serviced[BTH4])
Noname_277[BTH5] = ENDVAL (Accum_of_ML_Serviced[BTH5])
Noname_277[BTH6] = ENDVAL (Accum_of_ML_Serviced[BTH6])
Noname_277[BTH7] = ENDVAL (Accum_of_ML_Serviced[BTH7])
Noname_277[BTH8] = ENDVAL (Accum_of_ML_Serviced[BTH8])

Noname_277[BTH9] = ENDVAL (Accum_of_ML_Serviced[BTH9])
Noname_277[BTH10] = ENDVAL (Accum_of_ML_Serviced[BTH10])
Noname_277[BTH11] = ENDVAL (Accum_of_ML_Serviced[BTH11])
Noname_277[BTH12] = ENDVAL (Accum_of_ML_Serviced[BTH12])
Noname_277[BTH13] = ENDVAL (Accum_of_ML_Serviced[BTH13])
Noname_277[BTH14] = ENDVAL (Accum_of_ML_Serviced[BTH14])
Noname_277[BTH15] = ENDVAL (Accum_of_ML_Serviced[BTH15])
Noname_278[BTH1] = ENDVAL (Accum_of_FD_Serviced[BTH1])
Noname_278[BTH2] = ENDVAL (Accum_of_FD_Serviced[BTH2])
Noname_278[BTH3] = ENDVAL (Accum_of_FD_Serviced[BTH3])
Noname_278[BTH4] = ENDVAL (Accum_of_FD_Serviced[BTH4])
Noname_278[BTH5] = ENDVAL (Accum_of_FD_Serviced[BTH5])
Noname_278[BTH6] = ENDVAL (Accum_of_FD_Serviced[BTH6])
Noname_278[BTH7] = ENDVAL (Accum_of_FD_Serviced[BTH7])
Noname_278[BTH8] = ENDVAL (Accum_of_FD_Serviced[BTH8])
Noname_278[BTH9] = ENDVAL (Accum_of_FD_Serviced[BTH9])
Noname_278[BTH10] = ENDVAL (Accum_of_FD_Serviced[BTH10])
Noname_278[BTH11] = ENDVAL (Accum_of_FD_Serviced[BTH11])
Noname_278[BTH12] = ENDVAL (Accum_of_FD_Serviced[BTH12])
Noname_278[BTH13] = ENDVAL (Accum_of_FD_Serviced[BTH13])
Noname_278[BTH14] = ENDVAL (Accum_of_FD_Serviced[BTH14])
Noname_278[BTH15] = ENDVAL (Accum_of_FD_Serviced[BTH15])
Noname_279[BTH1] = Noname_277[BTH1]+Noname_278[BTH1]
Noname_279[BTH2] = Noname_277[BTH2]+Noname_278[BTH2]
Noname_279[BTH3] = Noname_277[BTH3]+Noname_278[BTH3]
Noname_279[BTH4] = Noname_277[BTH4]+Noname_278[BTH4]
Noname_279[BTH5] = Noname_277[BTH5]+Noname_278[BTH5]
Noname_279[BTH6] = Noname_277[BTH6]+Noname_278[BTH6]
Noname_279[BTH7] = Noname_277[BTH7]+Noname_278[BTH7]
Noname_279[BTH8] = Noname_277[BTH8]+Noname_278[BTH8]
Noname_279[BTH9] = Noname_277[BTH9]+Noname_278[BTH9]
Noname_279[BTH10] = Noname_277[BTH10]+Noname_278[BTH10]
Noname_279[BTH11] = Noname_277[BTH11]+Noname_278[BTH11]
Noname_279[BTH12] = Noname_277[BTH12]+Noname_278[BTH12]
Noname_279[BTH13] = Noname_277[BTH13]+Noname_278[BTH13]
Noname_279[BTH14] = Noname_277[BTH14]+Noname_278[BTH14]
Noname_279[BTH15] = Noname_277[BTH15]+Noname_278[BTH15]
Numb_of_FD_Pro = ARRAYSUM(FD_Being_Processed[*])
Numb_of_ML_Pro = ARRAYSUM(ML_Being_Processed[*])
Numb_of_VS_in_Process = Numb_of_ML_Pro+Numb_of_FD_Pro
Run_Daily__Annually = 1
Total_FD_Served = ROUND (ARRAYSUM(Noname_278[*]))
Total_Feeder_in_Q = Noname_386
Total_MainLine_in_Q = Noname_384
Total_ML_Served = ROUND (ARRAYSUM(Noname_277[*]))
Total_Q_time_for_FD = Total_Feeder_in_Q* 0.1042
Total_Q_time_for_ML = Total_MainLine_in_Q* 0.1042
Total_Vessel_Enter_CT = Numb_of_ML_Enter_CT+Numb_of_FD_Enter_CT
Total_Vessel_Queuing = Total_Feeder_in_Q+Total_MainLine_in_Q
Total_Vessel_Served = Total_FD_Served+Total_ML_Served

Total_VS_Queue_Time = Total_Q_time_for_ML+Total_Q_time_for_FD
Vessel_%_of_Change = 0
Vessel_Arrival_PoisDist = POISSON (9.0855 , 28)
Vessel_stil_in_Queue = ML_wt_for_BTH+ FD_wt_for_BTH
Yr_Vessel_Arrival = GRAPH(TIME)
(1.00, 2734), (2.00, 2622), (3.00, 2691), (4.00, 2952), (5.00, 3146), (6.00, 3200), (7.00, 3300), (8.00, 3400), (9.00, 3600), (10.0, 4000)

CONTAINER YARD: UnLoading and Loading Process
Accum_Cont_Loaded[Berth](t) = Accum_Cont_Loaded[Berth](t - dt) + (Cont_Loading_Complete[Berth]) * dt
INIT Accum_Cont_Loaded[Berth] = 0
INFLOWS:
Cont_Loading_Complete[BTH1] = CONVEYOR OUTFLOW
    TRANSIT TIME = Cont_Load_Pro[BTH1]
Cont_Loading_Complete[BTH2] = CONVEYOR OUTFLOW
    TRANSIT TIME = Cont_Load_Pro[BTH2]
Cont_Loading_Complete[BTH3] = CONVEYOR OUTFLOW
    TRANSIT TIME = Cont_Load_Pro[BTH3]
Cont_Loading_Complete[BTH4] = CONVEYOR OUTFLOW
    TRANSIT TIME = Cont_Load_Pro[BTH4]
Cont_Loading_Complete[BTH5] = CONVEYOR OUTFLOW
    TRANSIT TIME = Cont_Load_ProT[BTH5]
Cont_Loading_Complete[BTH6] = CONVEYOR OUTFLOW
    TRANSIT TIME = Cont_Load_Pro[BTH6]
Cont_Loading_Complete[BTH7] = CONVEYOR OUTFLOW
    TRANSIT TIME = Cont_Load_Pro[BTH7]
Cont_Loading_Complete[BTH8] = CONVEYOR OUTFLOW
    TRANSIT TIME = Cont_Load_Pro[BTH8]
Cont_Loading_Complete[BTH9] = CONVEYOR OUTFLOW
    TRANSIT TIME = Cont_Load_ProV[BTH9]
Cont_Loading_Complete[BTH10] = CONVEYOR OUTFLOW
    TRANSIT TIME = Cont_Load_Pro[BTH10]
Cont_Loading_Complete[BTH11] = CONVEYOR OUTFLOW
    TRANSIT TIME = Cont_Load_ProX[BTH11]
Cont_Loading_Complete[BTH12] = CONVEYOR OUTFLOW
    TRANSIT TIME = Cont_Load_Pro[BTH12]
Cont_Loading_Complete[BTH13] = CONVEYOR OUTFLOW
    TRANSIT TIME = Cont_Load_Pro[BTH13]
Cont_Loading_Complete[BTH14] = CONVEYOR OUTFLOW
    TRANSIT TIME = Cont_Load_ProQ[BTH14]
Cont_Loading_Complete[BTH15] = CONVEYOR OUTFLOW
    TRANSIT TIME = Cont_Load_Pro[BTH15]

Accum_Cont_UnL[Berth](t) = Accum_Cont_UnL[Berth](t - dt) + (Noname_177[Berth]) * dt
INIT Accum_Cont_UnL[Berth] = 0
INFLOWS:
Noname_177[BTH1] = Transfering_UnL_Cont_to_BLK[BTH1]
Noname_177[BTH2] = Transfering_UnL_Cont_to_BLK[BTH2]

Noname_177[BTH3] = Transfering_UnL_Cont_to_BLK[BTH3]
Noname_177[BTH4] = Transfering_UnL_Cont_to_BLK[BTH4]
Noname_177[BTH5] = Transfering_UnL_Cont_to_BLK[BTH5]
Noname_177[BTH6] = Transfering_UnL_Cont_to_BLK[BTH6]
Noname_177[BTH7] = Transfering_UnL_Cont_to_BLK[BTH7]
Noname_177[BTH8] = Transfering_UnL_Cont_to_BLK[BTH8]
Noname_177[BTH9] = Transfering_UnL_Cont_to_BLK[BTH9]
Noname_177[BTH10] = Transfering_UnL_Cont_to_BLK[BTH10]
Noname_177[BTH11] = Transfering_UnL_Cont_to_BLK[BTH11]
Noname_177[BTH12] = Transfering_UnL_Cont_to_BLK[BTH12]
Noname_177[BTH13] = Transfering_UnL_Cont_to_BLK[BTH13]
Noname_177[BTH14] = Transfering_UnL_Cont_to_BLK[BTH14]
Noname_177[BTH15] = Transfering_UnL_Cont_to_BLK[BTH15]

Accum_UnL_time(t) = Accum_UnL_time(t - dt) + (Noname_245) * dt
INIT Accum_UnL_time = 0
INFLOWS:
Noname_245 = ARRAYSUM(Cont_UnL_Pro[*])
Cont_in_BLK_1(t) = Cont_in_BLK_1(t - dt) + (Cont_Enter_BLK_1[BTH1] +
Cont_Enter_BLK_1[BTH2] + Cont_Enter_BLK_1[BTH3] +
Cont_Enter_BLK_1[BTH4] + Cont_Enter_BLK_1[BTH5] +
Cont_Enter_BLK_1[BTH6] + Cont_Enter_BLK_1[BTH7] +
Cont_Enter_BLK_1[BTH8] + Cont_Enter_BLK_1[BTH9] +
Cont_Enter_BLK_1[BTH10] + Cont_Enter_BLK_1[BTH11] +
Cont_Enter_BLK_1[BTH12] + Cont_Enter_BLK_1[BTH13] +
Cont_Enter_BLK_1[BTH14] + Cont_Enter_BLK_1[BTH15] +
Cont_Enter_BLK_1[Berth] - Cont_Leave_BLK_1[BTH1] -
Cont_Leave_BLK_1[BTH2] - Cont_Leave_BLK_1[BTH3] -
Cont_Leave_BLK_1[BTH4] - Cont_Leave_BLK_1[BTH5] -
Cont_Leave_BLK_1[BTH6] - Cont_Leave_BLK_1[BTH7] -
Cont_Leave_BLK_1[BTH8] - Cont_Leave_BLK_1[BTH9] -
Cont_Leave_BLK_1[BTH10] - Cont_Leave_BLK_1[BTH11] -
Cont_Leave_BLK_1[BTH12] - Cont_Leave_BLK_1[BTH13] -
Cont_Leave_BLK_1[BTH14] - Cont_Leave_BLK_1[BTH15] –
Cont_Leave_BLK_1[Berth]) * dt

INIT Cont_in_BLK_1 = 3780
INFLOWS:
Cont_Enter_BLK_1[BTH1] = Sorting_UnL_Cont_to_BLK[BTH1]*
to_BLK_1[BTH1]
Cont_Enter_BLK_1[BTH2] = Sorting_UnL_Cont_to_BLK[BTH2]*
to_BLK_1[BTH2]
Cont_Enter_BLK_1[BTH3] = Sorting_UnL_Cont_to_BLK[BTH3]*
to_BLK_1[BTH3]
Cont_Enter_BLK_1[BTH4] = Sorting_UnL_Cont_to_BLK[BTH4]*
to_BLK_1[BTH4]
Cont_Enter_BLK_1[BTH5] = Sorting_UnL_Cont_to_BLK[BTH5]*
to_BLK_1[BTH5]
Cont_Enter_BLK_1[BTH6] = Sorting_UnL_Cont_to_BLK[BTH6]*
to_BLK_1[BTH6]

Cont_Enter_BLK_1[BTH7] = Sorting_UnL_Cont_to_BLK[BTH7]*
to_BLK_1[BTH7]
Cont_Enter_BLK_1[BTH8] = Sorting_UnL_Cont_to_BLK[BTH8]*
to_BLK_1[BTH8]
Cont_Enter_BLK_1[BTH9] = Sorting_UnL_Cont_to_BLK[BTH9]*
to_BLK_1[BTH9]
Cont_Enter_BLK_1[BTH10] = Sorting_UnL_Cont_to_BLK[BTH10]*
to_BLK_1[BTH10]
Cont_Enter_BLK_1[BTH11] = Sorting_UnL_Cont_to_BLK[BTH11]*
to_BLK_1[BTH11]
Cont_Enter_BLK_1[BTH12] = Sorting_UnL_Cont_to_BLK[BTH12]*
to_BLK_1[BTH12]
Cont_Enter_BLK_1[BTH13] = Sorting_UnL_Cont_to_BLK[BTH13]*
to_BLK_1[BTH13]
Cont_Enter_BLK_1[BTH14] = Sorting_UnL_Cont_to_BLK[BTH14]*
to_BLK_1[BTH14]
Cont_Enter_BLK_1[BTH15] = Sorting_UnL_Cont_to_BLK[BTH15]*
to_BLK_1[BTH15]

OUTFLOWS:
Cont_Leave_BLK_1[BTH1] = IF (Cont_Leaving_BLK_Rate[BTH1] > 1)
THEN (Cont_Leaving_BLK_Rate[BTH1]*frm_BLK_1[BTH1])ELSE (0)
Cont_Leave_BLK_1[BTH2] = IF (Cont_Leaving_BLK_Rate[BTH2] > 1)
THEN (Cont_Leaving_BLK_Rate[BTH2]*frm_BLK_1[BTH2])ELSE (0)
Cont_Leave_BLK_1[BTH3] = IF (Cont_Leaving_BLK_Rate[BTH3] > 1)
THEN (Cont_Leaving_BLK_Rate[BTH3]*frm_BLK_1[BTH3])ELSE (0)
Cont_Leave_BLK_1[BTH4] = IF (Cont_Leaving_BLK_Rate[BTH4] > 1)
THEN (Cont_Leaving_BLK_Rate[BTH4]*frm_BLK_1S[BTH4])ELSE (0)
Cont_Leave_BLK_1[BTH5] = IF (Cont_Leaving_BLK_Rate[BTH5] > 1)
THEN (Cont_Leaving_BLK_Rate[BTH5]*frm_BLK_1[BTH5])ELSE (0)
Cont_Leave_BLK_1[BTH6] = IF (Cont_Leaving_BLK_Rate[BTH6] > 1)
THEN (Cont_Leaving_BLK_Rate[BTH6]*frm_BLK_1[BTH6])ELSE (0)
Cont_Leave_BLK_1[BTH7] = IF (Cont_Leaving_BLK_Rate[BTH7] > 1)
THEN (Cont_Leaving_BLK_Rate[BTH7]*frm_BLK_1[BTH7])ELSE (0)
Cont_Leave_BLK_1[BTH8] = IF (Cont_Leaving_BLK_Rate[BTH8] > 1)
THEN (Cont_Leaving_BLK_Rate[BTH8]*frm_BLK_1[BTH8])ELSE (0)
Cont_Leave_BLK_1[BTH9] = IF (Cont_Leaving_BLK_Rate[BTH9] > 1)
THEN (Cont_Leaving_BLK_Rate[BTH9]*frm_BLK_1M[BTH9])ELSE (0)
Cont_Leave_BLK_1[BTH10] = IF (Cont_Leaving_BLK_Rate[BTH10] > 1)
THEN (Cont_Leaving_BLK_Rate[BTH10]*frm_BLK_1[BTH10])ELSE (0)
Cont_Leave_BLK_1[BTH11] = IF (Cont_Leaving_BLK_Rate[BTH11] > 1)
THEN (Cont_Leaving_BLK_Rate[BTH11]*frm_BLK_1[BTH11])ELSE (0)
Cont_Leave_BLK_1[BTH12] = IF (Cont_Leaving_BLK_Rate[BTH12] > 1)
THEN (Cont_Leaving_BLK_Rate[BTH12]*frm_BLK_1[BTH12])ELSE (0)
Cont_Leave_BLK_1[BTH13] = IF (Cont_Leaving_BLK_Rate[BTH13] > 1)
THEN (Cont_Leaving_BLK_Rate[BTH13]*frm_BLK_1[BTH13])ELSE (0)
Cont_Leave_BLK_1[BTH14] = IF (Cont_Leaving_BLK_Rate[BTH14] > 1)
THEN (Cont_Leaving_BLK_Rate[BTH14]*frm_BLK_1[BTH14])ELSE (0)
Cont_Leave_BLK_1[BTH15] = IF (Cont_Leaving_BLK_Rate[BTH15] > 1)
THEN (Cont_Leaving_BLK_Rate[BTH15]*frm_BLK_1[BTH15])ELSE (0)

Cont_in_BLK_10(t) = Cont_in_BLK_10(t - dt) + (Cont_Enter_BLK_10[BTH1] + Cont_Enter_BLK_10[BTH2] + Cont_Enter_BLK_10E[BTH3] + Cont_Enter_BLK_10[BTH4] + Cont_Enter_BLK_10[BTH5] + Cont_Enter_BLK_10[BTH6] + Cont_Enter_BLK_10[BTH7] + Cont_Enter_BLK_10[BTH8] + Cont_Enter_BLK_10[BTH9] + Cont_Enter_BLK_10[BTH10] + Cont_Enter_BLK_10[BTH11] + Cont_Enter_BLK_10[BTH12] + Cont_Enter_BLK_10[BTH13] + Cont_Enter_BLK_10[BTH14] + Cont_Enter_BLK_10N[BTH15] + Cont_Enter_BLK_10[Berth] - Cont_Leave_BLK_10[BTH1] - Cont_Leave_BLK_10[BTH2] - Cont_Leave_BLK_10[BTH3] - Cont_Leave_BLK_10[BTH4] - Cont_Leave_BLK_10[BTH5] - Cont_Leave_BLK_10[BTH6] - Cont_Leave_BLK_10[BTH7] - Cont_Leave_BLK_10[BTH8] - Cont_Leave_BLK_10[BTH9] - Cont_Leave_BLK_10[BTH10] - Cont_Leave_BLK_10[BTH11] - Cont_Leave_BLK_10[BTH12] - Cont_Leave_BLK_10[BTH13] - Cont_Leave_BLK_10[BTH14] - Cont_Leave_BLK_10[BTH15] - Cont_Leave_BLK_10[Berth]) * dt

INIT Cont_in_BLK_10 = 3780
INFLOWS:
Cont_Enter_BLK_10[BTH1] = Sorting_UnL_Cont_to_BLK[BTH1]* to_BLK_10[BTH1]
Cont_Enter_BLK_10[BTH2] = Sorting_UnL_Cont_to_BLK[BTH2]* to_BLK_10[BTH2]
Cont_Enter_BLK_10[BTH3] = Sorting_UnL_Cont_to_BLK[BTH3]* to_BLK_10[BTH3]
Cont_Enter_BLK_10[BTH4] = Sorting_UnL_Cont_to_BLK[BTH4]* to_BLK_10[BTH4]
Cont_Enter_BLK_10[BTH5] = Sorting_UnL_Cont_to_BLK[BTH5]* to_BLK_10[BTH5]
Cont_Enter_BLK_10[BTH6] = Sorting_UnL_Cont_to_BLK[BTH6]* to_BLK_10[BTH6]
Cont_Enter_BLK_10[BTH7] = Sorting_UnL_Cont_to_BLK[BTH7]* to_BLK_10[BTH7]
Cont_Enter_BLK_10[BTH8] = Sorting_UnL_Cont_to_BLK[BTH8]* to_BLK_10[BTH8]
Cont_Enter_BLK_10[BTH9] = Sorting_UnL_Cont_to_BLK[BTH9]* to_BLK_10[BTH9]
Cont_Enter_BLK_10[BTH10] = Sorting_UnL_Cont_to_BLK[BTH10]* to_BLK_10[BTH10]
Cont_Enter_BLK_10[BTH11] = Sorting_UnL_Cont_to_BLK[BTH11]* to_BLK_10[BTH11]
Cont_Enter_BLK_10[BTH12] = Sorting_UnL_Cont_to_BLK[BTH12]* to_BLK_10[BTH12]
Cont_Enter_BLK_10[BTH13] = Sorting_UnL_Cont_to_BLK[BTH13]* to_BLK_10[BTH13]
Cont_Enter_BLK_10[BTH14] = Sorting_UnL_Cont_to_BLK[BTH14]* to_BLK_10[BTH14]
Cont_Enter_BLK_10[BTH15] = Sorting_UnL_Cont_to_BLK[BTH15]* to_BLK_10[BTH15]

OUTFLOWS:
Cont_Leave_BLK_10[BTH1] = IF (Cont_Leaving_BLK_Rate[BTH1] > 1) THEN (Cont_Leaving_BLK_Rate[BTH1]*frm_BLK_10[BTH1])ELSE (0)
Cont_Leave_BLK_10[BTH2] = IF (Cont_Leaving_BLK_Rate[BTH2] > 1) THEN (Cont_Leaving_BLK_Rate[BTH2]*frm_BLK_10[BTH2])ELSE (0)
Cont_Leave_BLK_10[BTH3] = IF (Cont_Leaving_BLK_Rate[BTH3] > 1) THEN (Cont_Leaving_BLK_Rate[BTH3]*frm_BLK_10[BTH3])ELSE (0)
Cont_Leave_BLK_10[BTH4] = IF (Cont_Leaving_BLK_Rate[BTH4] > 1) THEN (Cont_Leaving_BLK_Rate[BTH4]*frm_BLK_10[BTH4])ELSE (0)
Cont_Leave_BLK_10[BTH5] = IF (Cont_Leaving_BLK_Rate[BTH5] > 1) THEN (Cont_Leaving_BLK_Rate[BTH5]*frm_BLK_10[BTH5])ELSE (0)
Cont_Leave_BLK_10[BTH6] = IF (Cont_Leaving_BLK_Rate[BTH6] > 1) THEN (Cont_Leaving_BLK_Rate[BTH6]*frm_BLK_10[BTH6])ELSE (0)
Cont_Leave_BLK_10[BTH7] = IF (Cont_Leaving_BLKT_Rate[BTH7] > 1) THEN (Cont_Leaving_BLK_Rate[BTH7]*frm_BLK_10[BTH7])ELSE (0)
Cont_Leave_BLK_10[BTH8] = IF (Cont_Leaving_BLK_Rate[BTH8] > 1) THEN (Cont_Leaving_BLK_Rate[BTH8]*frm_BLK_10[BTH8])ELSE (0)
Cont_Leave_BLK_10[BTH9] = IF (Cont_Leaving_BLK_Rate[BTH9] > 1) THEN (Cont_Leaving_BLK_Rate[BTH9]*frm_BLK_10[BTH9])ELSE (0)
Cont_Leave_BLK_10[BTH10] = IF (Cont_Leaving_BLK_Rate[BTH10] > 1 THEN (Cont_Leaving_BLK_Rate[BTH10]*frm_BLK_10[BTH10])ELSE (0)
Cont_Leave_BLK_10[BTH11] = IF (Cont_Leaving_BLK_Rate[BTH11] > 1 THEN (Cont_Leaving_BLK_Rate[BTH11]*frm_BLK_10[BTH11])ELSE (0)
Cont_Leave_BLK_10[BTH12] = IF (Cont_Leaving_BLK_Rate[BTH12] > 1 THEN (Cont_Leaving_BLK_Rate[BTH12]*frm_BLK_10[BTH12])ELSE (0)
Cont_Leave_BLK_10[BTH13] = IF (Cont_Leaving_BLK_Rate[BTH13] > 1 THEN (Cont_Leaving_BLK_Rate[BTH13]*frm_BLK_10[BTH13])ELSE (0)
Cont_Leave_BLK_10[BTH14] = IF (Cont_Leaving_BLK_Rate[BTH14] > 1 THEN (Cont_Leaving_BLK_Rate[BTH14]*frm_BLK_10[BTH14])ELSE (0)
Cont_Leave_BLK_10[BTH15] = IF (Cont_Leaving_BLK_Rate[BTH15] > 1 THEN (Cont_Leaving_BLK_Rate[BTH15]*frm_BLK_10[BTH15] ELSE (0)

Cont_in_BLK_11(t) = Cont_in_BLK_11(t - dt) + (Cont_Enter_BLK_11[BTH1] + Cont_Enter_BLK_11[BTH2] + Cont_Enter_BLK_11[BTH3] + Cont_Enter_BLK_11[BTH4] + Cont_Enter_BLK_11[BTH5] + Cont_Enter_BLK_11[BTH6] + Cont_Enter_BLK_11[BTH7] + Cont_Enter_BLK_11[BTH8] + Cont_Enter_BLK_11[BTH9] + Cont_Enter_BLK_11[BTH10] + Cont_Enter_BLK_11[BTH11] + Cont_Enter_BLK_11[BTH12] + Cont_Enter_BLK_11[BTH13] + Cont_Enter_BLK_11[BTH14] + Cont_Enter_BLK_11[BTH15] + Cont_Enter_BLK_11[Berth] - Cont_Leave_BLK_11[BTH1] - Cont_Leave_BLK_11[BTH2] - Cont_Leave_BLK_11[BTH3] - Cont_Leave_BLK_11[BTH4] - Cont_Leave_BLK_11[BTH5] - Cont_Leave_BLK_11[BTH6] - Cont_Leave_BLK_11[BTH7] - Cont_Leave_BLK_11[BTH8] - Cont_Leave_BLK_11[BTH9] - Cont_Leave_BLK_11[BTH10] - Cont_Leave_BLK_11[BTH11] - Cont_Leave_BLK_11[BTH12] - Cont_Leave_BLK_11[BTH13] - Cont_Leave_BLK_11[BTH14] - Cont_Leave_BLK_11[BTH15] - Cont_Leave_BLK_11[Berth]) * dt
INIT Cont_in_BLK_11 = 3780

INFLOWS:
Cont_Enter_BLK_11[BTH1] =
Sorting_UnL_Cont_to_BLK[BTH1]*to_BLK_11[BTH1]
Cont_Enter_BLK_11[BTH2] =
Sorting_UnL_Cont_to_BLK[BTH2]*to_BLK_11[BTH2]
Cont_Enter_BLK_11[BTH3] =
Sorting_UnL_Cont_to_BLK[BTH3]*to_BLK_11[BTH3]
Cont_Enter_BLK_11[BTH4] =
Sorting_UnL_Cont_to_BLK[BTH4]*to_BLK_11[BTH4]
Cont_Enter_BLK_11[BTH5] =
Sorting_UnL_Cont_to_BLK[BTH5]*to_BLK_11[BTH5]
Cont_Enter_BLK_11[BTH6] =
Sorting_UnL_Cont_to_BLK[BTH6]*to_BLK_11[BTH6]
Cont_Enter_BLK_11[BTH7] =
Sorting_UnL_Cont_to_BLK[BTH7]*to_BLK_11[BTH7]
Cont_Enter_BLK_11[BTH8] =
Sorting_UnL_Cont_to_BLK[BTH8]*to_BLK_11[BTH8]
Cont_Enter_BLK_11[BTH9] =
Sorting_UnL_Cont_to_BLK[BTH9]*to_BLK_11[BTH9]
Cont_Enter_BLK_11[BTH10] =
Sorting_UnL_Cont_to_BLK[BTH10]*to_BLK_11[BTH10]
Cont_Enter_BLK_11[BTH11] =
Sorting_UnL_Cont_to_BLK[BTH11]*to_BLK_11[BTH11]
Cont_Enter_BLK_11[BTH12] =
Sorting_UnL_Cont_to_BLK[BTH12]*to_BLK_11[BTH12]
Cont_Enter_BLK_11[BTH13] =
Sorting_UnL_Cont_to_BLK[BTH13]*to_BLK_11[BTH13]
Cont_Enter_BLK_11[BTH14] =
Sorting_UnL_Cont_to_BLK[BTH14]*to_BLK_11[BTH14]
Cont_Enter_BLK_11[BTH15] =
Sorting_UnL_Cont_to_BLK[BTH15]*to_BLK_11[BTH15]

OUTFLOWS:
Cont_Leave_BLK_11[BTH1] = IF (Cont_Leaving_BLK_Rate[BTH1] > 1) THEN
(Cont_Leaving_BLK_Rate[BTH1]*frm_BLK_11[BTH1])ELSE (0)
Cont_Leave_BLK_11[BTH2] = IF (Cont_Leaving_BLK_Rate[BTH2] > 1) THEN
(Cont_Leaving_BLK_Rate[BTH2]*frm_BLK_11[BTH2])ELSE (0)
Cont_Leave_BLK_11[BTH3] = IF (Cont_Leaving_BLK_Rate[BTH3] > 1) THEN
(Cont_Leaving_BLK_Rate[BTH3]*frm_BLK_11[BTH3])ELSE (0)
Cont_Leave_BLK_11[BTH4] = IF (Cont_Leaving_BLK_Rate[BTH4] > 1) THEN
(Cont_Leaving_BLK_Rate[BTH4]*frm_BLK_11[BTH4])ELSE (0)
Cont_Leave_BLK_11[BTH5] = IF (Cont_Leaving_BLK_Rate[BTH5] > 1) THEN
(Cont_Leaving_BLK_Rate[BTH5]*frm_BLK_11[BTH5])ELSE (0)
Cont_Leave_BLK_11[BTH6] = IF (Cont_Leaving_BLK_Rate[BTH6] > 1) THEN
(Cont_Leaving_BLK_Rate[BTH6]*frm_BLK_11[BTH5])ELSE (0)
Cont_Leave_BLK_11[BTH7] = IF (Cont_Leaving_BLK_Rate[BTH7] > 1) THEN
(Cont_Leaving_BLK_Rate[BTH7]*frm_BLK_11[BTH7])ELSE (0)
Cont_Leave_BLK_11[BTH8] = IF (Cont_Leaving_BLK_Rate[BTH8] > 1) THEN
(Cont_Leaving_BLK_Rate[BTH8]*frm_BLK_11[BTH8])ELSE (0)

Cont_Leave_BLK_11[BTH9] = IF (Cont_Leaving_BLK_Rate[BTH9] > 1) THEN (Cont_Leaving_BLK_Rate[BTH9]*frm_BLK_11[BTH9])ELSE (0)
Cont_Leave_BLK_11[BTH10] = IF (Cont_Leaving_BLK_Rate[BTH10] > 1 THEN (Cont_Leaving_BLK_Rate[BTH10]*frm_BLK_11[BTH10])ELSE (0)
Cont_Leave_BLK_11[BTH11] = IF (Cont_Leaving_BLK_Rate[BTH11] > 1 THEN (Cont_Leaving_BLK_Rate[BTH11]*frm_BLK_11[BTH11])ELSE (0)
Cont_Leave_BLK_11[BTH12] = IF (Cont_Leaving_BLK_Rate[BTH12] > 1 THEN (Cont_Leaving_BLK_Rate[BTH12]*frm_BLK_11[BTH12])ELSE (0)
Cont_Leave_BLK_11[BTH13] = IF (Cont_Leaving_BLK_Rate[BTH13] > 1 THEN (Cont_Leaving_BLK_Rate[BTH13]*frm_BLK_11[BTH13])ELSE (0)
Cont_Leave_BLK_11[BTH14] = IF (Cont_Leaving_BLK_Rate[BTH14] > 1 THEN (Cont_Leaving_BLK_Rate[BTH14]*frm_BLK_11[BTH14])ELSE (0)
Cont_Leave_BLK_11[BTH15] = IF (Cont_Leaving_BLK_Rate[BTH15] > 1 THEN (Cont_Leaving_BLK_Rate[BTH15]*frm_BLK_11[BTH15])ELSE (0)

Cont_in_BLK_12(t) = Cont_in_BLK_12(t - dt) + (Cont_Enter_BLK_12[BTH1] + Cont_Enter_BLK_12[BTH2] + Cont_Enter_BLK_12[BTH3] + Cont_Enter_BLK_12[BTH4] + Cont_Enter_BLK_12[BTH5] + Cont_Enter_BLK_12[BTH6] + Cont_Enter_BLK_12[BTH7] + Cont_Enter_BLK_12[BTH8] + Cont_Enter_BLK_12[BTH9] + Cont_Enter_BLK_12[BTH10] + Cont_Enter_BLK_12[BTH11] + Cont_Enter_BLK_12[BTH12] + Cont_Enter_BLK_12[BTH13] + Cont_Enter_BLK_12[BTH14] + Cont_Enter_BLK_12[BTH15] + Cont_Enter_BLK_12[Berth] - Cont_Leave_BLK_12[BTH1] - Cont_Leave_BLK_12[BTH2] - Cont_Leave_BLK_12[BTH3] - Cont_Leave_BLK_12[BTH4] - Cont_Leave_BLK_12[BTH5] - Cont_Leave_BLK_12[BTH6] - Cont_Leave_BLK_12[BTH7] - Cont_Leave_BLK_12[BTH8] - Cont_Leave_BLK_12[BTH9] - Cont_Leave_BLK_12[BTH10] - Cont_Leave_BLK_12[BTH11] - Cont_Leave_BLK_12[BTH12] - Cont_Leave_BLK_12[BTH13] - Cont_Leave_BLK_12[BTH14] - Cont_Leave_BLK_12[BTH15] - Cont_Leave_BLK_12[Berth]) * dt
INIT Cont_in_BLK_12 = 3780
INFLOWS:
Cont_Enter_BLK_12[BTH1] = Sorting_UnL_Cont_to_BLK[BTH1]*to_BLK_12[BTH1]
Cont_Enter_BLK_12[BTH2] = Sorting_UnL_Cont_to_BLK[BTH2]*to_BLK_12[BTH2]
Cont_Enter_BLK_12[BTH3] = Sorting_UnL_Cont_to_BLK[BTH3]*to_BLK_12[BTH3]
Cont_Enter_BLK_12[BTH4] = Sorting_UnL_Cont_to_BLK[BTH4]*to_BLK_12[BTH4]
Cont_Enter_BLK_12[BTH5] = Sorting_UnL_Cont_to_BLK[BTH5]*to_BLK_12[BTH5]
Cont_Enter_BLK_12[BTH6] = Sorting_UnL_Cont_to_BLK[BTH6]*to_BLK_12[BTH6]
Cont_Enter_BLK_12[BTH7] = Sorting_UnL_Cont_to_BLK[BTH7]*to_BLK_12[BTH7]
Cont_Enter_BLK_12[BTH8] = Sorting_UnL_Cont_to_BLK[BTH8]*to_BLK_12[BTH8]

Cont_Enter_BLK_12[BTH9] =
Sorting_UnL_Cont_to_BLK[BTH9]*to_BLK_12[BTH9]
Cont_Enter_BLK_12[BTH10] =
Sorting_UnL_Cont_to_BLK[BTH10]*to_BLK_12[BTH10]
Cont_Enter_BLK_12[BTH11] =
Sorting_UnL_Cont_to_BLK[BTH11]*to_BLK_12[BTH11]
Cont_Enter_BLK_12[BTH12] =
Sorting_UnL_Cont_to_BLK[BTH12]*to_BLK_12[BTH12]
Cont_Enter_BLK_12[BTH13] =
Sorting_UnL_Cont_to_BLK[BTH13]*to_BLK_12[BTH13]
Cont_Enter_BLK_12[BTH14] =
Sorting_UnL_Cont_to_BLK[BTH14]*to_BLK_12[BTH14]
Cont_Enter_BLK_12[BTH15] =
Sorting_UnL_Cont_to_BLK[BTH15]*to_BLK_12[BTH15]
OUTFLOWS:
Cont_Leave_BLK_12[BTH1] = IF (Cont_Leaving_BLK_Rate[BTH1] > 1) THEN
(Cont_Leaving_BLK_Rate[BTH1]*frm_BLK_12[BTH1])ELSE (0)
Cont_Leave_BLK_12[BTH2] = IF (Cont_Leaving_BLK_Rate[BTH2] > 1) THEN
(Cont_Leaving_BLK_Rate[BTH2]*frm_BLK_12[BTH2])ELSE (0)
Cont_Leave_BLK_12[BTH3] = IF (Cont_Leaving_BLK_Rate[BTH3] > 1) THEN
(Cont_Leaving_BLK_Rate[BTH3]*frm_BLK_12[BTH3])ELSE (0)
Cont_Leave_BLK_12[BTH4] = IF (Cont_Leaving_BLK_Rate[BTH4] > 1) THEN
(Cont_Leaving_BLK_Rate[BTH4]*frm_BLK_12[BTH4])ELSE (0)
Cont_Leave_BLK_12[BTH5] = IF (Cont_Leaving_BLK_Rate[BTH5] > 1) THEN
(Cont_Leaving_BLK_Rate[BTH5]*frm_BLK_12[BTH5])ELSE (0)
Cont_Leave_BLK_12[BTH6] = IF (Cont_Leaving_BLK_Rate[BTH6] > 1) THEN
(Cont_Leaving_BLK_Rate[BTH6]*frm_BLK_12[BTH6])ELSE (0)
Cont_Leave_BLK_12[BTH7] = IF (Cont_Leaving_BLK_Rate[BTH7] > 1) THEN
(Cont_Leaving_BLK_Rate[BTH7]*frm_BLK_12[BTH7])ELSE (0)
Cont_Leave_BLK_12[BTH8] = IF (Cont_Leaving_BLK_Rate[BTH8] > 1) THEN
(Cont_Leaving_BLK_Rate[BTH8]*frm_BLK_12[BTH8])ELSE (0)
Cont_Leave_BLK_12[BTH9] = IF (Cont_Leaving_BLK_Rate[BTH9] > 1) THEN
(Cont_Leaving_BLK_Rate[BTH9]*frm_BLK_12[BTH9])ELSE (0)
Cont_Leave_BLK_12[BTH10] = IF (Cont_Leaving_BLK_Rate[BTH10] > 1 THEN
(Cont_Leaving_BLK_Rate[BTH10]*frm_BLK_12[BTH10])ELSE (0)
Cont_Leave_BLK_12[BTH11] = IF (Cont_Leaving_BLK_Rate[BTH11] > 1 THEN
(Cont_Leaving_BLK_Rate[BTH11]*frm_BLK_12[BTH11])ELSE (0)
Cont_Leave_BLK_12[BTH12] = IF (Cont_Leaving_BLK_RateE[BTH12] > 1
THEN (Cont_Leaving_BLK_Rate[BTH12]*frm_BLK_12[BTH12])ELSE (0)
Cont_Leave_BLK_12[BTH13] = IF (Cont_Leaving_BLK_Rate[BTH13] > 1 THEN
(Cont_Leaving_BLK_Rate[BTH13]*frm_BLK_12[BTH13])ELSE (0)
Cont_Leave_BLK_12[BTH14] = IF (Cont_Leaving_BLK_Rate[BTH14] > 1 THEN
(Cont_Leaving_BLK_Rate[BTH14]*frm_BLK_12[BTH14])ELSE (0)
Cont_Leave_BLK_12[BTH15] = IF (Cont_Leaving_BLK_Rate[BTH15] > 1 THEN
(Cont_Leaving_BLK_Rate[BTH15]*frm_BLK_12[BTH15])ELSE (0)

Cont_in_BLK_2(t) = Cont_in_BLK_2(t - dt) + (Cont_Enter_BLK_2[BTH1] +
Cont_Enter_BLK_2[BTH2] + Cont_Enter_BLK_2[BTH3] +
Cont_Enter_BLK_2[BTH4] + Cont_Enter_BLK_2[BTH5] +
Cont_Enter_BLK_2[BTH6] + Cont_Enter_BLK_2[BTH7] +

Cont_Enter_BLK_2[BTH8] + Cont_Enter_BLK_2[BTH9] +
Cont_Enter_BLK_2[BTH10] + Cont_Enter_BLK_2[BTH11] +
Cont_Enter_BLK_2[BTH12] + Cont_Enter_BLK_2[BTH13] +
Cont_Enter_BLK_2[BTH14] + Cont_Enter_BLK_2[BTH15] +
Cont_Enter_BLK_2[Berth] - Cont_Leave_BLK_2[BTH1] -
Cont_Leave_BLK_2[BTH2] - Cont_Leave_BLK_2[BTH3] -
Cont_Leave_BLK_2[BTH4] - Cont_Leave_BLK_2[BTH5] -
Cont_Leave_BLK_2[BTH6] - Cont_Leave_BLK_2[BTH7] -
Cont_Leave_BLK_2[BTH8] - Cont_Leave_BLK_2[BTH9] -
Cont_Leave_BLK_2[BTH10] - Cont_Leave_BLK_2[BTH11] -
Cont_Leave_BLK_2[BTH12] - Cont_Leave_BLK_2[BTH13] -
Cont_Leave_BLK_2[BTH14] - Cont_Leave_BLK_2[BTH15] -
Cont_Leave_BLK_2[Berth]) * dt
INIT Cont_in_BLK_2 = 3780
INFLOWS:
Cont_Enter_BLK_2[BTH1] =
Sorting_UnL_Cont_to_BLK[BTH1]*to_BLK_2[BTH1]
Cont_Enter_BLK_2[BTH2] =
Sorting_UnL_Cont_to_BLK[BTH2]*to_BLK_2[BTH2]
Cont_Enter_BLK_2[BTH3] =
Sorting_UnL_Cont_to_BLK[BTH3]*to_BLK_2[BTH3]
Cont_Enter_BLK_2[BTH4] =
Sorting_UnL_Cont_to_BLK[BTH4]*to_BLK_2[BTH4]
Cont_Enter_BLK_2[BTH5] =
Sorting_UnL_Cont_to_BLK[BTH5]*to_BLK_2[BTH5]
Cont_Enter_BLK_2[BTH6] =
Sorting_UnL_Cont_to_BLK[BTH6]*to_BLK_2[BTH6]
Cont_Enter_BLK_2[BTH7] =
Sorting_UnL_Cont_to_BLK[BTH7]*to_BLK_2[BTH7]
Cont_Enter_BLK_2[BTH8] =
Sorting_UnL_Cont_to_BLK[BTH8]*to_BLK_2[BTH8]
Cont_Enter_BLK_2[BTH9] =
Sorting_UnL_Cont_to_BLK[BTH9]*to_BLK_2[BTH9]
Cont_Enter_BLK_2[BTH10] =
Sorting_UnL_Cont_to_BLK[BTH10]*to_BLK_2[BTH10]
Cont_Enter_BLK_2R[BTH11] =
Sorting_UnL_Cont_to_BLK[BTH11]*to_BLK_2[BTH11]
Cont_Enter_BLK_2[BTH12] =
Sorting_UnL_Cont_to_BLK[BTH12]*to_BLK_2[BTH12]
Cont_Enter_BLK_2[BTH13] =
Sorting_UnL_Cont_to_BLK[BTH13]*to_BLK_2[BTH13]
Cont_Enter_BLK_2[BTH14] =
Sorting_UnL_Cont_to_BLK[BTH14]*to_BLK_2[BTH14]
Cont_Enter_BLK_2[BTH15] =
Sorting_UnL_Cont_to_BLK[BTH15]*to_BLK_2[BTH15]

OUTFLOWS:
Cont_Leave_BLK_2[BTH1] = IF (Cont_Leaving_BLK_Rate[BTH1] > 1) THEN (
Cont_Leaving_BLK_Rate[BTH1]*frm_BLK_2[BTH1])ELSE (0)
Cont_Leave_BLK_2[BTH2] = IF (Cont_Leaving_BLK_Rate[BTH2] > 1) THEN (

Cont_Leaving_BLK_Rate[BTH2]*frm_BLK_2[BTH2])ELSE (0)
Cont_Leave_BLK_2[BTH3] = IF (Cont_Leaving_BLK_Rate[BTH3] > 1) THEN (Cont_Leaving_BLK_Rate[BTH3]*frm_BLK_2[BTH3])ELSE (0)
Cont_Leave_BLK_2[BTH4] = IF (Cont_Leaving_BLK_Rate[BTH4] > 1) THEN (Cont_Leaving_BLK_Rate[BTH4]*frm_BLK_2[BTH4])ELSE (0)
Cont_Leave_BLK_2[BTH5] = IF (Cont_Leaving_BLK_Rate[BTH5] > 1) THEN (Cont_Leaving_BLK_Rate[BTH5]*frm_BLK_2[BTH5])ELSE (0)
Cont_Leave_BLK_2[BTH6] = IF (Cont_Leaving_BLK_Rate[BTH6] > 1) THEN (Cont_Leaving_BLK_Rate[BTH6]*frm_BLK_2[BTH6])ELSE (0)
Cont_Leave_BLK_2[BTH7] = IF (Cont_Leaving_BLK_Rate[BTH7] > 1) THEN (Cont_Leaving_BLK_Rate[BTH7]*frm_BLK_2[BTH7])ELSE (0)
Cont_Leave_BLK_2[BTH8] = IF (Cont_Leaving_BLK_Rate[BTH8] > 1) THEN (Cont_Leaving_BLK_Rate[BTH8]*frm_BLK_2[BTH8])ELSE (0)
Cont_Leave_BLK_2[BTH9] = IF (Cont_Leaving_BLK_Rate[BTH9] > 1) THEN (Cont_Leaving_BLK_Rate[BTH9]*frm_BLK_2[BTH9])ELSE (0)
Cont_Leave_BLK_2[BTH10] = IF (Cont_Leaving_BLK_Rate[BTH10] > 1) THEN (Cont_Leaving_BLK_Rate[BTH10]*frm_BLK_2[BTH10])ELSE (0)
Cont_Leave_BLK_2[BTH11] = IF (Cont_Leaving_BLK_Rate[BTH11] > 1) THEN (Cont_Leaving_BLK_Rate[BTH11]*frm_BLK_2[BTH11])ELSE (0)
Cont_Leave_BLK_2[BTH12] = IF (Cont_Leaving_BLK_Rate[BTH12] > 1) THEN (Cont_Leaving_BLK_Rate[BTH12]*frm_BLK_2[BTH12])ELSE (0)
Cont_Leave_BLK_2[BTH13] = IF (Cont_Leaving_BLK_Rate[BTH13] > 1) THEN (Cont_Leaving_BLK_Rate[BTH13]*frm_BLK_2[BTH13])ELSE (0)
Cont_Leave_BLK_2[BTH14] = IF (Cont_Leaving_BLK_Rate[BTH14] > 1) THEN (Cont_Leaving_BLK_Rate[BTH14]*frm_BLK_2[BTH14])ELSE (0)
Cont_Leave_BLK_2[BTH15] = IF (Cont_Leaving_BLK_Rate[BTH15] > 1) THEN (Cont_Leaving_BLK_Rate[BTH15]*frm_BLK_2[BTH15])ELSE (0)

Cont_in_BLK_3(t) = Cont_in_BLK_3(t - dt) + (Cont_Enter_BLK_3[BTH1] + Cont_Enter_BLK_3[BTH2] + Cont_Enter_BLK_3[BTH3] + Cont_Enter_BLK_3[BTH4] + Cont_Enter_BLK_3[BTH5] + Cont_Enter_BLK_3[BTH6] + Cont_Enter_BLK_3[BTH7] + Cont_Enter_BLK_3[BTH8] + Cont_Enter_BLK_3[BTH9] + Cont_Enter_BLK_3I[BTH10] + Cont_Enter_BLK_3[BTH11] + Cont_Enter_BLK_3[BTH12] + Cont_Enter_BLK_3[BTH13] + Cont_Enter_BLK_3[BTH14] + Cont_Enter_BLK_3[BTH15] + Cont_Enter_BLK_3[Berth] - Cont_Leave_BLK_3[BTH1] - Cont_Leave_BLK_3[BTH2] - Cont_Leave_BLK_3[BTH3] - Cont_Leave_BLK_3[BTH4] - Cont_Leave_BLK_3[BTH5] - Cont_Leave_BLK_3[BTH6] - Cont_Leave_BLK_3[BTH7] - Cont_Leave_BLK_3[BTH8] - Cont_Leave_BLK_3[BTH9] - Cont_Leave_BLK_3[BTH10] - Cont_Leave_BLK_3[BTH11] - Cont_Leave_BLK_3[BTH12] - Cont_Leave_BLK_3[BTH13] - Cont_Leave_BLK_3[BTH14] - Cont_Leave_BLK_3[BTH15] - Cont_Leave_BLK_3[Berth]) * dt
INIT Cont_in_BLK_3 = 3780
INFLOWS:
Cont_Enter_BLK_3[BTH1] = Sorting_UnL_Cont_to_BLK[BTH1]*to_BLK_3[BTH1]

Cont_Enter_BLK_3[BTH2] =
Sorting_UnL_Cont_to_BLK[BTH2]*to_BLK_3[BTH2]
Cont_Enter_BLK_3[BTH3] =
Sorting_UnL_Cont_to_BLK[BTH3]*to_BLK_3[BTH3]
Cont_Enter_BLK_3[BTH4] =
Sorting_UnL_Cont_to_BLK[BTH4]*to_BLK_3[BTH4]
Cont_Enter_BLK_3[BTH5] =
Sorting_UnL_Cont_to_BLK[BTH5]*to_BLK_3[BTH5]
Cont_Enter_BLK_3[BTH6] =
Sorting_UnL_Cont_to_BLK[BTH6]*to_BLK_3[BTH6]
Cont_Enter_BLK_3T[BTH7] =
Sorting_UnL_Cont_to_BLK[BTH7]*to_BLK_3[BTH7]
Cont_Enter_BLK_3[BTH8] =
Sorting_UnL_Cont_to_BLK[BTH8]*to_BLK_3[BTH8]
Cont_Enter_BLK_3[BTH9] =
Sorting_UnL_Cont_to_BLK[BTH9]*to_BLK_3[BTH9]
Cont_Enter_BLK_3[BTH10] =
Sorting_UnL_Cont_to_BLK[BTH10]*to_BLK_3[BTH10]
Cont_Enter_BLK_3[BTH11] =
Sorting_UnL_Cont_to_BLK[BTH11]*to_BLK_3[BTH11]
Cont_Enter_BLK_3[BTH12] =
Sorting_UnL_Cont_to_BLK[BTH12]*to_BLK_3[BTH12]
Cont_Enter_BLK_3A[BTH13] =
Sorting_UnL_Cont_to_BLK[BTH13]*to_BLK_3[BTH13]
Cont_Enter_BLK_3[BTH14] =
Sorting_UnL_Cont_to_BLK[BTH14]*to_BLK_3[BTH14]
Cont_Enter_BLK_3[BTH15] =
Sorting_UnL_Cont_to_BLK[BTH15]*to_BLK_3[BTH15]

OUTFLOWS:
Cont_Leave_BLK_3[BTH1] = IF (Cont_Leaving_BLK_Rate[BTH1] > 1) THEN (Cont_Leaving_BLK_Rate[BTH1]*frm_BLK_3[BTH1])ELSE (0)
Cont_Leave_BLK_3[BTH2] = IF (Cont_Leaving_BLK_Rate[BTH2] > 1) THEN (Cont_Leaving_BLK_Rate[BTH2]*frm_BLK_3[BTH2])ELSE (0)
Cont_Leave_BLK_3[BTH3] = IF (Cont_Leaving_BLK_Rate[BTH3] > 1) THEN (Cont_Leaving_BLK_Rate[BTH3]*frm_BLK_3[BTH3])ELSE (0)
Cont_Leave_BLK_3[BTH4] = IF (Cont_Leaving_BLK_Rate[BTH4] > 1) THEN (Cont_Leaving_BLK_Rate[BTH4]*frm_BLK_3[BTH4])ELSE (0)
Cont_Leave_BLK_3[BTH5] = IF (Cont_Leaving_BLK_Rate[BTH5] > 1) THEN (Cont_Leaving_BLK_Rate[BTH5]*frm_BLK_3[BTH5])ELSE (0)
Cont_Leave_BLK_3[BTH6] = IF (Cont_Leaving_BLK_Rate[BTH6] > 1) THEN (Cont_Leaving_BLK_Rate[BTH6N]*frm_BLK_3[BTH6])ELSE (0)
Cont_Leave_BLK_3[BTH7] = IF (Cont_Leaving_BLK_Rate[BTH7] > 1) THEN (Cont_Leaving_BLK_Rate[BTH7]*frm_BLK_3[BTH7])ELSE (0)
Cont_Leave_BLK_3[BTH8] = IF (Cont_Leaving_BLK_Rate[BTH8] > 1) THEN (Cont_Leaving_BLK_Rate[BTH8]*frm_BLK_3[BTH8])ELSE (0)
Cont_Leave_BLK_3[BTH9] = IF (Cont_Leaving_BLK_Rate[BTH9] > 1) THEN (Cont_Leaving_BLK_Rate[BTH9]*frm_BLK_3[BTH9])ELSE (0)
Cont_Leave_BLK_3[BTH10] = IF (Cont_Leaving_BLK_Rate[BTH10] > 1) THEN (Cont_Leaving_BLK_Rate[BTH10]*frm_BLK_3[BTH10])ELSE (0)

Cont_Leave_BLK_3[BTH11] = IF (Cont_Leaving_BLK_Rate[BTH11] > 1) THEN (Cont_Leaving_BLK_Rate[BTH11]*frm_BLK_3[BTH11])ELSE (0)
Cont_Leave_BLK_3[BTH12] = IF (Cont_Leaving_BLK_Rate[BTH12] > 1) THEN (Cont_Leaving_BLK_Rate[BTH12]*frm_BLK_3[BTH12])ELSE (0)
Cont_Leave_BLK_3[BTH13] = IF (Cont_Leaving_BLK_Rate[BTH13] > 1) THEN (Cont_Leaving_BLK_Rate[BTH13]*frm_BLK_3[BTH13])ELSE (0)
Cont_Leave_BLK_3[BTH14] = IF (Cont_Leaving_BLK_Rate[BTH14] > 1) THEN (Cont_Leaving_BLK_Rate[BTH14]*frm_BLK_3[BTH14])ELSE (0)
Cont_Leave_BLK_3[BTH15] = IF (Cont_Leaving_BLK_Rate[BTH15] > 1) THEN (Cont_Leaving_BLK_Rate[BTH15]*frm_BLK_3[BTH15])ELSE (0)

Cont_in_BLK_4(t) = Cont_in_BLK_4(t - dt) + (Cont_Enter_BLK_4[BTH1] + Cont_Enter_BLK_4[BTH2] + Cont_Enter_BLK_4[BTH3] + Cont_Enter_BLK_4[BTH4] + Cont_Enter_BLK_4[BTH5] + Cont_Enter_BLK_4[BTH6] + Cont_Enter_BLK_4[BTH7] + Cont_Enter_BLK_4[BTH8] + Cont_Enter_BLK_4[BTH9] + Cont_Enter_BLK_4[BTH10] + Cont_Enter_BLK_4[BTH11] + Cont_Enter_BLK_4[BTH12] + Cont_Enter_BLK_4[BTH13] + Cont_Enter_BLK_4[BTH14] + Cont_Enter_BLK_4[BTH15] + Cont_Enter_BLK_4[Berth] - Cont_Leave_BLK_4[BTH1] - Cont_Leave_BLK_4[BTH2] - Cont_Leave_BLK_4[BTH3] - Cont_Leave_BLK_4[BTH4] - Cont_Leave_BLK_4[BTH5] - Cont_Leave_BLK_4[BTH6] - Cont_Leave_BLK_4[BTH7] - Cont_Leave_BLK_4[BTH8] - Cont_Leave_BLK_4[BTH9] - Cont_Leave_BLK_4[BTH10] - Cont_Leave_BLK_4[BTH11] - Cont_Leave_BLK_4[BTH12] - Cont_Leave_BLK_4[BTH13] - Cont_Leave_BLK_4[BTH14] - Cont_Leave_BLK_4[BTH15] - Cont_Leave_BLK_4[Berth]) * dt
INIT Cont_in_BLK_4 = 3780M
INFLOWS:
Cont_Enter_BLK_4[BTH1] = Sorting_UnL_Cont_to_BLK[BTH1]*to_BLK_4[BTH1]
Cont_Enter_BLK_4[BTH2] = Sorting_UnL_Cont_to_BLK[BTH2]*to_BLK_4[BTH2]
Cont_Enter_BLK_4[BTH3] = Sorting_UnL_Cont_to_BLK[BTH3]*to_BLK_4[BTH3]
Cont_Enter_BLK_4[BTH4] = Sorting_UnL_Cont_to_BLK[BTH4]*to_BLK_4[BTH4]
Cont_Enter_BLK_4[BTH5] = Sorting_UnL_Cont_to_BLK[BTH5]*to_BLK_4[BTH5]
Cont_Enter_BLK_4[BTH6] = Sorting_UnL_Cont_to_BLK[BTH6]*to_BLK_4[BTH6]
Cont_Enter_BLK_4[BTH7] = Sorting_UnL_Cont_to_BLK[BTH7]*to_BLK_4[BTH7]
Cont_Enter_BLK_4[BTH8] = Sorting_UnL_Cont_to_BLK[BTH8]*to_BLK_4[BTH8]
Cont_Enter_BLK_4[BTH9] = Sorting_UnL_Cont_to_BLK[BTH9]*to_BLK_4[BTH9]
Cont_Enter_BLK_4[BTH10] = Sorting_UnL_Cont_to_BLK[BTH10]*to_BLK_4[BTH10]

Cont_Enter_BLK_4[BTH11] =
Sorting_UnL_Cont_to_BLK[BTH11]*to_BLK_4[BTH11]
Cont_Enter_BLK_4[BTH12] =
Sorting_UnL_Cont_to_BLK[BTH12]*to_BLK_4[BTH12]
Cont_Enter_BLK_4[BTH13] =
Sorting_UnL_Cont_to_BLK[BTH13]*to_BLK_4[BTH13]
Cont_Enter_BLK_4[BTH14] =
Sorting_UnL_Cont_to_BLK[BTH14]*to_BLK_4[BTH14]
Cont_Enter_BLK_4[BTH15] =
Sorting_UnL_Cont_to_BLK[BTH15]*to_BLK_4[BTH15]

OUTFLOWS:
Cont_Leave_BLK_4[BTH1] = IF (Cont_Leaving_BLK_Rate[BTH1] > 1) THEN (Cont_Leaving_BLK_Rate[BTH1]*frm_BLK_4[BTH1])ELSE (0)
Cont_Leave_BLK_4[BTH2] = IF (Cont_Leaving_BLK_Rate[BTH2] > 1) THEN (Cont_Leaving_BLK_Rate[BTH2]*frm_BLK_4[BTH2])ELSE (0)
Cont_Leave_BLK_4[BTH3] = IF (Cont_Leaving_BLK_Rate[BTH3] > 1) THEN (Cont_Leaving_BLK_Rate[BTH3]*frm_BLK_4[BTH3])ELSE (0)
Cont_Leave_BLK_4[BTH4] = IF (Cont_Leaving_BLK_Rate[BTH4] > 1) THEN (Cont_Leaving_BLK_Rate[BTH4]*frm_BLK_4[BTH4])ELSE (0)
Cont_Leave_BLK_4[BTH5] = IF (Cont_Leaving_BLK_Rate[BTH5] > 1) THEN (Cont_Leaving_BLK_Rate[BTH5]*frm_BLK_4[BTH5])ELSE (0)
Cont_Leave_BLK_4[BTH6] = IF (Cont_Leaving_BLK_Rate[BTH6] > 1) THEN (Cont_Leaving_BLK_Rate[BTH6]*frm_BLK_4[BTH6])ELSE (0)
Cont_Leave_BLK_4[BTH7] = IF (Cont_Leaving_BLK_Rate[BTH7] > 1) THEN (Cont_Leaving_BLK_Rate[BTH7]*frm_BLK_4[BTH7])ELSE (0)
Cont_Leave_BLK_4[BTH8] = IF (Cont_Leaving_BLK_Rate[BTH8] > 1) THEN (Cont_Leaving_BLK_Rate[BTH8]*frm_BLK_4[BTH8])ELSE (0)
Cont_Leave_BLK_4[BTH9] = IF (Cont_Leaving_BLK_Rate[BTH9] > 1) THEN (Cont_Leaving_BLK_Rate[BTH9]*frm_BLK_4[BTH9])ELSE (0)
Cont_Leave_BLK_4[BTH10] = IF (Cont_Leaving_BLK_Rate[BTH10] > 1) THEN (Cont_Leaving_BLK_Rate[BTH10]*frm_BLK_4[BTH10])ELSE (0)
Cont_Leave_BLK_4[BTH11] = IF (Cont_Leaving_BLK_Rate[BTH11] > 1) THEN (Cont_Leaving_BLK_Rate[BTH11]*frm_BLK_4[BTH11])ELSE (0)
Cont_Leave_BLK_4[BTH12] = IF (Cont_Leaving_BLK_Rate[BTH12] > 1) THEN (Cont_Leaving_BLK_Rate[BTH12]*frm_BLK_4[BTH12])ELSE (0)
Cont_Leave_BLK_4[BTH13] = IF (Cont_Leaving_BLK_Rate[BTH13] > 1) THEN (Cont_Leaving_BLK_Rate[BTH13]*frm_BLK_4[BTH13])ELSE (0)
Cont_Leave_BLK_4[BTH14] = IF (Cont_Leaving_BLK_Rate[BTH14] > 1) THEN (Cont_Leaving_BLK_Rate[BTH14]*frm_BLK_4[BTH14])ELSE (0)
Cont_Leave_BLK_4[BTH15] = IF (Cont_Leaving_BLK_Rate[BTH15] > 1) THEN (Cont_Leaving_BLK_Rate[BTH15]*frm_BLK_4[BTH15])ELSE (0)

Cont_in_BLK_5(t) = Cont_in_BLK_5(t - dt) + (Cont_Enter_BLK_5[BTH1] + Cont_Enter_BLK_5[BTH2] + Cont_Enter_BLK_5[BTH3] + Cont_Enter_BLK_5[BTH4] + Cont_Enter_BLK_5[BTH5] + Cont_Enter_BLK_5[BTH6] + Cont_Enter_BLK_5[BTH7] + Cont_Enter_BLK_5[BTH8] + Cont_Enter_BLK_5[BTH9] + Cont_Enter_BLK_5[BTH10] + Cont_Enter_BLK_5[BTH11] + Cont_Enter_BLK_5[BTH12] + Cont_Enter_BLK_5[BTH13] +

Cont_Enter_BLK_5[BTH14] + Cont_Enter_BLK_5[BTH15] +
Cont_Enter_BLK_5[Berth] - Cont_Leave_BLK_5[BTH1] -
Cont_Leave_BLK_5[BTH2] - Cont_Leave_BLK_5[BTH3] -
Cont_Leave_BLK_5[BTH4] - Cont_Leave_BLK_5[BTH5] -
Cont_Leave_BLK_5[BTH6] - Cont_Leave_BLK_5[BTH7] -
Cont_Leave_BLK_5[BTH8] - Cont_Leave_BLK_5[BTH9] -
Cont_Leave_BLK_5[BTH10] - Cont_Leave_BLK_5[BTH11] -
Cont_Leave_BLK_5[BTH12] - Cont_Leave_BLK_5[BTH13] -
Cont_Leave_BLK_5[BTH14] - Cont_Leave_BLK_5[BTH15] -
Cont_Leave_BLK_5[Berth]) * dt
INIT Cont_in_BLK_5 = 3780
INFLOWS:
Cont_Enter_BLK_5[BTH1] =
Sorting_UnL_Cont_to_BLK[BTH1]*to_BLK_5[BTH1]
Cont_Enter_BLK_5[BTH2] =
Sorting_UnL_Cont_to_BLK[BTH2]*to_BLK_5[BTH2]
Cont_Enter_BLK_5[BTH3] =
Sorting_UnL_Cont_to_BLK[BTH3]*to_BLK_5[BTH3]
Cont_Enter_BLK_5[BTH4] =
Sorting_UnL_Cont_to_BLK[BTH4]*to_BLK_5[BTH4]
Cont_Enter_BLK_5E[BTH5] =
Sorting_UnL_Cont_to_BLK[BTH5]*to_BLK_5[BTH5]
Cont_Enter_BLK_5[BTH6] =
Sorting_UnL_Cont_to_BLK[BTH6]*to_BLK_5[BTH6]
Cont_Enter_BLK_5[BTH7] =
Sorting_UnL_Cont_to_BLK[BTH7]*to_BLK_5[BTH7]
Cont_Enter_BLK_5[BTH8] =
Sorting_UnL_Cont_to_BLK[BTH8]*to_BLK_5[BTH8]
Cont_Enter_BLK_5[BTH9] =
Sorting_UnL_Cont_to_BLK[BTH9]*to_BLK_5[BTH9]
Cont_Enter_BLK_5[BTH10] =
Sorting_UnL_Cont_to_BLK[BTH10]*to_BLK_5[BTH10]
Cont_Enter_BLK_5[BTH11] =
Sorting_UnL_Cont_to_BLK[BTH11]*to_BLK_5[BTH11]
Cont_Enter_BLK_5[BTH12] =
Sorting_UnL_Cont_to_BLK[BTH12]*to_BLK_5[BTH12]
Cont_Enter_BLK_5[BTH13] =
Sorting_UnL_Cont_to_BLK[BTH13]*to_BLK_5[BTH13]
Cont_Enter_BLK_5[BTH14] =
Sorting_UnL_Cont_to_BLK[BTH14]*to_BLK_5[BTH14]
Cont_Enter_BLK_5[BTH15] =
Sorting_UnL_Cont_to_BLK[BTH15]*to_BLK_5[BTH15]

OUTFLOWS:
Cont_Leave_BLK_5[BTH1] = IF (Cont_Leaving_BLK_Rate[BTH1] > 1) THEN (
Cont_Leaving_BLK_Rate[BTH1]*frm_BLK_5[BTH1])ELSE (0)
Cont_Leave_BLK_5[BTH2] = IF (Cont_Leaving_BLK_Rate[BTH2] > 1) THEN (
Cont_Leaving_BLK_Rate[BTH2]*frm_BLK_5[BTH2])ELSE (0)
Cont_Leave_BLK_5[BTH3] = IF (Cont_Leaving_BLK_Rate[BTH3] > 1) THEN (
Cont_Leaving_BLK_Rate[BTH3]*frm_BLK_5[BTH3])ELSE (0)

Cont_Leave_BLK_5[BTH4] = IF (Cont_Leaving_BLK_Rate[BTH4] > 1) THEN (Cont_Leaving_BLK_Rate[BTH4]*frm_BLK_5[BTH4])ELSE (0)
Cont_Leave_BLK_5[BTH5] = IF (Cont_Leaving_BLK_Rate[BTH5] > 1) THEN (Cont_Leaving_BLK_Rate[BTH5]*frm_BLK_5[BTH5])ELSE (0)
Cont_Leave_BLK_5[BTH6] = IF (Cont_Leaving_BLK_Rate[BTH6] > 1) THEN (Cont_Leaving_BLK_Rate[BTH6]*frm_BLK_5[BTH6])ELSE (0)
Cont_Leave_BLK_5[BTH7] = IF (Cont_Leaving_BLK_Rate[BTH7] > 1) THEN (Cont_Leaving_BLK_Rate[BTH7]*frm_BLK_5[BTH7])ELSE (0)
Cont_Leave_BLK_5[BTH8] = IF (Cont_Leaving_BLK_Rate[BTH8] > 1) THEN (Cont_Leaving_BLK_Rate[BTH8]*frm_BLK_5[BTH8])ELSE (0)
Cont_Leave_BLK_5[BTH9] = IF (Cont_Leaving_BLK_Rate[BTH9] > 1) THEN (Cont_Leaving_BLK_Rate[BTH9]*frm_BLK_5[BTH9])ELSE (0)
Cont_Leave_BLK_5[BTH10] = IF (Cont_Leaving_BLK_Rate[BTH10] > 1) THEN (Cont_Leaving_BLK_Rate[BTH10]*frm_BLK_5[BTH10])ELSE (0)
Cont_Leave_BLK_5[BTH11] = IF (Cont_Leaving_BLK_Rate[BTH11] > 1) THEN (Cont_Leaving_BLK_Rate[BTH11]*frm_BLK_5[BTH11])ELSE (0)
Cont_Leave_BLK_5[BTH12] = IF (Cont_Leaving_BLK_Rate[BTH12] > 1) THEN (Cont_Leaving_BLK_Rate[BTH12]*frm_BLK_5[BTH12])ELSE (0)
Cont_Leave_BLK_5[BTH13] = IF (Cont_Leaving_BLK_Rate[BTH13] > 1) THEN (Cont_Leaving_BLK_Rate[BTH13]*frm_BLK_5[BTH13])ELSE (0)
Cont_Leave_BLK_5[BTH14] = IF (Cont_Leaving_BLK_Rate[BTH14] > 1) THEN (Cont_Leaving_BLK_Rate[BTH14]*frm_BLK_5[BTH14])ELSE (0)
Cont_Leave_BLK_5[BTH15] = IF (Cont_Leaving_BLK_Rate[BTH15] > 1) THEN (Cont_Leaving_BLK_Rate[BTH15]*frm_BLK_5[BTH15])ELSE (0)

Cont_in_BLK_6(t) = Cont_in_BLK_6(t - dt) + (Cont_Enter_BLK_6[BTH1] + Cont_Enter_BLK_6[BTH2] + Cont_Enter_BLK_6[BTH3] + Cont_Enter_BLK_6[BTH4] + Cont_Enter_BLK_6[BTH5] + Cont_Enter_BLK_6[BTH6] + Cont_Enter_BLK_6[BTH7] + Cont_Enter_BLK_6[BTH8] + Cont_Enter_BLK_6[BTH9] + Cont_Enter_BLK_6[BTH10] + Cont_Enter_BLK_6[BTH11] + Cont_Enter_BLK_6[BTH12] + Cont_Enter_BLK_6[BTH13] + Cont_Enter_BLK_6[BTH14] + Cont_Enter_BLK_6[BTH15] + Cont_Enter_BLK_6[Berth] - Cont_Leave_BLK_6[BTH1] - Cont_Leave_BLK_6[BTH2] - Cont_Leave_BLK_6[BTH3] - Cont_Leave_BLK_6[BTH4] - Cont_Leave_BLK_6[BTH5] - Cont_Leave_BLK_6[BTH6] - Cont_Leave_BLK_6[BTH7] - Cont_Leave_BLK_6[BTH8] - Cont_Leave_BLK_6[BTH9] - Cont_Leave_BLK_6[BTH10] - Cont_Leave_BLK_6[BTH11] - Cont_Leave_BLK_6[BTH12] - Cont_Leave_BLK_6[BTH13] - Cont_Leave_BLK_6[BTH14] - Cont_Leave_BLK_6[BTH15] - Cont_Leave_BLK_6[Berth]) * dt
INIT Cont_in_BLK_6 = 3780
INFLOWS:
Cont_Enter_BLK_6[BTH1] = Sorting_UnL_Cont_to_BLK[BTH1]*to_BLK_6[BTH1]
Cont_Enter_BLK_6[BTH2] = Sorting_UnL_Cont_to_BLK[BTH2]*to_BLK_6[BTH2]
Cont_Enter_BLK_6[BTH3] = Sorting_UnL_Cont_to_BLK[BTH3]* to_BLK_6[BTH3]

Cont_Enter_BLK_6[BTH4] =
Sorting_UnL_Cont_to_BLK[BTH4]*to_BLK_6[BTH4]
Cont_Enter_BLK_6[BTH5] =
Sorting_UnL_Cont_to_BLK[BTH5]*to_BLK_6[BTH5]
Cont_Enter_BLK_6[BTH6] =
Sorting_UnL_Cont_to_BLK[BTH6]*to_BLK_6[BTH6]
Cont_Enter_BLK_6[BTH7] =
Sorting_UnL_Cont_to_BLK[BTH7]*to_BLK_6[BTH7]
Cont_Enter_BLK_6[BTH8] =
Sorting_UnL_Cont_to_BLK[BTH8]*to_BLK_6[BTH8]
Cont_Enter_BLK_6[BTH9] =
Sorting_UnL_Cont_to_BLK[BTH9]*to_BLK_6[BTH9]
Cont_Enter_BLK_6[BTH10] =
Sorting_UnL_Cont_to_BLK[BTH10]*to_BLK_6[BTH10]
Cont_Enter_BLK_6[BTH11] =
Sorting_UnL_Cont_to_BLK[BTH11]*to_BLK_6[BTH11]
Cont_Enter_BLK_6[BTH12] =
Sorting_UnL_Cont_to_BLK[BTH12]*to_BLK_6[BTH12]
Cont_Enter_BLK_6[BTH13] =
Sorting_UnL_Cont_to_BLK[BTH13]*to_BLK_6[BTH13]
Cont_Enter_BLK_6[BTH14] =
Sorting_UnL_Cont_to_BLK[BTH14]*to_BLK_6[BTH14]
Cont_Enter_BLK_6[BTH15] =
Sorting_UnL_Cont_to_BLK[BTH15]*to_BLK_6[BTH15]

OUTFLOWS:
Cont_Leave_BLK_6[BTH1] = IF (Cont_Leaving_BLK_Rate[BTH1] > 1) THEN (Cont_Leaving_BLK_Rate[BTH1]*frm_BLK_6[BTH1])ELSE (0)
Cont_Leave_BLK_6[BTH2] = IF (Cont_Leaving_BLK_Rate[BTH2] > 1) THEN (Cont_Leaving_BLK_Rate[BTH2]*frm_BLK_6[BTH2])ELSE (0)
Cont_Leave_BLK_6[BTH3] = IF (Cont_Leaving_BLK_Rate[BTH3] > 1) THEN (Cont_Leaving_BLK_Rate[BTH3]*frm_BLK_6[BTH3])ELSE (0)
Cont_Leave_BLK_6[BTH4] = IF (Cont_Leaving_BLK_Rate[BTH4] > 1) THEN (Cont_Leaving_BLK_Rate[BTH4]*frm_BLK_6[BTH4])ELSE (0)
Cont_Leave_BLK_6[BTH5] = IF (Cont_Leaving_BLK_Rate[BTH5] > 1) THEN (Cont_Leaving_BLK_Rate[BTH5]*frm_BLK_6[BTH5])ELSE (0)
Cont_Leave_BLK_6[BTH6] = IF (Cont_Leaving_BLK_Rate[BTH6] > 1) THEN (Cont_Leaving_BLK_Rate[BTH6]*frm_BLK_6[BTH6])ELSE (0)
Cont_Leave_BLK_6[BTH7] = IF (Cont_Leaving_BLK_Rate[BTH7] > 1) THEN (Cont_Leaving_BLK_Rate[BTH7]*frm_BLK_6[BTH7])ELSE (0)
Cont_Leave_BLK_6[BTH8] = IF (Cont_Leaving_BLK_Rate[BTH8] > 1) THEN (Cont_Leaving_BLK_Rate[BTH8]*frm_BLK_6[BTH8])ELSE (0)
Cont_Leave_BLK_6[BTH9] = IF (Cont_Leaving_BLK_Rate[BTH9] > 1) THEN (Cont_Leaving_BLK_Rate[BTH9]*frm_BLK_6[BTH9])ELSE (0)
Cont_Leave_BLK_6[BTH10] = IF (Cont_Leaving_BLK_Rate[BTH10] > 1) THEN (Cont_Leaving_BLK_Rate[BTH10]*frm_BLK_6[BTH10])ELSE (0)
Cont_Leave_BLK_6[BTH11] = IF (Cont_Leaving_BLK_Rate[BTH11] > 1) THEN (Cont_Leaving_BLK_Rate[BTH11]*frm_BLK_6[BTH11])ELSE (0)
Cont_Leave_BLK_6[BTH12] = IF (Cont_Leaving_BLK_Rate[BTH12] > 1) THEN (Cont_Leaving_BLK_Rate[BTH12]*frm_BLK_6[BTH12])ELSE (0)

Cont_Leave_BLK_6[BTH13] = IF (Cont_Leaving_BLK_Rate[BTH13] > 1) THEN (Cont_Leaving_BLK_Rate[BTH13]*frm_BLK_6[BTH13])ELSE (0)
Cont_Leave_BLK_6[BTH14] = IF (Cont_Leaving_BLK_Rate[BTH14] > 1) THEN (Cont_Leaving_BLK_Rate[BTH14]*frm_BLK_6[BTH14])ELSE (0)
Cont_Leave_BLK_6[BTH15] = IF (Cont_Leaving_BLK_Rate[BTH15] > 1) THEN (Cont_Leaving_BLK_Rate[BTH15]*frm_BLK_6[BTH15])ELSE (0)

Cont_in_BLK_7(t) = Cont_in_BLK_7(t - dt) + (Cont_Enter_BLK_7[BTH1] + Cont_Enter_BLK_7[BTH2] + Cont_Enter_BLK_7[BTH3] + Cont_Enter_BLK_7[BTH4] + Cont_Enter_BLK_7[BTH5] + Cont_Enter_BLK_7[BTH6] + Cont_Enter_BLK_7[BTH7] + Cont_Enter_BLK_7[BTH8] + Cont_Enter_BLK_7[BTH9] + Cont_Enter_BLK_7[BTH10] + Cont_Enter_BLK_7[BTH11] + Cont_Enter_BLK_7[BTH12] + Cont_Enter_BLK_7[BTH13] + Cont_Enter_BLK_7[BTH14] + Cont_Enter_BLK_7[BTH15] + Cont_Enter_BLK_7[Berth] - Cont_Leave_BLK_7[BTH1] - Cont_Leave_BLK_7[BTH2] - Cont_Leave_BLK_7[BTH3] - Cont_Leave_BLK_7[BTH4] - Cont_Leave_BLK_7[BTH5] - Cont_Leave_BLK_7[BTH6] - Cont_Leave_BLK_7[BTH7] - Cont_Leave_BLK_7[BTH8] - Cont_Leave_BLK_7[BTH9] - Cont_Leave_BLK_7[BTH10] - Cont_Leave_BLK_7[BTH11] - Cont_Leave_BLK_7[BTH12] - Cont_Leave_BLK_7[BTH13] - Cont_Leave_BLK_7[BTH14] - Cont_Leave_BLK_7[BTH15] - Cont_Leave_BLK_7[Berth]) * dt
INIT Cont_in_BLK_7 = 3780
INFLOWS:
Cont_Enter_BLK_7[BTH1] = Sorting_UnL_Cont_to_BLK[BTH1]*to_BLK_7[BTH1]
Cont_Enter_BLK_7[BTH2] = Sorting_UnL_Cont_to_BLK[BTH2]*to_BLK_7[BTH2]
Cont_Enter_BLK_7[BTH3] = Sorting_UnL_Cont_to_BLK[BTH3]*to_BLK_7[BTH3]
Cont_Enter_BLK_7[BTH4] = Sorting_UnL_Cont_to_BLK[BTH4]*to_BLK_7[BTH4]
Cont_Enter_BLK_7[BTH5] = Sorting_UnL_Cont_to_BLK[BTH5]*to_BLK_7[BTH5]
Cont_Enter_BLK_7[BTH6] = Sorting_UnL_Cont_to_BLK[BTH6]*to_BLK_7[BTH6]
Cont_Enter_BLK_7[BTH7] = Sorting_UnL_Cont_to_BLK[BTH7]*to_BLK_7[BTH7]
Cont_Enter_BLK_7[BTH8] = Sorting_UnL_Cont_to_BLK[BTH8]*to_BLK_7[BTH8]
Cont_Enter_BLK_7[BTH9] = Sorting_UnL_Cont_to_BLK[BTH9]*to_BLK_7[BTH9]
Cont_Enter_BLK_7[BTH10] = Sorting_UnL_Cont_to_BLK[BTH10]*to_BLK_7[BTH10]
Cont_Enter_BLK_7[BTH11] = Sorting_UnL_Cont_to_BLK[BTH11]*to_BLK_7[BTH11]
Cont_Enter_BLK_7[BTH12] = Sorting_UnL_Cont_to_BLK[BTH12]*to_BLK_7[BTH12]

Cont_Enter_BLK_7[BTH13] =
Sorting_UnL_Cont_to_BLK[BTH13]*to_BLK_7[BTH13]
Cont_Enter_BLK_7[BTH14] =
Sorting_UnL_Cont_to_BLK[BTH14]*to_BLK_7[BTH14]
Cont_Enter_BLK_7[BTH15] =
Sorting_UnL_Cont_to_BLK[BTH15]*to_BLK_7[BTH15]

OUTFLOWS:
Cont_Leave_BLK_7[BTH1] = IF (Cont_Leaving_BLK_Rate[BTH1] > 1) THEN (Cont_Leaving_BLK_Rate[BTH1]*frm_BLK_7[BTH1])ELSE (0)
Cont_Leave_BLK_7[BTH2] = IF (Cont_Leaving_BLK_Rate[BTH2] > 1) THEN (Cont_Leaving_BLK_Rate[BTH2]*frm_BLK_7[BTH2])ELSE (0)
Cont_Leave_BLK_7[BTH3] = IF (Cont_Leaving_BLK_Rate[BTH3] > 1) THEN (Cont_Leaving_BLK_Rate[BTH3]*frm_BLK_7[BTH3])ELSE (0)
Cont_Leave_BLK_7[BTH4] = IF (Cont_Leaving_BLK_Rate[BTH4] > 1) THEN (Cont_Leaving_BLK_Rate[BTH4]*frm_BLK_7[BTH4])ELSE (0)
Cont_Leave_BLK_7[BTH5] = IF (Cont_Leaving_BLK_Rate[BTH5] > 1) THEN (Cont_Leaving_BLK_Rate[BTH5]*frm_BLK_7[BTH5])ELSE (0)
Cont_Leave_BLK_7[BTH6] = IF (Cont_Leaving_BLK_Rate[BTH6] > 1) THEN (Cont_Leaving_BLK_Rate[BTH6]*frm_BLK_7[BTH6])ELSE (0)
Cont_Leave_BLK_7[BTH7] = IF (Cont_Leaving_BLK_Rate[BTH7] > 1) THEN (Cont_Leaving_BLK_Rate[BTH7]*frm_BLK_7[BTH7])ELSE (0)
Cont_Leave_BLK_7[BTH8] = IF (Cont_Leaving_BLK_Rate[BTH8] > 1) THEN (Cont_Leaving_BLK_Rate[BTH8]*frm_BLK_7[BTH8])ELSE (0)
Cont_Leave_BLK_7[BTH9] = IF (Cont_Leaving_BLK_Rate[BTH9] > 1) THEN (Cont_Leaving_BLK_Rate[BTH9]*frm_BLK_7[BTH9])ELSE (0)
Cont_Leave_BLK_7[BTH10] = IF (Cont_Leaving_BLK_Rate[BTH10] > 1) THEN (Cont_Leaving_BLK_Rate[BTH10]*frm_BLK_7[BTH10])ELSE (0)
Cont_Leave_BLK_7[BTH11] = IF (Cont_Leaving_BLK_Rate[BTH11] > 1) THEN (Cont_Leaving_BLK_Rate[BTH11]*frm_BLK_7[BTH11])ELSE (0)
Cont_Leave_BLK_7[BTH12] = IF (Cont_Leaving_BLK_Rate[BTH12] > 1) THEN (Cont_Leaving_BLK_Rate[BTH12]*frm_BLK_7[BTH12])ELSE (0)
Cont_Leave_BLK_7[BTH13] = IF (Cont_Leaving_BLK_Rate[BTH13] > 1) THEN (Cont_Leaving_BLK_Rate[BTH13]*frm_BLK_7[BTH13])ELSE (0)
Cont_Leave_BLK_7[BTH14] = IF (Cont_Leaving_BLK_Rate[BTH14] > 1) THEN (Cont_Leaving_BLK_Rate[BTH14]*frm_BLK_7[BTH14])ELSE (0)
Cont_Leave_BLK_7[BTH15] = IF (Cont_Leaving_BLK_Rate[BTH15] > 1) THEN (Cont_Leaving_BLK_Rate[BTH15]*frm_BLK_7[BTH15])ELSE (0)

Cont_in_BLK_8(t) = Cont_in_BLK_8(t - dt) + (Cont_Enter_BLK_8[BTH1] + Cont_Enter_BLK_8[BTH2] + Cont_Enter_BLK_8[BTH3] + Cont_Enter_BLK_8[BTH4] + Cont_Enter_BLK_8[BTH5] + Cont_Enter_BLK_8[BTH6] + Cont_Enter_BLK_8[BTH7] + Cont_Enter_BLK_8[BTH8] + Cont_Enter_BLK_8[BTH9] + Cont_Enter_BLK_8[BTH10] + Cont_Enter_BLK_8[BTH11] + Cont_Enter_BLK_8[BTH12] + Cont_Enter_BLK_8[BTH13] + Cont_Enter_BLK_8[BTH14] + Cont_Enter_BLK_8[BTH15] + Cont_Enter_BLK_8[Berth] - Cont_Leave_BLK_8[BTH1] - Cont_Leave_BLK_8[BTH2] - Cont_Leave_BLK_8[BTH3] - Cont_Leave_BLK_8[BTH4] - Cont_Leave_BLK_8[BTH5] -

Cont_Leave_BLK_8[BTH6] - Cont_Leave_BLK_8[BTH7] -
Cont_Leave_BLK_8[BTH8] - Cont_Leave_BLK_8[BTH9] -
Cont_Leave_BLK_8[BTH10] - Cont_Leave_BLK_8[BTH11] -
Cont_Leave_BLK_8[BTH12] - Cont_Leave_BLK_8[BTH13] -
Cont_Leave_BLK_8[BTH14] - Cont_Leave_BLK_8[BTH15] -
Cont_Leave_BLK_8[Berth]) * dt
INIT Cont_in_BLK_8 = 3780
INFLOWS:
Cont_Enter_BLK_8[BTH1] =
Sorting_UnL_Cont_to_BLK[BTH1]*to_BLK_8[BTH1]
Cont_Enter_BLK_8[BTH2] =
Sorting_UnL_Cont_to_BLK[BTH2]*to_BLK_8[BTH2]
Cont_Enter_BLK_8[BTH3] =
Sorting_UnL_Cont_to_BLK[BTH3]*to_BLK_8[BTH3]
Cont_Enter_BLK_8[BTH4] =
Sorting_UnL_Cont_to_BLK[BTH4]*to_BLK_8[BTH4]
Cont_Enter_BLK_8[BTH5] =
Sorting_UnL_Cont_to_BLK[BTH5]*to_BLK_8[BTH5]
Cont_Enter_BLK_8[BTH6] =
Sorting_UnL_Cont_to_BLK[BTH6]*to_BLK_8[BTH6]
Cont_Enter_BLK_8[BTH7] =
Sorting_UnL_Cont_to_BLK[BTH7]*to_BLK_8[BTH7]
Cont_Enter_BLK_8[BTH8] =
Sorting_UnL_Cont_to_BLK[BTH8]*to_BLK_8[BTH8]
Cont_Enter_BLK_8[BTH9] =
Sorting_UnL_Cont_to_BLK[BTH9]*to_BLK_8[BTH9]
Cont_Enter_BLK_8[BTH10] =
Sorting_UnL_Cont_to_BLK[BTH10]*to_BLK_8[BTH10]
Cont_Enter_BLK_8[BTH11] =
Sorting_UnL_Cont_to_BLK[BTH11]*to_BLK_8[BTH11]
Cont_Enter_BLK_8[BTH12] =
Sorting_UnL_Cont_to_BLK[BTH12]*to_BLK_8[BTH12]
Cont_Enter_BLK_8[BTH13] =
Sorting_UnL_Cont_to_BLK[BTH13]*to_BLK_8[BTH13]
Cont_Enter_BLK_8[BTH14] =
Sorting_UnL_Cont_to_BLK[BTH14]*to_BLK_8[BTH14]
Cont_Enter_BLK_8[BTH15] =
Sorting_UnL_Cont_to_BLK[BTH15]*to_BLK_8[BTH15]

OUTFLOWS:
Cont_Leave_BLK_8[BTH1] = IF (Cont_Leaving_BLK_Rate[BTH1] > 1) THEN (
Cont_Leaving_BLK_Rate[BTH1]*frm_BLK_8[BTH1])ELSE (0)
Cont_Leave_BLK_8[BTH2] = IF (Cont_Leaving_BLK_Rate[BTH2] > 1) THEN (
Cont_Leaving_BLK_Rate[BTH2]*frm_BLK_8[BTH2])ELSE (0)
Cont_Leave_BLK_8[BTH3] = IF (Cont_Leaving_BLK_Rate[BTH3] > 1) THEN
(Cont_Leaving_BLK_Rate[BTH3]*frm_BLK_8[BTH3])ELSE (0)
Cont_Leave_BLK_8[BTH4] = IF (Cont_Leaving_BLK_Rate[BTH4] > 1) THEN (
Cont_Leaving_BLK_Rate[BTH4]*frm_BLK_8[BTH4])ELSE (0)
Cont_Leave_BLK_8[BTH5] = IF (Cont_Leaving_BLK_Rate[BTH5] > 1) THEN (
Cont_Leaving_BLK_Rate[BTH5]*frm_BLK_8[BTH5])ELSE (0)

Cont_Leave_BLK_8[BTH6] = IF (Cont_Leaving_BLK_Rate[BTH6] > 1) THEN (Cont_Leaving_BLK_Rate[BTH6]*frm_BLK_8[BTH6])ELSE (0)
Cont_Leave_BLK_8[BTH7] = IF (Cont_Leaving_BLK_Rate[BTH7] > 1) THEN (Cont_Leaving_BLK_Rate[BTH7]*frm_BLK_8[BTH7])ELSE (0)
Cont_Leave_BLK_8[BTH8] = IF (Cont_Leaving_BLK_Rate[BTH8] > 1) THEN (Cont_Leaving_BLK_Rate[BTH8]*frm_BLK_8[BTH8])ELSE (0)
Cont_Leave_BLK_8[BTH9] = IF (Cont_Leaving_BLK_Rate[BTH9] > 1) THEN (Cont_Leaving_BLK_Rate[BTH9]*frm_BLK_8[BTH9])ELSE (0)
Cont_Leave_BLK_8[BTH10] = IF (Cont_Leaving_BLK_Rate[BTH10] > 1) THEN (Cont_Leaving_BLK_Rate[BTH10]*frm_BLK_8[BTH10])ELSE (0)
Cont_Leave_BLK_8[BTH11] = IF (Cont_Leaving_BLK_Rate[BTH11] > 1) THEN (Cont_Leaving_BLK_Rate[BTH11]*frm_BLK_8[BTH11])ELSE (0)
Cont_Leave_BLK_8[BTH12] = IF (Cont_Leaving_BLK_Rate[BTH12] > 1) THEN (Cont_Leaving_BLK_Rate[BTH12]*frm_BLK_8[BTH12])ELSE (0)
Cont_Leave_BLK_8[BTH13] = IF (Cont_Leaving_BLK_Rate[BTH13] > 1) THEN (Cont_Leaving_BLK_Rate[BTH13]*frm_BLK_8[BTH13])ELSE (0)
Cont_Leave_BLK_8[BTH14] = IF (Cont_Leaving_BLK_Rate[BTH14] > 1) THEN (Cont_Leaving_BLK_Rate[BTH14]*frm_BLK_8[BTH14])ELSE (0)
Cont_Leave_BLK_8[BTH15] = IF (Cont_Leaving_BLK_Rate[BTH15] > 1) THEN (Cont_Leaving_BLK_Rate[BTH15]*frm_BLK_8[BTH15])ELSE (0)

Cont_in_BLK_9(t) = Cont_in_BLK_9(t - dt) + (Cont_Enter_BLK_9[BTH1] + Cont_Enter_BLK_9[BTH2] + Cont_Enter_BLK_9[BTH3] + Cont_Enter_BLK_9[BTH4] + Cont_Enter_BLK_9[BTH5] + Cont_Enter_BLK_9[BTH6] + Cont_Enter_BLK_9[BTH7] + Cont_Enter_BLK_9[BTH8] + Cont_Enter_BLK_9[BTH9] + Cont_Enter_BLK_9[BTH10] + Cont_Enter_BLK_9[BTH11] + Cont_Enter_BLK_9[BTH12] + Cont_Enter_BLK_9[BTH13] + Cont_Enter_BLK_9[BTH14] + Cont_Enter_BLK_9[BTH15] + Cont_Enter_BLK_9[Berth] - Cont_Leave_BLK_9[BTH1] - Cont_Leave_BLK_9[BTH2] - Cont_Leave_BLK_9[BTH3] - Cont_Leave_BLK_9[BTH4] - Cont_Leave_BLK_9[BTH5] - Cont_Leave_BLK_9[BTH6] - Cont_Leave_BLK_9[BTH7] - Cont_Leave_BLK_9[BTH8] - Cont_Leave_BLK_9[BTH9] - Cont_Leave_BLK_9[BTH10] - Cont_Leave_BLK_9[BTH11] - Cont_Leave_BLK_9[BTH12] - Cont_Leave_BLK_9[BTH13] - Cont_Leave_BLK_9[BTH14] - Cont_Leave_BLK_9[BTH15] - Cont_Leave_BLK_9[Berth]) * dt
INIT Cont_in_BLK_9 = 3780
INFLOWS:
Cont_Enter_BLK_9[BTH1] = Sorting_UnL_Cont_to_BLK[BTH1]*to_BLK_9[BTH1]
Cont_Enter_BLK_9[BTH2] = Sorting_UnL_Cont_to_BLK[BTH2*to_BLK_9[BTH2]
Cont_Enter_BLK_9[BTH3] = Sorting_UnL_Cont_to_BLK[BTH3]* to_BLK_9[BTH3]
Cont_Enter_BLK_9[BTH4] = Sorting_UnL_Cont_to_BLK[BTH4]*to_BLK_9[BTH4]
Cont_Enter_BLK_9[BTH5] = Sorting_UnL_Cont_to_BLK[BTH5]*to_BLK_9[BTH5]

Cont_Enter_BLK_9[BTH6] =
Sorting_UnL_Cont_to_BLK[BTH6]*to_BLK_9[BTH6]
Cont_Enter_BLK_9[BTH7] =
Sorting_UnL_Cont_to_BLK[BTH7]*to_BLK_9[BTH7]
Cont_Enter_BLK_9[BTH8] =
Sorting_UnL_Cont_to_BLK[BTH8]*to_BLK_9[BTH8]
Cont_Enter_BLK_9[BTH9] =
Sorting_UnL_Cont_to_BLK[BTH9]*to_BLK_9[BTH9]
Cont_Enter_BLK_9[BTH10] =
Sorting_UnL_Cont_to_BLK[BTH10]*to_BLK_9[BTH10]
Cont_Enter_BLK_9[BTH11] =
Sorting_UnL_Cont_to_BLK[BTH11]*to_BLK_9[BTH11]
Cont_Enter_BLK_9[BTH12] =
Sorting_UnL_Cont_to_BLK[BTH12]*to_BLK_9[BTH12]
Cont_Enter_BLK_9[BTH13] =
Sorting_UnL_Cont_to_BLK[BTH13]*to_BLK_9[BTH13]
Cont_Enter_BLK_9[BTH14] =
Sorting_UnL_Cont_to_BLK[BTH14]*to_BLK_9[BTH14]
Cont_Enter_BLK_9[BTH15] =
Sorting_UnL_Cont_to_BLK[BTH15]*to_BLK_9[BTH15]

OUTFLOWS:
Cont_Leave_BLK_9[BTH1] = IF (Cont_Leaving_BLK_Rate[BTH1] > 1) THEN (
Cont_Leaving_BLK_Rate[BTH1]*
frm_BLK_9[BTH1])ELSE (0)
Cont_Leave_BLK_9[BTH2] = IF (Cont_Leaving_BLK_Rate[BTH2] > 1) THEN (
Cont_Leaving_BLK_Rate[BTH2]*frm_BLK_9[BTH2])ELSE (0)
Cont_Leave_BLK_9[BTH3] = IF (Cont_Leaving_BLK_Rate[BTH3] > 1) THEN (
Cont_Leaving_BLK_Rate[BTH3]*frm_BLK_9[BTH3])ELSE (0)
Cont_Leave_BLK_9[BTH4] = IF (Cont_Leaving_BLK_Rate[BTH4] > 1) THEN (
Cont_Leaving_BLK_Rate[BTH4]*frm_BLK_9[BTH4])ELSE (0)
Cont_Leave_BLK_9[BTH5] = IF (Cont_Leaving_BLK_Rate[BTH5] > 1) THEN (
Cont_Leaving_BLK_Rate[BTH5]*frm_BLK_9[BTH5])ELSE (0)
Cont_Leave_BLK_9[BTH6] = IF (Cont_Leaving_BLK_Rate[BTH6] > 1) THEN (
Cont_Leaving_BLK_Rate[BTH6]*frm_BLK_9[BTH6])ELSE (0)
Cont_Leave_BLK_9[BTH7] = IF (Cont_Leaving_BLK_Rate[BTH7] > 1) THEN (
Cont_Leaving_BLK_Rate[BTH7]*frm_BLK_9[BTH7])ELSE (0)
Cont_Leave_BLK_9[BTH8] = IF (Cont_Leaving_BLK_Rate[BTH8] > 1) THEN (
Cont_Leaving_BLK_Rate[BTH8]*frm_BLK_9[BTH8])ELSE (0)
Cont_Leave_BLK_9[BTH9] = IF (Cont_Leaving_BLK_Rate[BTH9] > 1) THEN (
Cont_Leaving_BLK_Rate[BTH9]*frm_BLK_9[BTH9])ELSE (0)
Cont_Leave_BLK_9[BTH10] = IF (Cont_Leaving_BLK_Rate[BTH10] > 1) THEN
(Cont_Leaving_BLK_Rate[BTH10]*frm_BLK_9[BTH10])ELSE (0)
Cont_Leave_BLK_9[BTH11] = IF (Cont_Leaving_BLK_Rate[BTH11] > 1) THEN
(Cont_Leaving_BLK_Rate[BTH11]*frm_BLK_9[BTH11])ELSE (0)
Cont_Leave_BLK_9[BTH12] = IF (Cont_Leaving_BLK_Rate[BTH12] > 1)
THEN (Cont_Leaving_BLK_Rate[BTH12]*frm_BLK_9[BTH12])ELSE (0)
Cont_Leave_BLK_9[BTH13] = IF (Cont_Leaving_BLK_Rate[BTH13] > 1) THEN
(Cont_Leaving_BLK_Rate[BTH13]*frm_BLK_9[BTH13])ELSE (0)

Cont_Leave_BLK_9[BTH14] = IF (Cont_Leaving_BLK_Rate[BTH14] > 1) THEN (Cont_Leaving_BLK_Rate[BTH14]*frm_BLK_9[BTH14])ELSE (0)
Cont_Leave_BLK_9[BTH15] = IF (Cont_Leaving_BLK_Rate[BTH15] > 1) THEN (Cont_Leaving_BLK_Rate[BTH15]*frm_BLK_9[BTH15])ELSE (0)

Cont_Loading_Process[Berth](t) = Cont_Loading_Process[Berth](t - dt) + (Transferring_Cont_to_BTH[Berth] - Cont_Loading_Complete[Berth]) * dt
INIT Cont_Loading_Process[Berth] = 0
INFLOWS:
Transferring_Cont_to_BTH[BTH1] = CONVEYOR OUTFLOW
TRANSIT TIME = Cont_Transf_to_BTH_Pro[BTH1]
Transferring_Cont_to_BTH[BTH2] = CONVEYOR OUTFLOW
TRANSIT TIME = Cont_Transf_to_BTH_Pro[BTH2]
Transferring_Cont_to_BTH[BTH3] = CONVEYOR OUTFLOW
TRANSIT TIME = Cont_Transf_to_BTH_Pro[BTH3]
Transferring_Cont_to_BTH[BTH4] = CONVEYOR OUTFLOW
TRANSIT TIME = Cont_Transf_to_BTH_Pro[BTH4]
Transferring_Cont_to_BTH[BTH5] = CONVEYOR OUTFLOW
TRANSIT TIME = Cont_Transf_to_BTH_Pro[BTH5]
Transferring_Cont_to_BTH[BTH6] = CONVEYOR OUTFLOW
TRANSIT TIME = Cont_Transf_to_BTH_Pro[BTH6]
Transferring_Cont_to_BTH[BTH7] = CONVEYOR OUTFLOW
TRANSIT TIME = Cont_Transf_to_BTH_Pro[BTH7]
Transferring_Cont_to_BTH[BTH8] = CONVEYOR OUTFLOW
TRANSIT TIME = Cont_Transf_to_BTH_Pro[BTH8]
Transferring_Cont_to_BTH[BTH9] = CONVEYOR OUTFLOW
TRANSIT TIME = Cont_Transf_to_BTH_Pro[BTH9]
Transferring_Cont_to_BTH[BTH10] = CONVEYOR OUTFLOW
TRANSIT TIME = Cont_Transf_to_BTH_Pro[BTH10]
Transferring_Cont_to_BTH[BTH11] = CONVEYOR OUTFLOW
TRANSIT TIME = Cont_Transf_to_BTH_Pro[BTH11]
Transferring_Cont_to_BTH[BTH12] = CONVEYOR OUTFLOW
TRANSIT TIME = Cont_Transf_to_BTH_Pro[BTH12]
Transferring_Cont_to_BTH[BTH13] = CONVEYOR OUTFLOW
TRANSIT TIME = Cont_Transf_to_BTH_Pro[BTH13]
Transferring_Cont_to_BTH[BTH14] = CONVEYOR OUTFLOW
TRANSIT TIME = Cont_Transf_to_BTH_Pro[BTH14]
Transferring_Cont_to_BTH[BTH15] = CONVEYOR OUTFLOW
TRANSIT TIME = Cont_Transf_to_BTH_Pro[BTH15]

OUTFLOWS:
Cont_Loading_Complete[BTH1] = CONVEYOR OUTFLOW
TRANSIT TIME = Cont_Load_Pro[BTH1]
Cont_Loading_Complete[BTH2] = CONVEYOR OUTFLOW
TRANSIT TIME = Cont_Load_Pro[BTH2]
Cont_Loading_Complete[BTH3] = CONVEYOR OUTFLOW
TRANSIT TIME = Cont_Load_Pro[BTH3]
Cont_Loading_Complete[BTH4] = CONVEYOR OUTFLOW
TRANSIT TIME = Cont_Load_Pro[BTH4]
Cont_Loading_Complete[BTH5] = CONVEYOR OUTFLOW

TRANSIT TIME = Cont_Load_Pro[BTH5]
Cont_Loading_Complete[BTH6] = CONVEYOR OUTFLOW
TRANSIT TIME = Cont_Load_Pro[BTH6]
Cont_Loading_Complete[BTH7] = CONVEYOR OUTFLOW
TRANSIT TIME = Cont_Load_Pro[BTH7]
Cont_Loading_Complete[BTH8] = CONVEYOR OUTFLOW
TRANSIT TIME = Cont_Load_Pro[BTH8]
Cont_Loading_Complete[BTH9] = CONVEYOR OUTFLOW
TRANSIT TIME = Cont_Load_Pro[BTH9]
Cont_Loading_Complete[BTH10] = CONVEYOR OUTFLOW
TRANSIT TIME = Cont_Load_Pro[BTH10]
Cont_Loading_Complete[BTH11] = CONVEYOR OUTFLOW
TRANSIT TIME = Cont_Load_Pro[BTH11]
Cont_Loading_Complete[BTH12] = CONVEYOR OUTFLOW
TRANSIT TIME = Cont_Load_Pro[BTH12]
Cont_Loading_Complete[BTH13] = CONVEYOR OUTFLOW
TRANSIT TIME = Cont_Load_Pro[BTH13]
Cont_Loading_Complete[BTH14] = CONVEYOR OUTFLOW
TRANSIT TIME = Cont_Load_Pro[BTH14]
Cont_Loading_Complete[BTH15] = CONVEYOR OUTFLOW
TRANSIT TIME = Cont_Load_Pro[BTH15]

Cont_Q_to_Enter_BTH[Berth](t) = Cont_Q_to_Enter_BTH[Berth](t - dt) + (Cont_Entering_BTH_for_Loading[Berth] - Transferring_Cont_to_BTH[Berth]) * dt
INIT Cont_Q_to_Enter_BTH[Berth] = 0
INFLOWS:
Cont_Entering_BTH_for_Loading[BTH1] = Cont_Leaving_BLK_Rate[BTH1]
Cont_Entering_BTH_for_Loading[BTH2] = Cont_Leaving_BLK_Rate[BTH2]
Cont_Entering_BTH_for_Loading[BTH3] = Cont_Leaving_BLK_Rate[BTH3]
Cont_Entering_BTH_for_Loading[BTH4] = Cont_Leaving_BLK_Rate[BTH4]
Cont_Entering_BTH_for_Loading[BTH5] = Cont_Leaving_BLK_Rate[BTH5]
Cont_Entering_BTH_for_Loading[BTH6] = Cont_Leaving_BLK_Rate[BTH6]
Cont_Entering_BTH_for_Loading[BTH7] = Cont_Leaving_BLK_Rate[BTH7]
Cont_Entering_BTH_for_Loading[BTH8] = Cont_Leaving_BLK_Rate[BTH8]
Cont_Entering_BTH_for_Loading[BTH9] = Cont_Leaving_BLK_Rate[BTH9]
Cont_Entering_BTH_for_Loading[BTH10] = Cont_Leaving_BLK_Rate[BTH10]
Cont_Entering_BTH_for_Loading[BTH11] = Cont_Leaving_BLK_Rate[BTH11]
Cont_Entering_BTH_for_Loading[BTH12] = Cont_Leaving_BLK_Rate[BTH12]
Cont_Entering_BTH_for_Loading[BTH13] = Cont_Leaving_BLK_Rate[BTH13]
Cont_Entering_BTH_for_Loading[BTH14] = Cont_Leaving_BLK_Rate[BTH14]
Cont_Entering_BTH_for_Loading[BTH15] = Cont_Leaving_BLK_Rate[BTH15]

OUTFLOWS:
Transferring_Cont_to_BTH[BTH1] = CONVEYOR OUTFLOW
TRANSIT TIME = Cont_Transf_to_BTH_Pro[BTH1]
Transferring_Cont_to_BTH[BTH2] = CONVEYOR OUTFLOW
TRANSIT TIME = Cont_Transf_to_BTH_Pro[BTH2]
Transferring_Cont_to_BTH[BTH3] = CONVEYOR OUTFLOW
TRANSIT TIME = Cont_Transf_to_BTH_Pro[BTH3]

Transferring_Cont_to_BTH[BTH4] = CONVEYOR OUTFLOW
TRANSIT TIME = Cont_Transf_to_BTH_Pro[BTH4]
Transferring_Cont_to_BTH[BTH5] = CONVEYOR OUTFLOW
TRANSIT TIME = Cont_Transf_to_BTH_Pro[BTH5]
Transferring_Cont_to_BTH[BTH6] = CONVEYOR OUTFLOW
TRANSIT TIME = Cont_Transf_to_BTH_Pro[BTH6]
Transferring_Cont_to_BTH[BTH7] = CONVEYOR OUTFLOW
TRANSIT TIME = Cont_Transf_to_BTH_Pro[BTH7]
Transferring_Cont_to_BTH[BTH8] = CONVEYOR OUTFLOW
TRANSIT TIME = Cont_Transf_to_BTH_Pro[BTH8]
Transferring_Cont_to_BTH[BTH9] = CONVEYOR OUTFLOW
TRANSIT TIME = Cont_Transf_to_BTH_Pro[BTH9]
Transferring_Cont_to_BTH[BTH10] = CONVEYOR OUTFLOW
TRANSIT TIME = Cont_Transf_to_BTH_Pro[BTH10]
Transferring_Cont_to_BTH[BTH11] = CONVEYOR OUTFLOW
TRANSIT TIME = Cont_Transf_to_BTH_Pro[BTH11]
Transferring_Cont_to_BTH[BTH12] = CONVEYOR OUTFLOW
TRANSIT TIME = Cont_Transf_to_BTH_Pro[BTH12]
Transferring_Cont_to_BTH[BTH13] = CONVEYOR OUTFLOW
TRANSIT TIME = Cont_Transf_to_BTH_Pro[BTH13]
Transferring_Cont_to_BTH[BTH14] = CONVEYOR OUTFLOW
TRANSIT TIME = Cont_Transf_to_BTH_Pro[BTH14]
Transferring_Cont_to_BTH[BTH15] = CONVEYOR OUTFLOW
TRANSIT TIME = Cont_Transf_to_BTH_Pro[BTH15]

Cont_UnL_Process[Berth](t) = Cont_UnL_Process[Berth](t - dt) + (UnL_Cont_Enter_BTH[Berth] - Cont_UnL_Complete[Berth]) * dt
INIT Cont_UnL_Process[Berth] = 0
INFLOWS:
UnL_Cont_Enter_BTH[BTH1] = IF (Increase\Decrease_in_UnL_Cont = 1) THEN (Increase_UnL[BTH1])ELSE (IF (Increase\Decrease_in_UnL_Cont = 0) THEN (Decrease_UnL[BTH1])ELSE (0))
UnL_Cont_Enter_BTH[BTH2] = IF (Increase\Decrease_in_UnL_Cont = 1) THEN (Increase_UnL[BTH2])ELSE (IF (Increase\Decrease_in_UnL_Cont = 0) THEN (Decrease_UnL[BTH2])ELSE (0))
UnL_Cont_Enter_BTH[BTH3] = IF (Increase\Decrease_in_UnL_Cont = 1) THEN (Increase_UnL[BTH3])ELSE (IF (Increase\Decrease_in_UnL_Cont = 0) THEN (Decrease_UnL[BTH3])ELSE (0))
UnL_Cont_Enter_BTH[BTH4] = IF (Increase\Decrease_in_UnL_Cont = 1) THEN (Increase_UnL[BTH4])ELSE (IF (Increase\Decrease_in_UnL_Cont = 0) THEN (Decrease_UnL[BTH4])ELSE (0))
UnL_Cont_Enter_BTH[BTH5] = IF (Increase\Decrease_in_UnL_Cont = 1) THEN (Increase_UnL[BTH5])ELSE (IF (Increase\Decrease_in_UnL_Cont = 0) THEN (Decrease_UnL[BTH5])ELSE (0))
UnL_Cont_Enter_BTH[BTH6] = IF (Increase\Decrease_in_UnL_Cont = 1) THEN (Increase_UnL[BTH6])ELSE (IF (Increase\Decrease_in_UnL_Cont = 0) THEN (Decrease_UnL[BTH6])ELSE (0))
UnL_Cont_Enter_BTH[BTH7] = IF (Increase\Decrease_in_UnL_Cont = 1) THEN (Increase_UnL[BTH7])ELSE (IF (Increase\Decrease_in_UnL_Cont = 0) THEN (Decrease_UnL[BTH7])ELSE (0))

UnL_Cont_Enter_BTH[BTH8] = IF (Increase\Decrease_in_UnL_Cont = 1) THEN (Increase_UnL[BTH8])ELSE (IF (Increase\Decrease_in_UnL_Cont = 0) THEN (Decrease_UnL[BTH8])ELSE (0))
UnL_Cont_Enter_BTH[BTH9] = IF (Increase\Decrease_in_UnL_Cont = 1) THEN (Increase_UnL[BTH9])ELSE (IF (Increase\Decrease_in_UnL_Cont = 0) THEN (Decrease_UnL[BTH9])ELSE (0))
UnL_Cont_Enter_BTH[BTH10] = IF (Increase\Decrease_in_UnL_Cont = 1) THEN (Increase_UnL[BTH10])ELSE (IF (Increase\Decrease_in_UnL_Cont = 0) THEN (Decrease_UnL[BTH10])ELSE (0))
UnL_Cont_Enter_BTH[BTH11] = IF (Increase\Decrease_in_UnL_Cont = 1) THEN (Increase_UnL[BTH11])ELSE (IF (Increase\Decrease_in_UnL_Cont = 0) THEN (Decrease_UnL[BTH11])ELSE (0))
UnL_Cont_Enter_BTH[BTH12] = IF (Increase\Decrease_in_UnL_Cont = 1) THEN (Increase_UnL[BTH12])ELSE (IF (Increase\Decrease_in_UnL_Cont = 0) THEN (Decrease_UnL[BTH12])ELSE (0))
UnL_Cont_Enter_BTH[BTH13] = IF (Increase\Decrease_in_UnL_Cont = 1) THEN (Increase_UnL[BTH13])ELSE (IF (Increase\Decrease_in_UnL_Cont = 0) THEN (Decrease_UnL[BTH13])ELSE (0))
UnL_Cont_Enter_BTH[BTH14] = IF (Increase\Decrease_in_UnL_Cont = 1) THEN (Increase_UnL[BTH14])ELSE (IF (Increase\Decrease_in_UnL_Cont = 0) THEN (Decrease_UnL[BTH14])ELSE (0))
UnL_Cont_Enter_BTH[BTH15] = IF (Increase\Decrease_in_UnL_Cont = 1) THEN (Increase_UnL[BTH15])ELSE (IF (Increase\Decrease_in_UnL_Cont = 0) THEN (Decrease_UnL[BTH15])ELSE (0))

OUTFLOWS:
Cont_UnL_Complete[BTH1] = CONVEYOR OUTFLOW
    TRANSIT TIME = Cont_UnL_Pro[BTH1]
Cont_UnL_Complete[BTH2] = CONVEYOR OUTFLOW
    TRANSIT TIME = Cont_UnL_Pro[BTH2]
Cont_UnL_Complete[BTH3] = CONVEYOR OUTFLOW
    TRANSIT TIME = Cont_UnL_Pro[BTH3]
Cont_UnL_Complete[BTH4] = CONVEYOR OUTFLOW
    TRANSIT TIME = Cont_UnL_Pro[BTH4]
Cont_UnL_Complete[BTH5] = CONVEYOR OUTFLOW
    TRANSIT TIME = Cont_UnL_Pro[BTH5]
Cont_UnL_Complete[BTH6] = CONVEYOR OUTFLOW
    TRANSIT TIME = Cont_UnL_Pro[BTH6]
Cont_UnL_Complete[BTH7] = CONVEYOR OUTFLOW
    TRANSIT TIME = Cont_UnL_Pro[BTH7]
Cont_UnL_Complete[BTH8] = CONVEYOR OUTFLOW
    TRANSIT TIME = Cont_UnL_Pro[BTH8]
Cont_UnL_Complete[BTH9] = CONVEYOR OUTFLOW
    TRANSIT TIME = Cont_UnL_Pro[BTH9]
Cont_UnL_Complete[BTH10] = CONVEYOR OUTFLOW
    TRANSIT TIME = Cont_UnL_Pro[BTH10]
Cont_UnL_Complete[BTH11] = CONVEYOR OUTFLOW
    TRANSIT TIME = Cont_UnL_Pro[BTH11]
Cont_UnL_Complete[BTH12] = CONVEYOR OUTFLOW
    TRANSIT TIME = Cont_UnL_Pro[BTH12]

Cont_UnL_Complete[BTH13] = CONVEYOR OUTFLOW
TRANSIT TIME = Cont_UnL_Pro[BTH13]
Cont_UnL_Complete[BTH14] = CONVEYOR OUTFLOW
TRANSIT TIME = Cont_UnL_Pro[BTH14]
Cont_UnL_Complete[BTH15] = CONVEYOR OUTFLOW
TRANSIT TIME = Cont_UnL_Pro[BTH15]

FD_UnL[Berth](t) = FD_UnL[Berth](t - dt) + (Noname_170[Berth]) * dt
INIT FD_UnL[Berth] = 0
INFLOWS:
Noname_170[BTH1] = FD_Cont_UnL_Rate[BTH1]
Noname_170[BTH2] = FD_Cont_UnL_Rate[BTH2]
Noname_170[BTH3] = FD_Cont_UnL_Rate[BTH3]
Noname_170[BTH4] = FD_Cont_UnL_Rate[BTH4]
Noname_170[BTH5] = FD_Cont_UnL_Rate[BTH5]
Noname_170[BTH6] = FD_Cont_UnL_Rate[BTH6]
Noname_170[BTH7] = FD_Cont_UnL_Rate[BTH7]
Noname_170[BTH8] = FD_Cont_UnL_Rate[BTH8]
Noname_170[BTH9] = FD_Cont_UnL_Rate[BTH9]
Noname_170[BTH10] = FD_Cont_UnL_Rate[BTH10]
Noname_170[BTH11] = FD_Cont_UnL_Rate[BTH11]
Noname_170[BTH12] = FD_Cont_UnL_Rate[BTH12]
Noname_170[BTH13] = FD_Cont_UnL_Rate[BTH13]
Noname_170[BTH14] = FD_Cont_UnL_Rate[BTH14]
Noname_170[BTH15] = FD_Cont_UnL_Rate[BTH15]

ML_UnL[Berth](t) = ML_UnL[Berth](t - dt) + (Noname_164[Berth]) * dt
INIT ML_UnL[Berth] = 0
INFLOWS:
Noname_164[BTH1] = ML_Cont_UnL_Rate[BTH1]
Noname_164[BTH2] = ML_Cont_UnL_Rate[BTH2]
Noname_164[BTH3] = ML_Cont_UnL_Rate[BTH3]
Noname_164[BTH4] = ML_Cont_UnL_Rate[BTH4]
Noname_164[BTH5] = ML_Cont_UnL_Rate[BTH5]
Noname_164[BTH6] = ML_Cont_UnL_Rate[BTH6]
Noname_164[BTH7] = ML_Cont_UnL_Rate[BTH7]
Noname_164[BTH8] = ML_Cont_UnL_Rate[BTH8]
Noname_164[BTH9] = ML_Cont_UnL_Rate[BTH9]
Noname_164[BTH10] = ML_Cont_UnL_Rate[BTH10]
Noname_164[BTH11] = ML_Cont_UnL_Rate[BTH11]
Noname_164[BTH12] = ML_Cont_UnL_Rate[BTH12]
Noname_164[BTH13] = ML_Cont_UnL_Rate[BTH13]
Noname_164[BTH14] = ML_Cont_UnL_Rate[BTH14]
Noname_164[BTH15] = ML_Cont_UnL_Rate[BTH15]

Noname_388(t) = Noname_388(t - dt) + (Noname_387) * dt
INIT Noname_388 = 0
INFLOWS:
Noname_387 = Noname_381

Sorting_Cont_to_BTH[Berth](t) = Sorting_Cont_to_BTH[Berth](t - dt) + (Cont_Leave_BLK_1[Berth] + Cont_Leave_BLK_2[Berth] + Cont_Leave_BLK_3[Berth] + Cont_Leave_BLK_4[Berth] + Cont_Leave_BLK_5[Berth] + Cont_Leave_BLK_6[Berth] + Cont_Leave_BLK_7[Berth] + Cont_Leave_BLK_8[Berth] + Cont_Leave_BLK_9[Berth] + Cont_Leave_BLK_10[Berth] + Cont_Leave_BLK_11[Berth] + Cont_Leave_BLK_12[Berth] - Cont_Entering_BTH_for_Loading[Berth]) * dt
INIT Sorting_Cont_to_BTH[Berth] = 0
INFLOWS:
Cont_Leave_BLK_1[BTH1] = IF (Cont_Leaving_BLK_Rate[BTH1] > 1) THEN (Cont_Leaving_BLK_Rate[BTH1]*frm_BLK_1[BTH1])ELSE (0)
Cont_Leave_BLK_1[BTH2] = IF (Cont_Leaving_BLK_Rate[BTH2] > 1) THEN (Cont_Leaving_BLK_Rate[BTH2]*frm_BLK_1[BTH2])ELSE (0)
Cont_Leave_BLK_1[BTH3] = IF (Cont_Leaving_BLK_Rate[BTH3] > 1) THEN (Cont_Leaving_BLK_Rate[BTH3]*frm_BLK_1[BTH3])ELSE (0)
Cont_Leave_BLK_1[BTH4] = IF (Cont_Leaving_BLK_Rate[BTH4] > 1) THEN (Cont_Leaving_BLK_Rate[BTH4]*frm_BLK_1[BTH4])ELSE (0)
Cont_Leave_BLK_1[BTH5] = IF (Cont_Leaving_BLK_Rate[BTH5] > 1) THEN (Cont_Leaving_BLK_Rate[BTH5]*frm_BLK_1[BTH5])ELSE (0)
Cont_Leave_BLK_1[BTH6] = IF (Cont_Leaving_BLK_Rate[BTH6] > 1) THEN (Cont_Leaving_BLK_Rate[BTH6]*frm_BLK_1[BTH6])ELSE (0)
Cont_Leave_BLK_1[BTH7] = IF (Cont_Leaving_BLK_Rate[BTH7] > 1) THEN (Cont_Leaving_BLK_Rate[BTH7]*frm_BLK_1[BTH7])ELSE (0)
Cont_Leave_BLK_1[BTH8] = IF (Cont_Leaving_BLK_Rate[BTH8] > 1) THEN (Cont_Leaving_BLK_Rate[BTH8]*frm_BLK_1[BTH8])ELSE (0)
Cont_Leave_BLK_1[BTH9] = IF (Cont_Leaving_BLK_Rate[BTH9] > 1) THEN (Cont_Leaving_BLK_Rate[BTH9]*frm_BLK_1[BTH9])ELSE (0)
Cont_Leave_BLK_1[BTH10] = IF (Cont_Leaving_BLK_Rate[BTH10] > 1) THEN (Cont_Leaving_BLK_Rate[BTH10]*frm_BLK_1[BTH10])ELSE (0)
Cont_Leave_BLK_1[BTH11] = IF (Cont_Leaving_BLK_Rate[BTH11] > 1) THEN (Cont_Leaving_BLK_Rate[BTH11]*frm_BLK_1[BTH11])ELSE (0)
Cont_Leave_BLK_1[BTH12] = IF (Cont_Leaving_BLK_Rate[BTH12] > 1) THEN (Cont_Leaving_BLK_Rate[BTH12]*frm_BLK_1[BTH12])ELSE (0)
Cont_Leave_BLK_1[BTH13] = IF (Cont_Leaving_BLK_Rate[BTH13] > 1) THEN (Cont_Leaving_BLK_Rate[BTH13]*frm_BLK_1[BTH13])ELSE (0)
Cont_Leave_BLK_1[BTH14] = IF (Cont_Leaving_BLK_Rate[BTH14] > 1) THEN (Cont_Leaving_BLK_Rate[BTH14]*frm_BLK_1[BTH14])ELSE (0)
Cont_Leave_BLK_1[BTH15] = IF (Cont_Leaving_BLK_Rate[BTH15] > 1) THEN (Cont_Leaving_BLK_Rate[BTH15]*frm_BLK_1[BTH15])ELSE (0)
Cont_Leave_BLK_2[BTH1] = IF (Cont_Leaving_BLK_Rate[BTH1] > 1) THEN (Cont_Leaving_BLK_Rate[BTH1]*frm_BLK_2[BTH1])ELSE (0)
Cont_Leave_BLK_2[BTH2] = IF (Cont_Leaving_BLK_Rate[BTH2] > 1) THEN (Cont_Leaving_BLK_Rate[BTH2]*frm_BLK_2[BTH2])ELSE (0)
Cont_Leave_BLK_2[BTH3] = IF (Cont_Leaving_BLK_Rate[BTH3] > 1) THEN (Cont_Leaving_BLK_Rate[BTH3]*frm_BLK_2[BTH3])ELSE (0)
Cont_Leave_BLK_2[BTH4] = IF (Cont_Leaving_BLK_Rate[BTH4] > 1) THEN (Cont_Leaving_BLK_Rate[BTH4]*frm_BLK_2[BTH4])ELSE (0)
Cont_Leave_BLK_2[BTH5] = IF (Cont_Leaving_BLK_Rate[BTH5] > 1) THEN (Cont_Leaving_BLK_Rate[BTH5]*frm_BLK_2[BTH5])ELSE (0)

Cont_Leave_BLK_2[BTH6] = IF (Cont_Leaving_BLK_Rate[BTH6] > 1) THEN (Cont_Leaving_BLK_Rate[BTH6]*frm_BLK_2[BTH6])ELSE (0)
Cont_Leave_BLK_2[BTH7] = IF (Cont_Leaving_BLK_Rate[BTH7] > 1) THEN (Cont_Leaving_BLK_Rate[BTH7]*frm_BLK_2[BTH7])ELSE (0)
Cont_Leave_BLK_2[BTH8] = IF (Cont_Leaving_BLK_Rate[BTH8] > 1) THEN (Cont_Leaving_BLK_Rate[BTH8]*frm_BLK_2[BTH8])ELSE (0)
Cont_Leave_BLK_2[BTH9] = IF (Cont_Leaving_BLK_Rate[BTH9] > 1) THEN (Cont_Leaving_BLK_Rate[BTH9]*frm_BLK_2[BTH9])ELSE (0)
Cont_Leave_BLK_2[BTH10] = IF (Cont_Leaving_BLK_Rate[BTH10] > 1) THEN (Cont_Leaving_BLK_Rate[BTH10]*frm_BLK_2[BTH10])ELSE (0)
Cont_Leave_BLK_2[BTH11] = IF (Cont_Leaving_BLK_Rate[BTH11] > 1) THEN (Cont_Leaving_BLK_Rate[BTH11]*frm_BLK_2[BTH11])ELSE (0)
Cont_Leave_BLK_2[BTH12] = IF (Cont_Leaving_BLK_Rate[BTH12] > 1) THEN (Cont_Leaving_BLK_Rate[BTH12]*frm_BLK_2[BTH12])ELSE (0)
Cont_Leave_BLK_2[BTH13] = IF (Cont_Leaving_BLK_Rate[BTH13] > 1) THEN (Cont_Leaving_BLK_Rate[BTH13]*frm_BLK_2[BTH13])ELSE (0)
Cont_Leave_BLK_2[BTH14] = IF (Cont_Leaving_BLK_Rate[BTH14] > 1) THEN (Cont_Leaving_BLK_Rate[BTH14]*frm_BLK_2[BTH14])ELSE (0)
Cont_Leave_BLK_2[BTH15] = IF (Cont_Leaving_BLK_Rate[BTH15] > 1) THEN (Cont_Leaving_BLK_Rate[BTH15]*frm_BLK_2[BTH15])ELSE (0)
Cont_Leave_BLK_3[BTH1] = IF (Cont_Leaving_BLK_Rate[BTH1] > 1) THEN (Cont_Leaving_BLK_Rate[BTH1]*frm_BLK_3[BTH1])ELSE (0)
Cont_Leave_BLK_3[BTH2] = IF (Cont_Leaving_BLK_Rate[BTH2] > 1) THEN (Cont_Leaving_BLK_Rate[BTH2]*frm_BLK_3[BTH2])ELSE (0)
Cont_Leave_BLK_3[BTH3] = IF (Cont_Leaving_BLK_Rate[BTH3] > 1) THEN (Cont_Leaving_BLK_Rate[BTH3]*frm_BLK_3[BTH3])ELSE (0)
Cont_Leave_BLK_3[BTH4] = IF (Cont_Leaving_BLK_Rate[BTH4] > 1) THEN (Cont_Leaving_BLK_Rate[BTH4]*frm_BLK_3[BTH4])ELSE (0)
Cont_Leave_BLK_3[BTH5] = IF (Cont_Leaving_BLK_Rate[BTH5] > 1) THEN (Cont_Leaving_BLK_Rate[BTH5]*frm_BLK_3[BTH5])ELSE (0)
Cont_Leave_BLK_3[BTH6] = IF (Cont_Leaving_BLK_Rate[BTH6] > 1) THEN (Cont_Leaving_BLK_Rate[BTH6]*frm_BLK_3[BTH6])ELSE (0)
Cont_Leave_BLK_3[BTH7] = IF (Cont_Leaving_BLK_Rate[BTH7] > 1) THEN (Cont_Leaving_BLK_Rate[BTH7]*frm_BLK_3[BTH7])ELSE (0)
Cont_Leave_BLK_3[BTH8] = IF (Cont_Leaving_BLK_Rate[BTH8] > 1) THEN (Cont_Leaving_BLK_Rate[BTH8]*frm_BLK_3[BTH8])ELSE (0)
Cont_Leave_BLK_3[BTH9] = IF (Cont_Leaving_BLK_Rate[BTH9] > 1) THEN (Cont_Leaving_BLK_Rate[BTH9]*frm_BLK_3[BTH9])ELSE (0)
Cont_Leave_BLK_3[BTH10] = IF (Cont_Leaving_BLK_Rate[BTH10] > 1) THEN (Cont_Leaving_BLK_Rate[BTH10]*frm_BLK_3[BTH10])ELSE (0)
Cont_Leave_BLK_3[BTH11] = IF (Cont_Leaving_BLK_Rate[BTH11] > 1) THEN (Cont_Leaving_BLK_Rate[BTH11]*frm_BLK_3[BTH11])ELSE (0)
Cont_Leave_BLK_3[BTH12] = IF (Cont_Leaving_BLK_Rate[BTH12] > 1) THEN (Cont_Leaving_BLK_Rate[BTH12]*frm_BLK_3[BTH12])ELSE (0)Cont_Leave_BLK_3[BTH13] = IF (Cont_Leaving_BLK_Rate[BTH13] > 1) THEN (Cont_Leaving_BLK_Rate[BTH13]*frm_BLK_3[BTH13])ELSE (0)
Cont_Leave_BLK_3[BTH14] = IF (Cont_Leaving_BLK_Rate[BTH14] > 1) THEN (Cont_Leaving_BLK_Rate[BTH14]*frm_BLK_3[BTH14])ELSE (0)
Cont_Leave_BLK_3[BTH15] = IF (Cont_Leaving_BLK_Rate[BTH15] > 1) THEN (Cont_Leaving_BLK_Rate[BTH15]*frm_BLK_3[BTH15])ELSE (0)

Cont_Leave_BLK_4[BTH1] = IF (Cont_Leaving_BLK_Rate[BTH1] > 1) THEN (Cont_Leaving_BLK_Rate[BTH1]*frm_BLK_4[BTH1])ELSE (0)
Cont_Leave_BLK_4[BTH2] = IF (Cont_Leaving_BLK_Rate[BTH2] > 1) THEN (Cont_Leaving_BLK_Rate[BTH2]*frm_BLK_4[BTH2])ELSE (0)
Cont_Leave_BLK_4[BTH3] = IF (Cont_Leaving_BLK_Rate[BTH3] > 1) THEN (Cont_Leaving_BLK_Rate[BTH3]*frm_BLK_4[BTH3])ELSE (0)
Cont_Leave_BLK_4[BTH4] = IF (Cont_Leaving_BLK_Rate[BTH4] > 1) THEN (Cont_Leaving_BLK_Rate[BTH4]*frm_BLK_4[BTH4])ELSE (0)
Cont_Leave_BLK_4[BTH5] = IF (Cont_Leaving_BLK_Rate[BTH5] > 1) THEN (Cont_Leaving_BLK_Rate[BTH5]*frm_BLK_4[BTH5])ELSE (0)
Cont_Leave_BLK_4[BTH6] = IF (Cont_Leaving_BLK_Rate[BTH6] > 1) THEN (Cont_Leaving_BLK_Rate[BTH6]*frm_BLK_4[BTH6])ELSE (0)
Cont_Leave_BLK_4[BTH7] = IF (Cont_Leaving_BLK_Rate[BTH7] > 1) THEN (Cont_Leaving_BLK_Rate[BTH7]*frm_BLK_4[BTH7])ELSE (0)
Cont_Leave_BLK_4[BTH8] = IF (Cont_Leaving_BLK_Rate[BTH8] > 1) THEN (Cont_Leaving_BLK_Rate[BTH8]*frm_BLK_4[BTH8])ELSE (0)
Cont_Leave_BLK_4[BTH9] = IF (Cont_Leaving_BLK_Rate[BTH9] > 1) THEN (Cont_Leaving_BLK_Rate[BTH9]*frm_BLK_4[BTH9])ELSE (0)
Cont_Leave_BLK_4[BTH10] = IF (Cont_Leaving_BLK_Rate[BTH10] > 1) THEN (Cont_Leaving_BLK_Rate[BTH10]*frm_BLK_4[BTH10])ELSE (0)
Cont_Leave_BLK_4[BTH11] = IF (Cont_Leaving_BLK_Rate[BTH11] > 1) THEN (Cont_Leaving_BLK_Rate[BTH11]*frm_BLK_4[BTH11])ELSE (0)
Cont_Leave_BLK_4[BTH12] = IF (Cont_Leaving_BLK_Rate[BTH12] > 1) THEN (Cont_Leaving_BLK_Rate[BTH12]*frm_BLK_4[BTH12])ELSE (0)
Cont_Leave_BLK_4[BTH13] = IF (Cont_Leaving_BLK_Rate[BTH13] > 1) THEN (Cont_Leaving_BLK_Rate[BTH13]*frm_BLK_4[BTH13])ELSE (0)
Cont_Leave_BLK_4[BTH14] = IF (Cont_Leaving_BLK_Rate[BTH14] > 1) THEN (Cont_Leaving_BLK_Rate[BTH14]*frm_BLK_4[BTH14])ELSE (0)
Cont_Leave_BLK_4[BTH15] = IF (Cont_Leaving_BLK_Rate[BTH15] > 1) THEN (Cont_Leaving_BLK_Rate[BTH15]*frm_BLK_4[BTH15])ELSE (0)
Cont_Leave_BLK_5[BTH1] = IF (Cont_Leaving_BLK_Rate[BTH1] > 1) THEN (Cont_Leaving_BLK_Rate[BTH1]*frm_BLK_5[BTH1])ELSE (0)
Cont_Leave_BLK_5[BTH2] = IF (Cont_Leaving_BLK_Rate[BTH2] > 1) THEN (Cont_Leaving_BLK_Rate[BTH2]*frm_BLK_5[BTH2])ELSE (0)
Cont_Leave_BLK_5[BTH3] = IF (Cont_Leaving_BLK_Rate[BTH3] > 1) THEN (Cont_Leaving_BLK_Rate[BTH3]*frm_BLK_5[BTH3])ELSE (0)
Cont_Leave_BLK_5[BTH4] = IF (Cont_Leaving_BLK_Rate[BTH4] > 1) THEN (Cont_Leaving_BLK_Rate[BTH4]*frm_BLK_5[BTH4])ELSE (0)
Cont_Leave_BLK_5[BTH5] = IF (Cont_Leaving_BLK_Rate[BTH5] > 1) THEN (Cont_Leaving_BLK_Rate[BTH5]*frm_BLK_5[BTH5])ELSE (0)
Cont_Leave_BLK_5[BTH6] = IF (Cont_Leaving_BLK_Rate[BTH6] > 1) THEN (Cont_Leaving_BLK_Rate[BTH6]*frm_BLK_5[BTH6])ELSE (0)
Cont_Leave_BLK_5[BTH7] = IF (Cont_Leaving_BLK_Rate[BTH7] > 1) THEN (Cont_Leaving_BLK_Rate[BTH7]*frm_BLK_5[BTH7])ELSE (0)
Cont_Leave_BLK_5[BTH8] = IF (Cont_Leaving_BLK_Rate[BTH8] > 1) THEN (Cont_Leaving_BLK_Rate[BTH8]*frm_BLK_5[BTH8])ELSE (0)
Cont_Leave_BLK_5[BTH9] = IF (Cont_Leaving_BLK_Rate[BTH9] > 1) THEN (Cont_Leaving_BLK_Rate[BTH9]*frm_BLK_5[BTH9])ELSE (0)
Cont_Leave_BLK_5[BTH10] = IF (Cont_Leaving_BLK_Rate[BTH10] > 1) THEN (Cont_Leaving_BLK_Rate[BTH10]*frm_BLK_5[BTH10])ELSE (0)

Cont_Leave_BLK_5[BTH11] = IF (Cont_Leaving_BLK_Rate[BTH11] > 1) THEN (Cont_Leaving_BLK_Rate[BTH11]*frm_BLK_5[BTH11])ELSE (0)
Cont_Leave_BLK_5[BTH12] = IF (Cont_Leaving_BLK_Rate[BTH12] > 1) THEN (Cont_Leaving_BLK_Rate[BTH12]*frm_BLK_5[BTH12])ELSE (0)
Cont_Leave_BLK_5[BTH13] = IF (Cont_Leaving_BLK_Rate[BTH13] > 1) THEN (Cont_Leaving_BLK_Rate[BTH13]*frm_BLK_5[BTH13])ELSE (0)
Cont_Leave_BLK_5[BTH14] = IF (Cont_Leaving_BLK_Rate[BTH14] > 1) THEN (Cont_Leaving_BLK_Rate[BTH14]*frm_BLK_5[BTH14])ELSE (0)
Cont_Leave_BLK_5[BTH15] = IF (Cont_Leaving_BLK_Rate[BTH15] > 1) THEN (Cont_Leaving_BLK_Rate[BTH15]*frm_BLK_5[BTH15])ELSE (0)
Cont_Leave_BLK_6[BTH1] = IF (Cont_Leaving_BLK_Rate[BTH1] > 1) THEN (Cont_Leaving_BLK_Rate[BTH1]*frm_BLK_6[BTH1])ELSE (0)
Cont_Leave_BLK_6[BTH2] = IF (Cont_Leaving_BLK_Rate[BTH2] > 1) THEN (Cont_Leaving_BLK_Rate[BTH2]*frm_BLK_6[BTH2])ELSE (0)
Cont_Leave_BLK_6[BTH3] = IF (Cont_Leaving_BLK_Rate[BTH3] > 1) THEN (Cont_Leaving_BLK_Rate[BTH3]*frm_BLK_6[BTH3])ELSE (0)
Cont_Leave_BLK_6[BTH4] = IF (Cont_Leaving_BLK_Rate[BTH4] > 1) THEN (Cont_Leaving_BLK_Rate[BTH4]*frm_BLK_6[BTH4])ELSE (0)
Cont_Leave_BLK_6[BTH5] = IF (Cont_Leaving_BLK_Rate[BTH5] > 1) THEN (Cont_Leaving_BLK_Rate[BTH5]*frm_BLK_6[BTH5])ELSE (0)
Cont_Leave_BLK_6[BTH6] = IF (Cont_Leaving_BLK_Rate[BTH6] > 1) THEN (Cont_Leaving_BLK_Rate[BTH6]*frm_BLK_6[BTH6])ELSE (0)
Cont_Leave_BLK_6[BTH7] = IF (Cont_Leaving_BLK_Rate[BTH7] > 1) THEN (Cont_Leaving_BLK_Rate[BTH7]*frm_BLK_6[BTH7])ELSE (0)
Cont_Leave_BLK_6[BTH8] = IF (Cont_Leaving_BLK_Rate[BTH8] > 1) THEN (Cont_Leaving_BLK_Rate[BTH8]*frm_BLK_6[BTH8])ELSE (0)
Cont_Leave_BLK_6[BTH9] = IF (Cont_Leaving_BLK_Rate[BTH9] > 1) THEN (Cont_Leaving_BLK_Rate[BTH9]*frm_BLK_6[BTH9])ELSE (0)
Cont_Leave_BLK_6[BTH10] = IF (Cont_Leaving_BLK_Rate[BTH10] > 1) THEN (Cont_Leaving_BLK_Rate[BTH10]*frm_BLK_6[BTH10])ELSE (0)
Cont_Leave_BLK_6[BTH11] = IF (Cont_Leaving_BLK_Rate[BTH11] > 1) THEN (Cont_Leaving_BLK_Rate[BTH11]*frm_BLK_6[BTH11])ELSE (0)
Cont_Leave_BLK_6[BTH12] = IF (Cont_Leaving_BLK_Rate[BTH12] > 1) THEN (Cont_Leaving_BLK_Rate[BTH12]*frm_BLK_6[BTH12])ELSE (0)
Cont_Leave_BLK_6[BTH13] = IF (Cont_Leaving_BLK_Rate[BTH13] > 1) THEN (Cont_Leaving_BLK_Rate[BTH13]*frm_BLK_6[BTH13])ELSE (0)
Cont_Leave_BLK_6[BTH14] = IF (Cont_Leaving_BLK_Rate[BTH14] > 1) THEN (Cont_Leaving_BLK_Rate[BTH14]*frm_BLK_6[BTH14])ELSE (0)
Cont_Leave_BLK_6[BTH15] = IF (Cont_Leaving_BLK_Rate[BTH15] > 1) THEN (Cont_Leaving_BLK_Rate[BTH15]*frm_BLK_6[BTH15])ELSE (0)
Cont_Leave_BLK_7[BTH1] = IF (Cont_Leaving_BLK_Rate[BTH1] > 1) THEN (Cont_Leaving_BLK_Rate[BTH1]*frm_BLK_7[BTH1])ELSE (0)
Cont_Leave_BLK_7[BTH2] = IF (Cont_Leaving_BLK_Rate[BTH2] > 1) THEN (Cont_Leaving_BLK_Rate[BTH2]*frm_BLK_7[BTH2])ELSE (0)
Cont_Leave_BLK_7[BTH3] = IF (Cont_Leaving_BLK_Rate[BTH3] > 1) THEN (Cont_Leaving_BLK_Rate[BTH3]*frm_BLK_7[BTH3])ELSE (0)
Cont_Leave_BLK_7[BTH4] = IF (Cont_Leaving_BLK_Rate[BTH4] > 1) THEN (Cont_Leaving_BLK_Rate[BTH4]*frm_BLK_7[BTH4])ELSE (0)
Cont_Leave_BLK_7[BTH5] = IF (Cont_Leaving_BLK_Rate[BTH5] > 1) THEN (Cont_Leaving_BLK_Rate[BTH5]*frm_BLK_7[BTH5])ELSE (0)

Cont_Leave_BLK_7[BTH6] = IF (Cont_Leaving_BLK_Rate[BTH6] > 1) THEN (Cont_Leaving_BLK_Rate[BTH6]*frm_BLK_7[BTH6])ELSE (0)
Cont_Leave_BLK_7[BTH7] = IF (Cont_Leaving_BLK_Rate[BTH7] > 1) THEN (Cont_Leaving_BLK_Rate[BTH7]*frm_BLK_7[BTH7])ELSE (0)
Cont_Leave_BLK_7[BTH8] = IF (Cont_Leaving_BLK_Rate[BTH8] > 1) THEN (Cont_Leaving_BLK_Rate[BTH8]*frm_BLK_7[BTH8])ELSE (0)
Cont_Leave_BLK_7[BTH9] = IF (Cont_Leaving_BLK_Rate[BTH9] > 1) THEN (Cont_Leaving_BLK_Rate[BTH9]*frm_BLK_7[BTH9])ELSE (0)
Cont_Leave_BLK_7[BTH10] = IF (Cont_Leaving_BLK_Rate[BTH10] > 1) THEN (Cont_Leaving_BLK_Rate[BTH10]*frm_BLK_7[BTH10])ELSE (0)
Cont_Leave_BLK_7[BTH11] = IF (Cont_Leaving_BLK_Rate[BTH11] > 1) THEN (Cont_Leaving_BLK_Rate[BTH11]*frm_BLK_7[BTH11])ELSE (0)
Cont_Leave_BLK_7[BTH12] = IF (Cont_Leaving_BLK_Rate[BTH12] > 1) THEN (Cont_Leaving_BLK_Rate[BTH12]*frm_BLK_7[BTH12])ELSE (0)
Cont_Leave_BLK_7[BTH13] = IF (Cont_Leaving_BLK_Rate[BTH13] > 1) THEN (Cont_Leaving_BLK_Rate[BTH13]*frm_BLK_7[BTH13])ELSE (0)
Cont_Leave_BLK_7[BTH14] = IF (Cont_Leaving_BLK_Rate[BTH14] > 1) THEN (Cont_Leaving_BLK_Rate[BTH14]*frm_BLK_7[BTH14])ELSE (0)
Cont_Leave_BLK_7[BTH15] = IF (Cont_Leaving_BLK_Rate[BTH15] > 1) THEN (Cont_Leaving_BLK_Rate[BTH15]*frm_BLK_7[BTH15])ELSE (0)
Cont_Leave_BLK_8[BTH1] = IF (Cont_Leaving_BLK_Rate[BTH1] > 1) THEN (Cont_Leaving_BLK_Rate[BTH1]*frm_BLK_8[BTH1])ELSE (0)
Cont_Leave_BLK_8[BTH2] = IF (Cont_Leaving_BLK_Rate[BTH2] > 1) THEN (Cont_Leaving_BLK_Rate[BTH2]*frm_BLK_8[BTH2])ELSE (0)
Cont_Leave_BLK_8[BTH3] = IF (Cont_Leaving_BLK_Rate[BTH3] > 1) THEN (Cont_Leaving_BLK_Rate[BTH3]*frm_BLK_8[BTH3])ELSE (0)
Cont_Leave_BLK_8[BTH4] = IF (Cont_Leaving_BLK_Rate[BTH4] > 1) THEN (Cont_Leaving_BLK_Rate[BTH4]*frm_BLK_8[BTH4])ELSE (0)
Cont_Leave_BLK_8[BTH5] = IF (Cont_Leaving_BLK_Rate[BTH5] > 1) THEN (Cont_Leaving_BLK_Rate[BTH5]*frm_BLK_8[BTH5])ELSE (0)
Cont_Leave_BLK_8[BTH6] = IF (Cont_Leaving_BLK_Rate[BTH6] > 1) THEN (Cont_Leaving_BLK_Rate[BTH6]*frm_BLK_8[BTH6])ELSE (0)
Cont_Leave_BLK_8[BTH7] = IF (Cont_Leaving_BLK_Rate[BTH7] > 1) THEN (Cont_Leaving_BLK_Rate[BTH7]*frm_BLK_8[BTH7])ELSE (0)
Cont_Leave_BLK_8[BTH8] = IF (Cont_Leaving_BLK_Rate[BTH8] > 1) THEN (Cont_Leaving_BLK_Rate[BTH8]*frm_BLK_8[BTH8])ELSE (0)
Cont_Leave_BLK_8[BTH9] = IF (Cont_Leaving_BLK_Rate[BTH9] > 1) THEN (Cont_Leaving_BLK_Rate[BTH9]*frm_BLK_8[BTH9])ELSE (0)
Cont_Leave_BLK_8[BTH10] = IF (Cont_Leaving_BLK_Rate[BTH10] > 1) THEN (Cont_Leaving_BLK_Rate[BTH10]*frm_BLK_8[BTH10])ELSE (0)
Cont_Leave_BLK_8[BTH11] = IF (Cont_Leaving_BLK_Rate[BTH11] > 1) THEN (Cont_Leaving_BLK_Rate[BTH11]*frm_BLK_8[BTH11])ELSE (0)
Cont_Leave_BLK_8[BTH12] = IF (Cont_Leaving_BLK_Rate[BTH12] > 1) THEN (Cont_Leaving_BLK_Rate[BTH12]*frm_BLK_8[BTH12])ELSE (0)
Cont_Leave_BLK_8[BTH13] = IF (Cont_Leaving_BLK_Rate[BTH13] > 1) THEN (Cont_Leaving_BLK_Rate[BTH13]*frm_BLK_8[BTH13])ELSE (0)
Cont_Leave_BLK_8[BTH14] = IF (Cont_Leaving_BLK_Rate[BTH14] > 1) THEN (Cont_Leaving_BLK_Rate[BTH14]*frm_BLK_8[BTH14])ELSE (0)
Cont_Leave_BLK_8[BTH15] = IF (Cont_Leaving_BLK_Rate[BTH15] > 1) THEN (Cont_Leaving_BLK_Rate[BTH15]*frm_BLK_8[BTH15])ELSE (0)

Cont_Leave_BLK_9[BTH1] = IF (Cont_Leaving_BLK_Rate[BTH1] > 1) THEN (Cont_Leaving_BLK_Rate[BTH1]*frm_BLK_9[BTH1])ELSE (0)
Cont_Leave_BLK_9[BTH2] = IF (Cont_Leaving_BLK_Rate[BTH2] > 1) THEN (Cont_Leaving_BLK_Rate[BTH2]*frm_BLK_9[BTH2])ELSE (0)
Cont_Leave_BLK_9[BTH3] = IF (Cont_Leaving_BLK_Rate[BTH3] > 1) THEN (Cont_Leaving_BLK_Rate[BTH3]*frm_BLK_9[BTH3])ELSE (0)
Cont_Leave_BLK_9[BTH4] = IF (Cont_Leaving_BLK_Rate[BTH4] > 1) THEN (Cont_Leaving_BLK_Rate[BTH4]*frm_BLK_9[BTH4])ELSE (0)
Cont_Leave_BLK_9[BTH5] = IF (Cont_Leaving_BLK_Rate[BTH5] > 1) THEN (Cont_Leaving_BLK_Rate[BTH5]*frm_BLK_9[BTH5])ELSE (0)
Cont_Leave_BLK_9[BTH6] = IF (Cont_Leaving_BLK_Rate[BTH6] > 1) THEN (Cont_Leaving_BLK_Rate[BTH6]*frm_BLK_9[BTH6])ELSE (0)
Cont_Leave_BLK_9[BTH7] = IF (Cont_Leaving_BLK_Rate[BTH7] > 1) THEN (Cont_Leaving_BLK_Rate[BTH7]*frm_BLK_9[BTH7])ELSE (0)
Cont_Leave_BLK_9[BTH8] = IF (Cont_Leaving_BLK_Rate[BTH8] > 1) THEN (Cont_Leaving_BLK_Rate[BTH8]*frm_BLK_9[BTH8])ELSE (0)
Cont_Leave_BLK_9[BTH9] = IF (Cont_Leaving_BLK_Rate[BTH9] > 1) THEN (Cont_Leaving_BLK_Rate[BTH9]*frm_BLK_9[BTH9])ELSE (0)
Cont_Leave_BLK_9[BTH10] = IF (Cont_Leaving_BLK_Rate[BTH10] > 1) THEN (Cont_Leaving_BLK_Rate[BTH10]*frm_BLK_9[BTH10])ELSE (0)
Cont_Leave_BLK_9[BTH11] = IF (Cont_Leaving_BLK_Rate[BTH11] > 1) THEN (Cont_Leaving_BLK_Rate[BTH11]*frm_BLK_9[BTH11])ELSE (0)
Cont_Leave_BLK_9[BTH12] = IF (Cont_Leaving_BLK_Rate[BTH12] > 1) THEN (Cont_Leaving_BLK_Rate[BTH12]*frm_BLK_9[BTH12])ELSE (0)
Cont_Leave_BLK_9[BTH13] = IF (Cont_Leaving_BLK_Rate[BTH13] > 1) THEN (Cont_Leaving_BLK_Rate[BTH13]*frm_BLK_9[BTH13])ELSE (0)
Cont_Leave_BLK_9[BTH14] = IF (Cont_Leaving_BLK_Rate[BTH14] > 1) THEN (Cont_Leaving_BLK_Rate[BTH14]*frm_BLK_9[BTH14])ELSE (0)
Cont_Leave_BLK_9[BTH15] = IF (Cont_Leaving_BLK_Rate[BTH15] > 1) THEN (Cont_Leaving_BLK_Rate[BTH15]*frm_BLK_9[BTH15])ELSE (0)
Cont_Leave_BLK_10[BTH1] = IF (Cont_Leaving_BLK_Rate[BTH1] > 1) THEN (Cont_Leaving_BLK_Rate[BTH1]*frm_BLK_10[BTH1])ELSE (0)
Cont_Leave_BLK_10[BTH2] = IF (Cont_Leaving_BLK_Rate[BTH2] > 1) THEN (Cont_Leaving_BLK_Rate[BTH2]*frm_BLK_10[BTH2])ELSE (0)
Cont_Leave_BLK_10[BTH3] = IF (Cont_Leaving_BLK_Rate[BTH3] > 1) THEN (Cont_Leaving_BLK_Rate[BTH3]*frm_BLK_10[BTH3])ELSE (0)Cont_Leave_BLK_10[BTH4] = IF (Cont_Leaving_BLK_Rate[BTH4] > 1THEN (Cont_Leaving_BLK_Rate[BTH4]*frm_BLK_10[BTH4])ELSE (0)
Cont_Leave_BLK_10[BTH5] = IF (Cont_Leaving_BLK_Rate[BTH5] > 1) THEN (Cont_Leaving_BLK_Rate[BTH5]*frm_BLK_10[BTH5])ELSE (0)
Cont_Leave_BLK_10[BTH6] = IF (Cont_Leaving_BLK_Rate[BTH6] > 1) THEN (Cont_Leaving_BLK_Rate[BTH6]*frm_BLK_10[BTH6])ELSE (0)
Cont_Leave_BLK_10[BTH7] = IF (Cont_Leaving_BLK_Rate[BTH7] > 1) THEN (Cont_Leaving_BLK_Rate[BTH7]*frm_BLK_10[BTH7])ELSE (0)
Cont_Leave_BLK_10[BTH8] = IF (Cont_Leaving_BLK_Rate[BTH8] > 1) THEN (Cont_Leaving_BLK_Rate[BTH8]*frm_BLK_10[BTH8])ELSE (0)
Cont_Leave_BLK_10[BTH9] = IF (Cont_Leaving_BLK_Rate[BTH9] > 1) THEN (Cont_Leaving_BLK_Rate[BTH9]*frm_BLK_10[BTH9])ELSE (0)
Cont_Leave_BLK_10[BTH10] = IF (Cont_Leaving_BLK_Rate[BTH10] > 1 THEN (Cont_Leaving_BLK_Rate[BTH10]*frm_BLK_10[BTH10])ELSE (0)

Cont_Leave_BLK_10[BTH11] = IF (Cont_Leaving_BLK_Rate[BTH11] > 1 THEN
(Cont_Leaving_BLK_Rate[BTH11]*frm_BLK_10[BTH11])ELSE (0)
Cont_Leave_BLK_10[BTH12] = IF (Cont_Leaving_BLK_Rate[BTH12] > 1 THEN
(Cont_Leaving_BLK_Rate[BTH12]*frm_BLK_10[BTH12])ELSE (0)
Cont_Leave_BLK_10[BTH13] = IF (Cont_Leaving_BLK_Rate[BTH13] > 1 THEN
(Cont_Leaving_BLK_Rate[BTH13]*frm_BLK_10[BTH13])ELSE (0)
Cont_Leave_BLK_10[BTH14] = IF (Cont_Leaving_BLK_Rate[BTH14] > 1 THEN
(Cont_Leaving_BLK_Rate[BTH14]*frm_BLK_10[BTH14])ELSE (0)
Cont_Leave_BLK_10[BTH15] = IF (Cont_Leaving_BLK_Rate[BTH15] > 1 THEN
(Cont_Leaving_BLK_Rate[BTH15]*frm_BLK_10[BTH15])ELSE (0)
Cont_Leave_BLK_11[BTH1] = IF (Cont_Leaving_BLK_Rate[BTH1] > 1) THEN
(Cont_Leaving_BLK_Rate[BTH1]*frm_BLK_11[BTH1])ELSE (0)
Cont_Leave_BLK_11[BTH2] = IF (Cont_Leaving_BLK_Rate[BTH2] > 1) THEN
(Cont_Leaving_BLK_Rate[BTH2]*frm_BLK_11[BTH2])ELSE (0)
Cont_Leave_BLK_11[BTH3] = IF (Cont_Leaving_BLK_Rate[BTH3] > 1) THEN
(Cont_Leaving_BLK_Rate[BTH3]*frm_BLK_11[BTH3])ELSE (0)
Cont_Leave_BLK_11[BTH4] = IF (Cont_Leaving_BLK_Rate[BTH4] > 1) THEN
(Cont_Leaving_BLK_Rate[BTH4]*frm_BLK_11[BTH4])ELSE (0)
Cont_Leave_BLK_11[BTH5] = IF (Cont_Leaving_BLK_Rate[BTH5] > 1) THEN
(Cont_Leaving_BLK_Rate[BTH5]*frm_BLK_11[BTH5])ELSE (0)
Cont_Leave_BLK_11[BTH6] = IF (Cont_Leaving_BLK_Rate[BTH6] > 1) THEN
(Cont_Leaving_BLK_Rate[BTH6]*frm_BLK_11[BTH5])ELSE (0)
Cont_Leave_BLK_11[BTH7] = IF (Cont_Leaving_BLK_Rate[BTH7] > 1) THEN
(Cont_Leaving_BLK_Rate[BTH7]*frm_BLK_11[BTH7])ELSE (0)
Cont_Leave_BLK_11[BTH8] = IF (Cont_Leaving_BLK_Rate[BTH8] > 1) THEN
(Cont_Leaving_BLK_Rate[BTH8]*frm_BLK_11[BTH8])ELSE (0)
Cont_Leave_BLK_11[BTH9] = IF (Cont_Leaving_BLK_Rate[BTH9] > 1) THEN
(Cont_Leaving_BLK_Rate[BTH9]*frm_BLK_11[BTH9])ELSE (0)
Cont_Leave_BLK_11[BTH10] = IF (Cont_Leaving_BLK_Rate[BTH10] > 1 THEN
(Cont_Leaving_BLK_Rate[BTH10]*frm_BLK_11[BTH10])ELSE (0)
Cont_Leave_BLK_11[BTH11] = IF (Cont_Leaving_BLK_Rate[BTH11] > 1 THEN
(Cont_Leaving_BLK_Rate[BTH11]*frm_BLK_11[BTH11])ELSE (0)
Cont_Leave_BLK_11[BTH12] = IF (Cont_Leaving_BLK_Rate[BTH12] > 1 THEN
(Cont_Leaving_BLK_Rate[BTH12]*frm_BLK_11[BTH12])ELSE (0)
Cont_Leave_BLK_11[BTH13] = IF (Cont_Leaving_BLK_Rate[BTH13] > 1 THEN
(Cont_Leaving_BLK_Rate[BTH13]*frm_BLK_11[BTH13])ELSE (0)
Cont_Leave_BLK_11[BTH14] = IF (Cont_Leaving_BLK_Rate[BTH14] > 1 THEN
(Cont_Leaving_BLK_Rate[BTH14]*frm_BLK_11[BTH14])ELSE (0)
Cont_Leave_BLK_11[BTH15] = IF (Cont_Leaving_BLK_Rate[BTH15] > 1 THEN
(Cont_Leaving_BLK_Rate[BTH15]*frm_BLK_11[BTH15])ELSE (0)
Cont_Leave_BLK_12[BTH1] = IF (Cont_Leaving_BLK_Rate[BTH1] > 1) THEN
(Cont_Leaving_BLK_Rate[BTH1]*frm_BLK_12[BTH1])ELSE (0)
Cont_Leave_BLK_12[BTH2] = IF (Cont_Leaving_BLK_Rate[BTH2] > 1) THEN
(Cont_Leaving_BLK_Rate[BTH2]*frm_BLK_12[BTH2])ELSE (0)
Cont_Leave_BLK_12[BTH3] = IF (Cont_Leaving_BLK_Rate[BTH3] > 1) THEN
(Cont_Leaving_BLK_Rate[BTH3]*frm_BLK_12[BTH3])ELSE (0)
Cont_Leave_BLK_12[BTH4] = IF (Cont_Leaving_BLK_Rate[BTH4] > 1) THEN
(Cont_Leaving_BLK_Rate[BTH4]*frm_BLK_12[BTH4])ELSE (0)
Cont_Leave_BLK_12[BTH5] = IF (Cont_Leaving_BLK_Rate[BTH5] > 1) THEN
(Cont_Leaving_BLK_Rate[BTH5]*frm_BLK_12[BTH5])ELSE (0)

```
Cont_Leave_BLK_12[BTH6] = IF ( Cont_Leaving_BLK_Rate[BTH6] > 1 ) THEN
(Cont_Leaving_BLK_Rate[BTH6]*frm_BLK_12[BTH6] )ELSE (0)
Cont_Leave_BLK_12[BTH7] = IF ( Cont_Leaving_BLK_Rate[BTH7] > 1 ) THEN
(Cont_Leaving_BLK_Rate[BTH7]*frm_BLK_12[BTH7] )ELSE (0)
Cont_Leave_BLK_12[BTH8] = IF ( Cont_Leaving_BLK_Rate[BTH8] > 1 ) THEN
(Cont_Leaving_BLK_Rate[BTH8]*frm_BLK_12[BTH8] )ELSE (0)
Cont_Leave_BLK_12[BTH9] = IF ( Cont_Leaving_BLK_Rate[BTH9] > 1 ) THEN
(Cont_Leaving_BLK_Rate[BTH9]*frm_BLK_12[BTH9] )ELSE (0)
Cont_Leave_BLK_12[BTH10] = IF ( Cont_Leaving_BLK_Rate[BTH10] > 1 THEN
(Cont_Leaving_BLK_Rate[BTH10]*frm_BLK_12[BTH10] )ELSE (0)
Cont_Leave_BLK_12[BTH11] = IF ( Cont_Leaving_BLK_Rate[BTH11] > 1 THEN
(Cont_Leaving_BLK_Rate[BTH11]*frm_BLK_12[BTH11] )ELSE (0)
Cont_Leave_BLK_12[BTH12] = IF ( Cont_Leaving_BLK_Rate[BTH12] > 1 THEN
(Cont_Leaving_BLK_Rate[BTH12]*frm_BLK_12[BTH12] )ELSE (0)
Cont_Leave_BLK_12[BTH13] = IF ( Cont_Leaving_BLK_Rate[BTH13] > 1 THEN
(Cont_Leaving_BLK_Rate[BTH13]*frm_BLK_12[BTH13] )ELSE (0)
Cont_Leave_BLK_12[BTH14] = IF ( Cont_Leaving_BLK_Rate[BTH14] > 1 THEN
(Cont_Leaving_BLK_Rate[BTH14]*frm_BLK_12[BTH14] )ELSE (0)
Cont_Leave_BLK_12[BTH15] = IF ( Cont_Leaving_BLK_Rate[BTH15] > 1 THEN
(Cont_Leaving_BLK_Rate[BTH15]*frm_BLK_12[BTH15] )ELSE (0)

OUTFLOWS:
Cont_Entering_BTH_for_Loading[BTH1] = Cont_Leaving_BLK_Rate[BTH1]
Cont_Entering_BTH_for_Loading[BTH2] = Cont_Leaving_BLK_Rate[BTH2]
Cont_Entering_BTH_for_Loading[BTH3] = Cont_Leaving_BLK_Rate[BTH3]
Cont_Entering_BTH_for_Loading[BTH4] = Cont_Leaving_BLK_Rate[BTH4]
Cont_Entering_BTH_for_Loading[BTH5] = Cont_Leaving_BLK_Rate[BTH5]
Cont_Entering_BTH_for_Loading[BTH6] = Cont_Leaving_BLK_Rate[BTH6]
Cont_Entering_BTH_for_Loading[BTH7] = Cont_Leaving_BLK_Rate[BTH7]
Cont_Entering_BTH_for_Loading[BTH8] = Cont_Leaving_BLK_Rate[BTH8]
Cont_Entering_BTH_for_Loading[BTH9] = Cont_Leaving_BLK_Rate[BTH9]
Cont_Entering_BTH_for_Loading[BTH10] = Cont_Leaving_BLK_Rate[BTH10]
Cont_Entering_BTH_for_Loading[BTH11] = Cont_Leaving_BLK_Rate[BTH11]
Cont_Entering_BTH_for_Loading[BTH12] = Cont_Leaving_BLK_Rate[BTH12]
Cont_Entering_BTH_for_Loading[BTH13] = Cont_Leaving_BLK_Rate[BTH13]
Cont_Entering_BTH_for_Loading[BTH14] = Cont_Leaving_BLK_Rate[BTH14]
Cont_Entering_BTH_for_Loading[BTH15] = Cont_Leaving_BLK_Rate[BTH15]

Sorting_UnL_Cont_to_BLK[Berth](t) = Sorting_UnL_Cont_to_BLK[Berth](t - dt) +
(Transfering_UnL_Cont_to_BLK[Berth] - Cont_Enter_BLK_1[Berth] -
Cont_Enter_BLK_2[Berth] - Cont_Enter_BLK_3[Berth] -
Cont_Enter_BLK_4[Berth] - Cont_Enter_BLK_5[Berth] -
Cont_Enter_BLK_6[Berth] - Cont_Enter_BLK_7[Berth] -
Cont_Enter_BLK_8[Berth] - Cont_Enter_BLK_9[Berth] -
Cont_Enter_BLK_10[Berth] - Cont_Enter_BLK_11[Berth] -
Cont_Enter_BLK_12[Berth]) * dt
INIT Sorting_UnL_Cont_to_BLK[Berth] = 0
INFLOWS:
Transfering_UnL_Cont_to_BLK[BTH1] = CONVEYOR OUTFLOW
    TRANSIT TIME = Cont_Trans_to_BLK_Pro[BTH1]
```

Transfering_UnL_Cont_to_BLK[BTH2] = CONVEYOR OUTFLOW
TRANSIT TIME = Cont_Trans_to_BLK_Pro[BTH2]
Transfering_UnL_Cont_to_BLK[BTH3] = CONVEYOR OUTFLOW
TRANSIT TIME = Cont_Trans_to_BLK_Pro[BTH3]
Transfering_UnL_Cont_to_BLK[BTH4] = CONVEYOR OUTFLOW
TRANSIT TIME = Cont_Trans_to_BLK_Pro[BTH4]
Transfering_UnL_Cont_to_BLK[BTH5] = CONVEYOR OUTFLOW
TRANSIT TIME = Cont_Trans_to_BLK_Pro[BTH5]
Transfering_UnL_Cont_to_BLK[BTH6] = CONVEYOR OUTFLOW
TRANSIT TIME = Cont_Trans_to_BLK_Pro[BTH6]
Transfering_UnL_Cont_to_BLK[BTH7] = CONVEYOR OUTFLOW
TRANSIT TIME = Cont_Trans_to_BLK_Pro[BTH7]
Transfering_UnL_Cont_to_BLK[BTH8] = CONVEYOR OUTFLOW
TRANSIT TIME = Cont_Trans_to_BLK_Pro[BTH8]
Transfering_UnL_Cont_to_BLK[BTH9] = CONVEYOR OUTFLOW
TRANSIT TIME = Cont_Trans_to_BLK_Pro[BTH9]
Transfering_UnL_Cont_to_BLK[BTH10] = CONVEYOR OUTFLOW
TRANSIT TIME = Cont_Trans_to_BLK_Pro[BTH10]
Transfering_UnL_Cont_to_BLK[BTH11] = CONVEYOR OUTFLOW
TRANSIT TIME = Cont_Trans_to_BLK_Pro[BTH11]
Transfering_UnL_Cont_to_BLK[BTH12] = CONVEYOR OUTFLOW
TRANSIT TIME = Cont_Trans_to_BLK_Pro[BTH12]
Transfering_UnL_Cont_to_BLK[BTH13] = CONVEYOR OUTFLOW
TRANSIT TIME = Cont_Trans_to_BLK_Pro[BTH13]
Transfering_UnL_Cont_to_BLK[BTH14] = CONVEYOR OUTFLOW
TRANSIT TIME = Cont_Trans_to_BLK_Pro[BTH14]
Transfering_UnL_Cont_to_BLK[BTH15] = CONVEYOR OUTFLOW
TRANSIT TIME = Cont_Trans_to_BLK_Pro[BTH15]

OUTFLOWS:
Cont_Enter_BLK_1[BTH1] =
Sorting_UnL_Cont_to_BLK[BTH1]*to_BLK_1[BTH1]
Cont_Enter_BLK_1[BTH2] =
Sorting_UnL_Cont_to_BLK[BTH2]*to_BLK_1[BTH2]
Cont_Enter_BLK_1[BTH3] =
Sorting_UnL_Cont_to_BLK[BTH3]*to_BLK_1[BTH3]
Cont_Enter_BLK_1[BTH4] =
Sorting_UnL_Cont_to_BLK[BTH4]*to_BLK_1[BTH4]
Cont_Enter_BLK_1[BTH5] =
Sorting_UnL_Cont_to_BLK[BTH5]*to_BLK_1[BTH5]
Cont_Enter_BLK_1[BTH6] =
Sorting_UnL_Cont_to_BLK[BTH6]*to_BLK_1[BTH6]
Cont_Enter_BLK_1[BTH7] =
Sorting_UnL_Cont_to_BLK[BTH7]*to_BLK_1[BTH7]
Cont_Enter_BLK_1[BTH8] =
Sorting_UnL_Cont_to_BLK[BTH8]*to_BLK_1[BTH8]
Cont_Enter_BLK_1[BTH9] =
Sorting_UnL_Cont_to_BLK[BTH9]*to_BLK_1[BTH9]
Cont_Enter_BLK_1[BTH10] =
Sorting_UnL_Cont_to_BLK[BTH10]*to_BLK_1[BTH10]

Cont_Enter_BLK_1[BTH11] =
Sorting_UnL_Cont_to_BLK[BTH11]*to_BLK_1[BTH11]
Cont_Enter_BLK_1[BTH12] =
Sorting_UnL_Cont_to_BLK[BTH12]*to_BLK_1[BTH12]
Cont_Enter_BLK_1[BTH13] =
Sorting_UnL_Cont_to_BLK[BTH13]*to_BLK_1[BTH13]
Cont_Enter_BLK_1[BTH14] =
Sorting_UnL_Cont_to_BLK[BTH14]*to_BLK_1[BTH14]
Cont_Enter_BLK_1[BTH15] =
Sorting_UnL_Cont_to_BLK[BTH15]*to_BLK_1[BTH15]
Cont_Enter_BLK_2[BTH1] =
Sorting_UnL_Cont_to_BLK[BTH1]*to_BLK_2[BTH1]
Cont_Enter_BLK_2[BTH2] =
Sorting_UnL_Cont_to_BLK[BTH2]*to_BLK_2[BTH2]
Cont_Enter_BLK_2[BTH3] =
Sorting_UnL_Cont_to_BLK[BTH3]*to_BLK_2[BTH3]
Cont_Enter_BLK_2[BTH4] =
Sorting_UnL_Cont_to_BLK[BTH4]*to_BLK_2[BTH4]
Cont_Enter_BLK_2[BTH5] =
Sorting_UnL_Cont_to_BLK[BTH5]*to_BLK_2[BTH5]
Cont_Enter_BLK_2[BTH6] =
Sorting_UnL_Cont_to_BLK[BTH6]*to_BLK_2[BTH6]
Cont_Enter_BLK_2[BTH7] =
Sorting_UnL_Cont_to_BLK[BTH7]*to_BLK_2[BTH7]
Cont_Enter_BLK_2[BTH8] =
Sorting_UnL_Cont_to_BLK[BTH8]*to_BLK_2[BTH8]
Cont_Enter_BLK_2[BTH9] =
Sorting_UnL_Cont_to_BLK[BTH9]*to_BLK_2[BTH9]
Cont_Enter_BLK_2[BTH10] =
Sorting_UnL_Cont_to_BLK[BTH10]*to_BLK_2[BTH10]
Cont_Enter_BLK_2[BTH11] =
Sorting_UnL_Cont_to_BLK[BTH11]*to_BLK_2[BTH11]
Cont_Enter_BLK_2[BTH12] =
Sorting_UnL_Cont_to_BLK[BTH12]*to_BLK_2[BTH12]
Cont_Enter_BLK_2[BTH13] =
Sorting_UnL_Cont_to_BLK[BTH13]*to_BLK_2[BTH13]
Cont_Enter_BLK_2[BTH14] =
Sorting_UnL_Cont_to_BLK[BTH14]*to_BLK_2[BTH14]
Cont_Enter_BLK_2[BTH15] =
Sorting_UnL_Cont_to_BLK[BTH15]*to_BLK_2[BTH15]
Cont_Enter_BLK_3[BTH1] =
Sorting_UnL_Cont_to_BLK[BTH1]*to_BLK_3[BTH1]
Cont_Enter_BLK_3[BTH2] =
Sorting_UnL_Cont_to_BLK[BTH2]*to_BLK_3[BTH2]
Cont_Enter_BLK_3[BTH3] =
Sorting_UnL_Cont_to_BLK[BTH3]*to_BLK_3[BTH3]
Cont_Enter_BLK_3[BTH4] =
Sorting_UnL_Cont_to_BLK[BTH4]*to_BLK_3[BTH4]
Cont_Enter_BLK_3[BTH5] =
Sorting_UnL_Cont_to_BLK[BTH5]*to_BLK_3[BTH5]

Cont_Enter_BLK_3[BTH6] =
Sorting_UnL_Cont_to_BLK[BTH6]*to_BLK_3[BTH6]
Cont_Enter_BLK_3[BTH7] =
Sorting_UnL_Cont_to_BLK[BTH7]*to_BLK_3[BTH7]
Cont_Enter_BLK_3[BTH8] =
Sorting_UnL_Cont_to_BLK[BTH8]*to_BLK_3[BTH8]
Cont_Enter_BLK_3[BTH9] =
Sorting_UnL_Cont_to_BLK[BTH9]*to_BLK_3[BTH9]
Cont_Enter_BLK_3[BTH10] =
Sorting_UnL_Cont_to_BLK[BTH10]*to_BLK_3[BTH10]
Cont_Enter_BLK_3[BTH11] =
Sorting_UnL_Cont_to_BLK[BTH11]*to_BLK_3[BTH11]
Cont_Enter_BLK_3[BTH12] =
Sorting_UnL_Cont_to_BLK[BTH12]*to_BLK_3[BTH12]
Cont_Enter_BLK_3[BTH13] =
Sorting_UnL_Cont_to_BLK[BTH13]*to_BLK_3[BTH13]
Cont_Enter_BLK_3[BTH14] =
Sorting_UnL_Cont_to_BLK[BTH14]*to_BLK_3[BTH14]
Cont_Enter_BLK_3[BTH15] =
Sorting_UnL_Cont_to_BLK[BTH15]*to_BLK_3[BTH15]
Cont_Enter_BLK_4[BTH1] =
Sorting_UnL_Cont_to_BLK[BTH1]*to_BLK_4[BTH1]
Cont_Enter_BLK_4[BTH2] =
Sorting_UnL_Cont_to_BLK[BTH2]*to_BLK_4[BTH2]
Cont_Enter_BLK_4[BTH3] =
Sorting_UnL_Cont_to_BLK[BTH3]*to_BLK_4[BTH3]
Cont_Enter_BLK_4[BTH4] =
Sorting_UnL_Cont_to_BLK[BTH4]*to_BLK_4[BTH4]
Cont_Enter_BLK_4[BTH5] =
Sorting_UnL_Cont_to_BLK[BTH5]*to_BLK_4[BTH5]
Cont_Enter_BLK_4[BTH6] =
Sorting_UnL_Cont_to_BLK[BTH6]*to_BLK_4[BTH6]
Cont_Enter_BLK_4[BTH7] =
Sorting_UnL_Cont_to_BLK[BTH7]*to_BLK_4[BTH7]
Cont_Enter_BLK_4[BTH8] =
Sorting_UnL_Cont_to_BLK[BTH8]*to_BLK_4[BTH8]
Cont_Enter_BLK_4[BTH9] =
Sorting_UnL_Cont_to_BLK[BTH9]*to_BLK_4[BTH9]
Cont_Enter_BLK_4[BTH10] =
Sorting_UnL_Cont_to_BLK[BTH10]*to_BLK_4[BTH10]
Cont_Enter_BLK_4[BTH11] =
Sorting_UnL_Cont_to_BLK[BTH11]*to_BLK_4[BTH11]
Cont_Enter_BLK_4[BTH12] =
Sorting_UnL_Cont_to_BLK[BTH12]*to_BLK_4[BTH12]
Cont_Enter_BLK_4[BTH13] =
Sorting_UnL_Cont_to_BLK[BTH13]*to_BLK_4[BTH13]
Cont_Enter_BLK_4[BTH14] =
Sorting_UnL_Cont_to_BLK[BTH14]*to_BLK_4[BTH14]
Cont_Enter_BLK_4[BTH15] =
Sorting_UnL_Cont_to_BLK[BTH15]*to_BLK_4[BTH15]

Cont_Enter_BLK_5[BTH1] =
Sorting_UnL_Cont_to_BLK[BTH1]*to_BLK_5[BTH1]
Cont_Enter_BLK_5[BTH2] =
Sorting_UnL_Cont_to_BLK[BTH2]*to_BLK_5[BTH2]
Cont_Enter_BLK_5[BTH3] =
Sorting_UnL_Cont_to_BLK[BTH3]*to_BLK_5[BTH3]
Cont_Enter_BLK_5[BTH4] =
Sorting_UnL_Cont_to_BLK[BTH4]*to_BLK_5[BTH4]
Cont_Enter_BLK_5[BTH5] =
Sorting_UnL_Cont_to_BLK[BTH5]*to_BLK_5[BTH5]
Cont_Enter_BLK_5[BTH6] =
Sorting_UnL_Cont_to_BLK[BTH6*to_BLK_5[BTH6]
Cont_Enter_BLK_5[BTH7] =
Sorting_UnL_Cont_to_BLK[BTH7]*to_BLK_5[BTH7]
Cont_Enter_BLK_5[BTH8] =
Sorting_UnL_Cont_to_BLK[BTH8]*to_BLK_5[BTH8]
Cont_Enter_BLK_5[BTH9] =
Sorting_UnL_Cont_to_BLK[BTH9]*to_BLK_5[BTH9]
Cont_Enter_BLK_5[BTH10] =
Sorting_UnL_Cont_to_BLK[BTH10]*to_BLK_5[BTH10]
Cont_Enter_BLK_5[BTH11] =
Sorting_UnL_Cont_to_BLK[BTH11]*to_BLK_5[BTH11]
Cont_Enter_BLK_5[BTH12] =
Sorting_UnL_Cont_to_BLK[BTH12]*to_BLK_5[BTH12]
Cont_Enter_BLK_5[BTH13] =
Sorting_UnL_Cont_to_BLK[BTH13]*to_BLK_5[BTH13]
Cont_Enter_BLK_5[BTH14] =
Sorting_UnL_Cont_to_BLK[BTH14]*to_BLK_5[BTH14]
Cont_Enter_BLK_5[BTH15] =
Sorting_UnL_Cont_to_BLK[BTH15]*to_BLK_5[BTH15]
Cont_Enter_BLK_6[BTH1] =
Sorting_UnL_Cont_to_BLK[BTH1]*to_BLK_6[BTH1]
Cont_Enter_BLK_6[BTH2] =
Sorting_UnL_Cont_to_BLK[BTH2]*to_BLK_6[BTH2]
Cont_Enter_BLK_6[BTH3] =
Sorting_UnL_Cont_to_BLK[BTH3]*to_BLK_6[BTH3]
Cont_Enter_BLK_6[BTH4] =
Sorting_UnL_Cont_to_BLK[BTH4]*to_BLK_6[BTH4]
Cont_Enter_BLK_6[BTH5] =
Sorting_UnL_Cont_to_BLK[BTH5]*to_BLK_6[BTH5]
Cont_Enter_BLK_6[BTH6] =
Sorting_UnL_Cont_to_BLK[BTH6]*to_BLK_6[BTH6]
Cont_Enter_BLK_6[BTH7] =
Sorting_UnL_Cont_to_BLK[BTH7]*to_BLK_6[BTH7]
Cont_Enter_BLK_6[BTH8] =
Sorting_UnL_Cont_to_BLK[BTH8]*to_BLK_6[BTH8]
Cont_Enter_BLK_6[BTH9] =
Sorting_UnL_Cont_to_BLK[BTH9]*to_BLK_6[BTH9]
Cont_Enter_BLK_6[BTH10] =
Sorting_UnL_Cont_to_BLK[BTH10]*to_BLK_6[BTH10]

Cont_Enter_BLK_6[BTH11] =
Sorting_UnL_Cont_to_BLK[BTH11]*to_BLK_6[BTH11]
Cont_Enter_BLK_6[BTH12] =
Sorting_UnL_Cont_to_BLK[BTH12]*to_BLK_6[BTH12]
Cont_Enter_BLK_6[BTH13] =
Sorting_UnL_Cont_to_BLK[BTH13]*to_BLK_6[BTH13]
Cont_Enter_BLK_6[BTH14] =
Sorting_UnL_Cont_to_BLK[BTH14]*to_BLK_6[BTH14]
Cont_Enter_BLK_6[BTH15] =
Sorting_UnL_Cont_to_BLK[BTH15]*to_BLK_6[BTH15]
Cont_Enter_BLK_7[BTH1] =
Sorting_UnL_Cont_to_BLK[BTH1]*to_BLK_7[BTH1]
Cont_Enter_BLK_7[BTH2] =
Sorting_UnL_Cont_to_BLK[BTH2]*to_BLK_7[BTH2]
Cont_Enter_BLK_7[BTH3] =
Sorting_UnL_Cont_to_BLK[BTH3]*to_BLK_7[BTH3]
Cont_Enter_BLK_7[BTH4] =
Sorting_UnL_Cont_to_BLK[BTH4]*to_BLK_7[BTH4]
Cont_Enter_BLK_7[BTH5] =
Sorting_UnL_Cont_to_BLK[BTH5]*to_BLK_7[BTH5]
Cont_Enter_BLK_7[BTH6] =
Sorting_UnL_Cont_to_BLK[BTH6]*to_BLK_7[BTH6]
Cont_Enter_BLK_7[BTH7] =
Sorting_UnL_Cont_to_BLK[BTH7]*to_BLK_7[BTH7]
Cont_Enter_BLK_7[BTH8] =
Sorting_UnL_Cont_to_BLK[BTH8]*to_BLK_7[BTH8]
Cont_Enter_BLK_7[BTH9] =
Sorting_UnL_Cont_to_BLK[BTH9]*to_BLK_7[BTH9]
Cont_Enter_BLK_7[BTH10] =
Sorting_UnL_Cont_to_BLK[BTH10]*to_BLK_7[BTH10]
Cont_Enter_BLK_7[BTH11] =
Sorting_UnL_Cont_to_BLK[BTH11]*to_BLK_7[BTH11]
Cont_Enter_BLK_7[BTH12] =
Sorting_UnL_Cont_to_BLK[BTH12]*to_BLK_7[BTH12]
Cont_Enter_BLK_7[BTH13] =
Sorting_UnL_Cont_to_BLK[BTH13]*to_BLK_7[BTH13]
Cont_Enter_BLK_7[BTH14] =
Sorting_UnL_Cont_to_BLK[BTH14]*to_BLK_7[BTH14]
Cont_Enter_BLK_7[BTH15] =
Sorting_UnL_Cont_to_BLK[BTH15]*to_BLK_7[BTH15]
Cont_Enter_BLK_8[BTH1] =
Sorting_UnL_Cont_to_BLK[BTH1]*to_BLK_8[BTH1]
Cont_Enter_BLK_8[BTH2] =
Sorting_UnL_Cont_to_BLK[BTH2]*to_BLK_8[BTH2]
Cont_Enter_BLK_8[BTH3] =
Sorting_UnL_Cont_to_BLK[BTH3]*to_BLK_8[BTH3]
Cont_Enter_BLK_8[BTH4] =
Sorting_UnL_Cont_to_BLK[BTH4]*to_BLK_8[BTH4]
Cont_Enter_BLK_8[BTH5] =
Sorting_UnL_Cont_to_BLK[BTH5]*to_BLK_8[BTH5]

Cont_Enter_BLK_8[BTH6] =
Sorting_UnL_Cont_to_BLK[BTH6]*to_BLK_8[BTH6]
Cont_Enter_BLK_8[BTH7] =
Sorting_UnL_Cont_to_BLK[BTH7]*to_BLK_8[BTH7]
Cont_Enter_BLK_8[BTH8] =
Sorting_UnL_Cont_to_BLK[BTH8]*to_BLK_8[BTH8]
Cont_Enter_BLK_8[BTH9] =
Sorting_UnL_Cont_to_BLK[BTH9]*to_BLK_8[BTH9]
Cont_Enter_BLK_8[BTH10] =
Sorting_UnL_Cont_to_BLK[BTH10]*to_BLK_8[BTH10]
Cont_Enter_BLK_8[BTH11] =
Sorting_UnL_Cont_to_BLK[BTH11]*to_BLK_8[BTH11]
Cont_Enter_BLK_8[BTH12] =
Sorting_UnL_Cont_to_BLK[BTH12]*to_BLK_8[BTH12]
Cont_Enter_BLK_8[BTH13] =
Sorting_UnL_Cont_to_BLK[BTH13]*to_BLK_8[BTH13]
Cont_Enter_BLK_8[BTH14] =
Sorting_UnL_Cont_to_BLK[BTH14]*to_BLK_8[BTH14]
Cont_Enter_BLK_8[BTH15] =
Sorting_UnL_Cont_to_BLK[BTH15]*to_BLK_8[BTH15]
Cont_Enter_BLK_9[BTH1] =
Sorting_UnL_Cont_to_BLK[BTH1]*to_BLK_9[BTH1]
Cont_Enter_BLK_9[BTH2] =
Sorting_UnL_Cont_to_BLK[BTH2]*to_BLK_9[BTH2]
Cont_Enter_BLK_9[BTH3] =
Sorting_UnL_Cont_to_BLK[BTH3]*to_BLK_9[BTH3]
Cont_Enter_BLK_9[BTH4] =
Sorting_UnL_Cont_to_BLK[BTH4]*to_BLK_9[BTH4]
Cont_Enter_BLK_9[BTH5] =
Sorting_UnL_Cont_to_BLK[BTH5]*to_BLK_9[BTH5]
Cont_Enter_BLK_9[BTH6] =
Sorting_UnL_Cont_to_BLK[BTH6]*to_BLK_9[BTH6]
Cont_Enter_BLK_9[BTH7] =
Sorting_UnL_Cont_to_BLK[BTH7]*to_BLK_9[BTH7]
Cont_Enter_BLK_9[BTH8] =
Sorting_UnL_Cont_to_BLK[BTH8]*to_BLK_9[BTH8]
Cont_Enter_BLK_9[BTH9] =
Sorting_UnL_Cont_to_BLK[BTH9]*to_BLK_9[BTH9]
Cont_Enter_BLK_9[BTH10] =
Sorting_UnL_Cont_to_BLK[BTH10]*to_BLK_9[BTH10]
Cont_Enter_BLK_9[BTH11] =
Sorting_UnL_Cont_to_BLK[BTH11]*to_BLK_9[BTH11]
Cont_Enter_BLK_9[BTH12] =
Sorting_UnL_Cont_to_BLK[BTH12]*to_BLK_9[BTH12]
Cont_Enter_BLK_9[BTH13] =
Sorting_UnL_Cont_to_BLK[BTH13]*to_BLK_9[BTH13]
Cont_Enter_BLK_9[BTH14] =
Sorting_UnL_Cont_to_BLK[BTH14]*to_BLK_9[BTH14]
Cont_Enter_BLK_9[BTH15] =
Sorting_UnL_Cont_to_BLK[BTH15]*to_BLK_9[BTH15]

Cont_Enter_BLK_10[BTH1] =
Sorting_UnL_Cont_to_BLK[BTH1]*to_BLK_10[BTH1]
Cont_Enter_BLK_10[BTH2] =
Sorting_UnL_Cont_to_BLK[BTH2]*to_BLK_10[BTH2]
Cont_Enter_BLK_10[BTH3] =
Sorting_UnL_Cont_to_BLK[BTH3]*to_BLK_10[BTH3]
Cont_Enter_BLK_10[BTH4] =
Sorting_UnL_Cont_to_BLK[BTH4]*to_BLK_10[BTH4]
Cont_Enter_BLK_10[BTH5] =
Sorting_UnL_Cont_to_BLK[BTH5]*to_BLK_10[BTH5]
Cont_Enter_BLK_10[BTH6] =
Sorting_UnL_Cont_to_BLK[BTH6]*to_BLK_10[BTH6]
Cont_Enter_BLK_10[BTH7] =
Sorting_UnL_Cont_to_BLK[BTH7]*to_BLK_10[BTH7]
Cont_Enter_BLK_10[BTH8] =
Sorting_UnL_Cont_to_BLK[BTH8]*to_BLK_10[BTH8]
Cont_Enter_BLK_10[BTH9] =
Sorting_UnL_Cont_to_BLK[BTH9]*to_BLK_10[BTH9]
Cont_Enter_BLK_10[BTH10] =
Sorting_UnL_Cont_to_BLK[BTH10]*to_BLK_10[BTH10]
Cont_Enter_BLK_10[BTH11] =
Sorting_UnL_Cont_to_BLK[BTH11]*to_BLK_10[BTH11]
Cont_Enter_BLK_10[BTH12] =
Sorting_UnL_Cont_to_BLK[BTH12]*to_BLK_10[BTH12]
Cont_Enter_BLK_10[BTH13] =
Sorting_UnL_Cont_to_BLK[BTH13]*to_BLK_10[BTH13]
Cont_Enter_BLK_10[BTH14] =
Sorting_UnL_Cont_to_BLK[BTH14]*to_BLK_10[BTH14]
Cont_Enter_BLK_10[BTH15] =
Sorting_UnL_Cont_to_BLK[BTH15]*to_BLK_10[BTH15]
Cont_Enter_BLK_11[BTH1] =
Sorting_UnL_Cont_to_BLK[BTH1]*to_BLK_11[BTH1]
Cont_Enter_BLK_11[BTH2] =
Sorting_UnL_Cont_to_BLK[BTH2]*to_BLK_11[BTH2]
Cont_Enter_BLK_11[BTH3] =
Sorting_UnL_Cont_to_BLK[BTH3]*to_BLK_11[BTH3]
Cont_Enter_BLK_11[BTH4] =
Sorting_UnL_Cont_to_BLK[BTH4]*to_BLK_11[BTH4]
Cont_Enter_BLK_11[BTH5] =
Sorting_UnL_Cont_to_BLK[BTH5]*to_BLK_11[BTH5]
Cont_Enter_BLK_11[BTH6] =
Sorting_UnL_Cont_to_BLK[BTH6]*to_BLK_11[BTH6]
Cont_Enter_BLK_11[BTH7] =
Sorting_UnL_Cont_to_BLK[BTH7]*to_BLK_11[BTH7]
Cont_Enter_BLK_11[BTH8] =
Sorting_UnL_Cont_to_BLK[BTH8]*to_BLK_11[BTH8]
Cont_Enter_BLK_11[BTH9] =
Sorting_UnL_Cont_to_BLK[BTH9]*to_BLK_11[BTH9]
Cont_Enter_BLK_11[BTH10] =
Sorting_UnL_Cont_to_BLK[BTH10]*to_BLK_11[BTH10]

```
Cont_Enter_BLK_11[BTH11] =
Sorting_UnL_Cont_to_BLK[BTH11]*to_BLK_11[BTH11]
Cont_Enter_BLK_11[BTH12] =
Sorting_UnL_Cont_to_BLK[BTH12]*to_BLK_11[BTH12]
Cont_Enter_BLK_11[BTH13] =
Sorting_UnL_Cont_to_BLK[BTH13]*to_BLK_11[BTH13]
Cont_Enter_BLK_11[BTH14] =
Sorting_UnL_Cont_to_BLK[BTH14]*to_BLK_11[BTH14]
Cont_Enter_BLK_11[BTH15] =
Sorting_UnL_Cont_to_BLK[BTH15]*to_BLK_11[BTH15]
Cont_Enter_BLK_12[BTH1] =
Sorting_UnL_Cont_to_BLK[BTH1]*to_BLK_12[BTH1]
Cont_Enter_BLK_12[BTH2] =
Sorting_UnL_Cont_to_BLK[BTH2]*to_BLK_12[BTH2]
Cont_Enter_BLK_12[BTH3] =
Sorting_UnL_Cont_to_BLK[BTH3]*to_BLK_12[BTH3]
Cont_Enter_BLK_12[BTH4] =
Sorting_UnL_Cont_to_BLK[BTH4]*to_BLK_12[BTH4]
Cont_Enter_BLK_12[BTH5] =
Sorting_UnL_Cont_to_BLK[BTH5]*to_BLK_12[BTH5]
Cont_Enter_BLK_12[BTH6] =
Sorting_UnL_Cont_to_BLK[BTH6]*to_BLK_12[BTH6]
Cont_Enter_BLK_12[BTH7] =
Sorting_UnL_Cont_to_BLK[BTH7]*to_BLK_12[BTH7]
Cont_Enter_BLK_12[BTH8] =
Sorting_UnL_Cont_to_BLK[BTH8]*to_BLK_12[BTH8]
Cont_Enter_BLK_12[BTH9] =
Sorting_UnL_Cont_to_BLK[BTH9]*to_BLK_12[BTH9]
Cont_Enter_BLK_12[BTH10] =
Sorting_UnL_Cont_to_BLK[BTH10]*to_BLK_12[BTH10]
Cont_Enter_BLK_12[BTH11] =
Sorting_UnL_Cont_to_BLK[BTH11]*to_BLK_12[BTH11]
Cont_Enter_BLK_12[BTH12] =
Sorting_UnL_Cont_to_BLK[BTH12]*to_BLK_12[BTH12]
Cont_Enter_BLK_12[BTH13] =
Sorting_UnL_Cont_to_BLK[BTH13]*to_BLK_12[BTH13]
Cont_Enter_BLK_12[BTH14] =
Sorting_UnL_Cont_to_BLK[BTH14]*to_BLK_12[BTH14]
Cont_Enter_BLK_12[BTH15] =
Sorting_UnL_Cont_to_BLK[BTH15]*to_BLK_12[BTH15]

Ttl_UnL_Cont[Berth](t) = Ttl_UnL_Cont[Berth](t - dt) + (Noname_175[Berth]) * dt
INIT Ttl_UnL_Cont[Berth] = 0
INFLOWS:
Noname_175[BTH1] = UnL_Cont_Enter_BTH[BTH1]
Noname_175[BTH2] = UnL_Cont_Enter_BTH[BTH2]
Noname_175[BTH3] = UnL_Cont_Enter_BTH[BTH3]
Noname_175[BTH4] = UnL_Cont_Enter_BTH[BTH4]
Noname_175[BTH5] = UnL_Cont_Enter_BTH[BTH5]
Noname_175[BTH6] = UnL_Cont_Enter_BTH[BTH6]
```

Noname_175[BTH7] = UnL_Cont_Enter_BTH[BTH7]
Noname_175[BTH8] = UnL_Cont_Enter_BTH[BTH8]
Noname_175[BTH9] = UnL_Cont_Enter_BTH[BTH9]
Noname_175[BTH10] = UnL_Cont_Enter_BTH[BTH10]
Noname_175[BTH11] = UnL_Cont_Enter_BTH[BTH11]
Noname_175[BTH12] = UnL_Cont_Enter_BTH[BTH12]
Noname_175[BTH13] = UnL_Cont_Enter_BTH[BTH13]
Noname_175[BTH14] = UnL_Cont_Enter_BTH[BTH14]
Noname_175[BTH15] = UnL_Cont_Enter_BTH[BTH15]

UnL_Cont_Q_to_Enter_BLK[Berth](t) = UnL_Cont_Q_to_Enter_BLK[Berth](t - dt) + (Cont_UnL_Complete[Berth] - Transfering_UnL_Cont_to_BLK[Berth]) * dt
INIT UnL_Cont_Q_to_Enter_BLK[Berth] = 0
INFLOWS:
Cont_UnL_Complete[BTH1] = CONVEYOR OUTFLOW
 TRANSIT TIME = Cont_UnL_Pro[BTH1]
Cont_UnL_Complete[BTH2] = CONVEYOR OUTFLOW
 TRANSIT TIME = Cont_UnL_Pro[BTH2]
Cont_UnL_Complete[BTH3] = CONVEYOR OUTFLOW
 TRANSIT TIME = Cont_UnL_Pro[BTH3]
Cont_UnL_Complete[BTH4] = CONVEYOR OUTFLOW
 TRANSIT TIME = Cont_UnL_Pro[BTH4]
Cont_UnL_Complete[BTH5] = CONVEYOR OUTFLOW
 TRANSIT TIME = Cont_UnL_Pro[BTH5]
Cont_UnL_Complete[BTH6] = CONVEYOR OUTFLOW
 TRANSIT TIME = Cont_UnL_Pro[BTH6]
Cont_UnL_Complete[BTH7] = CONVEYOR OUTFLOW
 TRANSIT TIME = Cont_UnL_Pro[BTH7]
Cont_UnL_Complete[BTH8] = CONVEYOR OUTFLOW
 TRANSIT TIME = Cont_UnL_Pro[BTH8]
Cont_UnL_Complete[BTH9] = CONVEYOR OUTFLOW
 TRANSIT TIME = Cont_UnL_Pro[BTH9]
Cont_UnL_Complete[BTH10] = CONVEYOR OUTFLOW
 TRANSIT TIME = Cont_UnL_Pro[BTH10]
Cont_UnL_Complete[BTH11] = CONVEYOR OUTFLOW
 TRANSIT TIME = Cont_UnL_Pro[BTH11]
Cont_UnL_Complete[BTH12] = CONVEYOR OUTFLOW
 TRANSIT TIME = Cont_UnL_Pro[BTH12]
Cont_UnL_Complete[BTH13] = CONVEYOR OUTFLOW
 TRANSIT TIME = Cont_UnL_Pro[BTH13]
Cont_UnL_Complete[BTH14] = CONVEYOR OUTFLOW
 TRANSIT TIME = Cont_UnL_Pro[BTH14]
Cont_UnL_Complete[BTH15] = CONVEYOR OUTFLOW
TRANSIT TIME = Cont_UnL_Pro[BTH15]
OUTFLOWS:
Transfering_UnL_Cont_to_BLK[BTH1] = CONVEYOR OUTFLOW
TRANSIT TIME = Cont_Trans_to_BLK_Pro[BTH1]
Transfering_UnL_Cont_to_BLK[BTH2] = CONVEYOR OUTFLOW
TRANSIT TIME = Cont_Trans_to_BLK_Pro[BTH2]
Transfering_UnL_Cont_to_BLK[BTH3] = CONVEYOR OUTFLOW

TRANSIT TIME = Cont_Trans_to_BLK_Pro[BTH3]
Transfering_UnL_Cont_to_BLK[BTH4] = CONVEYOR OUTFLOW
TRANSIT TIME = Cont_Trans_to_BLK_Pro[BTH4]
Transfering_UnL_Cont_to_BLK[BTH5] = CONVEYOR OUTFLOW
TRANSIT TIME = Cont_Trans_to_BLK_Pro[BTH5]
Transfering_UnL_Cont_to_BLK[BTH6] = CONVEYOR OUTFLOW
TRANSIT TIME = Cont_Trans_to_BLK_Pro[BTH6]
Transfering_UnL_Cont_to_BLK[BTH7] = CONVEYOR OUTFLOW
TRANSIT TIME = Cont_Trans_to_BLK_Pro[BTH7]
Transfering_UnL_Cont_to_BLK[BTH8] = CONVEYOR OUTFLOW
TRANSIT TIME = Cont_Trans_to_BLK_Pro[BTH8]
Transfering_UnL_Cont_to_BLK[BTH9] = CONVEYOR OUTFLOW
TRANSIT TIME = Cont_Trans_to_BLK_Pro[BTH9]
Transfering_UnL_Cont_to_BLK[BTH10] = CONVEYOR OUTFLOW
TRANSIT TIME = Cont_Trans_to_BLK_Pro[BTH10]
Transfering_UnL_Cont_to_BLK[BTH11] = CONVEYOR OUTFLOW
TRANSIT TIME = Cont_Trans_to_BLK_Pro[BTH11]
Transfering_UnL_Cont_to_BLK[BTH12] = CONVEYOR OUTFLOW
TRANSIT TIME = Cont_Trans_to_BLK_Pro[BTH12]
Transfering_UnL_Cont_to_BLK[BTH13] = CONVEYOR OUTFLOW
TRANSIT TIME = Cont_Trans_to_BLK_Pro[BTH13]
Transfering_UnL_Cont_to_BLK[BTH14] = CONVEYOR OUTFLOW
TRANSIT TIME = Cont_Trans_to_BLK_Pro[BTH14]
Transfering_UnL_Cont_to_BLK[BTH15] = CONVEYOR OUTFLOW
TRANSIT TIME = Cont_Trans_to_BLK_Pro[BTH15]

Accum_UnL_Cont = ARRAYSUM(Ttl_UnL_Cont[*])
Cont_Leaving_BLK_Rate[BTH1] = IF (Increase\Decrease_in_Load_Cont = 1) THEN (Increase_Loading[BTH1])ELSE (IF (Increase\Decrease_in_Load_Cont = 0) THEN (Decrease_Loading[BTH1])ELSE (0))
Cont_Leaving_BLK_Rate[BTH2] = IF (Increase\Decrease_in_Load_Cont = 1) THEN (Increase_Loading[BTH2])ELSE (IF (Increase\Decrease_in_Load_Cont = 0) THEN (Decrease_Loading[BTH2])ELSE (0))
Cont_Leaving_BLK_Rate[BTH3] = IF (Increase\Decrease_in_Load_Cont = 1) THEN (Increase_Loading[BTH3])ELSE (IF (Increase\Decrease_in_Load_Cont = 0) THEN (Decrease_Loading[BTH3])ELSE (0))
Cont_Leaving_BLK_Rate[BTH4] = IF (Increase\Decrease_in_Load_Cont = 1) THEN (Increase_Loading[BTH4])ELSE (IF (Increase\Decrease_in_Load_Cont = 0) THEN (Decrease_Loading[BTH4])ELSE (0))
Cont_Leaving_BLK_Rate[BTH5] = IF (Increase\Decrease_in_Load_Cont = 1) THEN (Increase_Loading[BTH5])ELSE (IF (Increase\Decrease_in_Load_Cont = 0) THEN (Decrease_Loading[BTH5])ELSE (0))
Cont_Leaving_BLK_Rate[BTH6] = IF (Increase\Decrease_in_Load_Cont = 1) THEN (Increase_Loading[BTH6])ELSE (IF (Increase\Decrease_in_Load_Cont = 0) THEN (Decrease_Loading[BTH6])ELSE (0))
Cont_Leaving_BLK_Rate[BTH7] = IF (Increase\Decrease_in_Load_Cont = 1) THEN (Increase_Loading[BTH7])ELSE (IF (Increase\Decrease_in_Load_Cont = 0) THEN (Decrease_Loading[BTH7])ELSE (0))

Cont_Leaving_BLK_Rate[BTH8] = IF (Increase\Decrease_in_Load_Cont = 1) THEN (Increase_Loading[BTH8])ELSE (IF (Increase\Decrease_in_Load_Cont = 0) THEN (Decrease_Loading[BTH8])ELSE (0))
Cont_Leaving_BLK_Rate[BTH9] = IF (Increase\Decrease_in_Load_Cont = 1) THEN (Increase_Loading[BTH9])ELSE (IF (Increase\Decrease_in_Load_Cont = 0) THEN (Decrease_Loading[BTH9])ELSE (0))
Cont_Leaving_BLK_Rate[BTH10] = IF (Increase\Decrease_in_Load_Cont = 1) THEN (Increase_Loading[BTH10])ELSE (IF (Increase\Decrease_in_Load_Cont = 0) THEN (Decrease_Loading[BTH10])ELSE (0))
Cont_Leaving_BLK_Rate[BTH11] = IF (Increase\Decrease_in_Load_Cont = 1) THEN (Increase_Loading[BTH11])ELSE (IF (Increase\Decrease_in_Load_Cont = 0) THEN (Decrease_Loading[BTH11])ELSE (0))
Cont_Leaving_BLK_Rate[BTH12] = IF (Increase\Decrease_in_Load_Cont = 1) THEN (Increase_Loading[BTH12])ELSE (IF (Increase\Decrease_in_Load_Cont = 0) THEN (Decrease_Loading[BTH12])ELSE (0))
Cont_Leaving_BLK_Rate[BTH13] = IF (Increase\Decrease_in_Load_Cont = 1) THEN (Increase_Loading[BTH13])ELSE (IF (Increase\Decrease_in_Load_Cont = 0) THEN (Decrease_Loading[BTH13])ELSE (0))
Cont_Leaving_BLK_Rate[BTH14] = IF (Increase\Decrease_in_Load_Cont = 1) THEN (Increase_Loading[BTH14])ELSE (IF (Increase\Decrease_in_Load_Cont = 0) THEN (Decrease_Loading[BTH14])ELSE (0))
Cont_Leaving_BLK_Rate[BTH15] = IF (Increase\Decrease_in_Load_Cont = 1) THEN (Increase_Loading[BTH15])ELSE (IF (Increase\Decrease_in_Load_Cont = 0) THEN (Decrease_Loading[BTH15])ELSE (0))
Cont_Load_Pro[BTH1] = IF (Transferring_Cont_to_BTH[BTH1] = 0)THEN (Transferring_Cont_to_BTH[BTH1] / 1)ELSE (Transferring_Cont_to_BTH[BTH1] / UnL_&_Load_Rate[BTH1])
Cont_Load_Pro[BTH2] = IF (Transferring_Cont_to_BTH[BTH2] = 0)THEN Transferring_Cont_to_BTH[BTH2] / 1)ELSE (Transferring_Cont_to_BTH[BTH2] / UnL_&_Load_Rate[BTH2])
Cont_Load_Pro[BTH3] = IF (Transferring_Cont_to_BTH[BTH3] = 0)THEN (Transferring_Cont_to_BTH[BTH3] / 1)ELSE (Transferring_Cont_to_BTH[BTH3] / UnL_&_Load_Rate[BTH3])
Cont_Load_Pro[BTH4] = IF (Transferring_Cont_to_BTH[BTH4] = 0)THEN (Transferring_Cont_to_BTH[BTH4] / 1)ELSE (Transferring_Cont_to_BTH[BTH4] / UnL_&_Load_Rate[BTH4])
Cont_Load_Pro[BTH5] = IF (Transferring_Cont_to_BTH[BTH5] = 0)THEN (Transferring_Cont_to_BTH[BTH5] / 1)ELSE (Transferring_Cont_to_BTH[BTH5] / UnL_&_Load_Rate[BTH5])
Cont_Load_Pro[BTH6] = IF (Transferring_Cont_to_BTH[BTH6] = 0)THEN (Transferring_Cont_to_BTH[BTH6] / 1)ELSE (Transferring_Cont_to_BTH[BTH6] / UnL_&_Load_Rate[BTH6])
Cont_Load_Pro[BTH7] = IF (Transferring_Cont_to_BTH[BTH7] = 0)THEN (Transferring_Cont_to_BTH[BTH7] / 1)ELSE (Transferring_Cont_to_BTH[BTH7] / UnL_&_Load_Rate[BTH7])
Cont_Load_Pro[BTH8] = IF (Transferring_Cont_to_BTH[BTH8] = 0)THEN (Transferring_Cont_to_BTH[BTH8] / 1)ELSE (Transferring_Cont_to_BTH[BTH8] / UnL_&_Load_Rate[BTH8])

Cont_Load_Pro[BTH9] = IF (Transferring_Cont_to_BTH[BTH9] = 0)THEN (Transferring_Cont_to_BTH[BTH9] / 1)ELSE (Transferring_Cont_to_BTH[BTH9] / UnL_&_Load_Rate[BTH9])
Cont_Load_Pro[BTH10] = IF (Transferring_Cont_to_BTH[BTH10] = 0)THEN (Transferring_Cont_to_BTH[BTH10] / 1)ELSE (Transferring_Cont_to_BTH[BTH10]/ UnL_&_Load_Rate[BTH10])
Cont_Load_Pro[BTH11] = IF (Transferring_Cont_to_BTH[BTH11] = 0)THEN (Transferring_Cont_to_BTH[BTH11] / 1)ELSE (Transferring_Cont_to_BTH[BTH11] / UnL_&_Load_Rate[BTH11])
Cont_Load_Pro[BTH12] = IF (Transferring_Cont_to_BTH[BTH12] = 0)THEN (Transferring_Cont_to_BTH[BTH12] / 1)ELSE (Transferring_Cont_to_BTH[BTH12] / UnL_&_Load_Rate[BTH12])
Cont_Load_Pro[BTH13] = IF (Transferring_Cont_to_BTH[BTH13] = 0)THEN (Transferring_Cont_to_BTH[BTH13] / 1)ELSE (Transferring_Cont_to_BTH[BTH13] / UnL_&_Load_Rate[BTH13])
Cont_Load_Pro[BTH14] = IF (Transferring_Cont_to_BTH[BTH14] = 0)THEN (Transferring_Cont_to_BTH[BTH14] / 1)ELSE (Transferring_Cont_to_BTH[BTH14] / UnL_&_Load_Rate[BTH14])
Cont_Load_Pro[BTH15] = IF (Transferring_Cont_to_BTH[BTH15] = 0)THEN (Transferring_Cont_to_BTH[BTH15] / 1)ELSE (Transferring_Cont_to_BTH[BTH15] / UnL_&_Load_Rate[BTH15])
Cont_Trans_to_BLK_Pro[BTH1] = IF (Cont_UnL_Complete[BTH1] = 0)THEN (Cont_UnL_Complete[BTH1] / 1)ELSE (Cont_UnL_Complete[BTH1] / PM_Transfer_Rate)
Cont_Trans_to_BLK_Pro[BTH2] = IF (Cont_UnL_Complete[BTH2] = 0)THEN (Cont_UnL_Complete[BTH2] / 1)ELSE (Cont_UnL_Complete[BTH2] / PM_Transfer_Rate)
Cont_Trans_to_BLK_Pro[BTH3] = IF (Cont_UnL_Complete[BTH3] = 0)THEN (Cont_UnL_Complete[BTH3] / 1)ELSE (Cont_UnL_Complete[BTH3] / PM_Transfer_Rate)
Cont_Trans_to_BLK_Pro[BTH4] = IF (Cont_UnL_Complete[BTH4] = 0)THEN (Cont_UnL_Complete[BTH4] / 1)ELSE (Cont_UnL_Complete[BTH4] / PM_Transfer_Rate)
Cont_Trans_to_BLK_Pro[BTH5] = IF (Cont_UnL_Complete[BTH5] = 0)THEN (Cont_UnL_Complete[BTH5] / 1)ELSE (Cont_UnL_Complete[BTH5] / PM_Transfer_Rate)
Cont_Trans_to_BLK_Pro[BTH6] = IF (Cont_UnL_Complete[BTH6] = 0)THEN (Cont_UnL_Complete[BTH6] / 1)ELSE (Cont_UnL_Complete[BTH6] / PM_Transfer_Rate)
Cont_Trans_to_BLK_Pro[BTH7] = IF (Cont_UnL_Complete[BTH7] = 0)THEN (Cont_UnL_Complete[BTH7] / 1)ELSE (Cont_UnL_Complete[BTH7] / PM_Transfer_Rate)
Cont_Trans_to_BLK_Pro[BTH8] = IF (Cont_UnL_Complete[BTH8] = 0)THEN (Cont_UnL_Complete[BTH8] / 1)ELSE (Cont_UnL_Complete[BTH8] / PM_Transfer_Rate)
Cont_Trans_to_BLK_Pro[BTH9] = IF (Cont_UnL_Complete[BTH9] = 0)THEN (Cont_UnL_Complete[BTH9] / 1)ELSE (Cont_UnL_Complete[BTH9] / PM_Transfer_Rate)

Cont_Trans_to_BLK_Pro[BTH10] = IF (Cont_UnL_Complete[BTH10] = 0)THEN (Cont_UnL_Complete[BTH10] / 1)ELSE (Cont_UnL_Complete[BTH10] / PM_Transfer_Rate)
Cont_Trans_to_BLK_Pro[BTH11] = IF (Cont_UnL_Complete[BTH11] = 0)THEN (Cont_UnL_Complete[BTH11] / 1)ELSE (Cont_UnL_Complete[BTH11] / PM_Transfer_Rate)
Cont_Trans_to_BLK_Pro[BTH12] = IF (Cont_UnL_Complete[BTH12] = 0)THEN (Cont_UnL_Complete[BTH12] / 1)ELSE (Cont_UnL_Complete[BTH12] / PM_Transfer_Rate)
Cont_Trans_to_BLK_Pro[BTH13] = IF (Cont_UnL_Complete[BTH13] = 0)THEN (Cont_UnL_Complete[BTH13] / 1)ELSE (Cont_UnL_Complete[BTH13] / PM_Transfer_Rate)
Cont_Trans_to_BLK_Pro[BTH14] = IF (Cont_UnL_Complete[BTH14] = 0)THEN (Cont_UnL_Complete[BTH14] / 1)ELSE (Cont_UnL_Complete[BTH14] / PM_Transfer_Rate)
Cont_Trans_to_BLK_Pro[BTH15] = IF (Cont_UnL_Complete[BTH15] = 0)THEN (Cont_UnL_Complete[BTH15] / 1)ELSE (Cont_UnL_Complete[BTH15] / PM_Transfer_Rate)
Cont_Transf_to_BTH_Pro[BTH1] = IF (Cont_Entering_BTH_for_Loading[BTH1] = 0)THEN (Cont_Entering_BTH_for_Loading[BTH1] / 1)ELSE (Cont_Entering_BTH_for_Loading[BTH1] / PM_Transfer_Rate)
Cont_Transf_to_BTH_Pro[BTH2] = IF (Cont_Entering_BTH_for_Loading[BTH2] = 0)THEN (Cont_Entering_BTH_for_Loading[BTH2] / 1)ELSE (Cont_Entering_BTH_for_Loading[BTH2] / PM_Transfer_Rate)
Cont_Transf_to_BTH_Pro[BTH3] = IF (Cont_Entering_BTH_for_Loading[BTH3] = 0)THEN (Cont_Entering_BTH_for_Loading[BTH3] / 1)ELSE (Cont_Entering_BTH_for_Loading[BTH3] / PM_Transfer_Rate)
Cont_Transf_to_BTH_Pro[BTH4] = IF (Cont_Entering_BTH_for_Loading[BTH4] = 0)THEN (Cont_Entering_BTH_for_Loading[BTH4] / 1)ELSE (Cont_Entering_BTH_for_Loading[BTH4] / PM_Transfer_Rate)
Cont_Transf_to_BTH_Pro[BTH5] = IF (Cont_Entering_BTH_for_Loading[BTH5] = 0)THEN (Cont_Entering_BTH_for_Loading[BTH5] / 1)ELSE (Cont_Entering_BTH_for_Loading[BTH5] / PM_Transfer_Rate)
Cont_Transf_to_BTH_Pro[BTH6] = IF (Cont_Entering_BTH_for_Loading[BTH6] = 0)THEN (Cont_Entering_BTH_for_Loading[BTH6] / 1)ELSE (Cont_Entering_BTH_for_Loading[BTH6] / PM_Transfer_Rate)
Cont_Transf_to_BTH_Pro[BTH7] = IF (Cont_Entering_BTH_for_Loading[BTH7] = 0)THEN (Cont_Entering_BTH_for_Loading[BTH7] / 1)ELSE (Cont_Entering_BTH_for_Loading[BTH7] / PM_Transfer_Rate)
Cont_Transf_to_BTH_Pro[BTH8] = IF (Cont_Entering_BTH_for_Loading[BTH8] = 0)THEN (Cont_Entering_BTH_for_Loading[BTH8] / 1)ELSE (Cont_Entering_BTH_for_Loading[BTH8] / PM_Transfer_Rate)
Cont_Transf_to_BTH_Pro[BTH9] = IF (Cont_Entering_BTH_for_Loading[BTH9] = 0)THEN (Cont_Entering_BTH_for_Loading[BTH9] / 1)ELSE (Cont_Entering_BTH_for_Loading[BTH9] / PM_Transfer_Rate)
Cont_Transf_to_BTH_Pro[BTH10] = IF (Cont_Entering_BTH_for_Loading[BTH10] = 0)THEN (Cont_Entering_BTH_for_Loading[BTH10] / 1)ELSE (Cont_Entering_BTH_for_Loading[BTH10] / PM_Transfer_Rate)

Cont_Transf_to_BTH_Pro[BTH11] = IF (
Cont_Entering_BTH_for_Loading[BTH11] = 0)THEN (
Cont_Entering_BTH_for_Loading[BTH11] / 1)ELSE (
Cont_Entering_BTH_for_Loading[BTH11] / PM_Transfer_Rate)
Cont_Transf_to_BTH_Pro[BTH12] = IF (
Cont_Entering_BTH_for_Loading[BTH12] = 0)THEN (
Cont_Entering_BTH_for_Loading[BTH12] / 1)ELSE (
Cont_Entering_BTH_for_Loading[BTH12] / PM_Transfer_Rate)
Cont_Transf_to_BTH_Pro[BTH13] = IF (
Cont_Entering_BTH_for_Loading[BTH13] = 0)THEN (
Cont_Entering_BTH_for_Loading[BTH13] / 1)ELSE (
Cont_Entering_BTH_for_Loading[BTH13] / PM_Transfer_Rate)
Cont_Transf_to_BTH_Pro[BTH14] = IF (
Cont_Entering_BTH_for_Loading[BTH14] = 0)THEN (
Cont_Entering_BTH_for_Loading[BTH14] / 1)ELSE (
Cont_Entering_BTH_for_Loading[BTH14] / PM_Transfer_Rate)
Cont_Transf_to_BTH_Pro[BTH15] = IF (
Cont_Entering_BTH_for_Loading[BTH15] = 0)THEN (
Cont_Entering_BTH_for_Loading[BTH15] / 1)ELSE (
Cont_Entering_BTH_for_Loading[BTH15] / PM_Transfer_Rate)
Cont_UnL_Pro[BTH1] = IF (UnL_Cont_Enter_BTH[BTH1] = 0)THEN (
UnL_Cont_Enter_BTH[BTH1] / 1)ELSE (UnL_Cont_Enter_BTH[BTH1] /
UnL_&_Load_Rate[BTH1])Cont_UnL_Pro[BTH2] = IF (
UnL_Cont_Enter_BTH[BTH1] = 0)THEN (UnL_Cont_Enter_BTH[BTH2] / 1)
ELSE (UnL_Cont_Enter_BTH[BTH2] / UnL_&_Load_Rate[BTH2])
Cont_UnL_Pro[BTH3] = IF (UnL_Cont_Enter_BTH[BTH1] = 0)THEN (
UnL_Cont_Enter_BTH[BTH3] / 1)ELSE (UnL_Cont_Enter_BTH[BTH3] /
UnL_&_Load_Rate[BTH3])
Cont_UnL_Pro[BTH4] = IF (UnL_Cont_Enter_BTH[BTH1] = 0)THEN (
UnL_Cont_Enter_BTH[BTH4] / 1)ELSE (UnL_Cont_Enter_BTH[BTH4] /
UnL_&_Load_Rate[BTH4])
Cont_UnL_Pro[BTH5] = IF (UnL_Cont_Enter_BTH[BTH1] = 0)THEN (
UnL_Cont_Enter_BTH[BTH5] / 1)ELSE (UnL_Cont_Enter_BTH[BTH5] /
UnL_&_Load_Rate[BTH5])
Cont_UnL_Pro[BTH6] = IF (UnL_Cont_Enter_BTH[BTH1] = 0)THEN (
UnL_Cont_Enter_BTH[BTH6] / 1)ELSE (UnL_Cont_Enter_BTH[BTH6] /
UnL_&_Load_Rate[BTH6])
Cont_UnL_Pro[BTH7] = IF (UnL_Cont_Enter_BTH[BTH7] = 0)THEN (
UnL_Cont_Enter_BTH[BTH7] / 1)ELSE (UnL_Cont_Enter_BTH[BTH7] /
UnL_&_Load_Rate[BTH7])
Cont_UnL_Pro[BTH8] = IF (UnL_Cont_Enter_BTH[BTH8] = 0)THEN (
UnL_Cont_Enter_BTH[BTH8] / 1)ELSE (UnL_Cont_Enter_BTH[BTH8] /
UnL_&_Load_Rate[BTH8])
Cont_UnL_Pro[BTH9] = IF (UnL_Cont_Enter_BTH[BTH9] = 0)THEN (
UnL_Cont_Enter_BTH[BTH9] / 1)ELSE (UnL_Cont_Enter_BTH[BTH9] /
UnL_&_Load_Rate[BTH9])
Cont_UnL_Pro[BTH10] = IF (UnL_Cont_Enter_BTH[BTH10] = 0)
THEN (UnL_Cont_Enter_BTH[BTH10] / 1)ELSE (
UnL_Cont_Enter_BTH[BTH10] / UnL_&_Load_Rate[BTH10])

Cont_UnL_Pro[BTH11] = IF (UnL_Cont_Enter_BTH[BTH11] = 0)THEN (UnL_Cont_Enter_BTH[BTH11] / 1)ELSE (UnL_Cont_Enter_BTH[BTH11] / UnL_&_Load_Rate[BTH11])
Cont_UnL_Pro[BTH12] = IF (UnL_Cont_Enter_BTH[BTH12] = 0)THEN (UnL_Cont_Enter_BTH[BTH12] / 1)ELSE (UnL_Cont_Enter_BTH[BTH12] / UnL_&_Load_Rate[BTH12])Cont_UnL_Pro[BTH13] = IF (UnL_Cont_Enter_BTH[BTH13] = 0)THEN (UnL_Cont_Enter_BTH[BTH13] / 1) ELSE (UnL_Cont_Enter_BTH[BTH13] / UnL_&_Load_Rate[BTH13])
Cont_UnL_Pro[BTH14] = IF (UnL_Cont_Enter_BTH[BTH14] = 0)THEN (UnL_Cont_Enter_BTH[BTH14] / 1)ELSE (UnL_Cont_Enter_BTH[BTH14] / UnL_&_Load_Rate[BTH14])
Cont_UnL_Pro[BTH15] = IF (UnL_Cont_Enter_BTH[BTH15] = 0)THEN (UnL_Cont_Enter_BTH[BTH15] / 1)ELSE (UnL_Cont_Enter_BTH[BTH15] / UnL_&_Load_Rate[BTH15])
Decrease_Loading[BTH1] = ML&FD_Load[BTH1]-ML&FD_Load[BTH1] * (Load_Cont_%_of_Change / 100)
Decrease_Loading[BTH2] = ML&FD_Load[BTH2]-ML&FD_Load[BTH2] * (Load_Cont_%_of_Change / 100)
Decrease_Loading[BTH3] = ML&FD_Load[BTH3]-ML&FD_Load[BTH3] * (Load_Cont_%_of_Change / 100)
Decrease_Loading[BTH4] = ML&FD_Load[BTH4]-ML&FD_Load[BTH4] * (Load_Cont_%_of_Change / 100)
Decrease_Loading[BTH5] = ML&FD_Load[BTH5]-ML&FD_Load[BTH5] * (Load_Cont_%_of_Change / 100)
Decrease_Loading[BTH6] = ML&FD_Load[BTH6]-ML&FD_Load[BTH6] * (Load_Cont_%_of_Change / 100)
Decrease_Loading[BTH7] = ML&FD_Load[BTH7]-ML&FD_Load[BTH7] * (Load_Cont_%_of_Change / 100)
Decrease_Loading[BTH8] = ML&FD_Load[BTH8]-ML&FD_Load[BTH8] * (Load_Cont_%_of_Change / 100)
Decrease_Loading[BTH9] = ML&FD_Load[BTH9]-ML&FD_Load[BTH9] * (Load_Cont_%_of_Change / 100)
Decrease_Loading[BTH10] = ML&FD_Load[BTH10]-ML&FD_Load[BTH10] * (Load_Cont_%_of_Change / 100)
Decrease_Loading[BTH11] = ML&FD_Load[BTH11]-ML&FD_Load[BTH11] * (Load_Cont_%_of_Change / 100)
Decrease_Loading[BTH12] = ML&FD_Load[BTH12]-ML&FD_Load[BTH12] * (Load_Cont_%_of_Change / 100)
Decrease_Loading[BTH13] = ML&FD_Load[BTH13]-ML&FD_Load[BTH13] * (Load_Cont_%_of_Change / 100)
Decrease_Loading[BTH14] = ML&FD_Load[BTH14]-ML&FD_Load[BTH14] * (Load_Cont_%_of_Change / 100)
Decrease_Loading[BTH15] = ML&FD_Load[BTH15]-ML&FD_Load[BTH15] * (Load_Cont_%_of_Change / 100)
Decrease_UnL[BTH1] = ML_&_FD_Cont_UnL[BTH1]-(ML_&_FD_Cont_UnL[BTH1] * (UnL_Cont_%_of_Change / 100))
Decrease_UnL[BTH2] = ML_&_FD_Cont_UnL[BTH2] -(ML_&_FD_Cont_UnL[BTH2] * (UnL_Cont_%_of_Change / 100))
Decrease_UnL[BTH3] = ML_&_FD_Cont_UnL[BTH3]-(ML_&_FD_Cont_UnL[BTH3] * (UnL_Cont_%_of_Change / 100))

Decrease_UnL[BTH4] = ML_&_FD_Cont_UnL[BTH4]-
(ML_&_FD_Cont_UnL[BTH4] * (UnL_Cont_%_of_Change / 100))
Decrease_UnL[BTH5] = ML_&_FD_Cont_UnL[BTH5]-
(ML_&_FD_Cont_UnL[BTH5] * (UnL_Cont_%_of_Change / 100))
Decrease_UnL[BTH6] = ML_&_FD_Cont_UnL[BTH6]-
(ML_&_FD_Cont_UnL[BTH6] * (UnL_Cont_%_of_Change / 100))
Decrease_UnL[BTH7] = ML_&_FD_Cont_UnL[BTH7]-
(ML_&_FD_Cont_UnL[BTH7] * (UnL_Cont_%_of_Change / 100))
Decrease_UnL[BTH8] = ML_&_FD_Cont_UnL[BTH8]-
(ML_&_FD_Cont_UnL[BTH8] * (UnL_Cont_%_of_Change / 100))
Decrease_UnL[BTH9] = ML_&_FD_Cont_UnL[BTH9]-
(ML_&_FD_Cont_UnL[BTH9] * (UnL_Cont_%_of_Change / 100))
Decrease_UnL[BTH10] = ML_&_FD_Cont_UnL[BTH10]-
(ML_&_FD_Cont_UnL[BTH10] * (UnL_Cont_%_of_Change / 100))
Decrease_UnL[BTH11] = ML_&_FD_Cont_UnL[BTH11]-
(ML_&_FD_Cont_UnL[BTH11] * (UnL_Cont_%_of_Change / 100))
Decrease_UnL[BTH12] = ML_&_FD_Cont_UnL[BTH12]-
(ML_&_FD_Cont_UnL[BTH12] * (UnL_Cont_%_of_Change / 100))
Decrease_UnL[BTH13] = ML_&_FD_Cont_UnL[BTH13]-
(ML_&_FD_Cont_UnL[BTH13] * (UnL_Cont_%_of_Change / 100))
Decrease_UnL[BTH14] = ML_&_FD_Cont_UnL[BTH14]-
(ML_&_FD_Cont_UnL[BTH14] * (UnL_Cont_%_of_Change / 100))
Decrease_UnL[BTH15] = ML_&_FD_Cont_UnL[BTH15]-
(ML_&_FD_Cont_UnL[BTH15] * (UnL_Cont_%_of_Change / 100))
FD_Cont_UnL_Rate[BTH1] = FD_Enter_BTH[BTH1]*FD_UnL_Rd[BTH1]
FD_Cont_UnL_Rate[BTH2] = FD_Enter_BTH[BTH2]*FD_UnL_Rd[BTH2]
FD_Cont_UnL_Rate[BTH3] = FD_Enter_BTH[BTH3]*FD_UnL_Rd[BTH3]
FD_Cont_UnL_Rate[BTH4] = FD_Enter_BTH[BTH4]*FD_UnL_Rd[BTH4]
FD_Cont_UnL_Rate[BTH5] = FD_Enter_BTH[BTH5]*FD_UnL_Rd[BTH5]
FD_Cont_UnL_Rate[BTH6] = FD_Enter_BTH[BTH6]*FD_UnL_Rd[BTH6]
FD_Cont_UnL_Rate[BTH7] = FD_Enter_BTH[BTH7]*FD_UnL_Rd[BTH7]
FD_Cont_UnL_Rate[BTH8] = FD_Enter_BTH[BTH8]*FD_UnL_Rd[BTH8]
FD_Cont_UnL_Rate[BTH9] = FD_Enter_BTH[BTH9]*FD_UnL_Rd[BTH9]
FD_Cont_UnL_Rate[BTH10] = FD_Enter_BTH[BTH10]*FD_UnL_Rd[BTH10]
FD_Cont_UnL_Rate[BTH11] = FD_Enter_BTH[BTH11]*FD_UnL_Rd[BTH11]
FD_Cont_UnL_Rate[BTH12] = FD_Enter_BTH[BTH12]*FD_UnL_Rd[BTH12]
FD_Cont_UnL_Rate[BTH13] = FD_Enter_BTH[BTH13]*FD_UnL_Rd[BTH13]
FD_Cont_UnL_Rate[BTH14] = FD_Enter_BTH[BTH14]*FD_UnL_Rd[BTH14]
FD_Cont_UnL_Rate[BTH15] = FD_Enter_BTH[BTH15]*FD_UnL_Rd[BTH15]
FD_Load_Rate[BTH1] = FD_Enter_BTH[BTH1]*FD_Load_Rd[BTH1]
FD_Load_Rate[BTH2] = FD_Enter_BTH[BTH2]*FD_Load_Rd[BTH2]
FD_Load_Rate[BTH3] = FD_Enter_BTH[BTH3]*FD_Load_Rd[BTH3]
FD_Load_Rate[BTH4] = FD_Enter_BTH[BTH4]*FD_Load_Rd[BTH4]
FD_Load_Rate[BTH5] = FD_Enter_BTH[BTH5]*FD_Load_Rd[BTH5]
FD_Load_Rate[BTH6] = FD_Enter_BTH[BTH6]*FD_Load_Rd[BTH6]
FD_Load_Rate[BTH7] = FD_Enter_BTH[BTH7]*FD_Load_Rd[BTH7]
FD_Load_Rate[BTH8] = FD_Enter_BTH[BTH8]*FD_Load_Rd[BTH8]
FD_Load_Rate[BTH9] = FD_Enter_BTH[BTH9]*FD_Load_Rd[BTH9]
FD_Load_Rate[BTH10] = FD_Enter_BTH[BTH10]*FD_Load_Rd[BTH10]
FD_Load_Rate[BTH11] = FD_Enter_BTH[BTH11]*FD_Load_Rd[BTH11]

FD_Load_Rate[BTH12] = FD_Enter_BTH[BTH12]*FD_Load_Rd[BTH12]
FD_Load_Rate[BTH13] = FD_Enter_BTH[BTH13]*FD_Load_Rd[BTH13]
FD_Load_Rate[BTH14] = FD_Enter_BTH[BTH14]*FD_Load_Rd[BTH14]
FD_Load_Rate[BTH15] = FD_Enter_BTH[BTH15]*FD_Load_Rd[BTH15]
FD_Load_Rd[BTH1] = RANDOM (38 , 1260 , 28)
FD_Load_Rd[BTH2] = RANDOM (108 , 1371 , 28)
FD_Load_Rd[BTH3] = RANDOM (85 , 443 , 28)
FD_Load_Rd[BTH4] = RANDOM (10 , 183 , 28)
FD_Load_Rd[BTH5] = RANDOM (35 , 929 , 28)
FD_Load_Rd[BTH6] = RANDOM (58 , 529 , 28)
FD_Load_Rd[BTH7] = RANDOM (252 , 1136 , 28)
FD_Load_Rd[BTH8] = RANDOM (280 , 1444 , 28)
FD_Load_Rd[BTH9] = RANDOM (24 , 922 , 28)
FD_Load_Rd[BTH10] = RANDOM (24 , 1444 , 28)
FD_Load_Rd[BTH11] = RANDOM (24 , 1444 , 28)
FD_Load_Rd[BTH12] = RANDOM (24 , 1444 , 28)
FD_Load_Rd[BTH13] = RANDOM (24 , 1444 , 28)
FD_Load_Rd[BTH14] = RANDOM (24 , 1444 , 28)
FD_Load_Rd[BTH15] = RANDOM (24 , 1444 , 28)
FD_UnL_Rd[BTH1] = RANDOM (214 , 1895 , 28)
FD_UnL_Rd[BTH2] = RANDOM (10 , 1262 , 28)
FD_UnL_Rd[BTH3] = RANDOM (25 , 500 , 28)
FD_UnL_Rd[BTH4] = RANDOM (24 , 251 , 28)
FD_UnL_Rd[BTH5] = RANDOM (43 , 1440 , 28)
FD_UnL_Rd[BTH6] = RANDOM (380 , 915 , 28)
FD_UnL_Rd[BTH7] = RANDOM (100 , 981 , 28)
FD_UnL_Rd[BTH8] = RANDOM (80 , 2458 , 28)
FD_UnL_Rd[BTH9] = RANDOM (8 , 3262 , 28)
FD_UnL_Rd[BTH10] = RANDOM (8 , 3262 , 28)
FD_UnL_Rd[BTH11] = RANDOM (8 , 3262 , 28)
FD_UnL_Rd[BTH12] = RANDOM (8 , 3262 , 28)
FD_UnL_Rd[BTH13] = RANDOM (8 , 3262 , 28)
FD_UnL_Rd[BTH14] = RANDOM (8 , 3262 , 28)
FD_UnL_Rd[BTH15] = RANDOM (8 , 3262 , 28)
frm_BLK_1[BTH1] = 20.49551224/100frm_BLK_1[BTH2] = 19.40455368/100
frm_BLK_1[BTH3] = 1.341240187/100frm_BLK_1[BTH4] = 6.684220501/100
frm_BLK_1[BTH5] = 2.659569304/100frm_BLK_1[BTH6] = 0.368416598/100
frm_BLK_1[BTH7] = 0.172181552/100frm_BLK_1[BTH8] = 4.951656992/100
frm_BLK_1[BTH9] = 4.951656992/100frm_BLK_1[BTH10] = 4.951656992/100
frm_BLK_1[BTH11] = 4.951656992/100frm_BLK_1[BTH12] = 4.951656992/100
frm_BLK_1[BTH13] = 4.951656992/100frm_BLK_1[BTH14] = 4.951656992/100
frm_BLK_1[BTH15] = 4.951656992/100frm_BLK_10[BTH1] = 2/100
frm_BLK_10[BTH2] = 0.103970393/100frm_BLK_10[BTH3] = 3/100
frm_BLK_10[BTH4] = 0.075134023/100frm_BLK_10[BTH5] = 4/100
frm_BLK_10[BTH6] = 14/100frm_BLK_10[BTH7] = 16/100
frm_BLK_10[BTH8] = 15/100frm_BLK_10[BTH9] = 15/100
frm_BLK_10[BTH10] = 15/100frm_BLK_10[BTH11] = 15/100
frm_BLK_10[BTH12] = 15/100frm_BLK_10[BTH13] = 15/100
frm_BLK_10[BTH14] = 15/100frm_BLK_10[BTH15] = 15/100
frm_BLK_11[BTH1] = 0.303809765/100frm_BLK_11[BTH2] = 0.346067517/100

frm_BLK_11[BTH3] = 1/100frm_BLK_11[BTH4] = 0.289029079/100
frm_BLK_11[BTH5] = 1/100frm_BLK_11[BTH6] = 7/100
frm_BLK_11[BTH7] = 14/100frm_BLK_11[BTH8] = 20/100
frm_BLK_11[BTH9] = 20/100frm_BLK_11[BTH10] = 20/100
frm_BLK_11[BTH11] = 20/100frm_BLK_11[BTH12] = 20/100
frm_BLK_11[BTH13] = 20/100frm_BLK_11[BTH14] = 20/100
frm_BLK_11[BTH15] = 20/100frm_BLK_12[BTH1] = 0.284480921/100
frm_BLK_12[BTH2] = 1/100frm_BLK_12[BTH3] = 0.437330163/100
frm_BLK_12[BTH4] = 1/100frm_BLK_12[BTH5] = 1/100
frm_BLK_12[BTH6] = 8/100frm_BLK_12[BTH7] = 12/100
frm_BLK_12[BTH8] = 12/100frm_BLK_12[BTH9] = 12/100
frm_BLK_12[BTH10] = 12/100frm_BLK_12[BTH11] = 12/100
frm_BLK_12[BTH12] = 12/100frm_BLK_12[BTH13] = 12/100
frm_BLK_12[BTH14] = 12/100frm_BLK_12[BTH15] = 12/100
frm_BLK_2[BTH1] = 16/100frm_BLK_2[BTH2] = 28/100
frm_BLK_2[BTH3] = 7/100frm_BLK_2[BTH4] = 15/100
frm_BLK_2[BTH5] = 6/100frm_BLK_2[BTH6] = 2/100
frm_BLK_2[BTH7] = 2/100frm_BLK_2[BTH8] = 5/100
frm_BLK_2[BTH9] = 5/100frm_BLK_2[BTH10] = 5/100
frm_BLK_2[BTH11] = 5/100frm_BLK_2[BTH12] = 5/100
frm_BLK_2[BTH13] = 5/100frm_BLK_2[BTH14] = 5/100
frm_BLK_2[BTH15] = 5/100frm_BLK_3[BTH1] = 17/100
frm_BLK_3[BTH2] = 15/100frm_BLK_3[BTH3] = 23/100
frm_BLK_3[BTH4] = 18/100frm_BLK_3[BTH5] = 11/100
frm_BLK_3[BTH6] = 0.448531693/100frm_BLK_3[BTH7] = 0.377550024/100
frm_BLK_3[BTH8] = 2/100frm_BLK_3[BTH9] = 2/100
frm_BLK_3[BTH10] = 2/100frm_BLK_3[BTH11] = 2/100
frm_BLK_3[BTH12] = 2/100frm_BLK_3[BTH13] = 2/100
frm_BLK_3[BTH14] = 2/100frm_BLK_3[BTH15] = 2/100
frm_BLK_4[BTH1] = 18/100frm_BLK_4[BTH2] = 13/100
frm_BLK_4[BTH3] = 15/100frm_BLK_4[BTH4] = 20/100
frm_BLK_4[BTH5] = 13/100frm_BLK_4[BTH6] = 3.299952044/100
frm_BLK_4[BTH7] = 0.511078575/100frm_BLK_4[BTH8] = 4/100
frm_BLK_4[BTH9] = 4/100frm_BLK_4[BTH10] = 4/100
frm_BLK_4[BTH11] = 4/100frm_BLK_4[BTH12] = 4/100
frm_BLK_4[BTH13] = 4/100frm_BLK_4[BTH14] = 4/100
frm_BLK_4[BTH15] = 4/100frm_BLK_5[BTH1] = 8/100
frm_BLK_5[BTH2] = 9/100frm_BLK_5[BTH3] = 11/100
frm_BLK_5[BTH4] = 12/100frm_BLK_5[BTH5] = 19/100
frm_BLK_5[BTH6] = 7/100frm_BLK_5[BTH7] = 2/100
frm_BLK_5[BTH8] = 7/100frm_BLK_5[BTH9] = 7/100
frm_BLK_5[BTH10] = 7/100frm_BLK_5[BTH11] = 7/100
frm_BLK_5[BTH12] = 7/100frm_BLK_5[BTH13] = 7/100
frm_BLK_5[BTH14] = 7/100frm_BLK_5[BTH15] = 7/100
frm_BLK_6[BTH1] = 8/100frm_BLK_6[BTH2] = 9/100
frm_BLK_6[BTH3] = 17/100frm_BLK_6[BTH4] = 16/100
frm_BLK_6[BTH5] = 11/100frm_BLK_6[BTH6] = 10/100
frm_BLK_6[BTH7] = 16/100frm_BLK_6[BTH8] = 4/100
frm_BLK_6[BTH9] = 4/100frm_BLK_6[BTH10] = 4/100
frm_BLK_6[BTH11] = 4/100frm_BLK_6[BTH12] = 4/100

frm_BLK_6[BTH13] = 4/100frm_BLK_6[BTH14] = 4/100
frm_BLK_6[BTH15] = 4/100frm_BLK_7[BTH1] = 4/100
frm_BLK_7[BTH2] = 4/100frm_BLK_7[BTH3] = 12/100
frm_BLK_7[BTH4] = 8/100frm_BLK_7[BTH5] = 11/100
frm_BLK_7[BTH6] = 20.81187057/100frm_BLK_7[BTH7] = 13.2193265/100
frm_BLK_7[BTH8] = 7/100frm_BLK_7[BTH9] = 7/100
frm_BLK_7[BTH10] = 7/100frm_BLK_7[BTH11] = 7/100
frm_BLK_7[BTH12] = 7/100frm_BLK_7[BTH13] = 7/100
frm_BLK_7[BTH14] = 7/100frm_BLK_7[BTH15] = 7/100
frm_BLK_8[BTH1] = 3.30176289/100frm_BLK_8[BTH2] = 1.642882345/100
frm_BLK_8[BTH3] = 7.859205163/100frm_BLK_8[BTH4] = 2.088184329/100
frm_BLK_8[BTH5] = 10.29300888/100frm_BLK_8[BTH6] = 14.76148834/100
frm_BLK_8[BTH7] = 12.38613958/100frm_BLK_8[BTH8] = 5.750451556/100
frm_BLK_8[BTH9] = 5.750451556/100frm_BLK_8[BTH10] = 5.750451556/100
frm_BLK_8[BTH11] = 5.750451556/100frm_BLK_8[BTH12] = 5.750451556/100
frm_BLK_8[BTH13] = 5.750451556/100frm_BLK_8[BTH14] = 5.750451556/100
frm_BLK_8[BTH15] = 5.750451556/100frm_BLK_9[BTH1] = 4/100
frm_BLK_9[BTH2] = 1/100frm_BLK_9[BTH3] = 3/100
frm_BLK_9[BTH4] = 1/100frm_BLK_9[BTH5] = 12/100
frm_BLK_9[BTH6] = 12/100frm_BLK_9[BTH7] = 11/100
frm_BLK_9[BTH8] = 13/100frm_BLK_9[BTH9] = 13/100
frm_BLK_9N[BTH10] = 13/100frm_BLK_9[BTH11] = 13/100
frm_BLK_9[BTH12] = 13/100frm_BLK_9[BTH13] = 13/100
frm_BLK_9[BTH14] = 13/100frm_BLK_9[BTH15] = 13/100
Increase\Decrease_in_Load_Cont = 1
Increase\Decrease_in_UnL_Cont = 0
Increase_Loading[BTH1] = (ML&FD_Load[BTH1] * (Load_Cont_%_of_Change / 100))+ML&FD_Load[BTH1]
Increase_Loading[BTH2] = (ML&FD_Load[BTH2] * (Load_Cont_%_of_Change / 100))+ML&FD_Load[BTH2]
Increase_Loading[BTH3] = (ML&FD_Load[BTH3] * (Load_Cont_%_of_Change / 100))+ML&FD_Load[BTH3]
Increase_Loading[BTH4] = (ML&FD_Load[BTH4] * (Load_Cont_%_of_Change / 100))+ML&FD_Load[BTH4]
Increase_Loading[BTH5] = (ML&FD_Load[BTH5] * (Load_Cont_%_of_Change / 100))+ML&FD_Load[BTH5]
Increase_Loading[BTH6] = (ML&FD_Load[BTH6] * (Load_Cont_%_of_Change / 100))+ML&FD_Load[BTH6]
Increase_Loading[BTH7] = (ML&FD_Load[BTH7] * (Load_Cont_%_of_Change / 100))+ML&FD_Load[BTH7]
Increase_Loading[BTH8] = (ML&FD_Load[BTH8] * (Load_Cont_%_of_Change / 100))+ML&FD_Load[BTH8]
Increase_Loading[BTH9] = (ML&FD_Load[BTH9] * (Load_Cont_%_of_Change / 100))+ML&FD_Load[BTH9]
Increase_Loading[BTH10] = (ML&FD_Load[BTH10] * (Load_Cont_%_of_Change / 100))+ML&FD_Load[BTH10]
Increase_Loading[BTH11] = (ML&FD_Load[BTH11] * (Load_Cont_%_of_Change / 100))+ML&FD_Load[BTH11]
Increase_Loading[BTH12] = (ML&FD_Load[BTH12] * (Load_Cont_%_of_Change / 100))+ML&FD_Load[BTH12]

```
Increase_Loading[BTH13] = ( ML&FD_Load[BTH13] * (
Load_Cont_%_of_Change / 100 ) )+ML&FD_Load[BTH13]
Increase_Loading[BTH14] = ( ML&FD_Load[BTH14] * (
Load_Cont_%_of_Change / 100 ) )+ML&FD_Load[BTH14]
Increase_Loading[BTH15] = ( ML&FD_Load[BTH15] * (
Load_Cont_%_of_Change / 100 ) )+ML&FD_Load[BTH15]
Increase_UnL[BTH1] = ( ML_&_FD_Cont_UnL[BTH1] * (
UnL_Cont_%_of_Change / 100 ) )+ML_&_FD_Cont_UnL[BTH1]
Increase_UnL[BTH2] = ( ML_&_FD_Cont_UnL[BTH2] * (
UnL_Cont_%_of_Change / 100 ) )+ML_&_FD_Cont_UnL[BTH2]
Increase_UnL[BTH3] = ( ML_&_FD_Cont_UnL[BTH3] * (
UnL_Cont_%_of_Change / 100 ) )+ML_&_FD_Cont_UnL[BTH3]
Increase_UnL[BTH4] = ( ML_&_FD_Cont_UnL[BTH4] * (
UnL_Cont_%_of_Change / 100 ) )+ML_&_FD_Cont_UnL[BTH4]
Increase_UnL[BTH5] = ( ML_&_FD_Cont_UnL[BTH5] * (
UnL_Cont_%_of_Change / 100 ) )+ML_&_FD_Cont_UnL[BTH5]
Increase_UnL[BTH6] = ( ML_&_FD_Cont_UnL[BTH6] * (
UnL_Cont_%_of_Change / 100 ) )+ML_&_FD_Cont_UnL[BTH6]
Increase_UnL[BTH7] = ( ML_&_FD_Cont_UnL[BTH7] * (
UnL_Cont_%_of_Change / 100 ) )+ML_&_FD_Cont_UnL[BTH7]
Increase_UnL[BTH8] = ( ML_&_FD_Cont_UnL[BTH8] * (
UnL_Cont_%_of_Change / 100 ) )+ML_&_FD_Cont_UnL[BTH8]
Increase_UnL[BTH9] = ( ML_&_FD_Cont_UnL[BTH9] * (
UnL_Cont_%_of_Change / 100 ) )+ML_&_FD_Cont_UnL[BTH9]
Increase_UnL[BTH10] = ( ML_&_FD_Cont_UnL[BTH10] * (
UnL_Cont_%_of_Change / 100 ) )+ML_&_FD_Cont_UnL[BTH10]
Increase_UnL[BTH11] = ( ML_&_FD_Cont_UnL[BTH11] * (
UnL_Cont_%_of_Change / 100 ) )+ML_&_FD_Cont_UnL[BTH11]
Increase_UnL[BTH12] = ( ML_&_FD_Cont_UnL[BTH12] * (
UnL_Cont_%_of_Change / 100 ) )+ML_&_FD_Cont_UnL[BTH12]
Increase_UnL[BTH13] = ( ML_&_FD_Cont_UnL[BTH13] * (
UnL_Cont_%_of_Change / 100 ) )+ML_&_FD_Cont_UnL[BTH13]
Increase_UnL[BTH14] = ( ML_&_FD_Cont_UnL[BTH14] * (
UnL_Cont_%_of_Change / 100 ) )+ML_&_FD_Cont_UnL[BTH14]
Increase_UnL[BTH15] = ( ML_&_FD_Cont_UnL[BTH15] * (
UnL_Cont_%_of_Change / 100 ) )+ML_&_FD_Cont_UnL[BTH15]
Load_Cont_%_of_Change = 0
ML&FD_Cont_UnL_Rate[BTH1] = ML&FD_Enter_BTH[BTH1]
*ML&FD_UnL_Rd[BTH1]
ML&FD_Cont_UnL_Rate[BTH2] = ML&FD_Enter_BTH[BTH2]
*ML&FD_UnL_Rd[BTH2]
ML&FD_Cont_UnL_Rate[BTH3] = ML&FD_Enter_BTH[BTH3]
*ML&FD_UnL_Rd[BTH3]
ML&FD_Cont_UnL_Rate[BTH4] = ML&FD_Enter_BTH[BTH4]
*ML&FD_UnL_Rd[BTH4]
ML&FD_Cont_UnL_Rate[BTH5] = ML&FD_Enter_BTH[BTH5]
*ML&FD_UnL_Rd[BTH5]
ML&FD_Cont_UnL_Rate[BTH6] = ML&FD_Enter_BTH[BTH6]
*ML&FD_UnL_Rd[BTH6]
ML&FD_Cont_UnL_Rate[BTH7] = ML&FD_Enter_BTH[BTH7]
```

```
*ML&FD_UnL_Rd[BTH7]
ML&FD_Cont_UnL_Rate[BTH8] = ML&FD_Enter_BTH[BTH8]
*ML&FD_UnL_Rd[BTH8]
ML&FD_Cont_UnL_Rate[BTH9] = ML&FD_Enter_BTH[BTH9]
*ML&FD_UnL_Rd[BTH9]
ML&FD_Cont_UnL_Rate[BTH10] = ML&FD_Enter_BTH[BTH10]
*ML&FD_UnL_Rd[BTH10]
ML&FD_Cont_UnL_Rate[BTH11] = ML&FD_Enter_BTH[BTH11]
*ML&FD_UnL_Rd[BTH11]
ML&FD_Cont_UnL_Rate[BTH12] = ML&FD_Enter_BTH[BTH12]
*ML&FD_UnL_Rd[BTH12]
ML&FD_Cont_UnL_Rate[BTH13] = ML&FD_Enter_BTH[BTH13]
*ML&FD_UnL_Rd[BTH13]
ML&FD_Cont_UnL_Rate[BTH14] = ML&FD_Enter_BTH[BTH14]
*ML&FD_UnL_Rd[BTH14]
ML&FD_Cont_UnL_Rate[BTH15] = ML&FD_Enter_BTH[BTH15]
*ML&FD_UnL_Rd[BTH15]
ML&FD_Enter_BTH[BTH1] = ML_Enter_BTH[BTH1]
+FD_Enter_BTH[BTH1]
ML&FD_Enter_BTH[BTH2] = ML_Enter_BTH[BTH2]
+FD_Enter_BTH[BTH2]
ML&FD_Enter_BTH[BTH3] = ML_Enter_BTH[BTH3]
+FD_Enter_BTH[BTH3]
ML&FD_Enter_BTH[BTH4] = ML_Enter_BTH[BTH4]
+FD_Enter_BTH[BTH4]
ML&FD_Enter_BTH[BTH5] = ML_Enter_BTH[BTH5]
+FD_Enter_BTH[BTH5]
ML&FD_Enter_BTH[BTH6] = ML_Enter_BTH[BTH6]
+FD_Enter_BTH[BTH6]
ML&FD_Enter_BTH[BTH7] = ML_Enter_BTH[BTH7]
+FD_Enter_BTH[BTH7]
ML&FD_Enter_BTH[BTH8] = ML_Enter_BTH[BTH8]
+FD_Enter_BTH[BTH8]
ML&FD_Enter_BTH[BTH9] = ML_Enter_BTH[BTH9]
+FD_Enter_BTH[BTH9]
ML&FD_Enter_BTH[BTH10] = ML_Enter_BTH[BTH10]
+FD_Enter_BTH[BTH10]
ML&FD_Enter_BTH[BTH11] = ML_Enter_BTH[BTH11]
+FD_Enter_BTH[BTH11]
ML&FD_Enter_BTH[BTH12] = ML_Enter_BTH[BTH12]
+FD_Enter_BTH[BTH12]
ML&FD_Enter_BTH[BTH13] = ML_Enter_BTH[BTH13]
+FD_Enter_BTH[BTH13]
ML&FD_Enter_BTH[BTH14] = ML_Enter_BTH[BTH14]
+FD_Enter_BTH[BTH14]
ML&FD_Enter_BTH[BTH15] = ML_Enter_BTH[BTH15]
+FD_Enter_BTH[BTH15]
ML&FD_Load[BTH1] = ML&FD_Load_Rate[BTH1]
ML&FD_Load[BTH2] = ML&FD_Load_Rate[BTH2]
ML&FD_Load[BTH3] = ML&FD_Load_Rate[BTH3]
```

ML&FD_Load[BTH4] = ML&FD_Load_Rate[BTH4]
ML&FD_Load[BTH5] = ML&FD_Load_Rate[BTH5]
ML&FD_Load[BTH6] = ML&FD_Load_Rate[BTH6]
ML&FD_Load[BTH7] = ML&FD_Load_Rate[BTH7]
ML&FD_Load[BTH8] = ML&FD_Load_Rate[BTH8]
ML&FD_Load[BTH9] = ML&FD_Load_Rate[BTH9]
ML&FD_Load[BTH10] = ML&FD_Load_Rate[BTH10]
ML&FD_Load[BTH11] = ML&FD_Load_Rate[BTH11]
ML&FD_Load[BTH12] = ML&FD_Load_Rate[BTH12]
ML&FD_Load[BTH13] = ML&FD_Load_Rate[BTH13]
ML&FD_Load[BTH14] = ML&FD_Load_Rate[BTH14]
ML&FD_Load[BTH15] = ML&FD_Load_Rate[BTH15]
ML&FD_Load_Rate[BTH1] = ML&FD_Enter_BTH[BTH1]
*ML&FD_Load_Rd[BTH1]
ML&FD_Load_Rate[BTH2] = ML&FD_Enter_BTH[BTH2]
*ML&FD_Load_Rd[BTH2]
ML&FD_Load_Rate[BTH3] = ML&FD_Enter_BTH[BTH3]
*ML&FD_Load_Rd[BTH3]
ML&FD_Load_Rate[BTH4] = ML&FD_Enter_BTH[BTH4]
*ML&FD_Load_Rd[BTH4]
ML&FD_Load_Rate[BTH5] = ML&FD_Enter_BTH[BTH5]
*ML&FD_Load_Rd[BTH5]
ML&FD_Load_Rate[BTH6] = ML&FD_Enter_BTH[BTH6]
*ML&FD_Load_Rd[BTH6]
ML&FD_Load_Rate[BTH7] = ML&FD_Enter_BTH[BTH7]
*ML&FD_Load_Rd[BTH7]
ML&FD_Load_Rate[BTH8] = ML&FD_Enter_BTH[BTH8]
*ML&FD_Load_Rd[BTH8]
ML&FD_Load_Rate[BTH9] = ML&FD_Enter_BTH[BTH9]
*ML&FD_Load_Rd[BTH9]
ML&FD_Load_Rate[BTH10] = ML&FD_Enter_BTH[BTH10]
*ML&FD_Load_Rd[BTH10]
ML&FD_Load_Rate[BTH11] = ML&FD_Enter_BTH[BTH11]
*ML&FD_Load_Rd[BTH11]
ML&FD_Load_Rate[BTH12] = ML&FD_Enter_BTH[BTH12]
*ML&FD_Load_Rd[BTH12]
ML&FD_Load_Rate[BTH13] = ML&FD_Enter_BTH[BTH13]
*ML&FD_Load_Rd[BTH13]
ML&FD_Load_Rate[BTH14] = ML&FD_Enter_BTH[BTH14]
*ML&FD_Load_Rd[BTH14]
ML&FD_Load_Rate[BTH15] = ML&FD_Enter_BTH[BTH15]
*ML&FD_Load_Rd[BTH15]
ML&FD_Load_Rd[BTH1] = RANDOM (80 , 1303 , 20)
ML&FD_Load_Rd[BTH2] = RANDOM (55 , 1998 , 20)
ML&FD_Load_Rd[BTH3] = RANDOM (85 , 1764 , 20)
ML&FD_Load_Rd[BTH4] = RANDOM (10 , 1572 , 20)
ML&FD_Load_Rd[BTH5] = RANDOM (35 , 2327 , 20)
ML&FD_Load_Rd[BTH6] = RANDOM (58 , 1267 , 20)
ML&FD_Load_Rd[BTH7] = RANDOM (55 , 2288 , 20)
ML&FD_Load_Rd[BTH8] = RANDOM (280 , 1172, 20)

ML&FD_Load_Rd[BTH9] = 1 ML&FD_Load_Rd[BTH10] = 1
ML&FD_Load_Rd[BTH11] = 1ML&FD_Load_Rd[BTH12] = 1
ML&FD_Load_Rd[BTH13] = 1ML&FD_Load_Rd[BTH14] = 1
ML&FD_Load_Rd[BTH15] = 1
ML&FD_UnL_Rd[BTH1] = RANDOM (156 , 1974 , 20)
ML&FD_UnL_Rd[BTH2] = RANDOM (10 , 1262 , 20)
ML&FD_UnL_Rd[BTH3] = RANDOM (25 , 1882 , 20)
ML&FD_UnL_Rd[BTH4] = RANDOM (24 , 1700 , 20)
ML&FD_UnL_Rd[BTH5] = RANDOM (43 , 1553 , 20)
ML&FD_UnL_Rd[BTH6] = RANDOM (50 , 1292 , 20)
ML&FD_UnL_Rd[BTH7] = RANDOM (54 , 1016 , 20)
ML&FD_UnL_Rd[BTH8] = RANDOM (4 , 2173 , 20)
ML&FD_UnL_Rd[BTH9] = 1ML&FD_UnL_Rd[BTH10] = 1
ML&FD_UnL_Rd[BTH11] = 1ML&FD_UnL_Rd[BTH12] = 1
ML&FD_UnL_Rd[BTH13] = 1ML&FD_UnL_Rd[BTH14] = 1
ML&FD_UnL_Rd[BTH15] = 1
ML_&_FD_Cont_UnL[BTH1] = ML&FD_Cont_UnL_Rate[BTH1]
ML_&_FD_Cont_UnL[BTH2] = ML&FD_Cont_UnL_Rate[BTH2]
ML_&_FD_Cont_UnL[BTH3] = ML&FD_Cont_UnL_Rate[BTH3]
ML_&_FD_Cont_UnL[BTH4] = ML&FD_Cont_UnL_Rate[BTH4]
ML_&_FD_Cont_UnL[BTH5] = ML&FD_Cont_UnL_Rate[BTH5]
ML_&_FD_Cont_UnL[BTH6] = ML&FD_Cont_UnL_Rate[BTH6]
ML_&_FD_Cont_UnL[BTH7] = ML&FD_Cont_UnL_Rate[BTH7]
ML_&_FD_Cont_UnL[BTH8] = ML&FD_Cont_UnL_Rate[BTH8]
ML_&_FD_Cont_UnL[BTH9] = ML&FD_Cont_UnL_Rate[BTH9]
ML_&_FD_Cont_UnL[BTH10] = ML&FD_Cont_UnL_Rate[BTH10]
ML_&_FD_Cont_UnL[BTH11] = ML&FD_Cont_UnL_Rate[BTH11]
ML_&_FD_Cont_UnL[BTH12] = ML&FD_Cont_UnL_Rate[BTH12]
ML_&_FD_Cont_UnL[BTH13] = ML&FD_Cont_UnL_Rate[BTH13]
ML_&_FD_Cont_UnL[BTH14] = ML&FD_Cont_UnL_Rate[BTH14]
ML_&_FD_Cont_UnL[BTH15] = ML&FD_Cont_UnL_Rate[BTH15]
ML_Cont_UnL_Rate[BTH1] = ML_Enter_BTH[BTH1]*ML_UnL_Rd[BTH1]
ML_Cont_UnL_Rate[BTH2] = ML_Enter_BTH[BTH2]*ML_UnL_Rd[BTH2]
ML_Cont_UnL_Rate[BTH3] = ML_Enter_BTH[BTH3]*ML_UnL_Rd[BTH3]
ML_Cont_UnL_Rate[BTH4] = ML_Enter_BTH[BTH4]*ML_UnL_Rd[BTH4]
ML_Cont_UnL_Rate[BTH5] = ML_Enter_BTH[BTH5]*ML_UnL_Rd[BTH5]
ML_Cont_UnL_Rate[BTH6] = ML_Enter_BTH[BTH6]*ML_UnL_Rd[BTH6]
ML_Cont_UnL_Rate[BTH7] = ML_Enter_BTH[BTH7]*ML_UnL_Rd[BTH7]
ML_Cont_UnL_Rate[BTH8] = ML_Enter_BTH[BTH8]*ML_UnL_Rd[BTH8]
ML_Cont_UnL_Rate[BTH9] = ML_Enter_BTH[BTH9]*ML_UnL_Rd[BTH9]
ML_Cont_UnL_Rate[BTH10] = ML_Enter_BTH[BTH10]*ML_UnL_Rd[BTH10]
ML_Cont_UnL_Rate[BTH11] = ML_Enter_BTH[BTH11]*ML_UnL_Rd[BTH11]
ML_Cont_UnL_Rate[BTH12] = ML_Enter_BTH[BTH12]*ML_UnL_Rd[BTH12]
ML_Cont_UnL_Rate[BTH13] = ML_Enter_BTH[BTH13]*ML_UnL_Rd[BTH13]
ML_Cont_UnL_Rate[BTH14] = ML_Enter_BTH[BTH14]*ML_UnL_Rd[BTH14]
ML_Cont_UnL_Rate[BTH15] = ML_Enter_BTH[BTH15]*ML_UnL_Rd[BTH15]
ML_Load_Rate[BTH1] = ML_Enter_BTH[BTH1]*ML_Load_Rd[BTH1]
ML_Load_Rate[BTH2] = ML_Enter_BTH[BTH2]*ML_Load_Rd[BTH2]
ML_Load_Rate[BTH3] = ML_Enter_BTH[BTH3]*ML_Load_Rd[BTH3]
ML_Load_Rate[BTH4] = ML_Enter_BTH[BTH4]*ML_Load_Rd[BTH4]

```
ML_Load_Rate[BTH5] = ML_Enter_BTH[BTH5]
*ML_Load_Rd[BTH5]
ML_Load_Rate[BTH6] = ML_Enter_BTH[BTH6]*ML_Load_Rd[BTH6]
ML_Load_Rate[BTH7] = ML_Enter_BTH[BTH7]*ML_Load_Rd[BTH7]
ML_Load_Rate[BTH8] = ML_Enter_BTH[BTH8]*ML_Load_Rd[BTH8]
ML_Load_Rate[BTH9] = ML_Enter_BTH[BTH9]*ML_Load_Rd[BTH9]
ML_Load_Rate[BTH10] = ML_Enter_BTH[BTH10]*ML_Load_Rd[BTH10]
ML_Load_Rate[BTH11] = ML_Enter_BTH[BTH11]*ML_Load_Rd[BTH11]
ML_Load_Rate[BTH12] = ML_Enter_BTH[BTH12]*ML_Load_Rd[BTH12]
ML_Load_Rate[BTH13] = ML_Enter_BTH[BTH13]*ML_Load_Rd[BTH13]
ML_Load_Rate[BTH14] = ML_Enter_BTH[BTH14]*ML_Load_Rd[BTH14]
ML_Load_Rate[BTH15] = ML_Enter_BTH[BTH15]*ML_Load_Rd[BTH15]
ML_Load_Rd[BTH1] = RANDOM ( 80 , 2700 , 28 )
ML_Load_Rd[BTH2] = RANDOM ( 119 , 2864 , 28 )
ML_Load_Rd[BTH3] = RANDOM ( 262 , 1764 , 28 )
ML_Load_Rd[BTH4] = RANDOM ( 120 , 1955 , 28 )
ML_Load_Rd[BTH5] = RANDOM ( 348 , 3032 , 28 )
ML_Load_Rd[BTH6] = RANDOM ( 270 , 1510 , 28 )
ML_Load_Rd[BTH7] = RANDOM ( 55 , 3110 , 28 )
ML_Load_Rd[BTH8] = RANDOM ( 103 , 1070 , 28 )
ML_Load_Rd[BTH9] = RANDOM ( 10 , 2125, 28 )
ML_Load_Rd[BTH10] = RANDOM ( 10 , 2125, 28 )
ML_Load_Rd[BTH11] = RANDOM ( 10 , 2125, 28 )
ML_Load_Rd[BTH12] = RANDOM ( 10 , 2125, 28 )
ML_Load_Rd[BTH13] = RANDOM ( 10 , 2125, 28 )
ML_Load_Rd[BTH14] = RANDOM ( 10 , 2125, 28 )
ML_Load_Rd[BTH15] = RANDOM ( 10 , 2125, 28 )
ML_UnL_Rd[BTH1] = RANDOM ( 251 , 1516, 28 )
ML_UnL_Rd[BTH2] = RANDOM ( 57 , 1448 , 28 )
ML_UnL_Rd[BTH3] = RANDOM ( 122 , 2423 , 28 )
ML_UnL_Rd[BTH4] = RANDOM ( 107 , 2084 , 28 )
ML_UnL_Rd[BTH5] = RANDOM ( 464 , 2009 , 28 )
ML_UnL_Rd[BTH6] = RANDOM ( 151 , 1705 , 28 )
ML_UnL_Rd[BTH7] = RANDOM ( 54 , 1387, 28 )
ML_UnL_Rd[BTH8] = RANDOM ( 72 , 665 , 28 )
ML_UnL_Rd[BTH9] = RANDOM ( 8 , 848 , 28 )
ML_UnL_Rd[BTH10] = RANDOM ( 8 , 848 , 28 )
ML_UnL_Rd[BTH11] = RANDOM ( 8 , 848 , 28 )
ML_UnL_Rd[BTH12] = RANDOM ( 8 , 848 , 28 )
ML_UnL_Rd[BTH13] = RANDOM ( 8 , 848 , 28 )
ML_UnL_Rd[BTH14] = RANDOM ( 8 , 848 , 28 )
ML_UnL_Rd[BTH15] = RANDOM ( 8 , 848 , 28 )
Nisbah_QC_to_PM = Quay_Crane_Assignment*7
Noname_166 = ARRAYSUM(ML_UnL[*])
Noname_172 = ARRAYSUM(FD_UnL[*])
Noname_304[BTH1] = FD_UnL[BTH1]+ML_UnL[BTH1]
Noname_304[BTH2] = FD_UnL[BTH2]+ML_UnL[BTH2]
Noname_304[BTH3] = FD_UnL[BTH3]+ML_UnL[BTH3]
Noname_304[BTH4] = FD_UnL[BTH4]+ML_UnL[BTH4]
Noname_304[BTH5] = FD_UnL[BTH5]+ML_UnL[BTH5]
```

Noname_304[BTH6] = FD_UnL[BTH6]+ML_UnL[BTH6]
Noname_304[BTH7] = FD_UnL[BTH7]+ML_UnL[BTH7]
Noname_304[BTH8] = FD_UnL[BTH8]+ML_UnL[BTH8]
Noname_304[BTH9] = FD_UnL[BTH9]+ML_UnL[BTH9]
Noname_304[BTH10] = FD_UnL[BTH10]+ML_UnL[BTH10]
Noname_304[BTH11] = FD_UnL[BTH11]+ML_UnL[BTH11]
Noname_304[BTH12] = FD_UnL[BTH12]+ML_UnL[BTH12]
Noname_304[BTH13] = FD_UnL[BTH13]+ML_UnL[BTH13]
Noname_304[BTH14] = FD_UnL[BTH14]+ML_UnL[BTH14]
Noname_304[BTH15] = FD_UnL[BTH15]+ML_UnL[BTH15]
Noname_305 = ARRAYSUM(Noname_304[*])
Noname_381 = ARRAYSUM(ML_&_FD_Cont_UnL[*])
Noname_382 = ARRAYSUM(Cont_Leaving_BLK_Rate[*])
Noname_405 = Quay_Crane_Moves_perhour*2
PM_Assignment = Nisbah_QC_to_PM
PM_Fraction = 5544
PM_Transfer_Rate = PM_Assignment*PM_Fraction
QC_Cal[BTH1] = Dec_on_Numb_of_BTH[BTH1]*Quay_Crane_Assignment
QC_Cal[BTH2] = Dec_on_Numb_of_BTH[BTH2]*Quay_Crane_Assignment
QC_Cal[BTH3] = Dec_on_Numb_of_BTH[BTH3]*Quay_Crane_Assignment
QC_Cal[BTH4] = Dec_on_Numb_of_BTH[BTH4]*Quay_Crane_Assignment
QC_Cal[BTH5] = Dec_on_Numb_of_BTH[BTH5]*Quay_Crane_Assignment
QC_Cal[BTH6] = Dec_on_Numb_of_BTH[BTH6]*Quay_Crane_Assignment
QC_Cal[BTH7] = Dec_on_Numb_of_BTH[BTH7]*Quay_Crane_Assignment
QC_Cal[BTH8] = Dec_on_Numb_of_BTH[BTH8]*Quay_Crane_Assignment
QC_Cal[BTH9] = Dec_on_Numb_of_BTH[BTH9]*Quay_Crane_Assignment
QC_Cal[BTH10] = Dec_on_Numb_of_BTH[BTH10]*Quay_Crane_Assignment
QC_Cal[BTH11] = Dec_on_Numb_of_BTH[BTH11]*Quay_Crane_Assignment
QC_Cal[BTH12] = Dec_on_Numb_of_BTH[BTH12]*Quay_Crane_Assignment
QC_Cal[BTH13] = Dec_on_Numb_of_BTH[BTH13]*Quay_Crane_Assignment
QC_Cal[BTH14] = Dec_on_Numb_of_BTH[BTH14]*Quay_Crane_Assignment
QC_Cal[BTH15] = Dec_on_Numb_of_BTH[BTH15]*Quay_Crane_Assignment
QC_Moves_perDay = Noname_405*24
QC_Used = ARRAYSUM(QC_Cal[*])
Quay_Crane_Assignment = 2
Quay_Crane_Moves_perhour = 33
to_BLK_1[BTH1] = 8.220390928/100 to_BLK_1[BTH2] = 6.212221954/100
to_BLK_1[BTH3] = 10.74345106/100 to_BLK_1[BTH4] = 9.805298436/100
to_BLK_1[BTH5] = 6.293462773/100 to_BLK_1[BTH6] = 6.242540582/100
to_BLK_1[BTH7] = 3.883931741/100 to_BLK_1[BTH8] = 2.384649308/100
to_BLK_1T[BTH9] = 2.384649308/100 to_BLK_1[BTH10] = 2.384649308/100
to_BLK_1[BTH11] = 2.384649308/100 to_BLK_1[BTH12] = 2.384649308/100
to_BLK_1[BTH13] = 2.384649308/100 to_BLK_1[BTH14] = 2.384649308/100
to_BLK_1[BTH15] = 2.384649308/100 to_BLK_10[BTH1] = 4.816984116/100
to_BLK_10[BTH2] = 4.186311251/100 to_BLK_10[BTH3] = 2.753031428/100
to_BLK_10[BTH4] = 2.94552612/100 to_BLK_10[BTH5] = 4.621223158/100
to_BLK_10[BTH6] = 7.696985374/100 to_BLK_10[BTH7] = 12.18314359/100
to_BLK_10[BTH8] = 16.3962441/100 to_BLK_10[BTH9] = 4.816984116/100
to_BLK_10[BTH10] = 4.816984116/100 to_BLK_10[BTH11] = 4.816984116/100
to_BLK_10[BTH12] = 4.816984116/100 to_BLK_10[BTH13] = 4.816984116/100

to_BLK_10[BTH14] = 4.816984116/100 to_BLK_10[BTH15] = 4.816984116/100
to_BLK_11[BTH1] = 2.076853807/100 to_BLK_11[BTH2] = 1.553936976/100
to_BLK_11[BTH3] = 1.567858027/100to_BLK_11[BTH4] = 0.86604958/100
to_BLK_11[BTH5] = 1.676503719/100to_BLK_11[BTH6] = 7.804083337/100
to_BLK_11[BTH7] = 11.74883435/100to_BLK_11[BTH8] = 16.99987137/100
to_BLK_11[BTH9] = 16.99987137/100to_BLK_11[BTH10] = 16.99987137/100
to_BLK_11[BTH11] = 16.99987137/100to_BLK_11[BTH12] = 16.99987137/100
to_BLK_11[BTH13] = 16.99987137/100to_BLK_11[BTH14] = 16.99987137/100
to_BLK_11[BTH15] = 16.99987137/100to_BLK_12[BTH1] = 1.417360969/100
to_BLK_12[BTH2] = 1.857168762/100to_BLK_12[BTH3] = 5.207780959/100
to_BLK_12[BTH4] = 0.564953718/100to_BLK_12[BTH5] = 1.063538709/100
to_BLK_12[BTH6] = 5.49920811/100to_BLK_12[BTH7] = 7.982059596/100
to_BLK_12[BTH8] = 12.65458187/100to_BLK_12[BTH9] = 1.417360969/100
to_BLK_12[BTH10] = 1.417360969/100to_BLK_12[BTH11] = 1.417360969/100
to_BLK_12[BTH12] = 1.417360969/100to_BLK_12[BTH13] = 1.417360969/100
to_BLK_12[BTH14] = 1.417360969/100to_BLK_12[BTH15] = 1.417360969/100
to_BLK_2[BTH1] = 10.58858695/100to_BLK_2[BTH2] = 9.807180851/100
to_BLK_2[BTH3] = 15.25612472/100to_BLK_2[BTH4] = 12.82476859/100
to_BLK_2[BTH5] = 9.899118396/100to_BLK_2[BTH6] = 8.632277329/100
to_BLK_2[BTH7] = 5.112305034/100to_BLK_2[BTH8] = 3.575825509/100
to_BLK_2[BTH9] = 10.58858695/100to_BLK_2[BTH10] = 10.58858695/100
to_BLK_2[BTH11] = 10.58858695/100to_BLK_2[BTH12] = 10.58858695/100
to_BLK_2[BTH13] = 10.58858695/100to_BLK_2[BTH14] = 10.58858695/100
to_BLK_2[BTH15] = 10.58858695/100to_BLK_3[BTH1] = 12.67489992/100
to_BLK_3[BTH2] = 13.47165135/100to_BLK_3[BTH3] = 10.48052109/100
to_BLK_3[BTH4] = 13.54984573/100to_BLK_3[BTH5] = 10.45380731/100
to_BLK_3[BTH6] = 10.30318707/100to_BLK_3[BTH7] = 8.531750728/100
to_BLK_3[BTH8] = 3.732015215/100to_BLK_3[BTH9] = 12.67489992/100
to_BLK_3[BTH10] = 12.67489992/100to_BLK_3[BTH11] = 12.67489992/100
to_BLK_3[BTH12] = 12.67489992/100to_BLK_3[BTH13] = 12.67489992/100
to_BLK_3[BTH14] = 12.67489992/100to_BLK_3[BTH15] = 12.67489992/100
to_BLK_4[BTH1] = 12.15351086/100to_BLK_4[BTH2] = 12.13934558/100
to_BLK_4[BTH3] = 9.437639198/100to_BLK_4[BTH4] = 12.25981487/100
to_BLK_4[BTH5] = 12.22092325/100to_BLK_4[BTH6] = 9.540340989/100
to_BLK_4[BTH7] = 8.776123642/100to_BLK_4[BTH8] = 4.388930744/100
to_BLK_4[BTH9] = 12.15351086/100to_BLK_4[BTH10] = 12.15351086/100
to_BLK_4[BTH11] = 12.15351086/100
to_BLK_4[BTH12] = 12.15351086/100to_BLK_4[BTH13] = 12.15351086/100
to_BLK_4[BTH14] = 12.15351086/100to_BLK_4[BTH15] = 12.15351086/100
to_BLK_5[BTH1] = 10.75696005/100to_BLK_5[BTH2] = 10.41364442/100
to_BLK_5[BTH3] = 7.708487998/100to_BLK_5[BTH4] = 10.21119268/100
to_BLK_5[BTH5] = 10.16171616/100to_BLK_5[BTH6] = 7.424702417/100
to_BLK_5[BTH7] = 9.679652553/100to_BLK_5[BTH8] = 7.489296411/100
to_BLK_5[BTH9] = 7.489296411/100to_BLK_5[BTH10] = 7.489296411/100
to_BLK_5[BTH11] = 7.489296411/100to_BLK_5[BTH12] = 7.489296411/100
to_BLK_5[BTH13] = 7.489296411/100to_BLK_5[BTH14] = 7.489296411/100
to_BLK_5[BTH15] = 7.489296411/100to_BLK_6[BTH1] = 13.06083377/100
to_BLK_6[BTH2] = 14.54757012/100to_BLK_6[BTH3] = 13.89154028/100
to_BLK_6[BTH4] = 10.84370678/100to_BLK_6[BTH5] = 14.62196938/100
to_BLK_6[BTH6] = 11.32197913/100to_BLK_6[BTH7] = 7.318169984/100

to_BLK_6[BTH8] = 5.292534132/100to_BLK_6[BTH9] = 5.292534132/100
to_BLK_6[BTH10] = 5.292534132/100to_BLK_6[BTH11] = 5.292534132/100
to_BLK_6[BTH12] = 5.292534132/100to_BLK_6[BTH13] = 5.292534132/100
to_BLK_6[BTH14] = 5.292534132/100to_BLK_6[BTH15] = 5.292534132/100
to_BLK_7[BTH1] = 10.46221254/100to_BLK_7[BTH2] = 11.40494036/100
to_BLK_7[BTH3] = 11.53356666/100to_BLK_7[BTH4] = 10.92935419/100
to_BLK_7[BTH5] = 11.97324985/100to_BLK_7[BTH6] = 9.461832737/100
to_BLK_7[BTH7] = 8.543584767/100to_BLK_7[BTH8] = 7.44887084/100
to_BLK_7[BTH9] = 7.44887084/100to_BLK_7[BTH10] = 7.44887084/100
to_BLK_7[BTH11] = 7.44887084/100to_BLK_7[BTH12] = 7.44887084/100
to_BLK_7[BTH13] = 7.44887084/100to_BLK_7[BTH14] = 7.44887084/100
to_BLK_7[BTH15] = 7.44887084/100to_BLK_8[BTH1] = 7.300960294/100
to_BLK_8[BTH2] = 8.338874113/100to_BLK_8[BTH3] = 6.361579524/100
to_BLK_8[BTH4] = 8.329077561/100to_BLK_8[BTH5] = 9.308184593/100
to_BLK_8[BTH6] = 8.212053966/100to_BLK_8[BTH7] = 6.844216705/100
to_BLK_8[BTH8] = 7.984050275/100to_BLK_8[BTH9] = 7.984050275/100
to_BLK_8[BTH10] = 7.984050275/100to_BLK_8[BTH11] = 7.984050275/100
to_BLK_8[BTH12] = 7.984050275/100to_BLK_8[BTH13] = 7.984050275/100
to_BLK_8[BTH14] = 7.984050275/100to_BLK_8[BTH15] = 7.984050275/100
to_BLK_9[BTH1] = 6.470445791/100to_BLK_9[BTH2] = 6.067154255/100
to_BLK_9[BTH3] = 5.058419062/100to_BLK_9[BTH4] = 6.870411746/100
to_BLK_9[BTH5] = 7.706302702/100to_BLK_9[BTH6] = 7.860808953/100
to_BLK_9[BTH7] = 9.396227308/100to_BLK_9[BTH8] = 11.65313023/100
to_BLK_9[BTH9] = 11.65313023/100to_BLK_9[BTH10] = 11.65313023/100
to_BLK_9[BTH11] = 11.65313023/100to_BLK_9[BTH12] = 11.65313023/100
to_BLK_9[BTH13] = 11.65313023/100to_BLK_9[BTH14] =
11.65313023/100to_BLK_9[BTH15] = 11.65313023/100
Total_Cont_Loaded = ARRAYSUM(Accum_Cont_Loaded[*])
Total_Cont_UnLoaded = ARRAYSUM(Accum_Cont_UnL[*])
Ttl_UnL_time_@hours = IF (Accum_UnL_time = 0 OR (Total_Vessel_Served = 0
))THEN (Accum_UnL_time / 1)ELSE ((Accum_UnL_time / Total_Vessel_Served
) * 24)
UnL_&_Load_Rate[BTH1] = Quay_Crane_Assignment*QC_Moves_perDay
UnL_&_Load_Rate[BTH2] = Quay_Crane_Assignment*QC_Moves_perDay
UnL_&_Load_Rate[BTH3] = Quay_Crane_Assignment*QC_Moves_perDay
UnL_&_Load_Rate[BTH4] = Quay_Crane_Assignment*QC_Moves_perDay
UnL_&_Load_Rate[BTH5] = Quay_Crane_Assignment*QC_Moves_perDay
UnL_&_Load_Rate[BTH6] = Quay_Crane_Assignment*QC_Moves_perDay
UnL_&_Load_Rate[BTH7] = Quay_Crane_Assignment*QC_Moves_perDay
UnL_&_Load_Rate[BTH8] = Quay_Crane_Assignment*QC_Moves_perDay
UnL_&_Load_Rate[BTH9] = Quay_Crane_Assignment*QC_Moves_perDay
UnL_&_Load_Rate[BTH10] = Quay_Crane_Assignment*QC_Moves_perDay
UnL_&_Load_Rate[BTH11] = Quay_Crane_Assignment*QC_Moves_perDay
UnL_&_Load_Rate[BTH12] = Quay_Crane_Assignment*QC_Moves_perDay
UnL_&_Load_Rate[BTH13] = Quay_Crane_Assignment*QC_Moves_perDay
UnL_&_Load_Rate[BTH14] = Quay_Crane_Assignment*QC_Moves_perDay
UnL_&_Load_Rate[BTH15] = Quay_Crane_Assignment*QC_Moves_perDay
UnL_Cont_%_of_Change = 0