

Ryan Alvita
4865170

MSc Civil Engineering
Hydraulic Engineering Track
Ports and Waterways Specialization

A Tool for Container Terminal Design

Developing a tool for container terminal design on a concept design phase while taking into account area limitation

Picture: Container Terminal at Port of Singapore by @chuttersnap

A Tool for Container Terminal Design

Developing a tool for container terminal design while taking into
account area limitations

By

Ryan Alvita

in partial fulfilment of the requirements for the degree of

Master of Science

in Hydraulic Engineering

Faculty of Civil Engineering and Geosciences

at the Delft University of Technology,

to be defended publicly on Wednesday, October 28th 2020 at 3:00 PM.

Thesis Committee:

Chairman:	Prof. dr. ir. Mark van Koningsveld	TU Delft
Supervisors:	Dr. Poonam Taneja,	TU Delft
	Dr. Bart Wiegmans,	TU Delft

An electronic version of this thesis is available at <http://repository.tudelft.nl/>.

This page is left intentionally blank.

Acknowledgements

This thesis is part of the Master Programme in Hydraulic Engineering from the Faculty of Civil Engineering and Geosciences at the Delft University of Technology. It marks the end of a two years phase of achieving knowledge, experiences, challenges, and gratitude. I would like to use this opportunity to sincerely express my thanks to all that have the support and guide me throughout the process.

I would like to say thank you to Prof. Mark van Koningsveld for his guidance, passion, and expertise in helping me throughout the process. Thank you for your trust and opportunity to let me be your graduate student, giving your time to discuss which always provide me with advice and insights in doing the Thesis. Further, I would like to say thank you to Poonam Taneja for her constructive feedback and for being the one who was always there to help me get through the process. Without a caring and supportive supervisor like her, it would not be easy for me to work on this Thesis. Next, I would like to say thank you to Bart Wiegman for his valuable remarks in urging me to improve my Thesis to be even better. Thank you for the coaching and brainstorming sessions with you along the journey. You all have admirable qualities that I look up for my professional career life.

The opportunity of studying abroad is one of the most challenging situations and has been a period of a life-changing state for me. Therefore, I would also like to express my gratitude to my loving family in Indonesia for their endless support for in achieving my dreams. To my parents, Papa and Mama, thank you for making it possible for me to experience this wonderful dream of studying abroad in Delft, the Netherlands. Thank you so much for always believe in me and relieve my pressure at every moment of my life. To my siblings, Kak Novie (and my brother-in-law Kak Wibi), Bang Fiqhi (and my sister-in-law Kak Rina), thank you for always being supportive and understandable throughout my study; also not forgetting to my niece Abi who has been my source of happiness during my tiring time. Furthermore, I would like to express my gratitude for all my Indonesian friends here in Delft. Thank you for being my second family here in Delft with all the silly jokes and late-night gathering and conversations; I am very fortunate to share my studying abroad experience in the same space and time with all of you. Thank you also for my Indonesian friends in Indonesia, who always be there even when I was on my deepest moments, and who always ask me when I will go back to Indonesia again. Once again, thank you, my family and friends, for your love over the past years.

I hope that this thesis can be worthwhile for the world of science & research and my country, Indonesia. At last, I would like to say that I am truly grateful for this journey.

Delft, September 2020

Ryan Alvita

This page is left intentionally blank.

Abstract

Due to the increase of global containerized trade and container vessel sizes, the necessity of container terminal development is unavoidable. On the other hand, designing a container terminal is a very complex process, and on a concept design phase, a quick assessment for the design alternatives should be done in a limited time, cost, and effort. Therefore, an automated tool is developed by (Koster, 2019) to ease and accelerate the work of the engineers. However, further development is required for the automated tool, especially for the layout generation tool to also consider area limitations.

This research aims to answer the research question: *'How can a container terminal be designed in the concept design phase while taking into account area limitations?'*. The premise is answered by developing an automated tool (i.e. design tool) for container terminal design that consider not only terminal throughput and design rules for the main input but also terminal shape and dimensions. The algorithms implemented within the design tool will correspond for at least four tasks: recognizing the terminal shape and dimension as the boundary condition, modifying container block configuration (either by enlarging/reducing its size), generating the container blocks, and determining the desired output. After the design tool is developed, it will be validated using case studies from existing terminals by comparing several validation parameters: terminal layout, terminal capacity, the number of blocks, blocks configuration, stacking density and stacking orientation. Other than that, the design tool is also reviewed by using three types of applicability: design alternatives for all types of equipment, element calculation & cost estimation from all design alternatives, and changing throughput demand.

In this thesis, we can find the development and validation of the design tool that will give us the answer to the main deliverables of a concept design phase. The design tool is capable of presenting container terminal design alternatives and comparing the layout, elements, land use, cost, and capacity for 4 main types of stacking equipment: RTG, RMG, SC, and RS. This would be very beneficial for container terminal stakeholders, not only for determining which one is the most effective and efficient stacking equipment (which depends on the main objective of the terminal; e.g., lower land use, higher throughput, least labour cost, more sustainable, etc.), but also for getting information of main deliverables at the concept design phase: land use and cost. Other than that, this thesis will also show the design tool's capability of generating different container terminal design for varying throughput demand and providing information on whether the container terminal has reached its area capacity or not. All of these are resulted from the process of an automated tool, making the design process faster and better, which will be very meaningful in this era of continuous uncertainty.

All in all, area limitations have a significant influence in the container terminal design process since it is closely related to container block configuration, which also related to the terminal capacity, land use, and costs. Area limitations will also make the design process even more complicated; and thus, the design tool that has been developed in this thesis for generating container terminal design alternatives is one of the answers in solving the problem.

Contents

Contents	v
1. Introduction	1
1.1. Background	1
1.1.1. The necessity of container terminal in port development	1
1.1.2. Concept design phase in the container terminal design cycle	2
1.2. Problem Statement	3
1.3. Research Question	4
1.4. Methodology	5
2. Literature Review	6
2.1. Earlier container terminal design research	6
2.2. Container Types	7
2.3. Container Terminal Layout	8
2.4. Container Terminal Equipment	10
2.4.1. Ship-to-shore equipment	10
2.4.2. Stacking equipment	11
2.4.3. Horizontal transport equipment	13
2.5. Container Terminal Capacity	14
2.6. Container terminal design guidelines	16
2.6.1. Apron	16
2.6.2. Storage yard	16
3. Design Tool Development	20
3.1. Overview of the design tool	20
3.2. Layout Generation	24
3.2.1. Overview of the layout generation process	24
3.2.2. Design rules	26
3.2.3. Layout generation design tool	28
3.2.4. Placement of reefer containers	33
3.2.5. The complementary capabilities of the design tool	34
3.2.6. Synchronization with the existing tool	34
4. Design tool validation	36
4.1. Terminal design validation	36
4.2. Applicability review of the design tool	41
4.2.1. Design alternatives using all equipment types	41
4.2.2. Changing terminal throughput demand	43
4.2.3. Terminal element calculation and cost estimation	45
5. Discussion, Conclusion and Recommendation	48

5.1. Discussion	48
5.2. Conclusion.....	49
5.3. Recommendation.....	53
Bibliography.....	54
List of Figures	59
List of Tables.....	62
Appendix	I

1. Introduction

1.1. Background

1.1.1. The necessity of container terminal in port development

Port development depends on the development of maritime transport, both in terms of volumes per commodity and concerning types and sizes of vessels (Ligteringen, 2017). Port development can refer to an entirely new port project, to port expansion, or a new terminal in an existing port (Taneja, 2013). One of the main commodities that ports handled is containers. Containerization plays a fundamental role in changing how industrial production and distribution occurs around the world, where not only did it lead to a decrease of sea transportation costs and more cost-effective transport, containers have also lead to advantages in terms of handling processes at terminals and protection of the goods inside (Mohseni, 2011). As can be shown in Figure 1, since 1966, the global containerized trade has been steadily increasing and is still expected to grow in the next years.

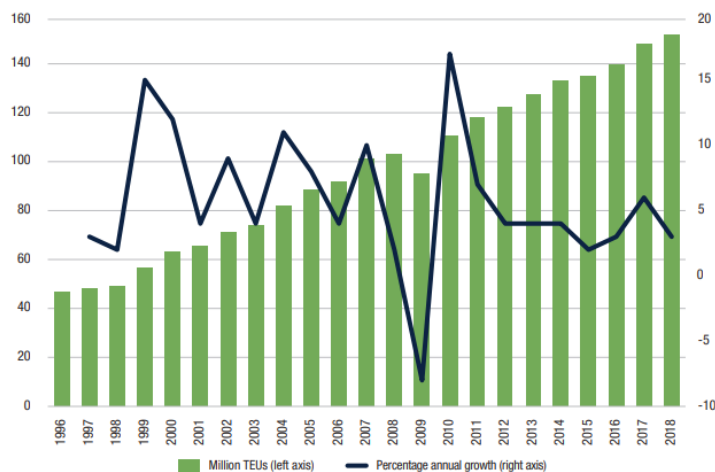


Figure 1 Global Containerized Trade 1966 – 2018 (UNCTAD, 2019)

Moreover, the size of the container vessels is constantly increasing for the last 50 years. The call for bigger and bigger container ships is increasing from many sides (Pawellek & Schonknecht, 2011). Container-carrying capacity has increased by almost 1,500% since 1968, which is ranging from ‘Encounter Bay’ vessels with a capacity of only 1.530 TEU, and the latest and largest is container vessels from OOCL Hongkong with a capacity of 21.413 TEU in 2017 (Allianz Global Corporate & Specialty, 2018). Figure 2 shows the illustration of the evolution of container ship growth.



Figure 2 50 Years of Container Ship Growth (Allianz Global Corporate & Specialty, 2018)

As a consequence, the necessity of container terminal in port developments to fulfil the demand for larger and faster container handling capacity, either by building new terminals or by increasing the capacity of existing container terminals, is unavoidable.

1.1.2. Concept design phase in the container terminal design cycle

The design cycle of a container terminal is a set of correlated practices, aiming to translate a cargo demand into a container terminal design (PIANC, 2014b). A container terminal project requires accurate planning in the early stages of its lifecycle, where at this stage, changes in the design still have considerable high influence. At the same time, the costs are still relatively low. Though this often adds additional time and cost to the early phase of a project, these costs are insignificant compared to the alternative of the costs and effort required to make modifications to the design at a later stage in the project (Yussef, Gibson Jr, Asmar, & Ramsey, 2018).

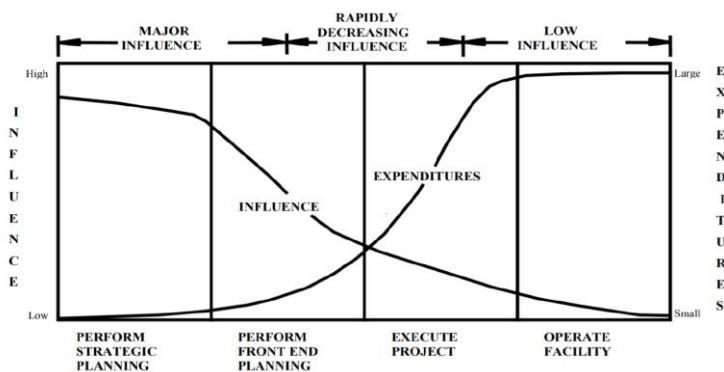


Figure 3 Influence and Expenditures Curve for Project Life Cycle (Yussef, Gibson Jr, Asmar, & Ramsey, 2018)

The container terminal design process consists of three phases: the feasibility study, the concept design, and the detailed design (Agerschou, et al., 2004). This report focuses on the concept design phase, where at this stage, there is a significant pressure to limit efforts because the project initiator does not have a proven positive business case yet and therefore minimizes costs during this phase. Furthermore, a quick assessment of the layout, element, and costs is required, as both are needed for further stakeholder processes such as financial planning, permits, and

environmental impact assessment. During the concept design phase, a first assessment of the project's technical and financial feasibility is made. The general purpose of the container terminal concept design is to (Koster, 2019) :

- (a) Investigate the total land use by generating and visualizing the terminal concept layout
- (b) make a first cost estimate (e.g., CAPEX, OPEX, NPV)
- (c) provide confidence on technical feasibility.

1.2. Problem Statement

As mentioned in the previous section, the necessity of container terminal in port developments is unavoidable. Moreover, designing a container terminal will involve various parameters, variables, and stakeholders, including the port authority (or governing bodies), the terminal operator, carriers, shippers, freight forwarders, and third parties such as logistic-, port-, and transport service providers (sometimes the roles may be combined) (Taneja, 2013). Therefore, different design alternatives should be studied quantitatively at the concept phase of a port development to give a more reliable assessment and feasibility. One of the parameters that most influence the container terminal design is the choice of the type of equipment for container handling, where it has a significant influence on the two primary deliverables of concept design: the land use and the cost estimate (Michele & Serra, 2014). A wide variety of equipment is used based on stakeholder's preferences, where it is usually a matter of economics, desired density, productivity, and scale of the projects. An illustration of different container terminal layout for a different container handling equipment can be shown in Figure 4.

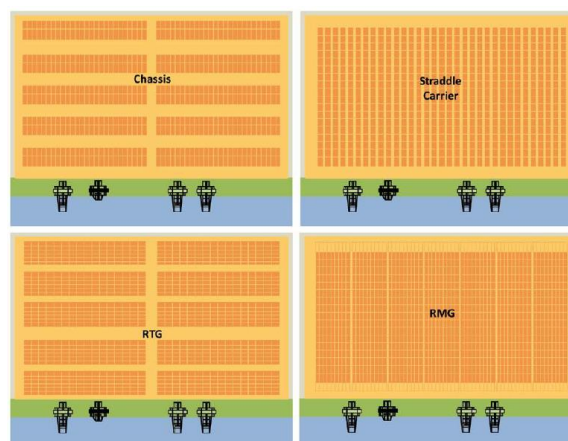


Figure 4 Container storage yard layout per equipment type illustration (Kox, 2017)

In addition, during the concept design phase, a quick assessment should be done in a limited time, cost, and effort. To increase the quality of the design during the concept design phase, automation in the design process would be preferred to ease and accelerate the work of the engineers. Automation makes it easier to generate several solutions and trying different what-if-conditions (Sandberg, Gerth, Lu, & Gustav, 2016). Several studies in accelerating the process of container terminal design have been done (either by automating the tasks or not), and the latest study has been held by (Koster, 2019) on how to accelerate the design process during the concept design phase of a container terminal by modelling the automatable tasks.

However, there are limitations to the latest study by (Koster, 2019) that needs to be considered, especially in the terminal layout generation part. In the latest study, the design tool is not capable of generating a container terminal design on a case where the container terminal layout is already defined since the design tool generates the layout starting from a 'blank canvas'. As a result, the model will always generate a rectangular-shape terminal and does not consider the limitation of the area available. While in reality, with the growing container traffic, physical expansion of a port is often constrained by the scarcity of land, especially for ports located in or near urban areas, such as in Singapore and Shanghai (Zhen, Jiang, Lee, & Chew, 2013). Moreover, in some cases, a container terminal is designed to have a non-rectangular shape. One of the reasons is that the terminal is designed to follow the existing bathymetry to produce the least expensive alternatives. Another reason might be that the desired design of the container terminal is set to be less vulnerable from wind, waves and current to the vessels. Therefore, the design tool from the latest study is not inadequate to solve this problem, and further research for the layout generation tool is necessary.

Besides, there is another reason why further development for the layout generation tool is required. The workflow of the design tool from the latest study is briefly explained as follows. First, the container terminal elements are calculated within a program called Python; then the output is imported to another program called Grasshopper to generate the container layout; finally, the output from container terminal elements and layout is imported back to Python to do the terminal cost estimation tasks. Moreover, the Grasshopper tool works more-or-less like a 'black box' type of program because the program is not a free-open-source program like Python. Thus, this research tried to have all the automated tasks in one environment, which is in Python, due to the advantages in keeping all automated tasks in one environment where it will remove barriers involved in the lifecycle of software development (Mulonda, 2019).

1.3. Research Question

Based on the problem statement, the following research question is defined:

“How can a container terminal be designed in the concept design phase while taking into account area limitations?”

To answer the research question, the following sub-questions are determined:

1. What does a typical container terminal design consist of?
2. What are the governing parameters and boundary conditions in the process of container terminal design?
3. How to develop an automated tool in generating container terminal design based on the governing parameters and boundary condition?
4. How to determine container terminal design using the automated tool?
5. How does the container terminal design generated by the automated tool compare with the actual project?

1.4. Methodology

To answer the research questions, a methodology is proposed that consists of generally three main steps: (a) Literature Review; (b) Design tool development; (c) Design tool validation.

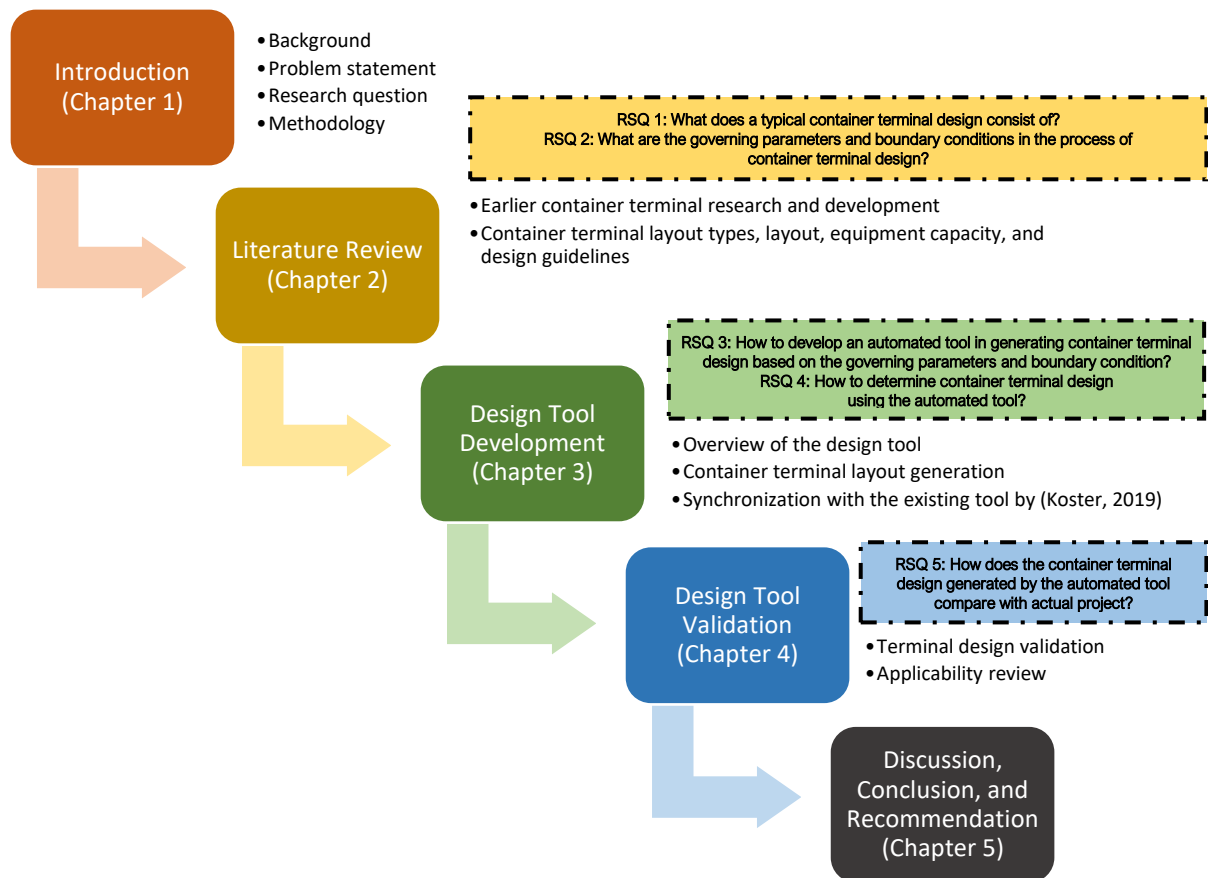


Figure 5 Research Approach

The overview of the research approach is presented in Figure 5. The introduction part will explain the background, problem statement, research question, and the methodology of the research. The literature review part will answer research sub-question 1 about the typical container terminal design process and research sub-question 2 about governing parameters and boundary conditions in the process of container terminal design. Next, the design tool development part will answer the research sub-question 3 about the development of the automated tool in generating container terminal design and research sub-question 4 about determining the container terminal design using the automated tool. The design tool development part is the core of this research, where the explanation of how the design tool will generate the container terminal layout is done. In this part also, synchronization with the previous tool by (Koster, 2019) for calculating the terminal elements and the terminal costs is explained. Furthermore, the design tool validation part will answer the research sub-question 5 about comparing the container terminal design generated by the automated tool with the actual project. Finally, this research will be enclosed by the conclusion and recommendation part.

2. Literature Review

2.1. Earlier Container Terminal Design Research

A summarizing overview of earlier research and development on container terminal design can be shown in Table 1, which is obtained from literature studies (including TU Delft repository). There, each reference is characterized with respect to its storage system, performance criteria, design problem, research approach, and how are the result is being presented. Some of the results also include simulation software related to container terminal design and operations, such as TOMAS (Duinkerken, Evers, & Ottjes, 2001), Desmo-J (Briskorn & Hartmann, 2005), Tecnomatix Plant Simulation (Kemme, 2013). There is none of the research that considers the terminal layout as a design problem and also considers area and cost as the performance indicator, thus concluding that a specific container terminal design tool must be developed.

Table 1 Overview of earlier research and development on container terminal design

Earlier container terminal design research and development	Stacking system				Performance criteria			Design problem				Approach		Result	
	RTG	RMG	SC	RS	Area	Operational	Cost	Layout	Throughput	Equipment	Block config	Analytical	Simulation	Calculation	Visualization
(Kim & Kim, 1998)	v	-	-	-	-	v	v	-	v	-	-	v	-	v	-
(Duinkerken, Evers, & Ottjes, 2001)	-	v	-	-	-	v	-	-	-	v	-	-	v	v	v
(Nam & Ha, 2001)	v	v	-	-	-	v	v	-	v	v	-	-	v	v	-
(Liu, Jula, & Iannou, 2002)	v	v	-	-	v	v	v	-	v	v	v	-	v	v	v
(van Valkengoed, 2004)	-	v	-	-	-	v	-	-	v	-	-	-	v	v	-
(Chu & Huang, 2005)	v		v	-	v	v	-	v	-	-	v	-	-	v	-
(Saanen & van Valkengoed, 2005)	-	v	-	-	-	v	v	-	-	v	-	-	v	v	-
(Zyngiridis, 2005)	-	v	-	-	-	v	-	-	-	v	v	v	-	v	-
(Briskorn & Hartmann, 2005)	-	v	-	-	-	v	-	-	-	v	-	-	v	v	-
(Vis I. F., 2006)	-	v	v	-	-	v	-	-	-	v	-	-	v	v	-
(Kim, Park, & Jin, 2008)	v	-	-	-	-	v	v	v	-	-	v	v	-	v	-
(Petering, 2009)	v	-	-	-	-	v	-	v	v	v	v	-	v	v	-
(Petering & Murty, 2009)	v	-	-	-	-	v	-	v	v	v	v	-	v	v	-

Earlier container terminal design research and development	Stacking system				Performance criteria			Design problem				Approach		Result	
	RTG	RMG	SC	RS	Area	Operational	Cost	Layout	Throughput	Equipment	Block config	Analytical	Simulation	Calculation	Visualization
(Lee & Kim, 2013)	v	v	-	-	-	v	v	v	-	-	v	v	-	v	-
(Wiese, Kliwer, & Suhl, 2010)	v	-	-	-	-	v	-	v	-	v	v	v	v	v	-
(Mohseni, 2011)	v	v	v	v	v	-	v	-	v	v	v	-	v	v	-
(Pirhoenen, 2011)	v	v	-	-	-	v	v	-	-	v	v	v	v	v	-
(Wiese, 2011)	-	-	v	-	-	v	-	v	-	v	v	v	v	v	-
(Kemme, 2013)	-	v	-	-	-	v	-	v	v	v	v	-	v	v	v
(Michele & Serra, 2014)	v	v	v	-	-	-	v	-	v	v	-	-	v	v	-
(Kox, 2017)	v	v	v	v	v	-	v	-	v	v	v	-	v	v	-
(Koster, 2019)	v	v	v	v	v	-	v	-	v	v	v	-	v	v	v
This thesis	v	v	v	v	v	-	v	v	v	v	v	-	v	v	v

2.2. Container Types

A regular container has a standard size which is often referred to as a single TEU. It is based on the volume of a 20-foot-long intermodal container, a standard-sized metal box that can be easily transferred between different modes of transportation because of its fixed dimension and secure lock system. The size of a container refers to its metrics in terms of length, width, and height, which are usually expressed in feet and inches. The length of a freight container is either 20', 40' or 45' and commonly used container heights are 8', 8'6" and 96". A standardized ISO-container is always 8' wide. A 9'6" high container is usually called high-cube. (Kemme, 2013). The double-length (40-foot) also occurs often in global trade, which is equal to two TEU. To standardize the total number of containers, the throughput in total TEU per year is corrected with a TEU factor, which is the ratio between 20-foot and 40-foot containers. This factor translates the throughput in TEU/annum, into total numbers of containers per year. The throughput can be subdivided into four categories: (a) Laden containers; (b) Reefers; (c) Empties; (d) Out-Of-Gauge (OOG) containers:

(a) Laden containers

A laden is the most common container type and thus often represents a substantial part of the total throughput. It always refers to a regular-sized and loaded container.

(b) Reefer containers

A reefer container is a dry container which is designed for carrying cargo that needs to be refrigerated (Kemme, 2013). It is equipped with a cooling unit and therefore requires external power to control the temperature inside the

container, allowing the containers to store perishable goods, see. The reefers are stored in sizeable so-called reefer racks, which are equipped with power plugs. For energy efficiency, most of the reefers used in global trade are 40-foot reefers.

(c) Empty containers

Containers with no cargo are called empties. Most of the container terminals have an empty depot where empty containers are stored according to the needs of the shipping lines (Steenken, Voß, & Stahlbock, 2004). Empty containers are stored on a container terminal for a much longer time than loaded containers. Also, less selectivity is needed when handling empties, as there is no specific cargo inside the container. The weight of an empty container is much less than that of a laden, allowing for faster and more energy-efficient handling.

(d) Out-of-gauge containers

The fourth type of container is the OOG container, which is a container with a deviating dimension. Before the time of containerized cargo, large OOG items could often relatively easily be stored on deck. Now that most of the load is containerized, these kinds of items require oversized containers, with a more flexible dimension. These container types are referred to as out of gauge containers. There are different types, such as the open-top container, the flat container with sides, or the flat without sides.

2.3. Container Terminal Layout

An idealized layout and the components of a container terminal are shown in Figure 6.

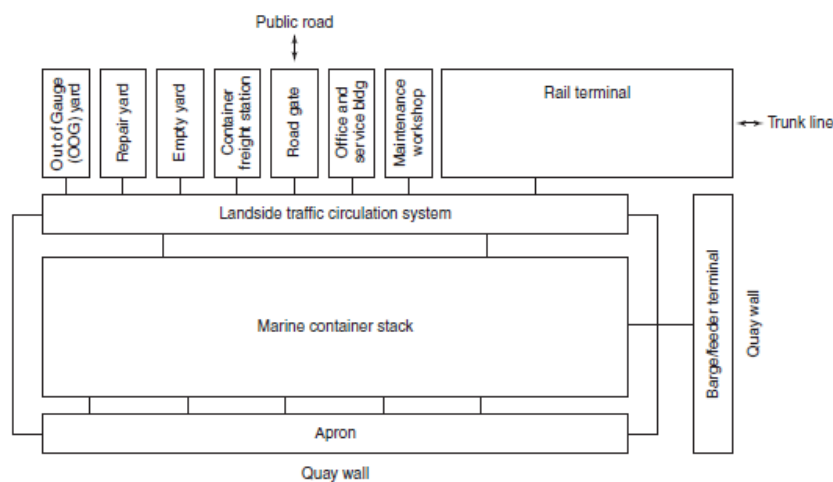


Figure 6 Idealized layout and the components of a container terminal (Thoresen, 2014)

In the idealized scheme shown in Figure 6, the quay wall is designed so that the berths can accommodate varying lengths of ships while maximizing the quay usage. The containers are stacked in a rectangular area directly behind the apron, giving the ability to optimize the stacking density and hence utilization of the stacking equipment. Other facilities are located at the back of the terminal, away from the quay and stack operations, where they do not impact on the terminal productivity. Similarly, landside traffic and the rail terminal are also separate from the core terminal operations at the quay and stack. In practice, these idealized concepts are subject to compromise as the available space for a new terminal rarely has the ideal size or shape. In general, the total area is usually divided into the following: (a) apron, (b) primary yard, (c) secondary yard.

(a) Apron

In the apron area, the ship-to-shore operation (loading and unloading of the container from/to the ship), is carried out (PIANC, 2014a). In general, the ship-to-shore operation is followed by the horizontal-transport operation, which is responsible for the transport of full and empty containers between the quay cranes and the storage / primary yard (Kempe, 2013). A typical apron arrangement consists of a quay line with room for bollards and fenders, the sea- and landside rail (or the operating space) for the equipment, and the back reach (Thoresen, 2014). The productivity at the quayside generally dictates the productivity of the container terminal, and to obtain excellent productivity, the space of the berth apron should always be kept clear (PIANC, 2014a).

(b) Primary Yard

The primary yard, or the storage area, is the area immediately adjacent to the apron, and it is used for storing inbound containers after they have been unloaded from a vessel and until they are collected for onward transport by road, rail, barge, or in the case of transshipment for loading to another vessel. It is also used for storing outbound containers to be loaded to a vessel after they have been delivered by road, rail, barge, or (in the case of transshipment) unloaded from another vessel (PIANC, 2014a). The primary yard is the place on the terminal where containers are temporarily stored (Kempe, 2013), in which the containers remain from a couple of hours to some weeks, and the time spent in the same position or area is referred to as the dwell time (PIANC, 2014a). Moreover, a container block can be defined by the number of bays, rows, and tiers, where a bay of a block is one column (one container in length) of the block, a row of a block is one row (one container in width) of the block, and a tier of a block is one high (one container in height) of the block (Wiese, 2012).

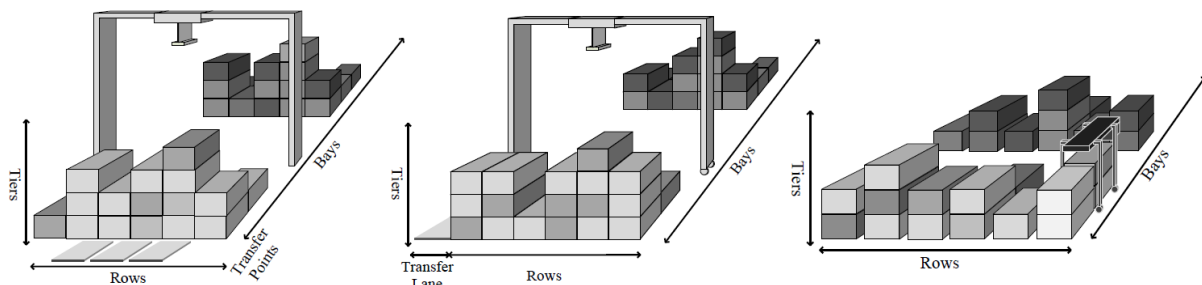


Figure 7 Structures of blocks for an RMG with transfer points, an RTG with transfer lane, and an SC (Wiese, 2012)

(c) Secondary Yard

The secondary yard is the area for storing empty containers, out of gauge containers, repair yard, container freight station, office and building, maintenance workshop, gate, and rail/barge terminal (PIANC, 2014a). One of the parameters that influence the secondary yard layout is the hinterland transport modes. The truck gate(s) is an essential feature for the secondary yard, and sufficient space must be allocated for these at the terminal boundary, which the gate should be segregated from the container yard. In the case of railway transport, the loading/unloading should take place outside the stacking area to avoid the yard equipment crossing with the railway/tracks, for reasons of efficiency and safety. In the case of inland transport, the barge terminal may be located within or outside the container terminal.

2.4. Container Terminal Equipment

Three main groups of terminal equipment can be identified: the crane in the quay for loading and unloading of the vessels (ship-to-shore), the horizontal transport equipment for moving containers around the terminal, and the stacking equipment for stacking containers in the storage area (Vis & Koster, 2003). This section will describe several types of container handling equipment, where the choice for the type of container handling equipment is one of the parameters that have a significant influence on the two primary deliverables of concept design: the land use and the cost estimate (Michele & Serra, 2014). Also, because of the rising number of container volumes that have to be stored on the storage yards and at the same time-space is an increasingly limited resource, the importance of choice for the container terminal equipment (especially stacking equipment) has continuously increased over the last few years, along with the growing traffic volume (Rijsenbrij & Wieschemann, 2011).

2.4.1. Ship-to-shore equipment

There are two main types of equipment for handling containers from (or to) ship-to-shore (STS) crane; there are ship-to-shore (STS) (or sometimes named rail-mounted gantry quay crane (RMGQC)) and mobile harbour crane (MHC). There are other types of ship-to-shore equipment such as wide span crane and onboard ship crane. However, those equipment are rarely used compared to STS crane and MHC.

(a) Ship-to-shore (STS) crane

This is the most commonly used crane for the loading and unloading of containers from or to the ships (Kempe, 2013). The crane consists of a rigid frame with a boom that extends horizontally over the ship. The crane travels parallel to the quay on fixed rails and containers removed from the ship are transferred to terminal vehicles either between the legs of the crane or in the back reach. Cranes are generally described by the spacing of the crane legs/rail span and outreach over the ship expressed in boxes (e.g. 16 wide, 30.48 m rail span crane). Today, modern QCs can technically perform around 50 loading and discharging moves per hour, while in operation usually only 22–30 moves per hour are realized ((Steenken, Voß, & Stahlbock, 2004) and (Saanen & van Valkengoed, 2005)).



Figure 8 STS Crane and Container Terminal using Straddle Carrier in Port of Teluk Lamong, Indonesia (Kone Cranes, 2020a), (<https://www.nauticexpo.com/prod/konecranes/product-30447-189113.html>)

(b) Mobile harbour crane (MHC)

This is a boom and jib type crane that can be easily moved between terminals and is usually a rubber-tired vehicle. In operating mode, it is supported on outriggers, as can be seen in Figure 9, and the crane can be used for a wide

variety of cargo types other than containers. These cranes are generally less efficient operationally than quay cranes but desirable where flexibility is more important than container throughput.



Figure 9 Mobile Harbor Crane and Container Terminal using Straddle Carrier in Port of Paramaribo, Suriname (Kone Cranes, 2020b), (<https://www.forkliftaction.com/news/newsdisplay.aspx?nwid=8272>)

2.4.2. Stacking equipment

There are four main types of stacking equipment for container handling in the container terminal yard: Rubber-Tired Gantry Crane (RTG), Rail-Mounted Gantry Crane (RMG), Straddle Carrier, and Reach Stacker (and Chassis-Based equipment) (Günther & Kim, 2006). However, within these four types, many different equipment specifications exist, based on stakeholder's preferences where it is usually a matter of economics, desired density, productivity, and scale of the projects.

(a) Rubber tired gantry (RTG) crane

This type of crane is similar to an RMG but operates directly on the yard pavement or concrete beams, as can be shown in Figure 10. They come with either eight wheels (2 per corner) or 16 wheels (4 per corner). The transfer of the containers is across the container stacks between the legs of the RTG. It is possible to move RTGs laterally between stacks with some effort, but typically it operates within one stack on a given operational shift.

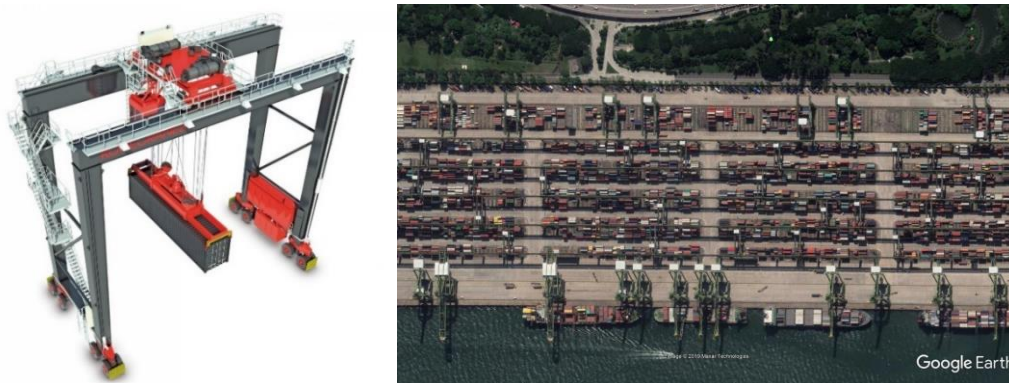


Figure 10 RTG Crane and Container Terminal using RMG in Port of Singapore, Singapore (Kone Cranes, 2020c), (Google Earth, 2017)

High equipment productivities and yard densities as well as low investment and operating costs (e.g., maintenance, energy) are offered by the RTGC system. Hence, it is most appropriate for large- to very large-sized terminals with scarce area resources. In addition, since the RTGC system performs relatively poorly in terms of labour costs and environmental friendliness, it is more appropriate for developing countries, where the labour force is not as costly

and environmental aspects are not considered to be that critical (Brinkmann, 2011). Nowadays, RTGC systems can be found at 77 container terminals of relevant size all around the world, but in particular in Asian countries, it seems to be the dominating yard system (Wiese, Kliewer, & Suhl, 2009).

(b) Rail-mounted gantry (RMG) crane

This is a large span gantry crane used in the container or intermodal yard and operates on two fixed parallel rails. Transfer of the containers is typically either at the end of the stack between the rails or parallel to the stack with cranes that cantilever outside the rails, as can be shown in Figure 11. RMG's can be equipped to perform automated stacking and sorting operations. RMGs run on fixed rails and once in place do not move to another stack.

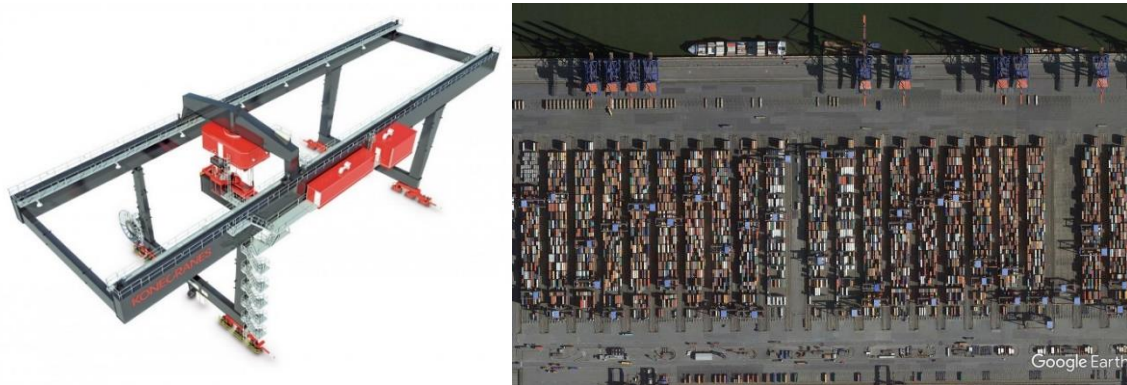


Figure 11 RMG Crane and Container Terminal using RMG Crane in Port of Hamburg, Germany (Kone Cranes, 2020d), (Google Earth, 2018a)

RMGC system is most appropriate in regions, where labour costs make up for a comparable significant fraction of the total terminal cost and where environmental aspects are rather important (Brinkmann, 2011). Nowadays, RMGC systems are mostly in operation in industrial countries in Europe, North-America, and parts of Asia (Wiese et al. 2009a).

(c) Straddle carrier

This is a versatile 4-wheeled vehicle, open in the centre which carries a container straddled between the vehicle legs. Straddle carriers can typically carry a container over either two or three stacked containers (PIANC, 2014a). They can move freely about the container yard, retrieve/deliver containers at the quay, stack them in the yard and load them on rail cars or truck chassis, as can be shown in Figure 12.

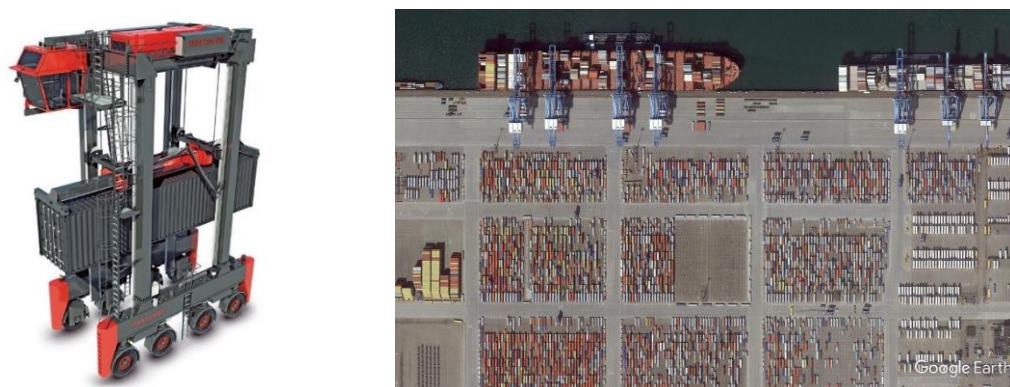


Figure 12 Straddle Carrier and Container Terminal using Straddle Carrier in Port of Rotterdam, the Netherlands (Kone Cranes, 2020e), (Google Earth, 2018b)

SC system may be most appropriate for medium-to-large-sized terminals that have sufficient storage area available (Brinkmann, 2011). Indeed, nowadays, SCs are deployed at 26 seaport container terminals of relevant sizes all around the world (Wiese, 2011).

(d) Reach stacker

Reach stacker a modified forklift fitted with a spreader for container handling, and also equipped with a telescopic boom that can be raised and extended, as can be shown in Figure 13. The boom allows for handling containers over an intervening stack or rail track. When operating at full outreach, outriggers may be required.



Figure 13 Reach Stacker and Container Terminal using Reach Stacker in Port of Aarhus, Denmark (Kone Cranes, 2020f), (Google Earth, 2018c)

2.4.3. Horizontal transport equipment

There are three main horizontal transport equipment, tractor-trailer system, straddle carrier, and automated guided vehicles (AGV). However, several other options exist, depending on the size of the terminal, the terminal throughput, and the operator's preference in automation. In this research, it will only consider the first two horizontal transport equipment, which is a tractor-trailer system and straddle carrier because the automated guided vehicles are considered to be an option for automation based on those types of equipment.

(a) Tractor-trailer system

Tractor-trailer (TT) system set consists of a purpose-designed tractor coupled to a trailer and is by far the most common method for the movement of containers within a terminal. Tractor-trailer operations are universal since they can be used for every type of ship-to-shore container handling operation and container stacking operation. Tractor-trailer units are also suitable for moving containers between the primary yard to the secondary yard, or even to the hinterland (PIANC, 2014a).



Figure 14 Tractor-trailer operating in Port of Cartagena, Colombia
 (<https://www.portstrategy.com/news101/port-operations/cargo-handling/Emissions-drive-tractor-development-agenda>)

(b) Straddle carrier

Straddle carrier (SC) is best suited for horizontal transport for terminals where the stacking operation is also SC-based. However, in some terminals with an SC stacking operation, TTs or low height SCs (shuttle carrier) is used for the horizontal transport between the apron area and the primary yard area (PIANC, 2014a). Commonly used SCs can stack containers up to three or four tiers high, which means that they can move laden over two or three containers, respectively (Steenken, Voß, & Stahlbock, 2004).



Figure 15 Straddle carrier operating in Port of Gothenburg, Sweden
 (<https://www.dvz.de/rubriken/see/detail/news/neue-straddle-carrier-fuer-goeteborg.html>)

2.5. Container Terminal Capacity

The (throughput) capacity of a container terminal is limited by the capacity of the container stacking area and the quay (Schutt, 2011). In this research, the capacity of the container stacking area is the more focused part for determining the capacity of the container terminal. For the container stacking area, the annual container throughput terminal capacity is usually expressed in terms of 20-foot-container equivalent units (TEUs). The annual container TEU movement/container yard capacity can be determined by the equation below (Thoresen, 2014):

$$C_{TEU} = \frac{A_T \cdot 365 \cdot H \cdot N \cdot L \cdot S}{D \cdot A_{TEU} \cdot (1 + B_f)}$$

where :

C_{TEU} = container movement/year (TEU/year)

A_T = total yard area needed (m²)

- H = ratio of average stacking height to maximum stacking height of the containers, varying between 0.5 and 0.8.
- N = primary yard area or container stacking area compared to total yard area, usually varying between 0.6 and 0.75 of the total yard area
- L = layout factor due to the shape of the terminal area, usually varying between 0.7 for triangular area shape to 1.0 for rectangular area shape
- S = segregation factor due to different container destinations, container maintenance systems (CMS), procedures, etc., usually varying from 0.8 to 1.0
- D = dwell time or the average number of days the container stays in the stacking area in transit. If no information is available, values of 7 days can be used for import containers and 5 days for export containers.
- A_{TEU} = area requirement/TEU, which depends on the container handling system (m^2/TEU)
- B_f = buffer storage factor in front of the storage or stacking area, between 0.05 and 0.1

In other references, the surface area requirements for the different stacks (laden, reefers, empties, oog) can be determined by the equation below (Ligteringen, 2017) :

$$A_T = \frac{N_c \cdot D \cdot A_{TEU}}{r_{st} \cdot 365 \cdot m_c}$$

where :

- A_T = area required (m^2)
- N_c = container movement per type of stack (TEU)
- D = average dwell time (days)
- A_{TEU} = required area per TEU inclusive of equipment travelling lanes (m^2)
- r_{st} = ratio average stacking height over nominal stacking height (0.6 to 0.9)
- m_c = acceptable occupancy rate (0.65 – 0.7)

Referencing from equations above, it can be assumed that acceptable occupancy rate (m_c) from (Ligteringen, 2017) is representing several values from (Thoresen, 2014) = $\frac{N \cdot L \cdot S}{(1+B_f)}$. Due to the possible stochastic arrival of container ships each week, it is advisable to adjust the assumed containers handled per week using a peak factor (Thoresen, 2014). Using these references, we can determine the TGS capacity of the container terminal as follows:

$$TGS = \frac{N_c \cdot D \cdot P}{H_{st} \cdot r_{st} \cdot 365 \cdot m_c}$$

where :

- TGS = Total TGS (TEU ground slots)
- N_c = container movement per type of stack (TEU)
- D = average dwell time (days)
- P = peak factor, typically varying between 1.1 and 1.3
- H = stacking height: no. of containers
- r_{st} = ratio average stacking height over nominal stacking height (0.6 to 0.9)
- m_c = acceptable occupancy rate (0.65 – 0.7)

2.6. Container Terminal Design Guidelines

2.6.1. Apron

The apron width consists of (PIANC, 2014a):

- The space between the edge of the quay wall and the centerline of the waterside rail of the quay cranes
- The space between the centerline of the waterside and landside rails (the rail span of the cranes)
- The back reach of quayside container cranes measured from the centerline of the landside rail
- The landside space not covered by the back reach

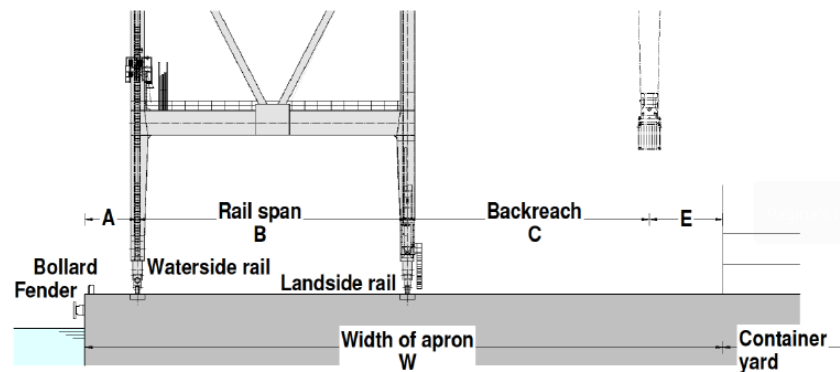


Figure 16 Typical apron width (PIANC, 2014a)

The typical average apron width is (PIANC, 2014a):

- 15 m to 20 m when ship to shore movements are carried out by onboard ship crane
- 25 m to 30 m for mobile harbour cranes
- 40 m to 55 m for STS cranes with a rail span of 15m to 20 m for low throughput terminals
- 55 m to 75 m for STS cranes of 30.48 m rail span and straddle carrier or tractor-trailer system

2.6.2. Storage yard

There are four main types of equipment that is being used as a container stacking equipment in the primary yard: Rubber Tired Gantry Crane (RTG), Rail Mounted Gantry Crane (RMG), Straddle Carrier (SC), and Reach Stacker (RS) system.

a) Rubber Tired Gantry Crane System

An RTGC-operated storage yard is usually subdivided into several yard blocks and driving lanes. The yard blocks, which are laid out parallel to the quay wall, consisting of several rows, in which the containers are stacked end to end, as well as an additional handover lane, which is reserved for trailer-tractor equipment that interacts with the RTGCs (Petering, 2009). Only 30–40 cm space is required to ensure safe crane operations (Chu & Huang, 2005). Block lengths in the range from 18 to 90 bays (Petering & Murty, 2009). The most common ones are the RTG that facilitates stacking heights of 1-over-4 and 1-over-5 (Chu & Huang, 2005), but 1-over-7 cranes are also available, leading to yard densities of up to 1,000 TEUs per hectare (Kempe, 2013). Typically, containers are stacked 6 rows wide in RTGC operated yard blocks; however, yard blocks up to 8 + 1 rows wide are also reported (Chu & Huang,

2005). A typical container stacking layout is shown in Figure 17, and a typical cross-section container stacking arrangement (for 6+1 spans and lifting 1 over 4) is shown in Figure 18.

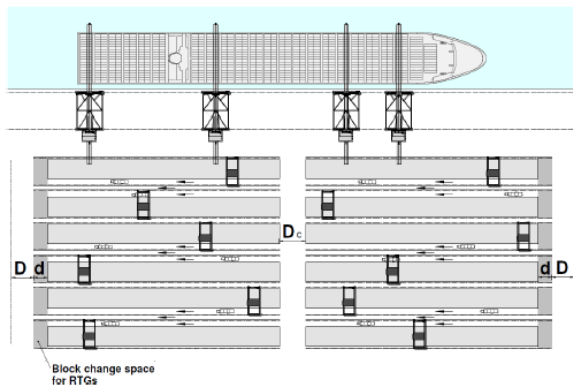


Figure 17 Typical container stacking layout for Rubber-Tired-Gantry Crane (PIANC, 2014a)

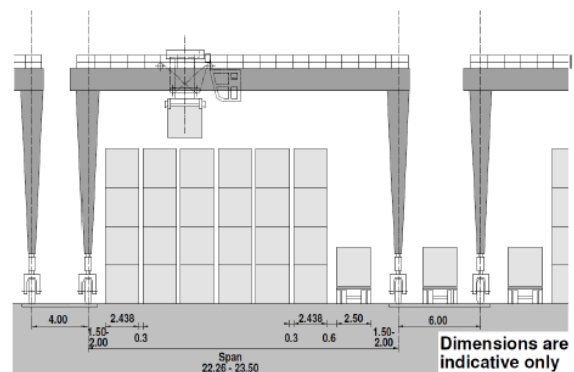


Figure 18 Typical cross-section container stacking arrangement for Rubber-Tired-Gantry (PIANC, 2014a)

In Figure 17, we can see that the typical container stacking layout for Rubber-Tired Gantry Crane used two types of block-truck lane arrangement, where there is one container block that has one truck lane at each side of the container block, and there are others container blocks that has one truck lane at each side of every two blocks. Some studies conduct the RTG-layout in parallel to quay orientation with one truck lane at each side of the container block ((Petering & Murty, 2009), (Petering, 2009), (Lee & Kim, Optimizing the Yard Layout in Container Terminals, 2013) (Alcalde, Kim, & Marchan, 2015)), and some studies conduct the RTG-layout in parallel to quay orientation with one truck lane at each side of every two blocks ((Wiese, Kliewer, & Suhl, 2010), (Jiang, Chew, Lee, & Tan, 2013), (Zhen, 2016), (Zhen, Xu, Wang, & Ding, 2016)). Nevertheless, research conducted in comparing both types of layout arrangement, resulting in higher throughput can be achieved if the yard provides more vehicular travelling space and accommodate a higher number of vehicles (Lee, Lee, & Chew, 2018). Therefore, the design rules will be set to generate a container terminal with one truck lane at each side of the container block.

b) Rail-Mounted Gantry Crane System

For the RMG system, the yard layout is either laid out parallel or perpendicular to the quay wall (Kempe, 2013). However, the perpendicular to the quay wall orientation, or front-end-loading systems, are mostly applied at large and very large terminals that require dense stacking operations (Kalmar, 2011), and such systems have been put into operation, for example, the ECT Delta Terminal in Rotterdam (Netherlands) and the CTA in Hamburg (Germany) (Dekker, Voogd, & van Asperen, 2006). A typical front-end-loading RMGC system consists of yard blocks, corresponding waterside, landside handover areas, and service lanes in between the yard blocks. Apart from their arrangements, the yard blocks are very similar to RTGC-operated yard blocks, as containers are stacked end to end in several rows that are separated by only 30–40cm clearance (Chu & Huang, 2005). Since the handover takes place at the front ends of the block, no handover lane is required inside the crane portal (Kempe, 2013). The dimension of yard blocks differs between terminals, but typically the order of magnitude for perpendicular RMGC yard blocks is 28–48 bays long and 6–10 rows wide (Petering & Murty, 2009). The most common ones are the RMG that facilitate stacking heights of 1-over-4 (Chu & Huang, 2005), but 1-over-6 cranes are also available, leading to high yard densities exceeding 1,200 TEUs per hectare (Saanen & van Valkengoed, 2005). A typical

container stacking layout is shown in Figure 19, and a typical cross-section container stacking arrangement is shown in Figure 20.

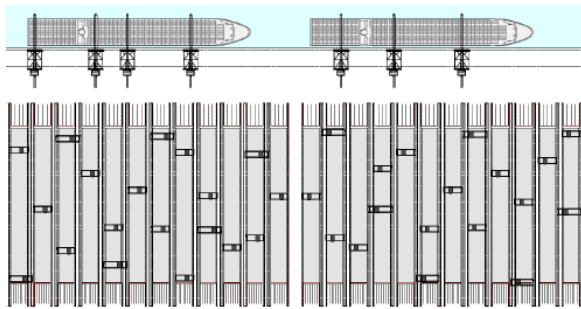


Figure 19 Typical container stacking layout (perpendicular) for Rail-Mounted Gantry Crane (PIANC, 2014a)

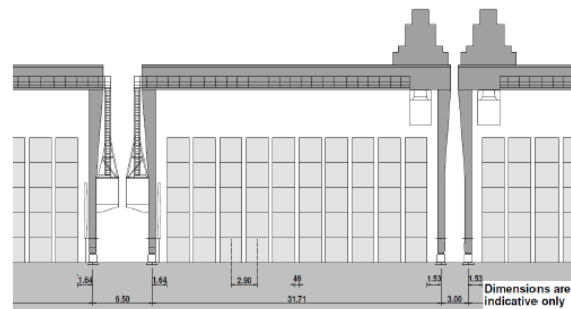


Figure 20 Typical cross-section container stacking arrangement for Rail-Mounted-Gantry Crane (PIANC, 2014a)

c) Straddle Carrier system

A typical SC-operated storage yard consists of several blocks and driving lanes as well as one handover area for loading and discharging trucks from the hinterland. Each block contains long rows with the containers placed end to end. The rows have to be separated from each other by a distance such that an SC can move along the row, straddling the containers to reach the required storage position. Considering the internal span of an SC, spaces for wheel travelling and some safety distance, there has to be a clearance of 1,5–2 m between two rows of a block (Kemme, 2013), where only around 1,4 – 1,5 m suggested in a design guidelines reference (PIANC, 2014a). The width of the driving lanes should be dimensioned such that an SC that usually has a turning radius of about 9,4m can manoeuvre in and out of the rows and travel between blocks (Chu and Huang 2005), where a width of about 20 m is suggested (Kemme, 2013). The container blocks can either be arranged parallel or perpendicular to the quay wall, that stacking orientation cannot be concluded that one option is superior to another (Wiese, 2011). The number of rows per block and the length of the rows are influenced by several factors and vary notably between different terminals. The stacking height of SC-operated storage yards is usually two or three layers high, which leads to yard densities of about 500–750 TEUs per hectare (Kalmar, 2011). For Straddle Carrier, a typical container stacking layout is shown in Figure 21, and the details on the container stacking layout are shown in Figure 22.

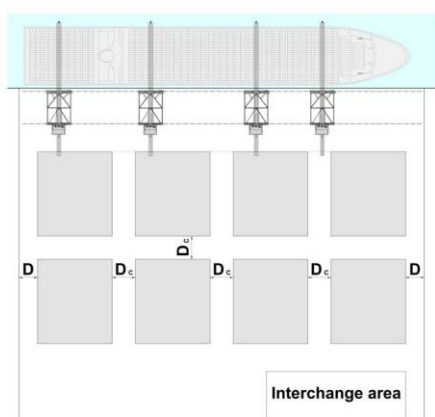


Figure 21 Typical container stacking layout for Straddle Carrier (PIANC, 2014a)

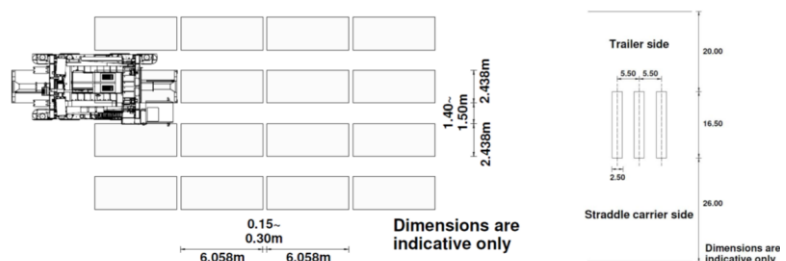


Figure 22 Details on typical container stacking layout for Straddle Carrier (PIANC, 2014a)

d) Reach Stacker

Reach stacker operations are generally very flexible and suitable for start-up phases of small terminals. The disadvantage of this system is its relatively low container stacking density. Reach stackers are commonly used in combination with tractor-trailer transport systems. In general, reach stackers place containers in narrow blocks, and able to stack 3-4 rows across and up to 5 tiers in height. The minimum space for the operation of a reach stacker between the block is optimally 18 m (PIANC, 2014a). For reach stacker, a typical container stacking arrangement is shown in Figure 23.

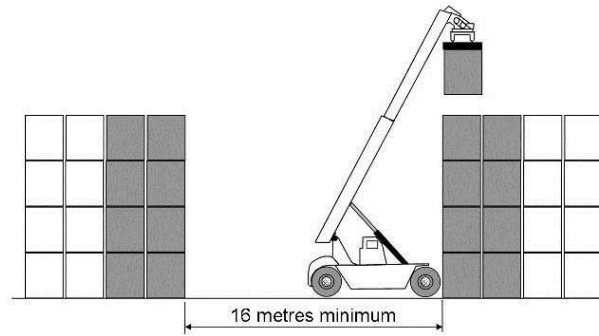


Figure 23 Typical cross-section container stacking arrangement for Reach Stacker (PIANC, 2014a)

3. Design Tool Development

3.1. Overview of the Design Tool

In Section 1.2, it is stated that area and cost are the primary deliverables for a concept design phase. Based on the literature study for earlier container terminal design research and development in Section 2.1, it can be seen that it is only a few numbers of references (e.g. (Liu, Jula, & Iannou, 2002), (Mohseni, 2011), (Kox, 2017), (Koster, 2019)) that are related to area and cost performance criteria, where it mostly related to operational performance criteria. It can also be seen that there only small numbers of references that show the results of the study as a visualization for the container terminal layout (e.g. (Duinkerken, Evers, & Ottjes, 2001), (Liu, Jula, & Iannou, 2002), (Kempe, 2013), (Koster, 2019)), where most of them show the results in the manner of calculation only. Other than that, In Section 1.2., it is also stated that automation is the preferred method to ease and accelerate the complex design process in designing container terminal, and based on the literature study in Section 2.1., several studies develop a design tool by modelling automatable tasks (e.g. (Mohseni, 2011) and (Koster, 2019)). So, this study will be based on the latest study by (Koster, 2019) because it is related to area and cost performance criteria, it shows the visualization of the container terminal layout as the results of the study, it develops a design tool by modelling automatable tasks, and it is the latest study concerning container terminal design. These reasons make it the most relevant study to the problem stated in this research. Besides, the design tool from this study requires further development when we would like to consider area limitations in determining the container terminal design.

The development of a model for the design tool will be coded in Python language to perform the terminal layout generation, and then integrated with the existing design tool from (Koster, 2019) to execute the terminal element calculation and cost estimation. Besides the problem regarding the area limitation in designing a container terminal, there is also another reason that triggers this research regarding the working environment of the previous design tool. The working study of the previous study by (Koster, 2019) is briefly explained as follows. First, the container terminal elements are calculated within a program called Python; then the output is imported to another program called Grasshopper to generate the container layout; finally, the output from container terminal elements and layout is imported back to Python to do the terminal cost estimation tasks. Thus, this research tried to have all the automated tasks in one environment, which is in Python, due to the advantages of keeping all automated tasks in one environment where it will boost barriers involved in the lifecycle of software development (Mulonda, 2019). Moreover, the Grasshopper tool works more-or-less like a 'black box' because it is not a free-open-source program like Python. The overview of the workflow of the design tool after being synchronized with the existing tool is shown in Figure 24.

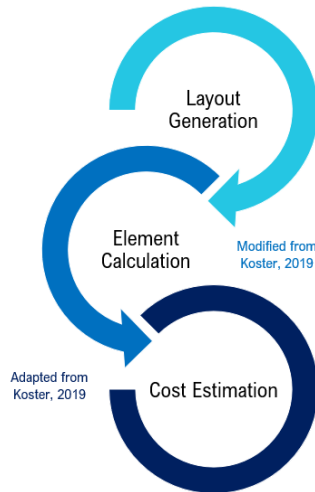


Figure 24 Workflow of the design tool
(Author's illustration)

This research is mainly focused on developing the design tool for generating the container terminal layout tasks based on the input of the terminal shape and dimension, the throughput demand, and the design rules. These design rules are mostly based on the ports and terminal design guidelines from (PIANC, 2014a), (Thoresen, 2014), (Ligteringen, 2017), and also the latest literature study from (Koster, 2019), (Chu & Huang, 2005), (Petering & Murty, 2009), and (Kemme, 2013). Then, output elements that can be determined from the generated container terminal layout, which are terminal throughput capacity, TGS capacity, number of blocks, block length, block width, stacking area, and stacking density (after it is validated by the actual projects and references from other studies), will be used to calculate the terminal element and estimate the terminal cost by modifying the existing design tool. The scope of the design tool can be incorporated in Table 2.

Table 2 Scope of the design tool

Aspects	Included in the tool	Outside of the scope
Layout Generation		
Governing parameters & boundary conditions	Throughput, terminal shape and dimension	-
Type of equipment	Apron: Ship-to-shore (STS) crane Yard: Rubber Tired Gantry Cranes (RTG), Rail Mounted Gantry Crane (RMG), Straddle Carrier (SC), Reach Stacker (RS) Horizontal Transport: Tractor-trailer system, Straddle Carrier (SC)	Apron: Mobile Harbor Crane (MHC) Yard: Forklift, Top Loader Horizontal transport: Automated Guided Vehicle (AGV)
Terminal shape generation	Apron, primary yard, secondary yard	-
Container block generation	Primary yard (laden & reefer stacks)	Secondary yard (empty & oog stacks)

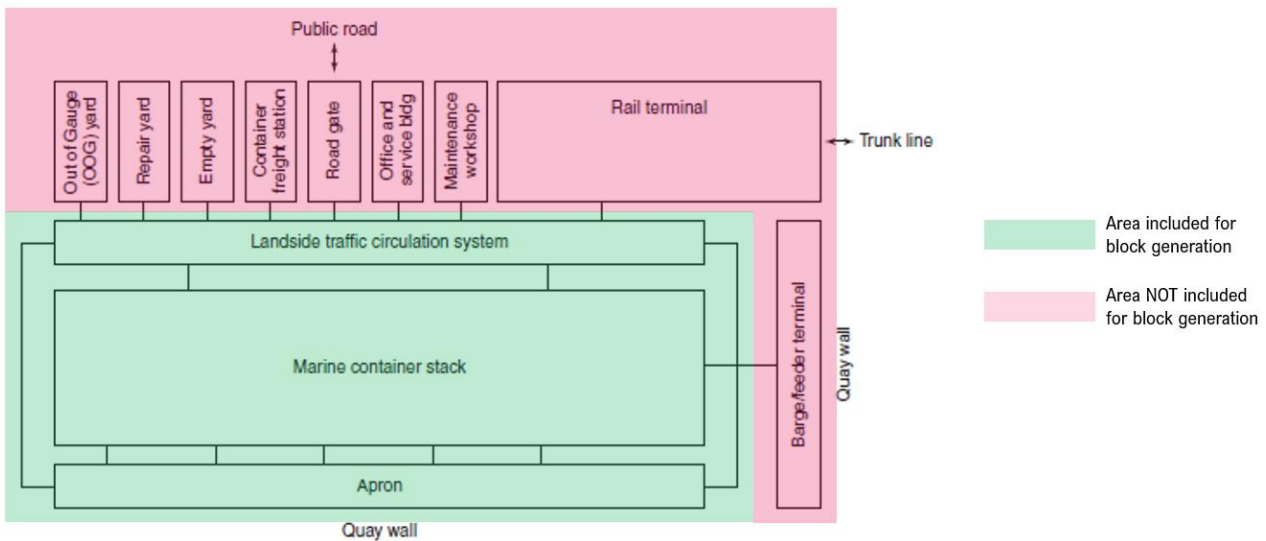


Figure 25 Area included and not included for block generation (Author's illustration)

Two of the most fundamental parameters of the concept design phase is the terminal throughput and stacking equipment (Koster, 2019). However, as can be seen from Table 2, the governing parameters and boundary conditions are including not only the terminal throughput but also the terminal shape and dimension. Also, the type of equipment that is included as an option within the design tool includes three types of mostly used container stacking equipment, which are RTG, RMG, and SC (Kempe, 2013), and RS is included as an alternative in designing small container terminals. Moreover, the area included for the terminal shape generation is including all terminal areas, including the apron, the primary yard, and the secondary yard. However, the container block that is generated by the design tool is only the container block in the primary yard, as can be seen by Figure 25, since the elements in the secondary yard location have a wide variety factor of preferences from its stakeholder. In most cases, it is easy to distinguish the primary yard of a container terminal, but hard to distinguish the secondary yard area. Therefore, the secondary yard is only included in the terminal shape generation but not for the container block generation, by implementing the ratio of the primary yard and the total yard area from literature, which is around 0.6 – 0.7 (Thoresen, 2014), (PIANC, 2014a)). Moreover, the input and the output of the design tool are incorporated in Table 3 below.

Table 3 Input and output for every task of the design tool

Tasks	Input	Output
Layout Generation		
Creating boundary condition in space	Terminal shape and dimension	Terminal shape and dimension as a boundary in space
Defining area for container stacks	Apron width, the ratio between the primary yard and terminal	Terminal area
Generating container stacking blocks	throughput demand, quay length (terminal shape and dimension), traffic lane width, crossing lane width,	Container stacking blocks on the terminal

Tasks	Input	Output
	light mast lane, equipment area requirement, maximum stacking blocks length, stack arrangement, and orientation	
Generating TGS in container stacking blocks	TGS dimension, container clearance, container stacking block length, container stacking block width	TGS capacity, TEU capacity, number of blocks, block configuration, stacking area, stacking density

Furthermore, this research will modify the automated tool that has been developed in the previous study for the calculation of the terminal element part. From the previous study, the terminal element tasks are done before the terminal layout generation, where the input is only based on the terminal throughput. In this research, the terminal element calculation is done after the terminal layout generation, due to the additional input based on the terminal shape and dimension, that will cause changes in the terminal throughput capacity. Finally, the terminal cost estimation can be performed using the existing algorithm within the design tool. Inefficiency on the operation side when non-rectangular container terminal is implemented is one of the issues that will affect the terminal element calculation and cost estimation. For example, with a non-rectangular container terminal layout, the driving lane might be not as ideal as the driving lane on a rectangular container terminal layout. Therefore, a non-rectangular terminal might have a larger number of terminal elements, leading to higher cost estimation. However, this inefficiency is not part of the design tool, because, on a concept design phase, the 'operational' performance criterion is not considered. This efficiency might be incorporated in the design process by increasing the cost estimation with an assumption (e.g. 5 or 10% higher than the rectangular container terminal). Nevertheless, until now, there is no insight from previous literature on how to specifically incorporate the inefficiency in a non-rectangular container terminal in a concept design phases. The workflow for terminal element calculations that also include the terminal shape and boundary can be schematized by the Figure 26, where it shows the relationship between the throughput demand, terminal shape and dimension, and the design tool's objects.

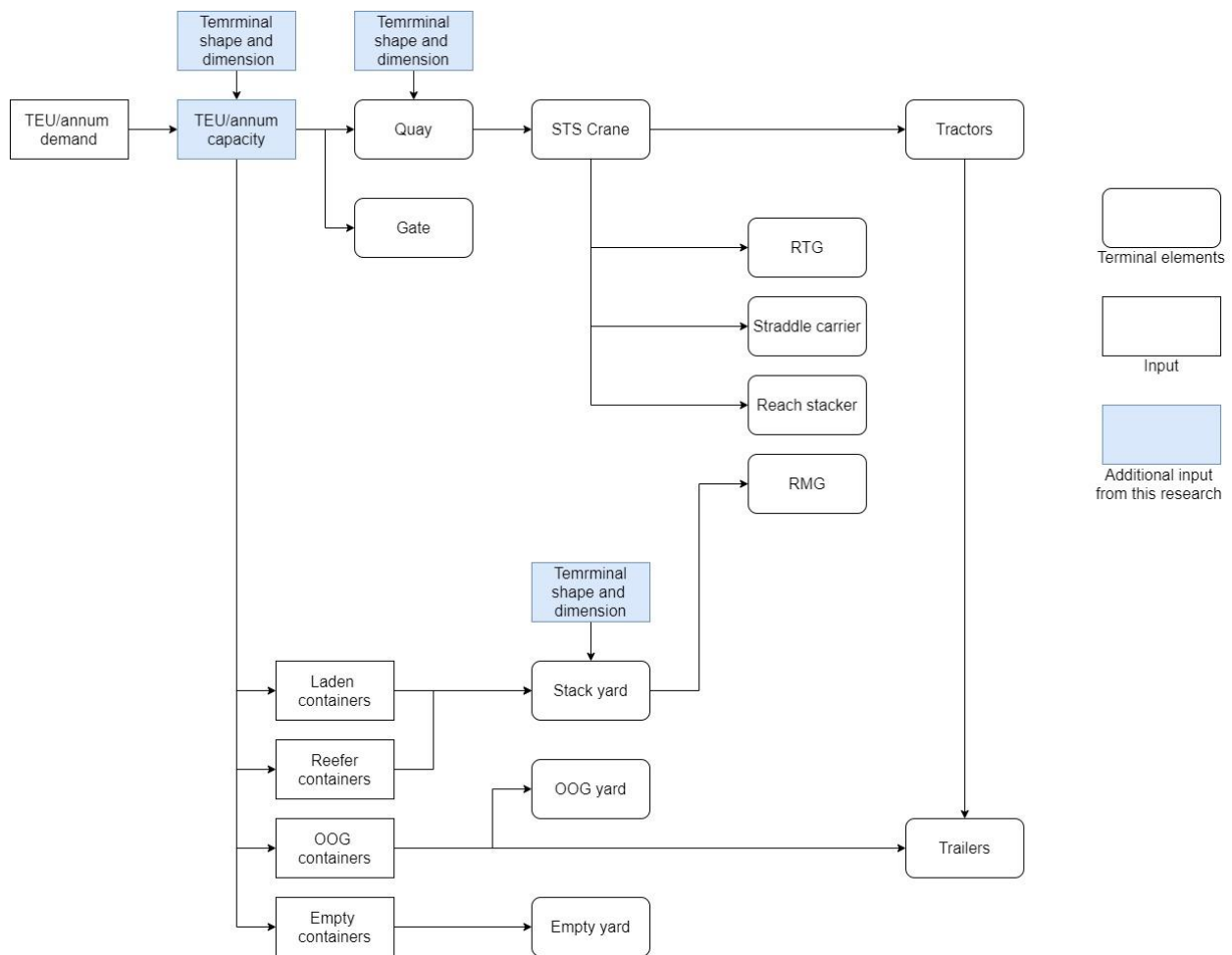


Figure 26 Relationship between the annual throughput [TEU/annum], terminal shape and dimension, and the various terminal objects (adapted from Koster, 2019)

3.2. Layout Generation

3.2.1. Overview of the layout generation process

In this process, a set of logic, in the form of algorithm-driven by input parameters, is established on the design tool to container terminal design alternatives, based on the terminal shape and dimension, terminal throughput, design rules, and type of stacking equipment chosen. The illustration for the design tool work flowchart for the non-rectangular container terminal is shown in Figure 27.

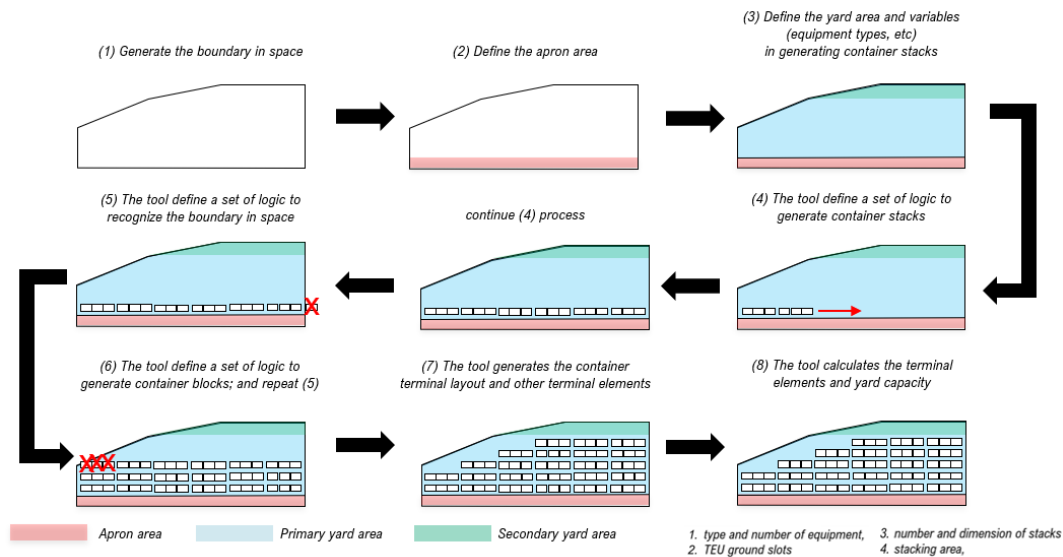


Figure 27 Illustration for the design tool work flowchart for non-rectangular-shape container terminal [pictures are not in scale] (Author's illustration)

The overview of the layout generation process is explained as follows. First, the design tool generates the container terminal shape and dimension and defines it as a boundary condition for the container terminal layout generation. Next, the design tool defines the apron area and the yard area (primary yard and secondary yard). It will be based on the design rules that the container terminal will generate more or less the idealized layout and elements location in a container terminal, as can be seen in Figure 6, where the apron is located as near as possible to the sea-side, then adjacent to the apron are the area for container stacking and traffic circulation system that is represented by the primary yard, and finally adjacent to the primary yard is the other terminal elements that are represented by the secondary yard. Furthermore, a set of logic is defined into the design tool to generate container block in quay-parallel direction, and another set of logic is defined into the design tool to recognize the container terminal shape and dimension as boundary conditions. Therefore, the design tool stops generating container block if they meet the boundary of the container terminal. Then, a set of logic is defined to the design tool to generate container block in quay-perpendicular direction, and another set of logic is defined to recognize the boundary of the primary yard area since the primary yard is the area assigned to stack the laden and reefer containers on the terminal yard. Finally, the tool calculates the terminal capacity, TGS capacity, number of container blocks, container block configuration, container block location, stacking area, and stacking density. The literature study regarding the design guidelines for container terminals (e.g. PIANC, 2014; Thoresen, 2014; Ligteringen, 2014) will give a lot of valuable information in designing the container terminal, which will serve as the primary sources of input for the tool. After the layout of the container terminal is generated, the terminal elements are calculated, and the terminal cost is estimated by using the existing tool from the previous study by (Koster, 2019). The detail explanation for terminal elements calculation and terminal costs estimation will not be explained in detail in this research since it already discussed in the previous study by (Koster, 2019). However, information regarding the terminal element calculation and cost estimation part can be found in Appendix F.

3.2.2. Design rules

For the apron side, by implementing the design guidelines from the literature study in Section 2.5, the design rules that will be used in the design tool development for the apron area can be shown in Figure 28 and Table 4.

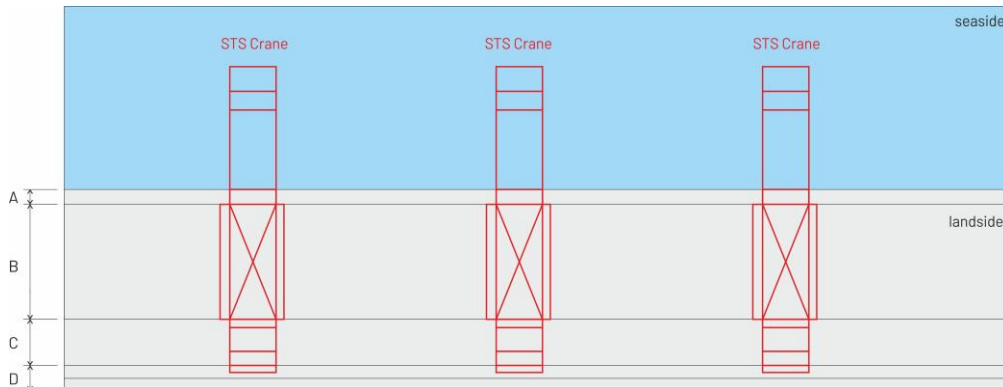


Figure 28 Illustration of the design rules applied for Apron Area
(Author's Illustration)

Table 4 Design Rules Applied for Apron Area

Element	Value	Source	Remarks	
A	Quay wall to rail	5 m	PIANC (2014a)	Minimal 4 m for large vessels
B	Crane rail gauge	35 m	PIANC (2014a)	Crane rail for the newest generation STS cranes capable handling Triple E class vessels
C	Hatch Cover area	15 m	PIANC (2014a)	Together with traffic lanes, should be at least 16 m
D	Traffic lane	2 x 3.5 m	PIANC (2014a)	For tractor-trailer systems. For SC systems two-way lanes of 5.5 m are advised

Using these values above, for the tractor-trailer systems the required apron width will be 62 meters, and for the straddle carrier systems, the required apron width will be 66 meters. For the primary yard or storage yard, the design tool for the layout generation will be generated for each type of container handling equipment. As discussed in the previous section, there will be four main equipment that will be used as stacking equipment, which is Rubber-Tired Gantry (RTG) Crane, Rail-Mounted Gantry (RMG) Crane, Straddle Carrier (SC), and Reach Stacker (RS) (to incorporate small-scale terminals). Implementing the design guidelines from the literature study, design rules that will be used in the design tool development for yard area for RTG crane can be shown in Figure 53 and Table 5 below.

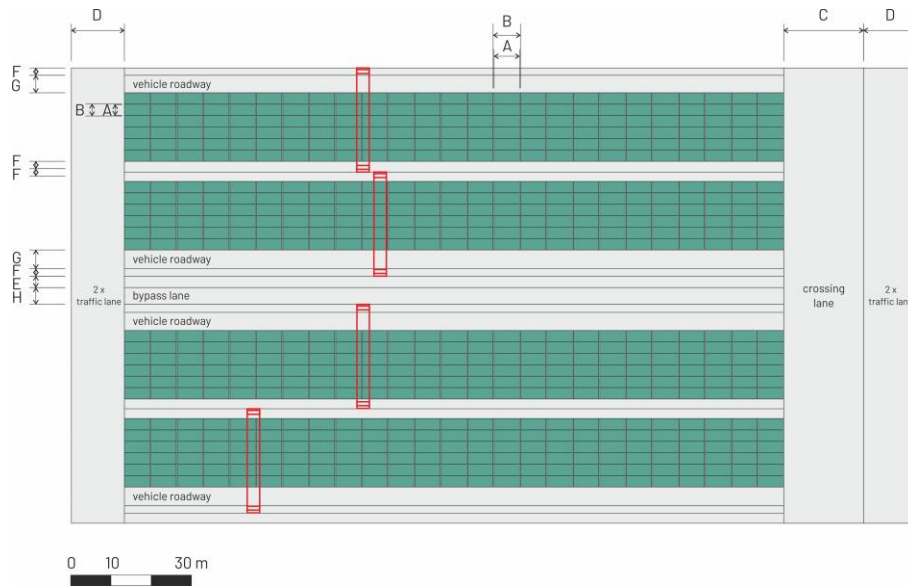


Figure 29 Illustration of the design rules applied for the apron area (Author's Illustration)

Table 5 Design Rules Applied for Yard Area (RTG Crane)

Element	Value	Source	Remarks	
A	Net TGS	2.44 x 6.10 m	PIANC (2014a)	
B	Gross TGS	2.79 x 6.45 m	Koster (2019)	
C	Crossing lane width	19.4 m	Koster (2019)	
D	2x Traffic lane	2 x 3.5 m	Koster (2019)	Two-way
E	Light mast lane	2.80 m	Koster (2019)	Requires free space to lower the mast for maintenance
F	Track width	1.90 m	Koster (2019)	
G	Vehicle roadway	4.35 m	Koster (2019)	One-way
Layout Rule	Value	Source	Remarks	
Maximum stack length	90 TEU	(Petering & Murty, 2009)	Block length in the range from 18 to 90 bays for RTGC systems	
Minimum stack length	18 TEU	(Petering & Murty, 2009)	Block length in the range from 18 to 90 bays for RTGC systems	
Stack arrangement	Back-to-back	(Wiese, Kliewer, & Suhl, 2010)	Most common stacking strategy	
Stack orientation	Parallel to quay	(Wiese, Kliewer, & Suhl, 2010)	Operational purposes	

The design rules will be the fundamental rules for the design tool, as it will represent the area required for the stacking equipment to be working properly. Thus, all operational point of view is assumed to be working properly using these design rules.

For RMG, SC, and RS, the design rules that will be used in the design tool development for yard area can be seen further in detail in Appendix A.

3.2.3. Layout generation design tool

The design tool generates the layout by means of placing the “physical” elements (terminal area such as apron, primary yard, secondary yard; container blocks) into a canvas based on the throughput demand and terminal dimension as an input. In generating the container terminal layout, a software called Python is used for generating the container terminal layout, since Python is simple enough for anyone to get started, yet powerful enough to handle complex and extensive design projects (Pajankar, 2017). Python is a very suitable tool for automating an engineering design process and is thus commonly used in the field. A reason to use Python is to have all the automated tasks in one environment, due to the advantages of keeping all automated tasks in one environment where it will boost barriers involved in the lifecycle of software development (Mulonda, 2019). Another advantage is the simplicity of Python, which is induced by the vast range of package options available for a large number of applicability, making it ideal for a wide variety of design automation projects (Pushkov, 2019).

In addition, one of the packages in Python that is highly valuable and important in the process of the layout generation in the design tool is a package called ‘Shapely’. Shapely is a python package for the manipulation and analysis of two-dimensional geospatial geometries. Shapely is based on the GEOS library, which implements a wide range of manipulations in C++. (Westra, 2013). The shapely package will provide function in several requirements to do the algorithm of the layout generation (i.e. shapely could determine whether a point is inside or outside a defined space), which is very beneficial to the layout generation tool. An overview of the python module: Shapely can be further explained through Appendix E.

In general, the flowchart of the layout generation tool can be shown in Figure 30.

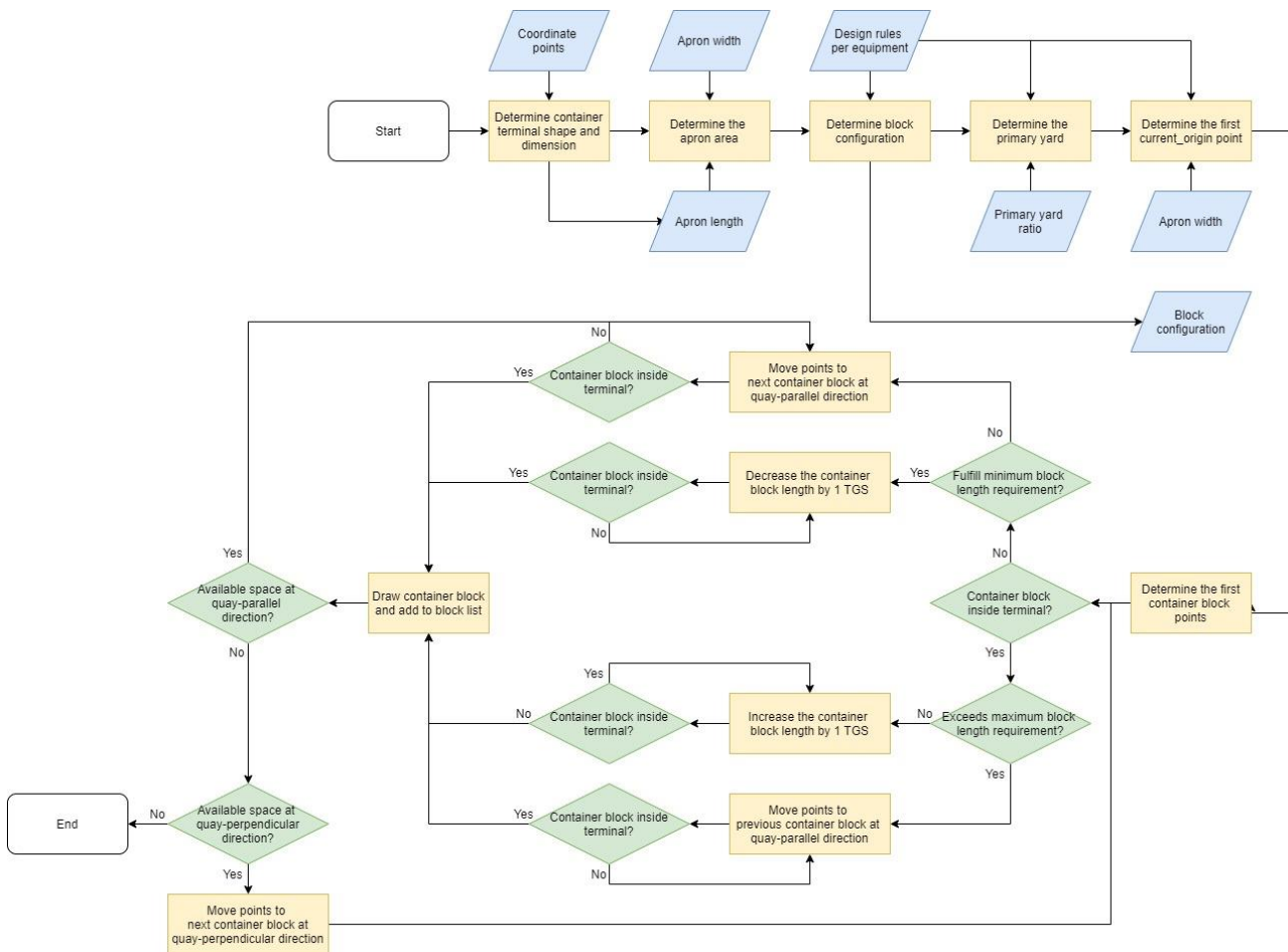


Figure 30 Flowchart for the layout generation tool.
(Author's Illustration)

First, the container terminal shape and dimension are defined by determining the number of points and the x-y coordinates of the points so that it can determine the boundary conditions of the terminal. Moreover, seven container terminal shapes are used for the pilot simulation of the design tool to cover all possibilities of the condition that could happen in generating the container blocks. In detail, all main shapes that are used for the pilot simulation of the design tool are discussed in Appendix C. However, in this section, to make it efficient in describing how the layout generation tool works, specific terminal shape and dimension is chosen, which is a trapezoidal container terminal that will represent a non-rectangular container terminal that can be shown in Figure 31. Next, based on the terminal dimension, the apron length is determined, while the apron width is determined from the design rules in the literature study. Therefore, the apron area could be determined based on the apron length and the apron width. For illustration, the apron area in the container terminal is shown in Figure 32.

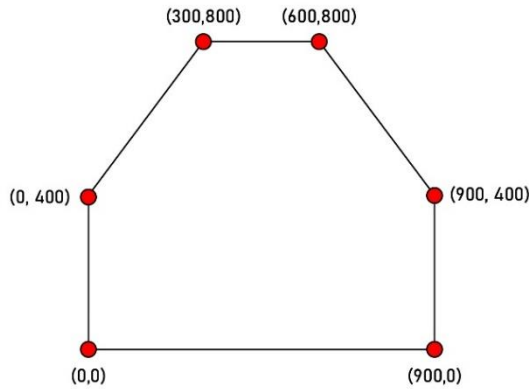


Figure 31 Container terminal shape and dimension determination (Author's illustration)

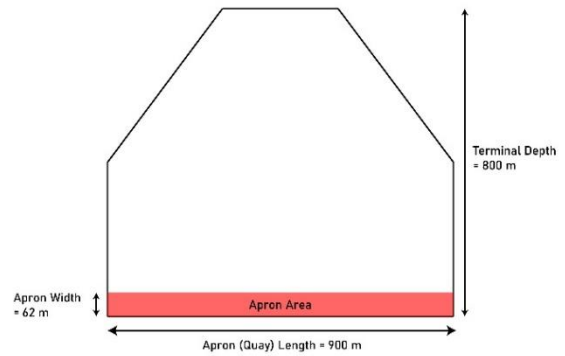


Figure 32 Apron (quay) length and apron width calculation (Author's illustration)

Next, the block configuration, which is the block length and the block width, is determined based on the design rules that differ for each type of equipment. As mentioned before, there will be 4 types of equipment that is included in the design tool, which are RTG, RMG, SC, and RS. In detail, the container terminal layout generation for all types of equipment are discussed in Appendix C. However, in this section, to make it efficient in describing how the layout generation tool works, an RTG system is chosen as an example. For RTG, the block length can be calculated using the equations below.

$$n_{\text{block}} \approx \frac{\text{quay length} - \text{traffic lane}}{\text{max. block length} + \text{crossing width lane}}$$

$$\text{block length} = \frac{\text{quay length} - \text{traffic lane} - n_{\text{block}} * \text{crossing width lane}}{n_{\text{block}}}$$

Then, as mentioned before, part of the yard that will be used for the stacking activities is the primary yard; therefore, the primary yard area should be determined. The primary yard area is determined based on the container terminal configuration; and also the design rules regarding the ratio of the primary yard area to the total terminal yard area, which should be around 0.6 – 0.75 (Thoresen, 2014) or 0.6 – 0.7 (PIANC, 2014a). For the design tool, the value 0.6 – 0.7 will be used. Also, another consideration where the design rules are applied is that the primary yard should be located as near as possible to the apron area, and the secondary yard should be located as near as possible to the hinterland, following the idealized container terminal layout that can be shown by Figure 6 in the literature review. The flowchart of the process in determining the area of the primary yard can be shown in Figure 32 below.

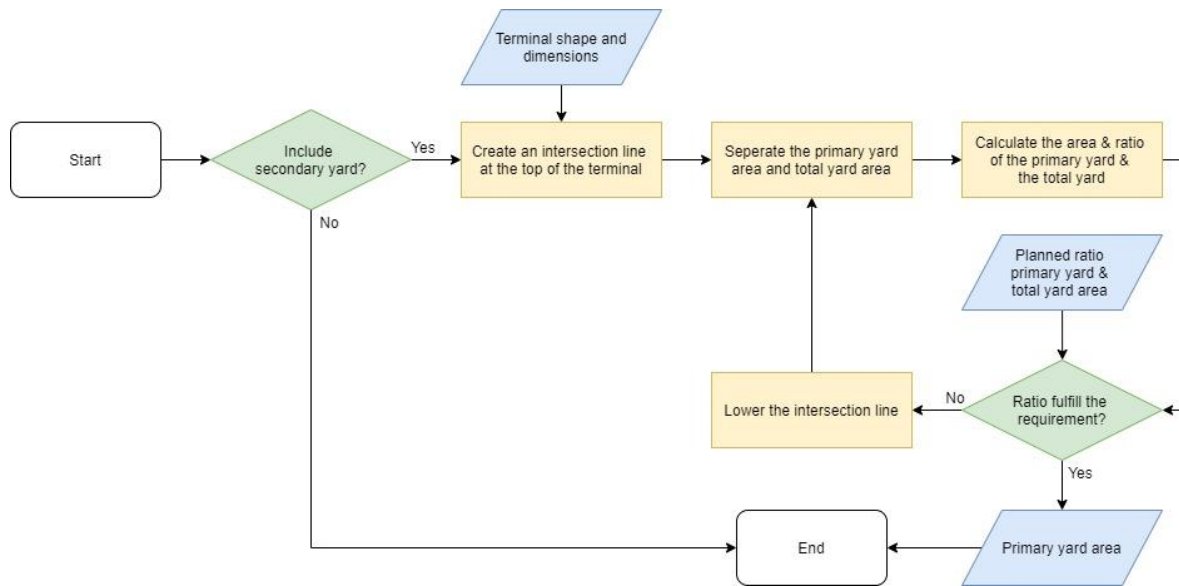


Figure 33 Flowchart for the process in determining the primary yard area (Author's illustration)

After that, the generation of the container blocks is started. First, the design tool will define the point (in coordinate) that represents the 'first point' for each container block, which the design tool called the 'current_origin'. And then, based on that point, the design tool defines the points (in coordinates) that will represent each edge location of a container block. There will be 5 points, where for RTG, point 0 and point 4 are the 'current_origin', point 1 is the 'current_origin' y-coordinate plus the block width, point 2 is the 'current_origin' x-coordinate plus the block length and y-coordinate plus the block width, point 3 is the 'current_origin' x-coordinate plus the block length. For illustration, the points on the container blocks can be shown in Figure 34. Then, using the shapely module, the design tool checks whether the points of the container blocks are inside the terminal or not. The term 'inside' the terminal means that if all container block coordinate points are inside the stacking area, and the term 'outside' the terminal means that if one of the container block coordinate points is outside the stacking area. If the container block is inside the terminal, the design tool checks whether it is possible to add either 1 TGS or 1 full container block. This algorithm is important because it will make the design tool to use the available space as much as possible so that the container terminal area is used efficiently. If it is possible to add either 1 TGS or 1 full container block, the design tool does the process until it reaches the condition when it is not possible anymore and defines the new block configuration. If it is not possible to add even 1 TGS, the design tool defines the initial block configuration. After the block configuration is defined, the design tool generates a container block in the form of a rectangle based on the block coordinate points. In generating the container block, a set of design rules regarding maximum and minimum container block length is applied. A container block will be generated if only the block length is less than the maximum block length, and more than the minimum block length. For illustration, this process can be shown in Figure 35.

Equipment type : RTG

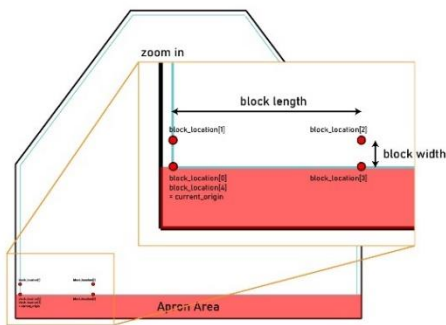


Figure 34 The current_origin and container_block points for RTG (Author's illustration)

Equipment type : RTG

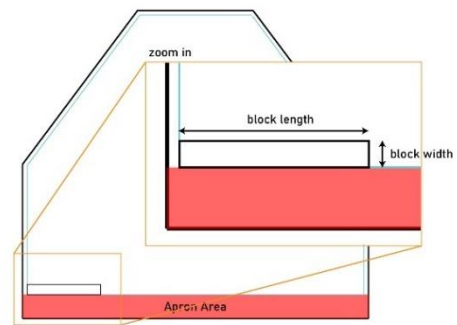


Figure 35 The process of generation of a container block for RTG (Author's illustration)

Next, the design tool checks whether there are available spaces in the quay-parallel direction. If there are available spaces in the quay-parallel direction, the design tool moves the container block coordinate points to the next container block location in quay-parallel direction. Design rules regarding the required space between the container block are applied in this algorithm. Accordingly, the design tool does the container block generation iteratively until it reaches the condition where there are no available spaces to generate a container block in quay-parallel direction. For illustration, this process can be shown in Figure 36. However, if there are no available spaces in the quay-parallel direction, the design tool checks whether there are available spaces in the quay-perpendicular direction. If there are available spaces in the quay-perpendicular direction, the design tool moves the container block coordinate points to the next container block in the quay-perpendicular direction. Design rules regarding the required space between the container block are applied in this algorithm. For illustration, this process can be shown in Figure 37.

Equipment type : RTG

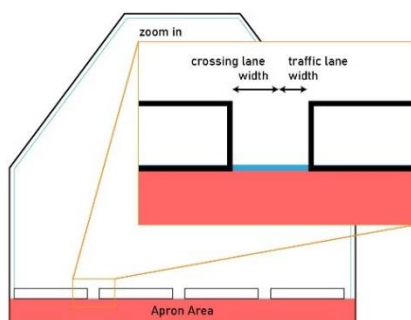


Figure 36 Container block generation in the quay-parallel direction (Author's illustration)

Equipment type : RTG

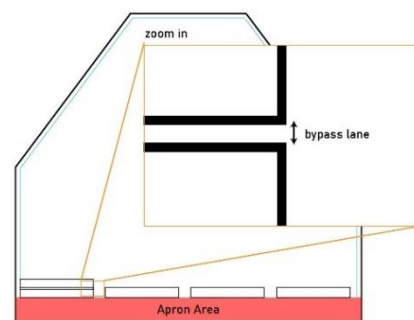


Figure 37 Container block generation in the quay-perpendicular direction (Author's illustration)

Then, the design tool checks whether the points of the container blocks are inside the terminal or not iteratively. However, if the container block is outside the terminal, meaning that one of the container block coordinate points is outside the stacking area, the design tool checks whether it is possible to subtract 1 TGS block. This algorithm is also important because it will make the design tool to use the available space as much as possible so that the container terminal area is used efficiently. If it is possible to subtract 1 TGS, the design tool does the process until it reaches the condition when it is not possible anymore and defines the new block configuration. However, a container block will be generated if only the block length is more than the minimum block length. If the container

block length is less than the minimum block length, the design tool moves the container block coordinate points to the next container block location in quay-parallel direction. For illustration, this process can be shown in Figure 38. Subsequently, the design tool continues the container block generation until it reaches the boundary of the stacking area, where there are no available spaces to generate container blocks in the quay-perpendicular direction. For illustration, this process can be shown in Figure 39.

Equipment type : RTG

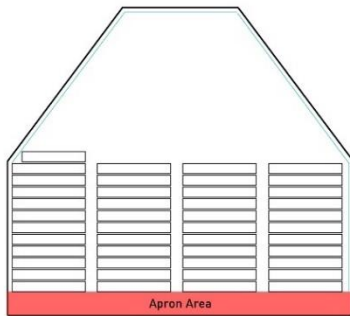


Figure 38 Generating container block and process of decreasing container block length for RTG (Author's illustration)

Equipment type : RTG

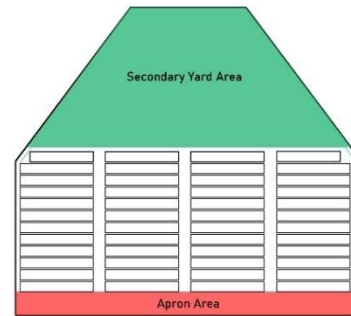


Figure 39 Generating container blocks further and defining secondary yard area for RTG (Author's illustration)

This section only explains in detail the layout generation process of the design tool for an RTG system terminal, and a specific trapezoidal terminal shape and dimension. As mentioned before, seven container terminal shapes are used for the pilot simulation of the design tool, to cover all possibilities of the condition that could happen in generating the container blocks. Moreover, there are three other types of equipment for the container terminal, which are RMG, SC, and RS system terminal. In detail, the process and the result of the layout generation for all shapes and types of equipment used for the pilot simulation of the design tool are discussed further in Appendix C.

3.2.4. Placement of reefer containers

There are no specific design guidelines in determining the placement of reefer containers in a container terminal. Design guidelines from (PIANC, 2014a) only discuss the necessity of reefer racks in container terminal since reefer containers require connection to an electrical power supply to maintain the function of the integral refrigeration plants while it is stored in the container terminal. A comparison study of 4 types of reefer container layouts arrangement in automated container terminal (which has perpendicular to quay orientation, same as RMG) shows that the distributed layout, in which reefer containers are distributed over all the blocks, is more effective in reducing the operation time (Choi, Park, Kwon, & Yoo, 2006). In addition, in perpendicular layout configuration, reefer racks are frequently positioned at the landside end of the blocks, which makes it easier for service workers to access the reefer containers (Wiese, 2012). This arrangement (distributed and land-sided) of reefer racks will be implemented in the design tool for the RMG-system terminal. For other terminals (RTG, SC, RS), the reefer containers will be to be positioned in the middle of the container terminals, assuming that it would be more effective for both import and export operations.

3.2.5. The complementary capabilities of the design tool

In developing the design tool, several complementary capabilities are developed to become a feature for the design tool. Firstly, the tool will have the capability to not including the secondary yard area in the algorithm, which will generate very different container terminal layout compared to the case if the tool includes the secondary yard. This capability will answer problems in a situation when the design tool would like to be validated with existing container terminals, and it is very really hard to determine the separation between primary and secondary yard areas of the container terminal since it might be varying for every container terminal design. Secondly, the tool will have the capability to determine the apron width. This capability will be valuable also when the design tool would like to be validated with the existing terminal since using the same apron width will increase the similarity of the layout generated by the design tool and the existing terminal. Nevertheless, if this feature is not activated, the design tool will determine the apron width based on the terminal dimension and the design rules, that are already defined in Section 3.2.2. Thirdly, the tool will have the capability to set a 'default' block configuration as the basic container block configuration in generating the container terminal. This capability will be beneficial also when the design tool would like to be validated with existing container terminals since using more alike block configuration will increase the similarity of the layout generated by the design tool and the existing terminal. However, if this feature is not activated, the design tool will determine the block configuration based on the terminal dimension and the design rules that are already defined in Section 3.2.2. Lastly, the tool will have the capability to generate a container terminal layout for each year, not only the container terminal layout at the end of the simulation year. This capability will be advantageous when the evolution of the container terminal would like to be assessed, which will define the period when the container terminal has reached its capacity in terms of its area/layout. In detail, the algorithm of the layout generation tool for each type of equipment can be seen in Appendix C.

3.2.6. Synchronization with the existing tool

As mentioned before, the main process in this research is generating the container terminal layout, based on the input of the terminal shape and dimension, the throughput demand, and the design rules, and then use the output from the generated container terminal layout to calculate the terminal element and estimate the terminal cost, as the previous tool also does. During the concept design, generally, four types of basic terminal elements are considered, each with their own distinct calculation method (Agerschou, et al., 2004). From the previous tool developed by (Koster, 2019), all four types included in the tool are : (a) number of berths and total quay length, (2) required number of equipment. (3) storage capacity, and (4) hinterland connectivity. By adding the terminal shape and dimension as the new boundary conditions, the terminal element calculation result will be different from the previously developed tool. The workflow of terminal element calculations that already include the terminal shape and dimension as the input is shown in Figure 26, and the in-depth workflow for the terminal element calculations can be shown in Figure 40.

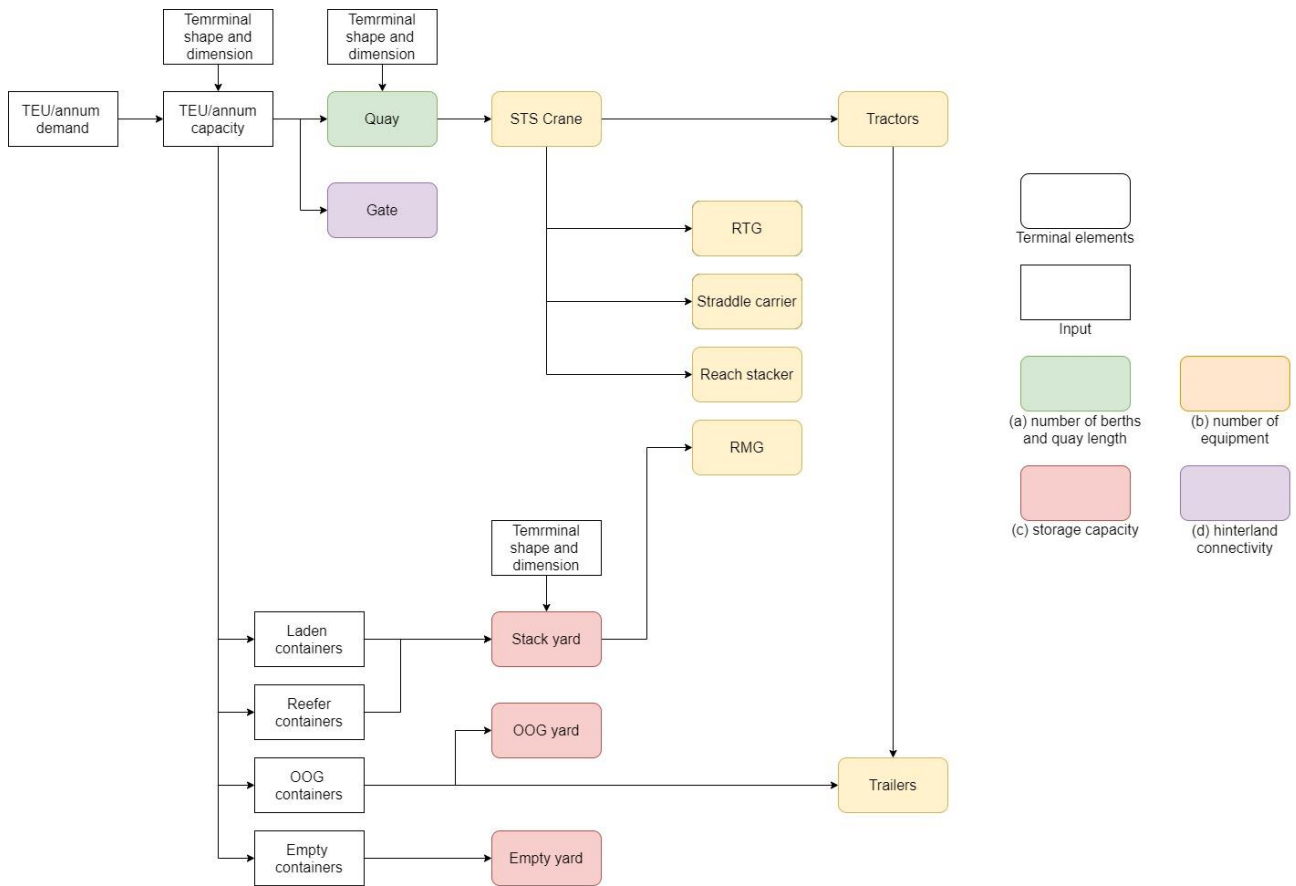


Figure 40 Terminal element calculations workflow (adapted from Koster, 2019)

After the terminal elements are calculated, the terminal cost estimation task can be done, where it will use the existing algorithm from the existing tool. Based on the existing tool, it can assess the concept terminal on several cost items, e.g., CAPEX, OPEX, and NPV, where these three outputs will allow the terminal planner to evaluate the various design options, based on their costs. The CAPEX will be estimated based on the calculated elements, the generated layout, and the unit costs. The OPEX estimation will include: (a) labor, (b) electrical power, (c) fuel, (d) maintenance. The process within the terminal cost estimation will not be elaborated further, since it is already discussed by (Koster, 2019). Further information regarding terminal element calculation and cost estimation can be seen in Appendix E.

4. Design tool validation

4.1. Terminal Design Validation

Before assessing the benefits of the tool, it is essential to validate that the design tool is working properly. Comparing the results of the design tool to an actual project can provide insights on the accuracy of the developed tool. Therefore, the terminal design from the design tool is validated using three case studies from several existing container terminal projects. These 3 case studies use RTG, RMG, and SC respectively as its stacking operation. It is chosen since RTG, RMG, and SC are three types of equipment that are mostly used as a container stacking equipment (Kemme, 2013). In comparing the terminal design from the actual projects and the design tool, two different scenarios are implemented. The first scenario uses an input for a 'default' container block configuration as the basis for the design tool in generating the container blocks. The terms 'default' here means the most used container block configuration based on the actual project. This scenario will show the capability of the design tool to re-generate a container terminal from the existing project. On the other hand, the second scenario does not use an input for a default container block configuration and let the design tool determine the most efficient default container block configuration, which should lead to a higher number of TGS and thus terminal capacity. This scenario will show the capability of the design tool to generate a container terminal design alternatives based on the container terminal shape and dimension input. The design tool is compared using several validation criteria, which are incorporated as follows:

- Terminal capacity (TEU/year)
- Storage capacity (TGS)
- Number of container blocks
- Container block configuration
- Stacking density
- Stacking orientation

The explanations for terminals that use RTG and SC as its stacking equipment can be seen in Section 4.1, and for terminals that use RMG as its stacking equipment can be seen in Appendix D.

(a) APM Terminal Rotterdam – Port of Rotterdam (SC Terminal)

To validate the output from the design tool for an SC system terminal, a case study of APM Terminal Rotterdam at Port of Rotterdam, the Netherlands (APMT-R) is done. In implementing the first scenario, the following block configuration is used as the default block configuration: width = 45 TEU, length = 20 TEU, height = 4 TEU. In detail, input for the design tool and selection of block configuration can be shown in Appendix D. APMT-R container terminal layout, and the generated container terminal layout from the design tool for scenario 1 and scenario 2 can be shown in Figure 41 and Figure 42.

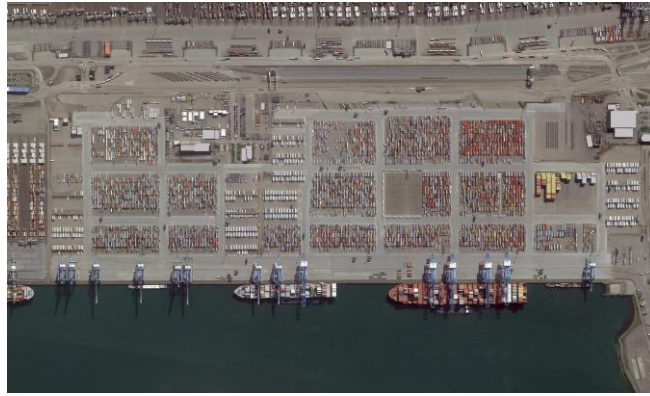


Figure 41 APM Terminal at Port of Rotterdam (Google Earth, 2018b)

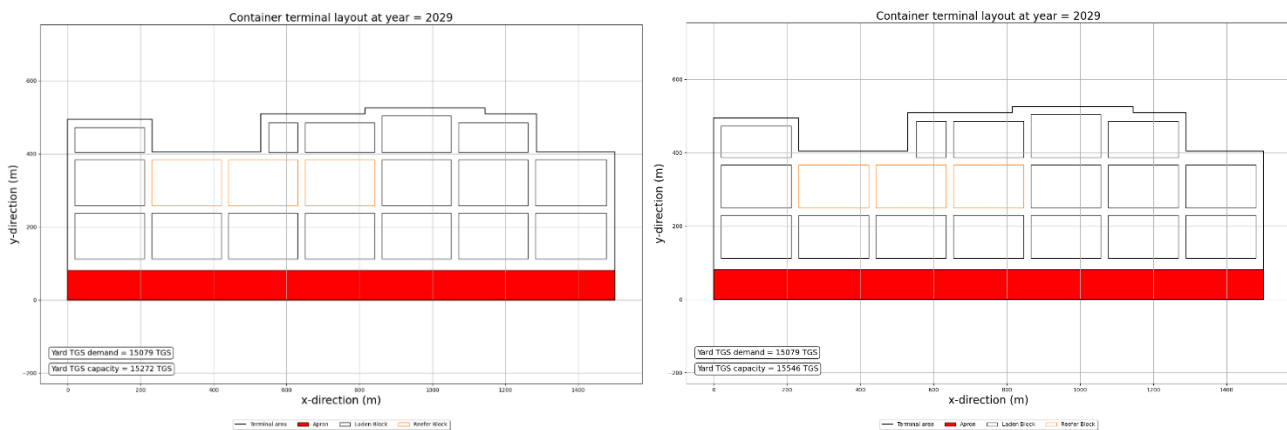


Figure 42 Results of the design tool for the APMT-R case study: scenario 1 (left) and scenario 2 (right) (Author's illustration)

Comparing qualitatively from Figure 41 and Figure 42, we can see that the design tool generates the same number of blocks compared to the actual project. However, for scenario 1, even though it uses the same 'default' container block configuration, the position of the container blocks is slightly different. The algorithm that is implemented on the design tool is based on an 'idealized' container terminal layout which can be seen in Figure 6, where it determines the container stacking area should be adjacent to the apron side and should be placed as near as possible to the quayside to have more efficient handling process. As a consequence, we can see from the output of the design tool that the default container block (length = 20 TEU) is located on the nearest side of the apron, and smaller container block due that is modified to area limitations (length = 14 TEU) is located on the furthest side of the apron. On the other hand, we can see from the actual project that the condition is 'vice-versa', where the smaller container block is located on the nearest side of the apron. For scenario 2, we can see from the output of the design tool that it has slightly smaller 'default' container block configuration (19 TEU). Comparing with the result for scenario 1, we can also see that the modified container block configuration for scenario 2 is slightly larger. This is caused by the condition where scenario 2 does not use a user's input for a 'default' container block configuration and let the design tool determine the most efficient one. Furthermore, a quantitative comparison for validating the terminal design from the actual project and the design tool for APMT-R can be seen in Table 6.

Table 6 Comparison of terminal design from the actual project and the design tool for APMT-R case study

Parameters	APMT-R	Scenario 1	Scenario 2	Unit	Source*
Capacity (TEU)	3.250.000	3.291.828	3.350.890	TEU	(APM Terminal, 2020)
TEU Ground Slots/TGS	15.079	15.272	15.546	TEU	
Number of Blocks	19	19	19	Blocks	(Google Earth, 2018b)
Default block length	20	20	19	TEU	(Google Earth, 2018b)
	126	126	117	m	
Default block width	45	45	45	TEU	(Google Earth, 2018b)
	190	190	190	m	
Default block height	4	4	4	TEU	(Google Earth, 2018b)
	10,36	10,36	10,36	m	
Stacking density	1000-1250	1.036	1.077	TEU/ha	(Saanen & van Valkengoed, 2005), (Brinkmann, 2011), (Kemme, 2013)
	250-300	258	269	TGS/ha	
Stacking orientation	Perpendicular to quay	Perpendicular to quay	Perpendicular to quay	ha	(Google Earth, 2018b)

* for the actual project

Based on Table 6, we can see that the terminal capacity resulted from the design tool is larger than the actual project. This means that the design tool can determine container terminal design that satisfies the terminal capacity from the actual project. We can also see that the terminal and TGS capacity from scenario 2 is larger than scenario 1, leading to a conclusion that the design tool can determine the most efficient default container block configuration. It also shows that the design tool can determine design alternatives for a container terminal based on its terminal shape and dimension input only (no block configuration input). Next, the number of blocks for scenario 1 resulted from the design tool is the same as the actual project, meaning that the design tool can produce a quite accurate result compared to what is happening on real life. Moreover, in container terminal design practice, stacking area and stacking density is considered as the most important indicator of land use which is one of the main deliverables on a concept design phase. For validation purposes, three benchmark references are used to verify the result from the design tool. For an RTG-system, the stacking density usually achieves 500-600 TEU/ha (Saanen & van Valkengoed, 2005), where this reference was not mentioning its stacking height. On the other reference, SC-system operations typically have densities around 750 TEU/ha for 3 stacking height (Brinkmann, 2011). Another reference stated that for SC-system operations leading to yard densities of up to 500-750 TEUs per hectare for 2-3 stacking height (Kemme, 2013). These references give a range of an SC-system stacking density of 250-300 TEU/ha for every stacking height, thus resulting 1000-1250 TEU/ha for 4 stacking height. These sources do not specify if the given stacking densities are including or excluding the infrastructure (roads, lanes, margin, etc.). However, if we compare it with the result from the design tool, it suggests that the given number are related to all stacking area, including the container terminal infrastructure (e.g. traffic lane, bypass lane, etc.). As can be seen from Table 6, the stacking density from the design tool is 1.077 TEU/ha, which is considered accurate because it is

well within the predetermined bandwidth from the literature. This concludes that the design tool can determine an accurate container terminal design for an SC-system.

(b) Container Terminal 2 - Port of Jebel Ali (RTG Terminal)

To validate the output from the design tool for an RTG-system terminal, a case study of Container Terminal 2 at Port of Jebel Ali, United Arab Emirates (CT2 Jebel Ali) is done. In implementing the first scenario, the following block configuration is used as the default block configuration: width = 10 TEU, length = 48 TEU, height = 5 TEU. In detail, input for the design tool and selection of block configuration can be shown in Appendix D. CT2 Jebel Ali container terminal layout and the generated container terminal layout from the design tool for scenario 1 and scenario 2 can be shown in Figure 43 and Figure 44.

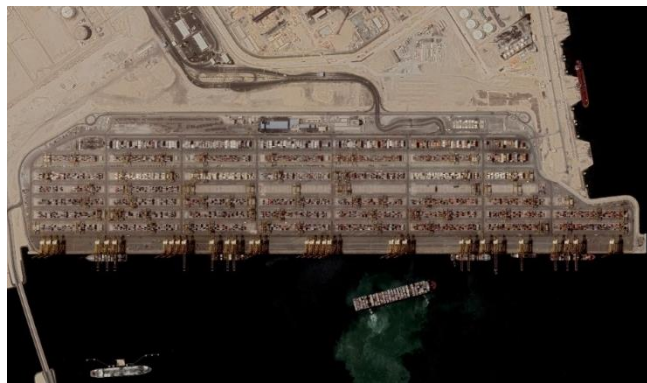


Figure 43 Container Terminal 2 at Port of Jebel Ali (Google Earth, 2019)

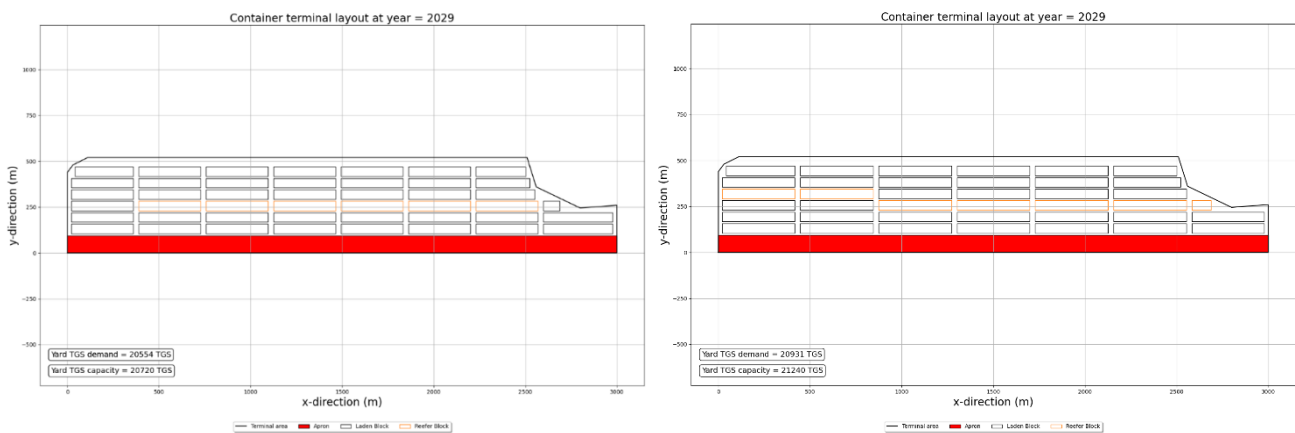


Figure 44 Results of the design tool for the CT2 Jebel Ali case study (Author's illustration)

Comparing qualitatively from Figure 41 and Figure 42, we can see that for scenario 1 the design tool generates the same block configuration and the number of blocks compared to the actual project. This shows that the design tool is capable to re-generate a container terminal from an existing project. However, scenario 2 generates different block configuration and number of blocks compared to the actual project. This is caused by the condition where scenario 2 does not use a user's input for a 'default' container block configuration and let the design tool determine the most efficient one. Furthermore, a quantitative comparison for validating the terminal design from the actual project and the design tool for APMT-R can be seen in Table 6.

Table 7 Comparison of terminal design from the actual project and the design tool for CT2 Jebel Ali case study

Parameters	CT2 Jebel Ali	Scenario 1	Scenario 2	Unit	Source*
Capacity (TEU)	5.500.000	5.677.262		TEU	(APM Terminal, 2020)
TEU Ground Slots/TGS	20.520	20.720	21.230	TEU	
Number of Blocks	45	45	39	Blocks	(Google Earth, 2018b)
Default block length	48	48	56	TEU	(Google Earth, 2018b)
	340	338	397	m	
Default block width	10	10	10	TEU	(Google Earth, 2018b)
	52	52	52	m	
Default block height	5	5	5	TEU	(Google Earth, 2018b)
	12,95	12,95	12,95	m	
Stacking density	900 - 1100	932	943	TEU/ha	(Saanen & van Valkengoed, 2005), (Brinkmann, 2011), (Kemme, 2013)
	180 - 220	186	188	TGS/ha	
Stacking orientation	Parallel to quay	Parallel to quay	Parallel to quay	ha	(Google Earth, 2018b)

* for the actual project

Based on Table 7, same as APMT-R case study, the terminal and TGS capacity from scenario 2 are larger than scenario 1, which gives another supporting evidence that the design tool can determine the most efficient default container block configuration. Scenario 2 uses a different default container block configuration, which caused a smaller number of blocks (39) required compared to scenario 1 and the actual project. From the area perspective, this should lead to a higher stacking density, since a smaller area is required for the infrastructure. In addition, from the operational perspective, a study is conducted in analysing the effect of block length on overall performance at a seaport container terminal (Petering & Murty, 2009). The study compares 12 different block length for 23 different container terminal layouts, and the number shows that 56 – 72 TEU block length resulting in the highest GCR (i.e. average quay crane work rate). It also stated that 60 TEU block length has a higher GCR than 40 TEU block length. From both area and operation perspective, the design tool can determine more efficient design alternatives than the actual project (and scenario 1). This is a very interesting result, while it is also proven that the design tool can determine design alternatives for a container terminal based on its terminal shape and dimension input only (no input for block configuration). Next, same as the previous case study, the stacking density is validated using three benchmark references, where for an RTG-system, the stacking density is 900-1.100 TEU/ha (Saanen & van Valkengoed, 2005), 1.000 TEU/ha for 4-5 stacking height (Brinkmann, 2011), and up to 1000 TEUs per hectare (Kemme, 2013). As can be seen from Table 7, the stacking density from the design tool is 1.036 TEU/ha and 258 TGS/ha, which is considered accurate because it is well within the predetermined bandwidth from the literature. This concludes that the design tool can determine an accurate container terminal design for an RTG-system.

4.2. Applicability Review of the Design Tool

4.2.1. Design alternatives using all equipment types

In Section 1.2, it is explained that one of the most influential choices in the early stages of the design cycle is the choice of the stacking equipment type. This applicability review demonstrates that the design tool is capable of considering all types of equipment throughout the design process and evaluating more design alternatives at the same time. It will use APMT-Rotterdam as a case study, where it will do the same steps as in the design tool validation for scenario 2, but now it is also done for all four types of equipment. The input values are kept the same, and only the stack equipment type was changed. This applicability review will show the differences of container terminal design alternatives between the four main types of stacking equipment (RTG, RMG, SC, and RS) by comparing its stacking layout, terminal (TEU and TGS) capacity, number of blocks, block configuration, stacking density, and stacking orientation. There will be no input for default block configuration and will let the design tool determine the most efficient one based on the terminal shape and dimensions input. The result of the applicability review can be shown in Figure 45, Figure 46, Figure 47, and Figure 48.

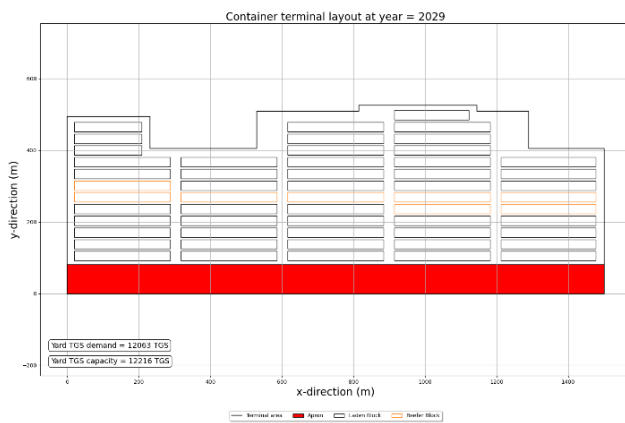


Figure 45 Result for RTG applicability review in APMT-R case study (Author's illustration)

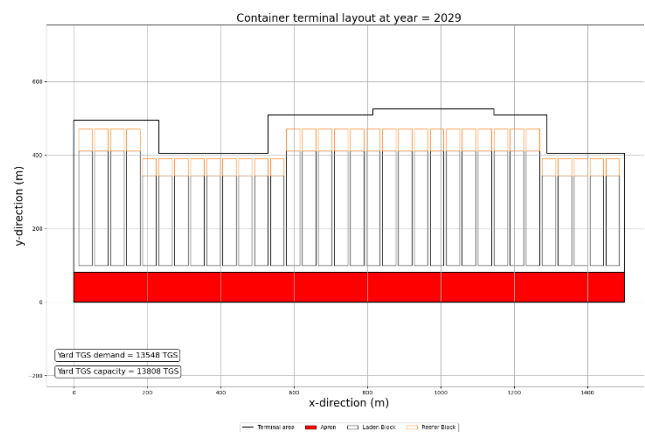


Figure 46 Result for RMG applicability review in APMT-R case study (Author's illustration)

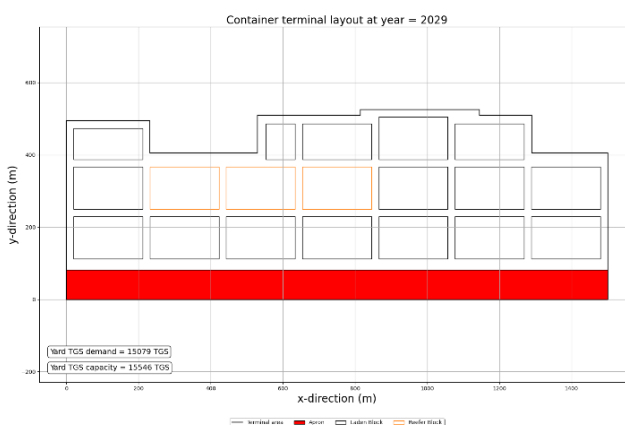


Figure 47 Result for SC applicability review in APMT-R case study (Author's illustration)

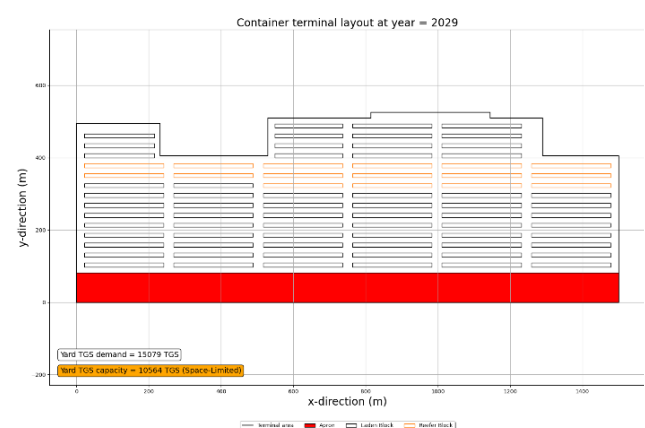


Figure 48 Result for RS applicability review in APMT-R case study (Author's illustration)

As we can see qualitatively, all four types of equipment are generating different container terminal layout. The block configuration and the number of container blocks are different for each type of equipment when it is applied to the same container terminal layout. Here, we can see the capability of the design tool to generate different container terminal design for a different type of stacking equipment based on its terminal shape and dimensions that have been set as the boundary conditions. Furthermore, we can see in detail the quantitative comparison of the terminal design for each type of equipment in Table 8.

Table 8 Comparison of the result from RTG, RMG, SC, and RS for applicability review in APMT-R case study

Parameters	Unit	RTG	RMG	SC	RS
Capacity (TEU)	TEU	3.291.401	3.720.339	3.350.892	2.277.035
TEU Ground Slots/TGS	TGS	12.216	13.808	15.546	10.564
Number of Blocks	Unit	55	34	19	81
Block length	TEU	30	43	45	20
	m	268	370	191	219
Block width	TEU	6	10	20	4
	m	26	37	117	11
Block height	TEU	5	5	4	4
	m	12,15	12,15	9,72	9,72
Stacking density	TEU/ha	1.035	1.097	1.053	716
	TGS/ha	207	219	263	179
Stacking orientation	-	Parallel to quay	Perpendicular to quay	Perpendicular to quay	Parallel to quay

Based on Table 8, we can see that RMG-system generates 10-15% larger terminal capacity compared to RTG and SC, even though SC-system is the one which has the highest TGS compared to other types of equipment. This is caused by the difference in the stacking height since RTG and RMG-system use 5 TEU as stacking height, and SC-system uses 4 TEU as stacking height. A large number of TGS for an SC-system can be explained by understanding the algorithm for an SC-system in the design tool, where it can maximize the land use of the container terminal by adapting not only its block length but also its block width. Therefore, for this specific terminal shape and dimensions, SC-system is the most preferred one in terms of terminal capacity. We can also see that RS-system produces the smallest terminal capacity and the largest number of blocks since it has a limited maximum block length and supposed to be implemented in relatively small-scale terminals. This is caused by the condition where RS cannot handle large stacks due to its limited reach and requires larger operating space. Moreover, the stacking density for three types of equipment is also considered accurate because it is well within the predetermined bandwidth from the literature ((Brinkmann, 2011), (Kempe, 2013), (Saanen & van Valkengoed, 2005), (Kalmar, 2011)), as discussed from the terminal design validation section. From the literature, we also learn that RMG-system generally has the largest stacking density compared to the other types of equipment. This condition is also shown within this applicability review. This concludes that the design tool can determine accurate container terminal design

alternatives for all types of equipment. This applicability review shows the capability of the design tool to produce different container terminal design alternatives for different types of stacking equipment. Based on the results and explanation, it concluded that the design tool produces reasonable output. This feature of the design tool will be very useful for container terminal stakeholders in choosing the most appropriate and efficient container stacking equipment type. However, while it is true that we can compare the design alternatives in terms of terminal capacity using the design tool, the design alternatives can be further compared with the terminal element calculation and cost estimation using the existing tool from the latest study by (Koster, 2019). This issue will be accommodated in the next applicability review regarding terminal element calculation and cost estimation.

4.2.2. Changing terminal throughput demand

In Section 2.1, it is mentioned that one of the most fundamental parameters in container terminal design is terminal throughput, as the container terminal design is basically aimed to fulfil the demand of the incoming throughput. This applicability review demonstrates that the design tool can be versatile in terms of changing terminal throughput demand. It will use APMT-Rotterdam as a case study, SC stacking system, same input values, and simulate in 10 simulation years (2020-2029) where at several simulation years the terminal throughput demand was changed. In this manner, the design tool's versatility to generate container terminal design in changing throughput can be observed. The result of this applicability review can be shown in Figure 49, Figure 50, Figure 51, and Figure 52.

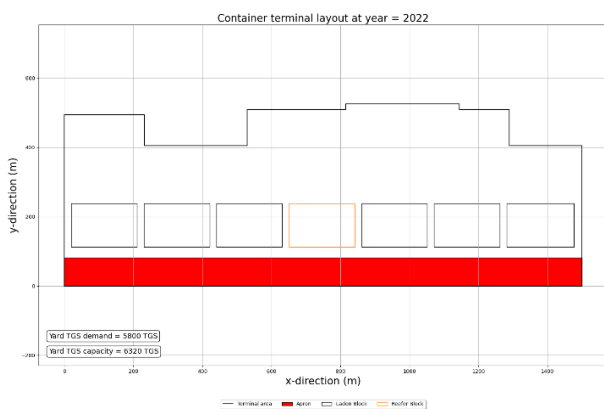


Figure 49 Result for year simulation: 2022 [1.250.000 TEU demand] (Author's illustration)

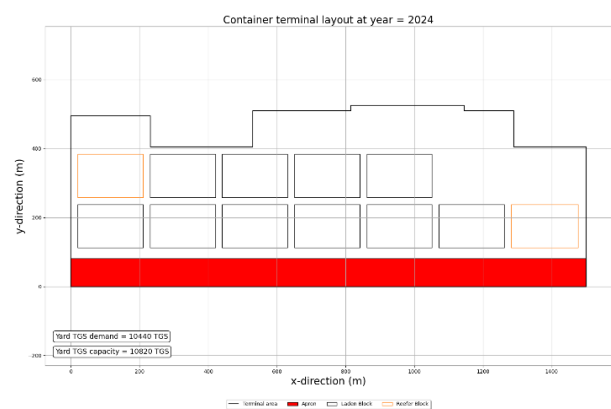


Figure 50 Result for year simulation: 2024 [2.250.000 TEU demand] (Author's illustration)

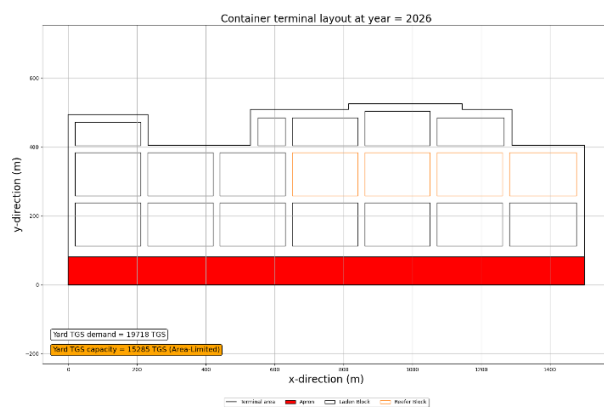


Figure 51 Result for year simulation: 2026 [4.250.000 TEU demand] (Author's illustration)

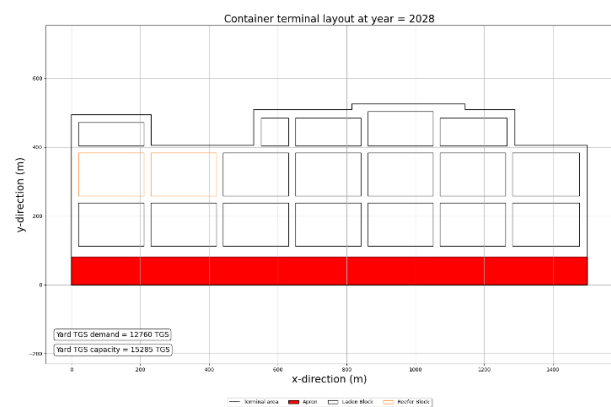


Figure 52 Result for year simulation: 2028 [2.750.000 TEU demand] (Author's illustration)

As we can see, the design tool shows its capability to generate different container terminal design to cope up with the changing of the terminal throughput demand. We can see the evolution of the container terminal design in terms of TEU ground slots, block configuration, and the number of blocks required to fulfil the throughput demand that changes per year. The throughput demand is set to change three times in 10 simulation years, which can be shown in details in Table 9. We can also see in Figure 51 (simulation year: 2026; throughput demand 4.250.000 TEU), the design tool shows an orange indicator for the TGS capacity with 'area-limited' annotation. This shows that the design tool can recognize the period when is the terminal has reached its area capacity because the throughput demand is greater than the throughput capacity, which is determined from the terminal shape and dimensions input. Therefore, after the design tool shows this sign, it means that an additional area or an upgrade in handling productivity is required to cope with the situation. However, we can also see that in the simulation year 2028 (throughput demand: 2.750.000 TEU) the throughput demand is reduced, and it resulted in the same container layout. This is because in the simulation year 2026, the container terminal layout is already designed to maximize the area used to fulfil the throughput demand. Thus, in the simulation year 2028, even though the throughput demand is reduced, the container terminal layout remains the same. Furthermore, we can see in detail the quantitative comparison of the result from each type of equipment, which can be shown in Table 9.

Table 9 Comparison of the result in changing throughput demand for applicability review in APMT-R case study

Parameters	Unit	Sim year 2022	Sim year 2024	Sim year 2026	Sim year 2028
Throughput demand	TEU	1.250.000	2.250.000	4.250.000	2.750.000
	TGS	5.800	10.440	19.718	12.760
Throughput capacity	TEU	1.362.255	2.332.216	3.294.631	3.294.631
	TGS	6.320	10.820	15.285	15.285
Number of Blocks	Unit	7	12	19	19
Land use	ha	23,2	39.9	59	59
Stacking density	TEU/ha	1.052	1.052	1.053	1.053
	TGS/ha	263	263	263	263

Based on Table 9, we can see that the number of blocks is increasing with the increasing of the throughput demand since more areas are required to store the containers. For example, to accommodate 2.250.000 TEU throughput demand in the simulation year 2024, it needs five additional blocks from the previous state of the terminal which has seven container blocks in the simulation year 2022 (throughput demand: 1.250.000 TEU). This shows the capability of the design tool to be versatile in determining the container terminal design. Along with the increasing number of blocks, the number of land use required is also increasing since the container blocks will need areas to store the containers. We can see that in the simulation year 2026, to accommodate 4.250.000 TEU throughput demand 59 ha area is required, where it is 9,1 ha larger than the previous state of the terminal in the simulation year 2024 (throughput demand 2.250.000 TEU). Moreover, we can see that even though it different throughput demand, the design tool produce an acceptable stacking density since it is well within the predetermined bandwidth from the literature ((Brinkmann, 2011), (Kemme, 2013), (Saanen & van Valkengoed, 2005), (Kalmar, 2011)) as discussed

in the terminal design validation section. This concludes that the design tool can determine an accurate container terminal design for every throughput demand. This applicability review shows the capability of the developed design tool to produce different container terminal design for a different container terminal throughput demand that is set on each simulation year in terms of its layout, capacity, number of blocks, and land use. This feature of the design tool will be very beneficial for container terminal stakeholders in answering how should they do to cope with the uncertainties in the future by using the versatility of the design tool in determining the container terminal design. It will deliver one of the main deliverables on a concept design phase, which is land use, and how much the land use is needed for an increasing/decreasing container throughput demand.

4.2.3. Terminal element calculation and cost estimation

This applicability review demonstrates that the design tool can be synchronized with the existing tool to do the terminal element calculation and cost estimation tasks for further design alternatives comparison. From the previous section (Section 4.2.1), it can be understood that the container terminal design using RS-system will always generate the least terminal capacity due to its requirement for larger operational space. However, it might be interesting if we could also compare it from another perspective, which is from cost-related perspective. RS-system is basically meant to be applied on a small-scale terminal, so it might be the cheaper alternatives if smaller throughput demand is implemented compared to other types of equipment. Same as the previous applicability review, this applicability review will use APMT-Rotterdam as a case study. It will use the same input values; however, smaller throughput demand is used: 1.250.000 TEU/year. This is done to have an apple-to-apple cost-related comparison for all types of equipment where all design alternatives can fulfil the throughput demand. The terminal element calculation and terminal cost estimation are done using the existing tool that is already synchronized to consider the terminal layout generated from the developed design tool. We can see in detail the quantitative comparison of the terminal elements and costs for each type of equipment in Table 8.

Table 10 Comparison of the result in element calculation and cost estimation for 1.250.000 TEU/year throughput demand

Parameters	Unit	RTG	RMG	SC	RS
Throughput demand	TEU/year	1.250.000	1.250.000	1.250.000	1.250.000
	TGS	4.641	4.641	5.800	5.800
Throughput capacity	TEU/year	1.283.577	1.291.126	1.318.716	1.251.894
	TGS	4.764	4.792	6.118	5.808
Stacking height	TEU	5	5	4	4
STS Crane	unit	5	5	5	5
Horizontal transport	unit	25	25	25	25
Stacking equipment	unit	20	19	25	20
Land use	ha	22,99	22.75	22.76	32.43
Stacking density	TEU/ha	1.035	1.052	1.075	716
	TGS/ha	207	210	268	179

Parameters	Unit	RTG	RMG	SC	RS
Total CAPEX	m USD	217,08	219,97	238,22	198,73
Total OPEX	m USD	179,68	164,06	178,48	183,35
Total Cost	m USD	396,75	384,03	416,70	382,09

Based on Table 10, we can see that the number of STS crane produced by the design tool for all types of equipment is the same. This is because the quay wall length that is calculated from the developed tool (which is determined from the terminal shape and dimensions input) is acting as an input for the existing tool. So, for the same number of throughput demand and quay wall length, it will produce the same number of STS crane required for each design alternative. Next, we can see that RS-system produces the largest land use, which is relevant to the condition where RS-system requires a larger operating space compared to the other equipment types. However, even though RS-system needs a larger area for the container handling operation, it produces the least cost estimation from all other design alternatives. Therefore, the premise that said an RS-system is more efficient for small terminals is true, as it can handle a limited number of containers with limited costs. Even though RS-system has relatively higher labour costs, it has the lowest maintenance and investment costs as the unit price for the RS system is very low.

However, it might be interesting to see how is the effect of the terminal throughput to the land use and terminal cost. Therefore, further simulation using the same input is done for a greater throughput demand: 2.250.000 TEU/year. This is also done by also considering that terminal capacity for all types of equipment can fulfil the throughput demand. We can see in detail the quantitative comparison of the terminal elements and costs for each type of equipment and the difference compared to the previous throughput demand in Table 11.

Table 11 Comparison of the result in element calculation and cost estimation for 2.250.000 TEU/year throughput demand

Parameters	Unit	RTG	RMG	SC	RS
Throughput demand	TEU/year	2.250.000 (+ 80%)	2.250.000 (+ 80%)	2.250.000 (+ 80%)	2.250.000 (+ 80%)
	TGS	8.352	8.352	10.440	10.440
Throughput capacity	TEU/year	2.261.621	2.292.339	2.260.656	2.277.037
	TGS	8.394	8.508	10.488	10.564
Land use	ha	40.51 (+ 76%)	38.51 (+ 69%)	39.86 (+ 75%)	59 (+82 %)
Stacking density	TEU/ha	1.036	1.104	1.053	716
	TGS/ha	207	220	263	179
Total CAPEX	m USD	368,42	371,73	406,64	335,43
Total OPEX	m USD	319,91	289,29	317,70	326,65
Total Cost	m USD	688,33 (+ 73%)	661,02 (+ 72%)	724,34 (+ 73%)	662,082 (+ 73%)

Based on Table 11, an interesting output can be seen. For 2.250.000 TEU/year throughput demand, RMG-system has the lowest cost estimation, not RS-system. This happens even though we also see that there is only a slight difference in the percentage of increase in the total cost estimation for all types of equipment (around 72-73%). This can be explained since all type of equipment has its advantages and disadvantages related to the terminal cost. For example, RS has low maintenance and capital cost, but it has a high labour cost. Further details on the qualitative comparison between stacking equipment can be seen in Appendix B. However, even though the percentage of increase in the total cost estimation for all types of equipment is similar, the percentage of increase in land use is more varying. We can see that RS-system has the highest percentage of increase in land use for its stacking area (82%) compared to other equipment types. Again, it is caused by the necessity for a larger operating space when using RS-system. Therefore, it is proven that RS-system is more efficient for small terminals in terms of cost and land use. We can also see from Table 10 and Table 11 that the design tool can produce accurate container terminal design alternatives for any given throughput. This can be seen from the stacking density that is resulted from the design tool, where it is well within the predetermined bandwidth from literature ((Brinkmann, 2011)), (Kempe, 2013), (Saanen & van Valkengoed, 2005), (Kalmar, 2011)) as discussed in the terminal design validation section.

This applicability review shows the capability of the developed design tool to be further compared with the terminal element calculation and cost estimation using the existing tool from the latest study by (Koster, 2019). Same as the first applicability review, this will be very useful for container terminal stakeholders in choosing the most appropriate and efficient container stacking equipment type. However, as can be seen from the previous two applicability review, the most appropriate and efficient container stacking equipment type depends on the design objective of the terminal. For example, if the terminal has low throughput demand and looking for the lowest cost alternatives, RS-system might be the answer. However, if the terminal has a high throughput demand and planning to expand in the future, RS-system might not be a good option. Thus, high-density stacking equipment such as RTG, RMG, and SC would be preferred for this case. Also, RS-system might not be a good solution for the case that has a high land price since RS-system requires larger land use to do the stacking operation. An analysis in determining the influence of local cost conditions (cost of land, labour, fuel, and electricity) on a concept design phase is also done by (Koster, 2019). It is concluded that the cost of land has considerable impact, where a reclamation project using an RS-system is approximately 21% more expensive than the original case, where RTG-system is only cost 13% more. Moreover, RS-system is not the preferred alternatives for the case that has high labour costs, as RS-system requires a higher number of labours compared to the other types of equipment. From the study by (Koster, 2019), the cost of labour is concluded as the most influential cost parameter, and RS-system is the most affected type of equipment compared to the other types of equipment. Thus, for this case also, choosing low-labours-personnel stacking equipment like RMG or SC would be preferred. For the cost of fuel and electrical power, it is concluded that it only has minimum influence and not expected to be decisive regarding the stacking equipment choice. Further information regarding terminal element calculation and cost estimation from the existing design tool can be seen in Appendix E.

5. Discussion, Conclusion and Recommendation

5.1. Discussion

This thesis aims at improving the quality of the design process of a concept level container terminal design. This is done by developing an automated tool that can accelerate the work in providing multiple design alternatives while also considering area limitations. However, this thesis comes with challenges and possible downsides. Several aspects are discussed here to be analysed from the proper perspective.

The automated tool is developed to become a solution for the complex container terminal design process on a concept design phase while also taking into account area limitations. Therefore, it can determine the container terminal design for a non-rectangular terminal. However, it can be argued that the container handling on a non-rectangular terminal might not be as ideal as a rectangular terminal. The container terminal handling on a non-rectangular terminal might lead to inefficiency due to the 'non-ideal' terminal infrastructure (e.g. traffic lane, crossing lane, etc.). Therefore, this condition might affect the number of elements and cost of the container terminal. However, in this research, the inefficiency due to a non-rectangular terminal is not yet discussed as no literature considers it in designing a container terminal on a concept design phase. This research only gives a disclaimer to further discuss the effect of the inefficiency to the number of elements and cost of the terminal, which might lead to an increase of 5 – 10% from a rectangular terminal. It is also difficult to determine in which part of the design tool that should incorporate this inefficiency, as the design tool is only based on the design rules from the literature that discuss a container terminal design on a concept design phase. Therefore, it can be argued that the design tool might give only an almost ideal result regarding the number of elements and cost of the terminal due to this issue.

Another discussion is concerning the design rules that are implemented within the design tool. The design rules is one of the most important parts of the design tool since it is a rule-based design. The design tool determines the container terminal design based on a set of rules that are implemented on the tool. In determining the design rules for each type of equipment, it is based on several container terminal design guidelines and literature such as (PIANC, 2014a), (Ligteringen, 2017), and (Thoresen, 2014). However, some parts of the design rules are based on the observation from several existing container terminals using Google Earth. This is because there is very limited research regarding several specific elements of the design rules (e.g. traffic lane, crossing lane, etc.) and those elements might be varying for all container terminals in the world. Also, it is understood that it is very hard to create a 'general' design tool that can be versatile enough to design a container terminal since each container terminal stakeholder will have different preferences and challenges. Therefore, the output from the design tool might not be the most representative one, but it can provide design alternatives for the container terminal stakeholder.

5.2. Conclusion

As stated in the introduction part of this research, to answer the research question, five sub-questions are determined, and these sections will address the answer to the research question and sub-questions:

“How can the container terminal be designed in the concept design phase while taking into account area limitation?”

Designing a container terminal involves various parameters, variables, and stakeholders, making it a very complex process. On a concept design phase, a quick assessment for container terminal design alternatives should be done in a limited time, cost, and effort. Thus, automation in the design process would be significant to ease and accelerate the work of the engineers. The latest study has been done by (Koster, 2019) to solve the problem by developing an automated tool (i.e. design tool) by modelling automatable tasks in determining the container terminal design. However, further development is required for the automated tool, especially for the layout generation tool, to also consider area limitations. Therefore, this thesis is done to solve the problem by developing a design tool for container terminal design which consider not only terminal throughput and stacking equipment choice as the main input but also terminal shape and dimensions. In a nutshell, a design tool is defined as a set of system that was implemented with some algorithms to transform an input into the desired output. The algorithms will correspond for at least four tasks: recognizing the terminal shape and dimension as the boundary condition, modifying container block configuration (either by enlarging/reducing its size, generating container blocks, and determining the desired output. After the design tool is developed, it is essential to validate the design tool to check whether it is working properly or not. The design tool is validated using case studies from existing terminals by comparing several validation criteria: terminal layout, terminal capacity, the number of blocks, blocks configuration, stacking density and stacking orientation. Other than that, the design tool is also reviewed by implementing three types of applicability: design alternatives for all types of equipment, element calculation & cost estimation from all design alternatives, and changing throughput demand. It is concluded that the design tool is capable of providing container terminal design alternatives and also the comparison of the layout, elements, land use, cost, and capacity for four main types of stacking equipment: RTG, RMG, SC, and RS. This would be very beneficial for container terminal stakeholders in determining which one is the most effective and efficient stacking equipment (which depends on the main objective of the terminal; e.g., lower land use, higher throughput, least labour cost, more sustainable, etc.). For example, RS-system is proven to be efficient for low throughput demand container terminal when considering cost as the main objective. However, RS-system might not be a good option for the case that has a high land price, since RS-system requires larger land use to do stacking operation. Thus, choosing high-density stacking equipment like RTG, RMG, or SC would be preferred for this case. Also, RS-system might not be a good solution for the case that has high labour costs, as RS-system requires a higher number of labours compared to other types of equipment. Other than that, this thesis will also show you the design tool's capability of generating different container terminal design for varying throughput demand and giving information whether the container terminal has reached its area capacity or not. It is concluded that the design tool is capable of being versatile enough in determining the container terminal design. This would be very beneficial for container terminal stakeholders in answering how they should do to cope with the uncertainties in the future in determining the container terminal design. Furthermore, from this applicability

review, we can also see the effect of increasing terminal throughput to terminal land use and terminal cost. Here, we learn that RS-system has the highest percentage of increase in land use compared to the other types of equipment, even though the percentage of increase in total cost estimation for all types of equipment is similar. All in all, area limitations have a significant influence in the container terminal design process since it is closely related to container block configuration, which also related to the terminal capacity, land use, and costs. Area limitations will also make the design process even more complicated; and thus, the design tool that has been developed in this thesis for generating container terminal design alternatives is one of the answers in solving the problem.

1. What does a typical container terminal design consist of?

The container terminal serves as the main intermediary between the seaside and the landside operations. In general, the terminal area is usually divided into three main areas, which are apron, primary yard, and secondary yard. In the apron area, ship-to-shore operation (loading and unloading of containers from/to vessels) is carried out. The quay is located in this area, which is defined as the interface between the container vessel and the other terminal elements. A typical apron arrangement consists of a quay with rooms for bollards and fenders, the sea- and landside rail (or the operating space) for the (un)loading equipment, and the back-reach area of the equipment. Adjacent to the apron area, lies the primary yard, or also called the stacking/storage area. It is used for storing inbound containers after they have been unloaded from a vessel until they are collected for onward transport to the hinterland, and also for outbound containers to be loaded into a vessel after they have been delivered from the hinterland (or another vessel in case of transshipment). It is used for storing laden and reefer stacks, while empty stacks and oog stacks are usually stored at the secondary yard. Containers are stored in the form of container block, which can be defined as a group of containers stacked in a number of bays, rows, and tiers. A bay of a block is one column (container length) of the block, a row of a block is one row (container width) of the block, and a tier of a block is one high (container height) of the block. Adjacent to the primary yard, lies the secondary yard, which is the area that also acts as a connector between the hinterland and the container terminal. This area is used for storing empty containers, out of gauge containers, repair yard, container freight station, office building, maintenance workshop, and gate or rail/barge terminal.

2. What are the governing parameters and boundary conditions in the process of container terminal design?

Two of the most fundamental parameters of a concept design phase is the terminal throughput and stacking equipment. The terminal throughput will be the primary input in determining the container terminal design, as the container terminal design is basically aimed to fulfil the demand of the incoming throughput. On the other hand, the type of stacking equipment has a great influence on the container terminal design since it has a significant influence on the two primary deliverables of the concept design phase: the land use and the cost estimate. The influence of the stacking equipment choice to the container terminal design is shown by the implementation of the design rules that differ for each type of equipment. It represents the area required for the stacking equipment to be working properly in handling the containers. There are four main types of stacking equipment that is considered in this research, which are Rubber-Tired Gantry Crane (RTG), Rail-Mounted Gantry Crane, Straddle Carrier (SC), and Reach Stacker (RS). However, other than terminal throughput and stacking equipment, this research will also

include terminal shape and dimension as one of the fundamental parameters since it also has a significant impact on the container terminal design. It will determine how the container terminal elements should be located and configured and influence the container terminal throughput that can be handled (i.e. container terminal capacity) since it is not only determined by the container handling capacity at the ship-to-shore operations but also at the container handling capacity at the primary yard/stacking area.

3. How to develop an automated tool in generating container terminal design based on the governing parameters and boundary condition?

In this thesis, the design tool in determining container terminal design is developed using Python. This is based on the fact that Python is a programming language that is simple enough for anyone to get started, yet powerful enough to handle complex and extensive design projects. Python is also a very suitable tool for automating an engineering design process and is thus commonly used in the field. Moreover, by using Python, it will keep all automated tasks in one environment since the terminal element calculation and cost estimation tasks from the existing tool are also developed in Python. The design tool generates the container terminal design by means of placing the “physical” elements into a blank canvas based on the governing parameters and boundary conditions, which are terminal throughput, type of stacking equipment, and terminal configuration. Algorithms of container block generation were developed and implemented within the design tool. These algorithms will correspond for at least four tasks: recognizing the terminal shape and dimension as the boundary condition, modifying container block configuration (either by enlarging/reducing its size, generating container blocks, and determining the desired output. The process in developing the algorithm, especially the algorithm in modifying block configuration, is done by using the principle that the design tool should accommodate all container terminal shape and dimension possibility so that the design tool will be as robust as possible. This can be done by implementing seven main shapes as the container terminal shape and dimension pilot projects. Therefore, the design tool could determine the container terminal design for any possible container terminal shape and dimensions.

4. How to determine container terminal design using the automated tool?

By using the input of the design tool and the algorithm that is implemented to the design tool, the design tool will come up with several outputs. First, the container terminal layout figure is generated for each simulated year, showing the container terminal shape and dimensions, apron, primary yard, secondary yard, and the generated container blocks. This figure is governed by the terminal shape and dimensions, types of stacking equipment, and terminal throughput demand that is defined per each simulated year. So, the design tool could show the container terminal layout evolution in time for changing container throughput demand. Next, annual terminal capacity and TEU ground slot (TGS) capacity is also presented by the design tool, where it will show whether the container terminal annual capacity is fulfilling the container terminal throughput demand, or it has reached its area capacity. Furthermore, the number of blocks and blocks configuration is also provided as one of the outputs from the design tool. Lastly, the stacking density is shown by the design tool to give an overview of how much container terminal can be stored per area units.

5. How does the container terminal design generated by the automated tool compare with the actual project?

The container terminal design resulted from the design tool is validated using case studies from several existing terminals by quantitatively comparing several validation criteria: terminal layout, terminal capacity, number of blocks, blocks configuration, stacking density and stacking orientation. These 3 case studies use RTG, RMG, and SC respectively as its stacking operation. It is chosen since RTG, RMG, and SC are three types of equipment that are mostly used as a container stacking equipment. The terminal design validation concluded that the design tool is not only capable to re-generate container terminal from the existing project, but also generate accurate container terminal design alternatives based on the terminal shape and dimensions input. Other than that, the design tool is also reviewed for three types of applicability: design alternatives for all types of equipment, element calculation & cost estimation from all design alternatives, and changing throughput demand. These applicability reviews use APM Terminal Rotterdam (APMT-R) as the case study. From these applicability reviews, it is concluded that the design tool is capable of providing container terminal design alternatives and also the comparison of the layout, elements, land use, cost, and capacity for four main types of stacking equipment: RTG, RMG, SC, and RS. This would be very beneficial for container terminal stakeholders in determining which one is the most effective and efficient stacking equipment (which depends on the main objective of the terminal; e.g., lower land use, higher throughput, least labour cost, more sustainable, etc.). For example, RS-system is proven to be efficient for low throughput demand container terminal when considering cost as the main objective. However, RS-system might not be a good option for the case that has a high land price, since RS-system requires larger land use to do stacking operation. Thus, choosing high-density stacking equipment would be preferred for this case. Also, RS-system might not be a good solution for the case that has high labour costs, as RS-system requires a higher number of labours compared to other types of equipment. Other than that, from these applicability reviews, it is also concluded that the design tool is capable of generating different container terminal design for varying throughput demand and providing information whether the container terminal has reached its area capacity or not. This would be very beneficial for container terminal stakeholders in answering how they should do to cope with the uncertainties in the future in determining the container terminal design.

5.3. Recommendation

Based on the conclusion, the developed design tool shows its capability to generate container terminal design alternatives while taking into account area limitation. Nevertheless, during the progress of this thesis, several valuable opportunities for further research and development for the design tool have been identified as follows:

- First, even though the design tool already considers all terminal areas (apron, primary yard, and secondary yard) in generating the terminal configuration, it only generates terminal elements on the primary yard and not on the secondary yard. If the generation of terminal elements on the secondary yard is included, the output of the design will be more realistic. However, the challenge for this potential tool development is that it is hard to distinguish the separation between the primary yard and the secondary yard. Moreover, if we see from all container terminals around the world, secondary yard terminal elements can be different with a high level of variation, starting from the element location, sizes, types, etc., since it is very closely related to the terminal stakeholder's preferences.
- Second, the design tool can be further developed by including more types of equipment (for the ship to shore operation, stacking operation, and horizontal transport operation) as an option in generating the container terminal design. It is not only about the type of the equipment, but also detail specification of the equipment such as level of automation, types of fuel, types of electrical power, etc. Using different equipment might lead to different design rules. For example, it can be projected that the design rules for an automated guided vehicle (AGV) is different than a tractor-trailer vehicle (TT), and different design rules will affect the resulted container terminal design.
- Third, details on stacking orientation can also be further developed for the design tool. In this research, there is only one stacking orientation that is considered, which is either parallel or perpendicular to quay stacking orientation. It might be interesting to see and compare the container terminal design that is, for example, stacked at 30° or 45° (or other numbers) orientation to the quay. Nevertheless, the challenge for this further tool development is that we should incorporate container terminal operation point of view since stacking orientation obviously will affect container handling process.
- Lastly, now, the developed design tool can provide the evolution of the container terminal design based on a different throughput demand for each simulation year. However, the developed design tool can be further developed to include options of an expansion of the container terminal. Now, the design tool only determines the container terminal design based on one fixed container terminal shape and dimensions. For that reason, we could not see the evolution of a container terminal that will be expanded in the future. The design tool will have to run a new simulation to determine the expanded container terminal design. Therefore, it might be interesting to further develop the design tool to be able to show the evolution of the container terminal design, not only by changing throughput demand but also by changing terminal shape and dimensions.

Bibliography

- Agerschou, H., Dand, I., Ernst, T., Ghoois, H., Jensen, O., Korsgaard, J., . . . Svendsen, H. (2004). *Planning and Design of Ports and Marine Terminals*. London: Thomas Telford Publishing.
- Alcalde, E. M., Kim, K. H., & Marchan, S. S. (2015). Optimal Space for Storage Yard Considering Yard Inventory Forecasts and Terminal Performance. *Transportation Research Part E*, 101-128.
- Alessandri, A. (2008). Modelling and Feedback Control for Resource Allocation and Performance Analysis in Container Terminals. *IEEE Transactions on Intelligent Transportation Systems Vol. 4 No. 4*, 601-613.
- Allianz Global Corporate & Specialty. (2018). *Safety and Shipping Review 2018*. Munich: Allianz Global Corporate & Specialty SE.
- APM Terminal. (2020, August 10). *Our Terminal*. Retrieved from APM Terminal: <https://www.apmterminals.com/en/rotterdam/about/our-terminal>
- Bose, W. J. (2011). *Handbook of Terminal Planning*. Berlin: Springer Science+Business Media.
- Brinkmann, B. (2011). Operations Systems of Container Terminals: A Compendious Overview. In J. W. Bose, R. Sharda, & S. Voß (Eds.), *Handbook of terminal planning, Volume 49 of Operations research/computer science interfaces series*, 25-39.
- Briskorn, D., & Hartmann, S. (2005). Simulating Dispatching Strategies for Automated Container Terminal. *Operations Research Proceedings*, 97-102.
- Caserta, M., Schwarze, S., & Voß, S. (2011). Container Rehandling at Maritime Container Terminals. In J. W. Bose, R. Sharda, & S. Voß (Eds.), *Handbook of terminal planning, Volume 49 of Operations research/computer science interfaces series*, 247-269.
- Choi, H. R., Park, B. J., Kwon, H. K., & Yoo, D. H. (2006). A Comparison of Layout of Reefer Containers in Automated Container Terminal. *Proceedings of the 6th WSEAS International Conference on Simulation, Modelling and Optimization, Lisbon*, 22-24.
- Chu, C.-Y., & Huang, W.-C. (2005). Determining Container Terminal Capacity on the Basis of an Adopted Yard Handling System. *Transport Reviews: A Transnational Transdisciplinary Journal* 25:2, 181-199.
- Copeland, T. E., Weston, J. F., & Shastri, K. (2005). *Financial Theory and Corporate Policy 4th Edition*. Amsterdam: Addison-Wesley Longman.
- Dekker, R., Voogd, P., & van Asperen, E. (2006). Advanced methods for container stacking. *OR Spectrum* 28, 563-586.
- Duinkerken, M. B., Evers, J. J., & Ottjes, J. A. (2001). A Simulation Model for Integrating Quay Transport and Stacking Policies on Automated Container Terminals. *Proceedings of the 15th European Simulation Multiconference*.
- Google Earth. (2017, June 30). *Port of Singapore 1°17'19.00"N, 103°46'02.53"E, Elevation 8m*. Retrieved from Google Earth: <http://www.earth.google.com> [viewed November 26, 2019]
- Google Earth. (2018a, March 19). *Port of Hamburg 53°30'13.48"N, 9°56'02.97"E, Elevation 8m*. Retrieved from Google Earth: www.earth.google.com [viewed 29 November 2019]

- Google Earth. (2018b, August 05). *Port of Rotterdam 51°57'19.89"N, 4°02'41.18"E, Elevation 5m*. Retrieved from Google Earth: www.earth.google.com [viewed 26 November 2019]
- Google Earth. (2018c, May 30). *Port of Aarhus 56°08'50.15"N, 10°14'07.24"E, Elevation 1 m*. Retrieved from Google Earth: www.earth.google.com [viewed 26 November 2019]
- Google Earth. (2018d, August 5). *Port of Rotterdam 51°57'10.37"N, 4°00'22.18"E, Elevation 5m*. Retrieved from Google Earth: www.earth.google.com [viewed 14 July 2020]
- Google Earth. (2018e, August 05). *Port of Rotterdam 51°57'17.42"N, 3°59'20.91"E, Elevation 0 m*. Retrieved from Google Earth: www.earth.google.com [viewed 16 August 2020]
- Google Earth. (2019, April 07). *Port of Jebel Ali 25°01'18.34"N, 55°03'34.91"E, Elevation 0 m*. Retrieved from Google Earth: www.earth.google.com [viewed 16 August 2020]
- Günther, H.-O., & Kim, K.-H. (2006). Container terminals and terminal operations. *OR Spectrum* 28, 437-445.
- Ijzermans, W. (2019). *Terminal Design Optimization*. Delft: Delft University of Technology.
- Jiang, X., Chew, E. P., Lee, L. H., & Tan, K. C. (2013). Flexible Space-Sharing Strategy for Storage Yard Management in a Transshipment Hub Port. *OR Spectrum* 35, 417-439.
- Kalmar. (2011). *Kalmar Container Handling Systems - Complete Range of Products and*. Tampere: Tampereen Offsetpalvelu Oy. Retrieved from Kalmar.
- Kemme, N. (2013). *Design and Operation of Automated Container Storage Systems*. Berlin: Springer-Verlag Berlin Heidelberg.
- Kim, K. H., & Kim, H. B. (1998). The Optimal Determination of the Space Requirement and the Number of Transfer Cranes for Import Containers. *Computers Industrial Engineering Vol. 35, Nos 3-4*, 427-430.
- Kim, K. H., Park, Y.-M., & Jin, M.-J. (2008). An Optimal Layout of Container Yards. *OR Spectrum* 30, 675-695.
- Kone Cranes. (2020a). *Ship to Shore Gantry Cranes | Kone Cranes*. Retrieved from Kone Cranes: <https://www.konecranes.com/equipment/container-handling-equipment/ship-to-shore-gantry-cranes> [accessed 22 February 2020]
- Kone Cranes. (2020b). *Mobile Harbour Cranes | Kone Cranes*. Retrieved from Kone Cranes: <https://www.konecranes.com/equipment/mobile-harbor-cranes> [accessed 22 February 2020]
- Kone Cranes. (2020c). *Rubber Tired Gantry Crane | Kone Cranes*. Retrieved from Kone Cranes: <https://www.konecranes.com/equipment/container-handling-equipment/rubber-tired-gantry-cranes> [accessed 22 February 2020]
- Kone Cranes. (2020d). *Rail Mounted Gantry Cranes | Kone Cranes*. Retrieved from Kone Cranes: <https://www.konecranes.com/equipment/container-handling-equipment/rail-mounted-gantry-cranes> [accessed 22 February 2020]
- Kone Cranes. (2020e). *Straddle Carrier | Kone Cranes*. Retrieved from Kone Cranes: <https://www.konecranes.com/equipment/container-handling-equipment/straddle-carriers> [accessed 22 February 2020]
- Kone Cranes. (2020f). *Reach Stackers | Kone Cranes*. Retrieved from Kone Cranes: <https://www.konecranes.com/equipment/lift-trucks/reach-stackers> [accessed 22 February 2020]
- Koster, P. H. (2019). *Concept Level Container Terminal*. Delft: Delft University of Technology.

- Kox, S. (2017). *A Tool for Estimating Marine Terminal Dimensions and Costs in a Project's Feasibility Phase*. Delft: Delft University of Technology.
- Lee, B. K., & Kim, K. H. (2013). Optimizing the Yard Layout in Container Terminals. *OR Spectrum* 35, 363-398.
- Lee, B. K., Lee, L. H., & Chew, E. P. (2018). Analysis on High Throughput Layout of Container Yards. *International Journal of Production Research*, 1-20.
- Ligteringen, H. (2017). *Ports and Terminals*. Delft: Delft Academic Press.
- Liu, C.-I., Jula, H., & Iannou, P. A. (2002). Design, Simulation, and Evaluation of Automated Container Terminals. *IEEE Transactions on Intelligent Transportation Systems Vol. 3 No. 1*, 12-26.
- Michele, A., & Serra, P. (2014). Strategic Determinants of Terminal Operating System Choice: An Empirical Approach using Multinomial Analysis. *Transportation Research Procedia* 3, 592-601.
- Mohseni, N. S. (2011). *Developing a Tool for Designing a Container Terminal Yard*. Delft: Delft University of Technology.
- Monfort, A. M. (2012). *Seaport capacity manual: Application to container terminals*. Valencia: Fundacion Valenciaport.
- Mulonda, Y. (2019, Mar 15). *How to Set Up the Development Environment*. Retrieved from Medium: <https://medium.com/dev-genius/how-to-set-up-the-development-environment-50d7249ac166>
- Nam, K.-C., & Ha, W.-I. (2001). Evaluation of Handling Systems for Container Terminals. *Journal of Waterway, Port, Coastal, and Ocean Engineering*, 171-175.
- Pajankar, A. (2017). *Introduction to Python. In: Python Unit Test Automation*. Berkeley: Apress.
- Pawellek, G., & Schonknecht, A. (2011). Cost and Performance Evaluation Impacts of Container Vessels on Seaport Container Terminals. In J. W. Bose, R. Sharda, & S. Voß (Eds.), *Handbook of terminal planning, Volume 49 of Operations research/computer science interfaces series*, 195-215.
- Petering, M. E. (2009). Effect of Block Width and Storage Yard Layout on Marine Container Terminal Performance. *Transportation Research Part E* 45, 591-610.
- Petering, M. E. (2010). Development and Simulation Analysis of Real-Time, Dual-Load Yard Truck Control Systems for Seaport Container Transshipment Terminals. *OR Spectrum* 32 (3), 633-661.
- Petering, M. E. (2011). Decision Support for Yard Capacity, Fleet Composition, Truck Substitutability, and Scalability Issues at Seaport Container Terminals. *Transportation Research Part E: Logistics and Transportation Review* 47 (1), 85-103.
- Petering, M. E., & Murty, K. G. (2009). Effect of Block Length and Yard Crane Deployment Systems on Overall Performance at a Seaport Container Transshipment Terminal. *Computers and Operations Research*, 1711-1725.
- PIANC. (2014a). *Report No 135: Design Principles for Small and Medium Marine Container Terminals*.
- PIANC. (2014b). *Report No 158: Masterplans for the Development of Existing Ports*.
- Pirhoenen, J. (2011). Automated Shuttle Carrier Concept. In J. W. Bose, R. Sharda, & S. Voß (Eds.), *Handbook of terminal planning, Volume 49 of Operations research/computer science interfaces series*, 41-59.

- Pushkov, G. (2019, April 11). *Why I think Python is the best programming language*. Retrieved from freeCodeCamp: <https://medium.com/free-code-camp/why-i-think-python-is-the-best-programming-language-for-startups-9c8f7d748b3e>
- Rijsenbrij, J. C., & Wieschemann, A. (2011). Sustainable Container Terminals: A Design. In J. W. Bose, R. Sharda, & S. Voß (Eds.), *Handbook of terminal planning, Volume 49 of Operations research/computer science interfaces series*, 61-82.
- RWG Terminal. (2020, August 10). *The process*. Retrieved from RWG Terminal: <https://www.rwg.nl/en/terminal/process>
- Saanan, Y. A., & van Valkengoed, M. (2005). Comparison of Three Automated Stacking Alternatives by Means of Simulation. *Proceedings of the 2005 Winter Simulation Conference*, 1567-1576.
- Sandberg, M., Gerth, R., Lu, W., & Gustav, J. (2016). Design Automation in Construction: An Overview. *33rd CIB W78 Conference 2016*. Brisbane.
- Schutt, H. (2011). Simulation Technology in Planning, Implementation and Operation of Container Terminal. In J. W. Bose, *Handbook of Container Terminal Planning* (pp. 103-132). Berlin: Springer.
- Steenken, D., Voß, S., & Stahlbock, R. (2004). Container terminal operation and operations. *OR Spectrum* 26, 3-49.
- Taneja, P. (2013). *The Flexible Port*. Delft: Delft University of Technology.
- Thoresen, C. A. (2014). *Port Designer's Handbook*. London: ICE Publishing.
- Tsinker, G. P. (2004). *Port Engineering: Planning, Construction, Maintenance, and Security*. Hoboken: John Wiley & Sons.
- UNCTAD. (2019). *Review of Maritime Transport 2019*. New York: United Nations Publications.
- van der Linden, L. P. (2018). *Parametric Engineering*.
- van Valkengoed, M. P. (2004). *How Passing Cranes Influence Stack Operations in a Container Terminal*. Amsterdam: Vrije Universiteit Amsterdam.
- Virdil, F., & Oguz, C. (2011). Maritime Logistics in the Global Economy: Current Trends and Approaches. In T. Blecker, C. Jahn, & W. Kersten, *Maritime Logistics in the Global Economy: Current Trends and Approaches* (pp. 95-108). Cologne: Josef Eul Verlag GmbH.
- Vis, I. F. (2006). A Comparative Analysis of Storage and Retrieval Equipment at a Container Terminal. *International Journal of Production Economics*, 680-693.
- Vis, I., & Koster, R. (2003). Transshipment of containers at a container terminal: An overview. *European Journal of Operational Research* 147, 1-16.
- Wang, Z. (2008). *Masterplan of Jebel Ali Port*. Delft: Delft University of Technology.
- Westra, E. (2013). *Python Geospatial Development, Second Edition*. Birmingham: Packt Publishing.
- Wiese, J. (2011). An Analytical Model for Designing Yard Layouts of a Straddle Carrier-Based Container Terminal. *Flexible Services and Manufacturing Journal* 25, 466-502.
- Wiese, J. (2012). *Quantitative Decision Support for the Layout Design of Container Terminals*. Paderborn: Universitat Paderborn.

- Wiese, J., Kliewer, N., & Suhl, L. (2009). A Survey of Container Terminal Characteristics and Equipment Types. *Working Paper 0901, DS & OR Lab*, University of Paderborn.
- Wiese, J., Kliewer, N., & Suhl, N. (2010). Mathematical models and solution methods for optimal container terminal yard layouts. *OR Spectrum* 32, 427-452.
- Williams, T. (2013, July 15). *System integration is a critical element in project design*. Retrieved from Control Engineering: <https://www.controleng.com/articles/system-integration-is-a-critical-element-in-project-design/>
- Yussef, A., Gibson Jr, E., Asmar, M., & Ramsey, D. (2018). Front End Engineering Design (FEED) for Large Industrial Projects: FEED Maturity and Its Impact on Project Cost and Schedule Performance. *Construction Research Congress*, 1-8.
- Zhen, L. (2016). Modelling of Yard Congestion and Optimization of Yard Template in Container Ports. *Transportation Research Part B*, 83-104.
- Zhen, L., Jiang, X., Lee, L. H., & Chew, E. P. (2013). A Review on Yard Management in Container Terminals. *Industrial Engineering & Management Systems Vol 12 No 4*, 289-305.
- Zhen, L., Xu, Z., Wang, K., & Ding, Y. (2016). Multi-Period Yard Template Planning in Container Terminals. *Transportation Research Part B*, 1-20.
- Zyngiridis, I. (2005). *Optimizing Container Movements using One and Two Automated Stacking Cranes*. Monterey: Naval Postgraduate School.

List of Figures

Figure 1 Global Containerized Trade 1966 – 2018	1
Figure 2 50 Years of Container Ship Growth	2
Figure 3 Influence and Expenditures Curve for Project Life Cycle	2
Figure 4 Container storage yard layout per equipment type illustration	3
Figure 5 Research Approach	5
Figure 6 Idealized layout and the components of a container terminal	8
Figure 7 Structures of blocks for an RMG with transfer points, an RTG with transfer lane, and an SC	9
Figure 8 STS Crane and Container Terminal using Straddle Carrier in Port of Teluk Lamong, Indonesia	10
Figure 9 Mobile Harbor Crane and Container Terminal using Straddle Carrier in Port of Paramaribo, Suriname ..	11
Figure 10 RTG Crane and Container Terminal using RMG in Port of Singapore, Singapore	11
Figure 11 RMG Crane and Container Terminal using RMG Crane in Port of Hamburg, Germany	12
Figure 12 Straddle Carrier and Container Terminal using Straddle Carrier in Port of Rotterdam, the Netherlands	12
Figure 13 Reach Stacker and Container Terminal using Reach Stacker in Port of Aarhus, Denmark	13
Figure 14 Tractor-trailer operating in Port of Cartagena, Colombia	14
Figure 15 Straddle carrier operating in Port of Gothenburg, Sweden	14
Figure 16 Typical apron width	16
Figure 17 Typical container stacking layout for Rubber-Tired-Gantry Crane	17
Figure 18 Typical cross-section container stacking arrangement for Rubber-Tired-Gantry	17
Figure 19 Typical container stacking layout (perpendicular) for Rail-Mounted Gantry Crane	18
Figure 20 Typical cross-section container stacking arrangement for Rail-Mounted-Gantry Crane	18
Figure 21 Typical container stacking layout for Straddle Carrier	18
Figure 22 Details on typical container stacking layout for Straddle Carrier	18
Figure 23 Typical cross-section container stacking arrangement for Reach Stacker	19
Figure 24 Workflow of the design tool	21
Figure 25 Area included and not included for block generation	22
Figure 26 Relationship between the annual throughput [TEU/annum], terminal shape and dimension, and the various terminal objects	24
Figure 27 Illustration for the design tool work flowchart for non-rectangular-shape container terminal	25
Figure 28 Illustration of the design rules applied for Apron Area	26
Figure 29 Illustration of the design rules applied for the apron area	27
Figure 30 Flowchart for the layout generation tool	29
Figure 31 Container terminal shape and dimension determination (Author's illustration)	30
Figure 32 Apron (quay) length and apron width calculation	30
Figure 33 Flowchart for the process in determining the primary yard area	31
Figure 34 The current_origin and container_block points for RTG	32
Figure 35 The process of generation of a container block for RTG	32

Figure 36 Container block generation in the quay-parallel direction.....	32
Figure 37 Container block generation in the quay-perpendicular direction	32
Figure 38 Generating container block and process of decreasing container block length for RTG.....	33
Figure 39 Generating container blocks further and defining secondary yard area for RTG.....	33
Figure 40 Terminal element calculations workflow.....	35
Figure 41 APM Terminal at Port of Rotterdam	37
Figure 42 Results of the design tool for the APMT-R case study: scenario 1 (left) and scenario 2 (right).....	37
Figure 43 Container Terminal 2 at Port of Jebel Ali.....	39
Figure 44 Results of the design tool for the CT2 Jebel Ali case study	39
Figure 45 Result for RTG applicability review in APMT-R case study	41
Figure 46 Result for RMG applicability review in APMT-R case study	41
Figure 47 Result for SC applicability review in APMT-R case study.....	41
Figure 48 Result for RS applicability review in APMT-R case study.....	41
Figure 49 Result for year simulation: 2022 [1.250.000 TEU demand].....	43
Figure 50 Result for year simulation: 2024 [2.250.000 TEU demand].....	43
Figure 51 Result for year simulation: 2026 [4.250.000 TEU demand].....	43
Figure 52 Result for year simulation: 2028 [2.750.000 TEU demand].....	43
Figure 53 Illustration of the design rules applied for yard area (RMG Crane)	II
Figure 54 Illustration of the design rules applied for yard area (SC)	III
Figure 55 Illustration of the design rules applied for yard area (RS)	IV
Figure 56 Flowchart for the layout generation tool.....	VII
Figure 57 Terminal shape and dimension for the design tool	VIII
Figure 58 Container terminal shape and dimension determination	IX
Figure 59 Apron (quay) length and apron width calculation.....	IX
Figure 60 The current_origin and container_block points for RTG	X
Figure 61 The process of generation of a container block for RTG	X
Figure 62 Container block generation in quay-parallel direction for RTG	XI
Figure 63 Container block generation in quay-perpendicular direction for RTG	XI
Figure 64 Generating container block and process of decreasing container block length for RTG.....	XI
Figure 65 Generating container blocks further and defining secondary yard for RTG.....	XI
Figure 66 The result of the design tool for RTG for several shapes of container terminals.....	XII
Figure 67 The current_origin and container_block points for RMG	XIII
Figure 68 The process of generation of a container block for RMG	XIII
Figure 69 Container block generation in quay-parallel direction for RMG.....	XIV
Figure 70 Container block generation in quay-perpendicular direction for RMG	XIV
Figure 71 Generating container block and process of decreasing container block length for RMG.....	XIV
Figure 72 Generating container blocks further and defining secondary yard area for RMG.....	XIV
Figure 73 The result of the design tool for RMG for several shapes of container terminals	XV

Figure 74 The current_origin and container_block points for SC	XVII
Figure 75 The process of generation of a container block for SC.....	XVII
Figure 76 Container block generation in quay-parallel direction for SC.....	XVII
Figure 77 Container block generation in quay-perpendicular direction for SC	XVII
Figure 78 Generating container block and process of decreasing container block length for SC	XVIII
Figure 79 Generating container blocks further and defining secondary yard area for SC	XVIII
Figure 80 The result of the design tool for Straddle Carrier for several shapes of container terminals	XIX
Figure 81 The current_origin and container_block points for RS.....	XX
Figure 82 The process of generation of a container block for RS.....	XX
Figure 83 Container block generation in quay-parallel direction for RS	XX
Figure 84 Container block generation in quay-perpendicular direction for RS.....	XX
Figure 85 Generating container block and process of decreasing container block length for RS	XXI
Figure 86 Generating container blocks further and defining secondary yard area for RS	XXI
Figure 87 The result of the design tool for Reach Stacker for several shapes of container terminals	XXII
Figure 88 Default Container Block for the first scenario of the APMT-R case study.....	XXIV
Figure 89 Default Container Block for CT2 Jebel Ali case study	XXV
Figure 90 RWG Terminal at Port of Rotterdam	XXVI
Figure 91 Results of the design tool for the RWG case study:scenario 1 (left) and scenario 2 (right)	XXVII
Figure 92 Schematic overview required berth calculation.....	XXIX
Figure 93 Schematic overview gate calculation.....	XXX
Figure 94 Organization of the shapely library.....	XXXIII
Figure 95 Examples for shapely code.....	XXXV
Figure 96 OpenTISim QR code	XXXVI
Figure 97 GitHub QR code.....	XXXVI

List of Tables

Table 1 Overview of earlier research and development on container terminal design	6
Table 2 Scope of the design tool	21
Table 3 Input and output for every task of the design tool.....	22
Table 4 Design Rules Applied for Apron Area	26
Table 5 Design Rules Applied for Yard Area (RTG Crane)	27
Table 6 Comparison of terminal design from the actual project and the design tool for APMT-R case study	38
Table 7 Comparison of terminal design from the actual project and the design tool for CT2 Jebel Ali case study	40
Table 8 Comparison of the result from RTG, RMG, SC, and RS for applicability review in APMT-R case study	42
Table 9 Comparison of the result in changing throughput demand for applicability review in APMT-R case study	44
Table 10 Comparison of the result in element calculation and cost estimation for 1.250.000 TEU/year throughput demand	45
Table 11 Comparison of the result in element calculation and cost estimation for 2.250.000 TEU/year throughput demand	46
Table 12 Design Rules Applied for Yard Area (RMG Crane)	II
Table 13 Design Rules Applied for Yard Area (SC).....	III
Table 14 Design Rules Applied for Yard Area (RS)	IV
Table 15 Stacking equipment qualitative comparison	V
Table 16 Inputs and defaults for the design tool validation APMT-R case study	XXIII
Table 17 Inputs and defaults for the design tool validation CT2 Jebel Ali case study	XXIV
Table 18 Inputs and defaults for the design tool validation RWG Terminal case study.....	XXV
Table 19 Comparison of the RWG Terminal with the design tool.....	XXVII
Table 20 Default unit rate of CAPEX calculation	XXXI
Table 21 Default unit rate of OPEX calculation	XXXII

Appendix

Appendix A

Container terminal design rules

Implementing the design guidelines from the literature study, design rules that will be used in the design tool development for yard area (for RMG crane) can be shown in Figure 53 and Table 12 below.

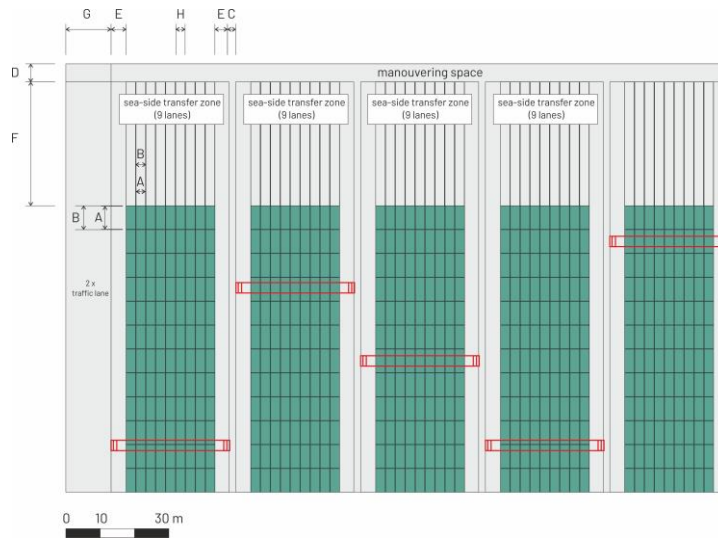


Figure 53 Illustration of the design rules applied for yard area (RMG Crane)
(Author's Illustration)

Table 12 Design Rules Applied for Yard Area (RMG Crane)

Element	Value	Source	Remarks	
A	Net TGS	2.44 x 6.10 m	PIANC (2014a)	
B	Gross TGS	2.90 x 6.70 m	Koster (2019)	
C	Margin parallel to the stack	2.0 m	Koster (2019)	Safety margin between two RMG tracks
D	Margin at stack head	5.0 m	Koster (2019)	Manoeuvring space for trucks
E	Track width	4.0 m	Koster (2019)	
F	Length parking area	35 m	Koster (2019)	Allowing for quay-stack and stack-land interaction
G	Width traffic lane	12.90 m	Koster (2019)	Two-way
H	Width parking lane	2.90 m	Koster (2019)	Similar to gross width TGS
Layout Rule	Value	Source	Remarks	
Maximum stack length	48 TEU	Koster (2019)	(Kemme, 2013)	
Minimum stack length	28 TEU	(Virdil & Oguz, 2011)	(Kemme, 2013)	
Stack – terminal vehicle interaction	End-loaded	PIANC (2014a)	Two loading points, one seaside and one landside, r	
Stack orientation	Perpendicular to quay	PIANC (2014a)	The easier separation between seaside and landside operation	

Implementing the design guidelines from the literature study, design rules that will be used in the design tool development for yard area (for SC) can be shown in Figure 54 and Table 13 below.



Figure 54 Illustration of the design rules applied for yard area (SC)
(Author's Illustration)

Table 13 Design Rules Applied for Yard Area (SC)

Element	Value	Source	Remarks	
A	Net TGS	2.44 x 6.10 m	PIANC (2014a)	
B	Gross TGS	2.90 x 6.70 m	Koster (2019)	
C	Width traffic lane	10.0 m	PIANC (2014a)	Manoeuvring space for straddle carriers between stacks
D	Margin at stack head	2.0 m	Koster (2019)	Accounting for the turning radius of SC handling 40 feet containers
E	Apron side truck lane	4.0 m	Koster (2019)	Two-way safety margin
Element	Value	Source	Remarks	
Maximum stack length	126 m	Koster (2019)	Only including gross TGS	
Minimum stack length	0.5 x Maximum stack length		Stack length should be at least 0.25 x maximum stack length in order to maximize the effectiveness of the block	
Maximum stack width	210 m	Koster (2019)	Only including gross TGS	
Minimum stack width	0.5 x Minimum stack width		Stack width should be at least 0.25 x maximum stack width in order to maximize the effectiveness of the block	
Stack – terminal vehicle interaction	Outside stack	PIANC (2014a)	Separate interchange area at landside for truck loading	
Stack orientation	Perpendicular to quay	PIANC (2014a)	Usually higher capacity	

Implementing the design guidelines from the literature study, design rules that will be used in the design tool development for yard area (for RS) can be shown in Figure 55 and Table 14 below.

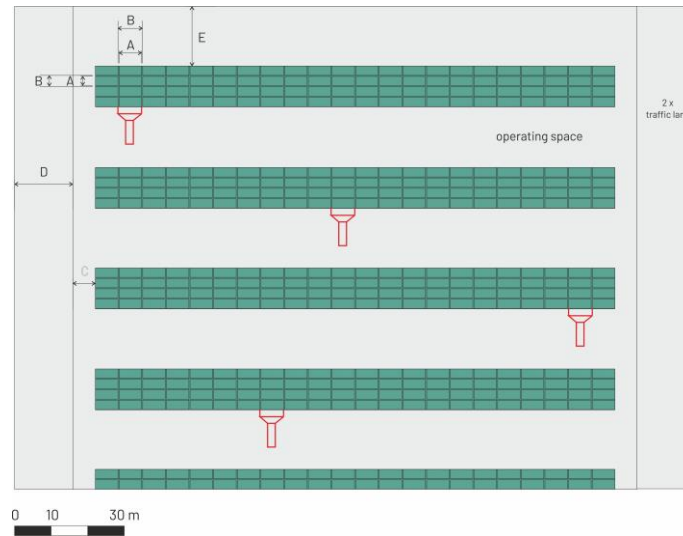


Figure 55 Illustration of the design rules applied for yard area (RS)
(Author's Illustration)

Table 14 Design Rules Applied for Yard Area (RS)

Element	Value	Source	Remarks	
A	Net TGS	2.44 x 6.10 m	PIANC (2014a)	
B	Gross TGS	2.79 x 6.45 m	Koster (2019)	
C	Margin at stack head	6.0 m	Koster (2019)	Additional manoeuvring space to improve operations
D	Traffic lane	16.0 m	Koster (2019)	Only including gross ground slots
E	Operating space	16.30 m	PIANC (2014a)	Allowing stacking up to 5 containers high
Layout Rule	Value	Source	Remarks	
Maximum stack length	220 m	Koster (2019)	Stack length should be maximized up to 220 m for operational purposes.	
Maximum stack length	220 m	Koster (2019)	Stack length should be maximized up to 220 m for operational purposes.	
Minimum stack length	10 x Gross TGS	(Virdil & Oguz, 2011)	Stack length should be at least can be placed for 10 TGS in order to maximize the effectiveness of the block	
Maximum stack width	4 x Gross TGS	PIANC (2014a)		
Stack – terminal vehicle interaction	Side-loaded	PIANC (2014a)	Standard loading procedure for RS	
Stack orientation	Parallel to quay	PIANC (2014a)		

Appendix B

Stacking equipment qualitative comparison

Table 15 Stacking equipment qualitative comparison (PIANC, 2014a)

Topic	Rubber Tired Gantry Crane (RTG)	Rail Mounted Gantry Crane (RMG)	Straddle Carrier (SC)	Reach Stacker (RS)
Stacking Density	Potential for high stacking densities	Potential for high speeds, stacking densities and precision	Potential for medium stacking densities and good accessibility to containers	Potential for high stacking densities for empty containers
Compatible with operational automation	Can be used with a degree of operational automation	Very suitable, including in conjunction with automated quay to stack operations	Can be designed for automated operation	Low technology facilitates rapid startup with minimal training
Stack width and height	Can be designed to span up to 9 rows and up to 6 tiers high	Can be designed for a wide range of spans and stack heights	Can be designed to stack 4 tiers high	Can be designed to stack full containers up to 5 tiers high (in first row) and to block stack empty containers 8 tiers high
Terminal shape	Suited to terminals with large rectangular stacking areas	Limited to terminals with large rectangular stacking areas	Can operate in irregularly shaped stacking areas	Can operate in irregularly shaped stacking areas
Flexibility of operation	Cranes can move between stacks and to/from remote maintenance facility	Cranes within a stack can be designed to nest (i.e. pass each other using two sets of rails and crane sizes), but in general cranes cannot move between stacks	Whole fleet can be deployed anywhere within the stack yard	Machine versatility enables them to be used for various functions at any location in the terminal that is capable of withstanding the high wheel loads
Traffic segregation	Terminal transfer vehicles and road vehicles need to share lanes, which requires good traffic management	Stacks oriented end-on to the quay can be designed for container transfers at the stack ends but this results in considerable unproductive longtravel; stacks parallel to the quay may require terminal and road vehicles to share lanes, which	Good control as terminal and road vehicles are handled at separate locations and are segregated from the stacks; with direct operations, tractor- trailers are not required	The machines serve both terminal and road vehicles within the stack yard, which requires strict control

Topic	Rubber Tired Gantry Crane (RTG)	Rail Mounted Gantry Crane (RMG)	Straddle Carrier (SC)	Reach Stacker (RS)
		requires good traffic management		
Rail tracks and travelling with containers	Rail tracks are not required; except for local shuffles, the cranes cannot carry containers while travelling from stack to stack	Medium/heavy rail tracks are required; cranes can carry full containers along the stack at high speed	Rail tracks are not required	Rail tracks are not required
Loadings	Specially designed runways may be required but can sometimes be avoided by using 16-wheel machines instead of 8-wheel, which allows flexibility in stack yard layout	Cantilevered spans may result in reduced rail spans and wheel loads but longer cranes	Wheel loads may require localized strengthening of pavement for runways, but uniform paving design allows flexibility in stack yard layout	Stack yard layout can be readily modified
Paving requirements	Paving of stack areas can be lighter duty if heavy vehicles are excluded	Paving of stack areas can be lighter duty if heavy vehicles are excluded	Entire stack yard generally has to accommodate the heaviest loadings	Entire stack yard generally has to accommodate the heaviest loadings
Power source	Usually powered by crane's diesel engine, avoiding the need for HV power supply, but fixed electrical power is also available for low emissions	Usually employ fixed HV electrical power supply, but diesel alternative exists if power supply is inadequate	No requirement for electrical power supply infrastructure	No requirement for electrical power supply infrastructure
Emissions	Medium air and noise emissions with diesel power, low or zero with electrical power	Zero air and low noise emissions with electrical power	Medium air and noise emissions	Medium
Stacking Density	Long lead time	Long lead time	Medium lead time	Short delivery lead time and low technology facilitate rapid start-up with minimal training
Capital costs	Medium	High, but long design life and low maintenance should help to minimize whole-life costs	Medium to low, but total fleet cost may be comparable to RTG system; relatively high maintenance costs for equipment and pavement	Relatively low, suitable for low budget terminals; relatively high pavement maintenance costs

Appendix C

Container terminal layout generation tool

The design tool generates the layout by means that placing the "physical" elements (terminal area such as apron, primary yard, secondary yard; container blocks) into a canvas based on the throughput demand and terminal dimension as an input. From the literature study, the container terminal consists in general of (a) Apron; (b) Primary yard; (c) Secondary yard. In general, the flowchart of the layout generation tool can be shown in Figure 30 below.

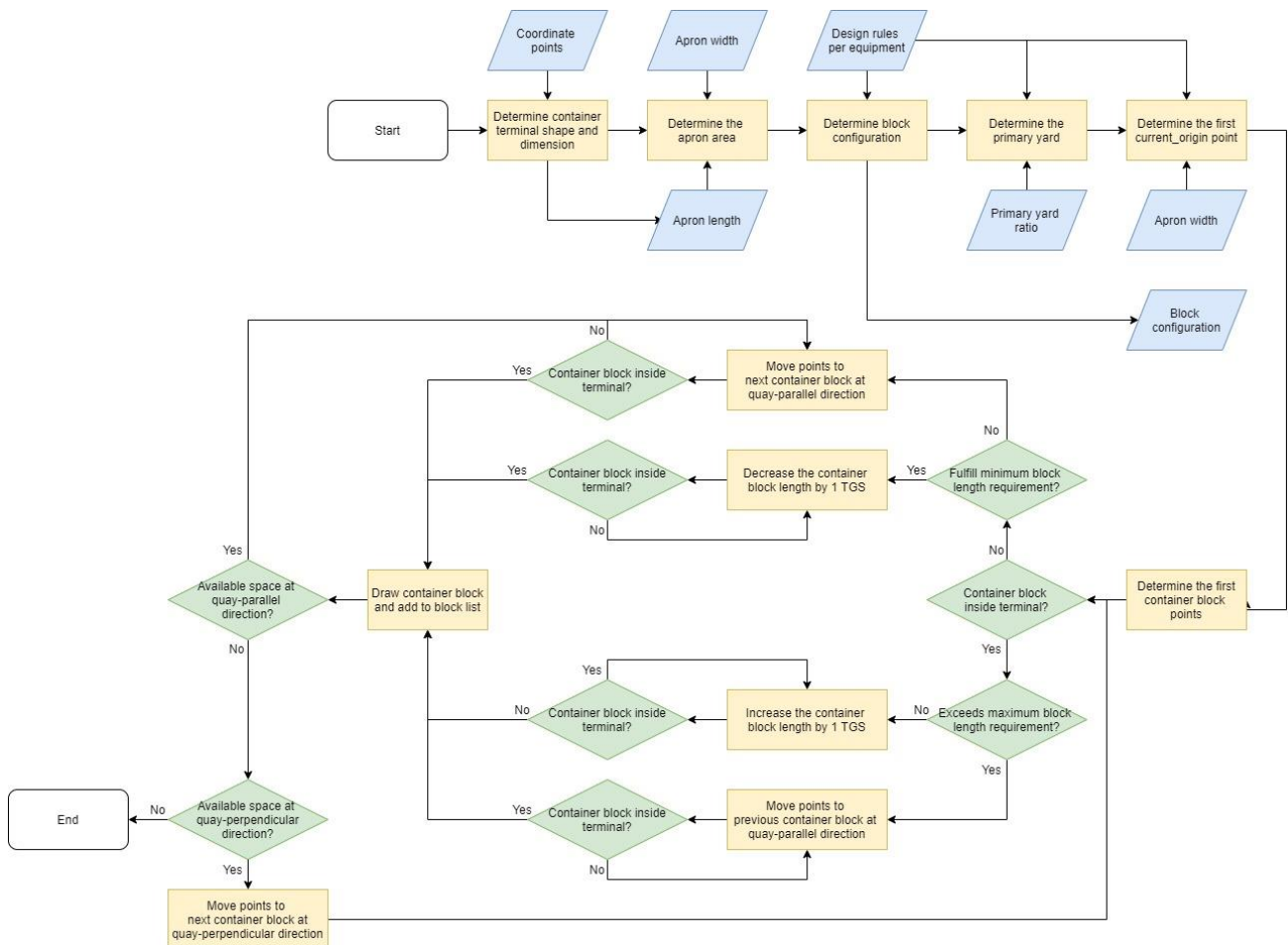


Figure 56 Flowchart for the layout generation tool.
(Author's Illustration)

First, the container terminal shape and dimension are defined by determining the number of points and the x-y coordinates of the points so that it can determine the boundary conditions for the terminal. Now, this method is considered, but in the future, there might be another method such as using (mouse) pointer to give more flexibility in determining the container shape and dimension. Moreover, at the moment, seven main shapes are being used for the design tool, in order to cover all possibilities of the condition that could happen in generating the container blocks. All the shapes and specifications for the shapes can be shown in Figure 57.

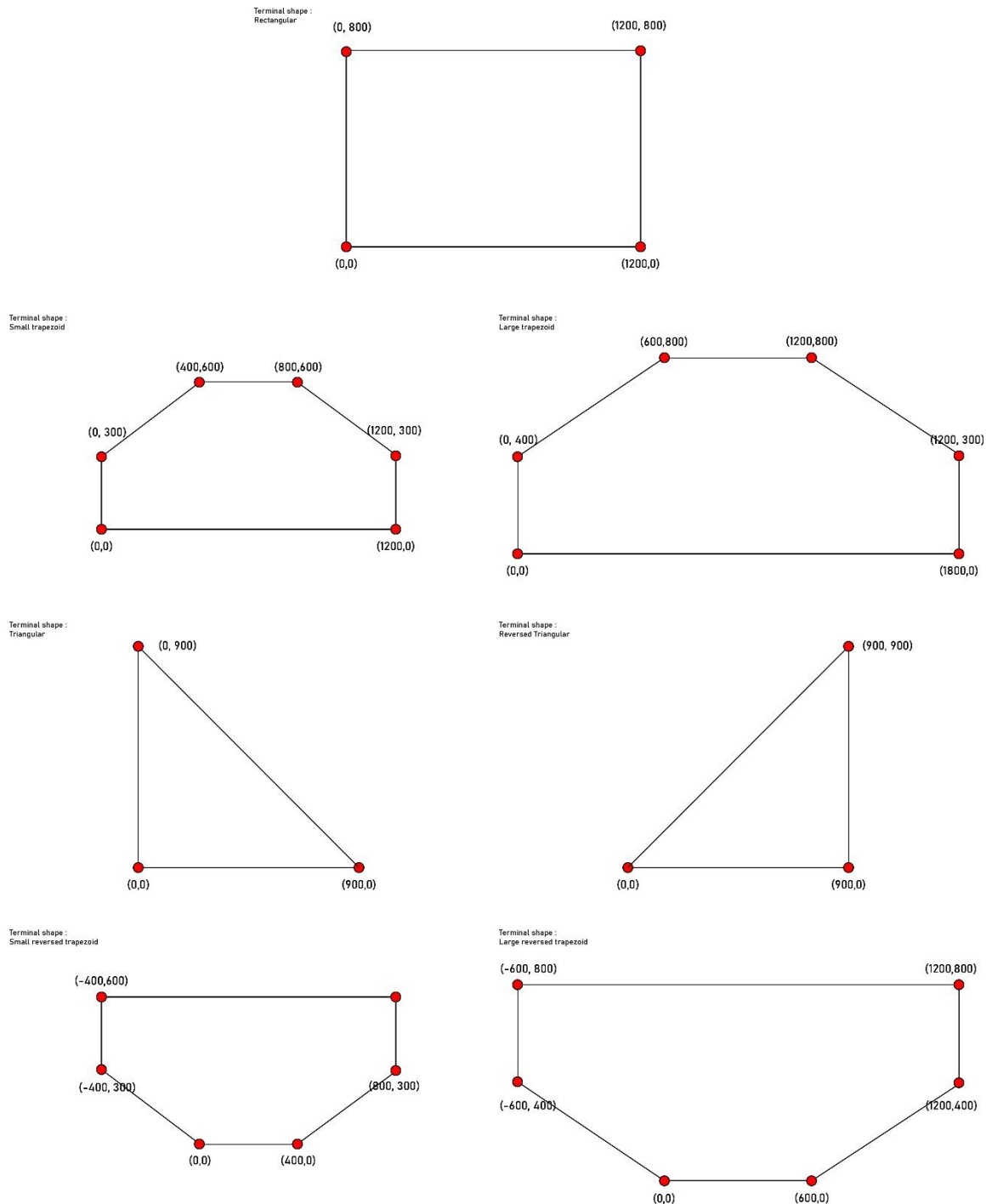


Figure 57 Terminal shape and dimension for the design tool
(Author's Illustration)

To describe in detail how the layout generation tool works, specific terminal shape and dimension is chosen for example, where it has a trapezoidal shape that will represent a non-rectangular container terminal, which can be shown in Figure 58. Next, based on the terminal dimension, the apron length is determined, while the apron width is determined from the design rules in the literature study. Therefore, the apron area could be determined based on the apron length and the apron width. For illustration, the apron area in the container terminal is shown in Figure 59.

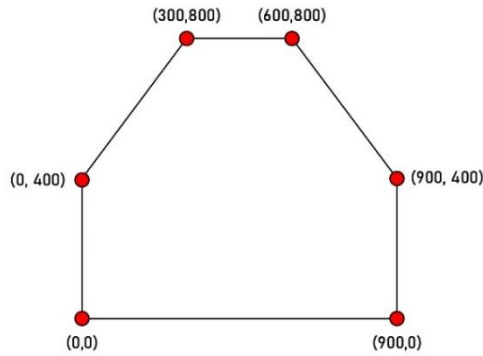


Figure 58 Container terminal shape and dimension determination (Author's illustration)

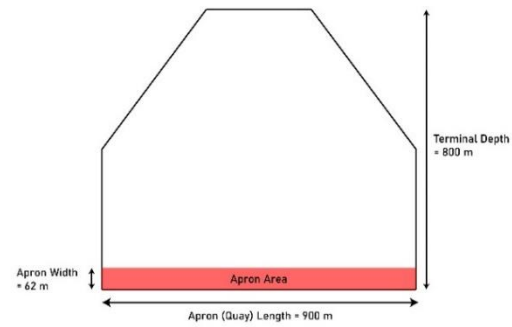


Figure 59 Apron (quay) length and apron width calculation (Author's illustration)

Next, the block configuration, which is the block length and the block width, is determined based on the design rules that differ for each type of equipment.

(a) Rubber-Tired Gantry Crane

For RTG, the block length can be calculated using the equations below.

$$n_{\text{block}} \approx \frac{\text{quay length} - \text{traffic lane}}{\text{max. block length} + \text{crossing width lane}}$$

$$\text{block length} = \frac{\text{quay length} - \text{traffic lane} - n_{\text{block}} * \text{crossing width lane}}{n_{\text{block}}}$$

Then, as mentioned before, part of the yard that will be used for the stacking activities is the primary yard; therefore, the primary yard area should be determined. The primary yard area is determined based on the container terminal configuration; and also the design rules regarding the ratio of the primary yard area to the total terminal yard area, which should be around 0.6 – 0.75 (Thoresen, 2014) or 0.6 – 0.7 (PIANC, 2014a). For the design tool, the value 0.6 – 0.7 will be used. Also, another the design rules are applied, that the primary yard should be located as near as possible to the apron area, and the secondary yard should be located as near as possible to the hinterland, following the idealized container terminal layout.

After that, the generation of the container blocks is started. First, the design tool will define the point (in coordinate) that represents the 'first point' for each container block, which the design tool called the 'current_origin'. And then, based on that point, the design tool defines the points (in coordinates) that will represent each edge location of a container block. There will be 5 points, where for RTG, point 0 and point 4 are the 'current_origin', point 1 is the 'current_origin' y-coordinate plus the block width, point 2 is the 'current_origin' x-coordinate plus the block length and y-coordinate plus the block width, point 3 is the 'current_origin' x-coordinate plus the block length. For illustration, the points on the container blocks can be shown in Figure 60. Then, using the shapely module, the design tool checks whether the points of the container blocks are inside the terminal or not. The term 'inside' the terminal means that if all container block coordinate points are inside the stacking area, and the term 'outside' the terminal means that if one of the container block coordinate points is outside the stacking area. If the container block is inside the terminal, the design tool checks whether it is possible to add either 1 TGS or 1 full container block. This algorithm is important because it will make the design tool to use the available space as much as possible so that the container terminal area is used efficiently. If it is possible to add either 1 TGS or 1 full container block,

the design tool does the process until it reaches the condition when it is not possible anymore and defines the new block configuration. If it is not possible to add even 1 TGS, the design tool defines the initial block configuration. After the block configuration is defined, the design tool generates a container block in the form of a rectangle based on the block coordinate points. In generating the container block, a set of design rules regarding maximum and minimum container block length is applied. A container block will be generated if only the block length is less than the maximum block length, and more than the minimum block length. For illustration, this process can be shown in Figure 61.

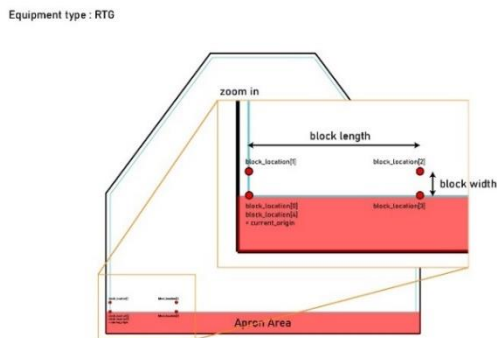


Figure 60 The current_origin and container_block points for RTG (Author's illustration)

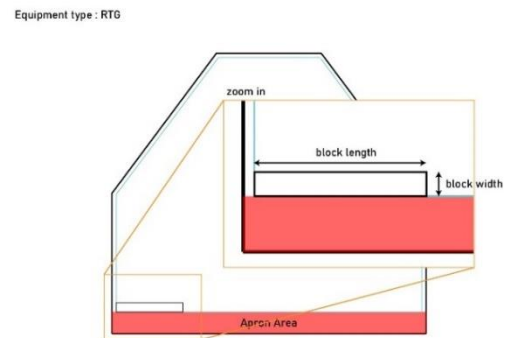


Figure 61 The process of generation of a container block for RTG (Author's illustration)

Next, the design tool checks whether there are available spaces in the quay-parallel direction. If there are available spaces in the quay-parallel direction, the design tool moves the container block coordinate points to the next container block location in quay-parallel direction. Design rules regarding the required space between the container block are applied in this algorithm. Accordingly, the design tool does the container block generation iteratively until it reaches the condition where there are no available spaces to generate a container block in quay-parallel direction. For illustration, this process can be shown in Figure 62. However, if there are no available spaces in the quay-parallel direction, the design tool checks whether there are available spaces in the quay-perpendicular direction. If there are available spaces in the quay-perpendicular direction, the design tool moves the container block coordinate points to the next container block in the quay-perpendicular direction. Design rules regarding the required space between the container block are applied in this algorithm. For illustration, this process can be shown in Figure 63.

Equipment type : RTG

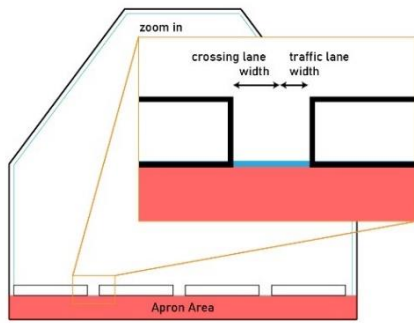


Figure 62 Container block generation in quay-parallel direction for RTG (Author's illustration)

Equipment type : RTG

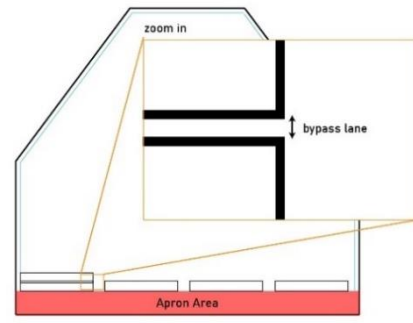


Figure 63 Container block generation in quay-perpendicular direction for RTG (Author's illustration)

Then, the design tool checks whether the points of the container blocks are inside the terminal or not iteratively. However, if the container block is outside the terminal, meaning that one of the container block coordinate points is outside the stacking area, the design tool checks whether it is possible to subtract 1 TGS block. This algorithm is also important because it will make the design tool to use the available space as much as possible so that the container terminal area is used efficiently. If it is possible to subtract 1 TGS, the design tool does the process until it reaches the condition when it is not possible anymore and defines the new block configuration. However, a container block will be generated if only the block length is more than the minimum block length. If the container block length is less than the minimum block length, the design tool moves the container block coordinate points to the next container block location in quay-parallel direction. For illustration, this process can be shown in Figure 64. Subsequently, the design tool continues the container block generation until it reaches the boundary of the stacking area, where there are no available spaces to generate container blocks in the quay-perpendicular direction. For illustration, this process can be shown in Figure 65.

Equipment type : RTG

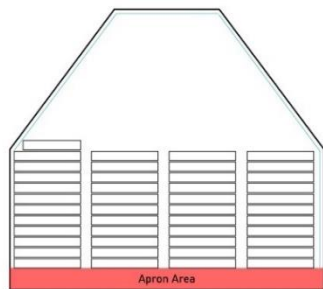


Figure 64 Generating container block and process of decreasing container block length for RTG (Author's illustration)

Equipment type : RTG

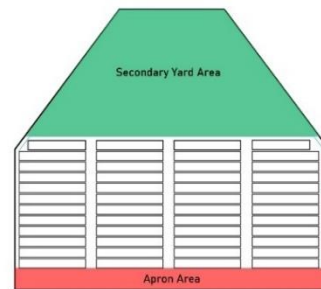


Figure 65 Generating container blocks further and defining secondary yard for RTG (Author's illustration)

The result of the design tool for several shapes of the container terminal is shown in Figure 66 below.

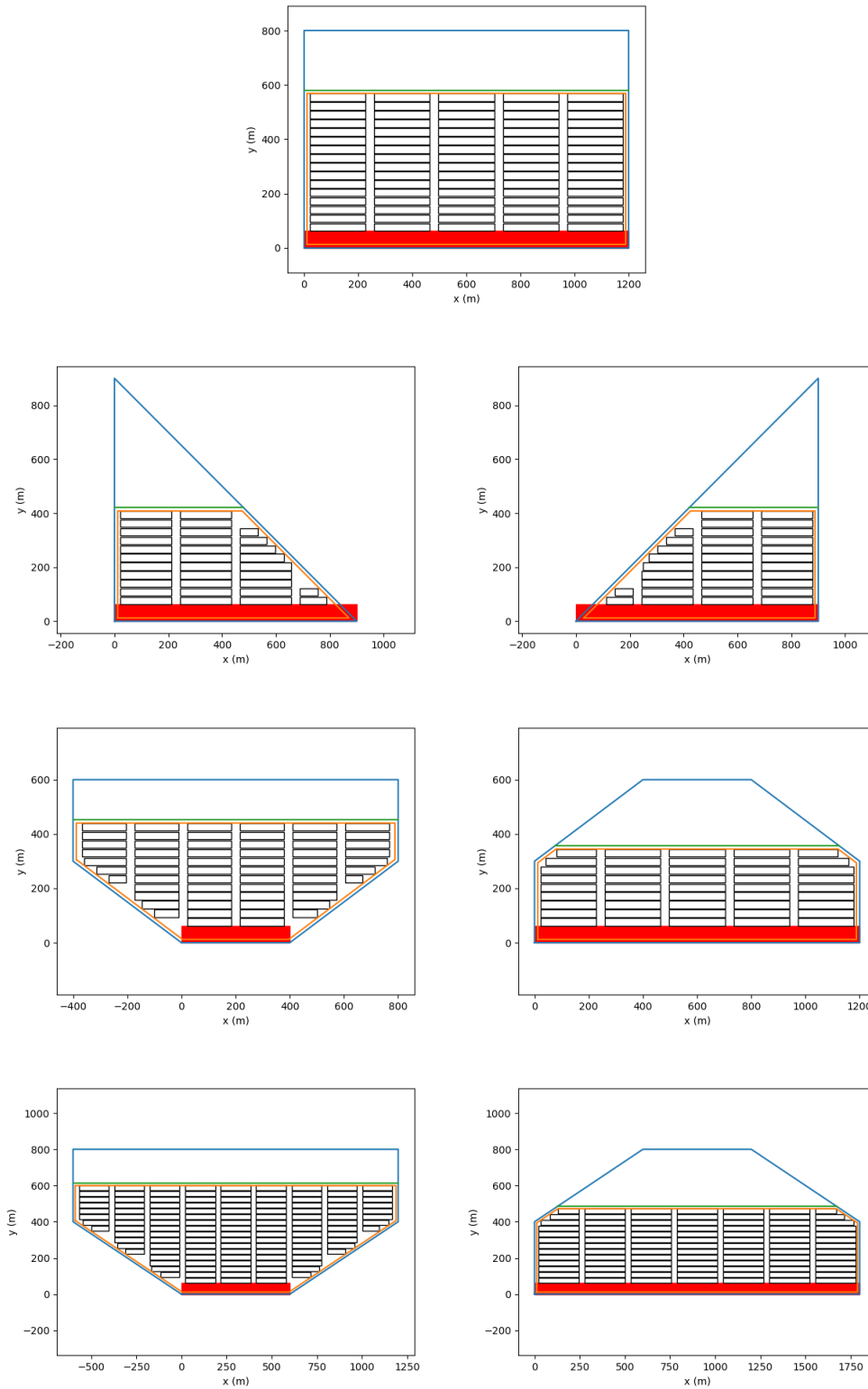


Figure 66 The result of the design tool for RTG for several shapes of container terminals
(Author's illustration)

(b) Rail-Mounted Gantry Crane (RMG)

For RMG, the block length can be calculated using the equations below.

$$n_{\text{block}} \approx \frac{\text{terminal depth} - \text{apron width}}{\text{max. block length} + \text{traffic lane} + 2 * \text{margin at stack}}$$

$$\text{block length} = \frac{\text{terminal depth} - \text{apron width} - n_{\text{block}} * (\text{traffic lane} + 2 * \text{margin at stack})}{n_{\text{block}}}$$

Then, the primary yard is determined based on the terminal configuration and the design rules. After that, the design tool will define the point (in coordinate) that represents the ‘first point’ for each container block, which the design tool called the ‘current_origin’. And then, based on that point, the design tool defines the points (in coordinates) that will represent each edge location of a container block. There will be 5 points, where for RMG, point 0 and point 4 are the ‘current_origin’, point 1 is the ‘current_origin’ y-coordinate plus the block length, point 2 is the ‘current_origin’ x-coordinate plus the block width and y-coordinate plus the block length, point 3 is the ‘current_origin’ x-coordinate plus the block width. For illustration, the points on the container blocks can be shown in Figure 67. Then, a slight difference algorithm also applied for RMG-system. The design tool checks whether the points of the container blocks are inside the terminal or not (the definition ‘inside’ and ‘outside’ is the same as explained in the RTG-system part). If the container block is inside the terminal, before the design tool checks whether it is possible to add either 1 TGS or 1 full container block, the design tool will check whether there will be only 1 container block in the quay-perpendicular area. If there will be only 1 container block in the quay-perpendicular area, and if it is possible to add either 1 TGS or 1 full container block, the design tool does the process until it reaches the condition when it is not possible anymore and defines the new block configuration. If there will be more than 1 container block in quay-perpendicular direction, the design tool defines the initial block configuration. After the block configuration is defined, the design tool generates a container block in the form of a rectangle based on the block coordinate points. In generating the container block, a set of design rules regarding maximum and minimum container block length is applied. A container block will be generated if only the block length is less than the maximum block length, and more than the minimum block length. For illustration, this process can be shown in Figure 68.

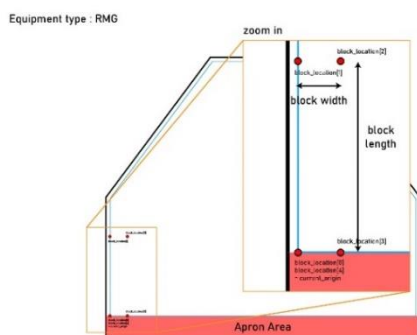


Figure 67 The current_origin and container_block points for RMG (Author's illustration)

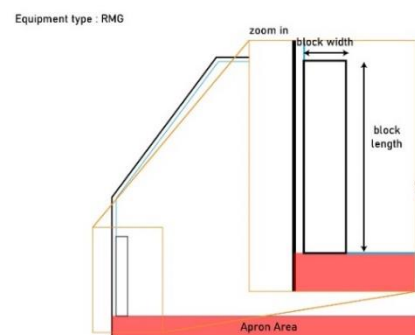


Figure 68 The process of generation of a container block for RMG (Author's illustration)

Next, the design tool checks whether there are available spaces at the quay-parallel direction, and If there are available spaces at the quay-parallel direction, the design tool moves the container block coordinate points to the next container block location in quay-parallel direction, and the design tool does the container block generation iteratively until it reaches the condition where there are no available spaces to generate a container block in quay-

parallel direction. For illustration, this process can be shown in Figure 62. If there are no available spaces at the quay-parallel direction, the design tool checks whether there are available spaces at the quay-perpendicular direction, and if there are available spaces at the quay-perpendicular direction, the design tool moves the container block coordinate points to the next container block in quay-perpendicular direction. Design rules regarding the required space between the container block are applied in this algorithm. For illustration, this process can be shown in Figure 63.

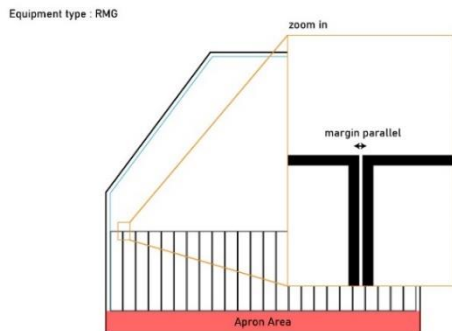


Figure 69 Container block generation in quay-parallel direction for RMG (Author's illustration)

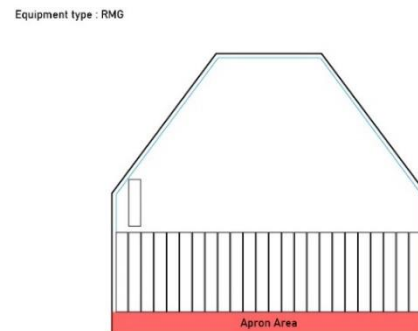


Figure 70 Container block generation in quay-perpendicular direction for RMG (Author's illustration)

Then, the design tool checks whether the points of the container blocks are inside the terminal or not iteratively, and if the container block is outside the terminal, meaning that one of the container block coordinate points is outside the stacking area, the design tool checks whether it is possible to subtract 1 TGS block, and if it is possible to subtract 1 TGS, the design tool does the process until it reaches the condition when it is not possible anymore and defines the new block configuration. A container block will be generated if only the block length is more than the minimum block length, and if the container block length is less than the minimum block length, the design tool moves the container block coordinate points to the next container block location in quay-parallel direction. For illustration, this process can be shown in Figure 71. Subsequently, the design tool continues the container block generation until it reaches the boundary of the stacking area, where there are no available spaces to generate container blocks in the quay-perpendicular direction. For illustration, this process can be shown in Figure 72.

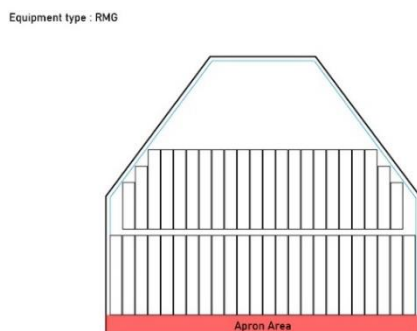


Figure 71 Generating container block and process of decreasing container block length for RMG (Author's illustration)

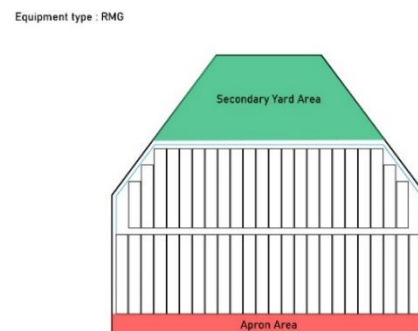


Figure 72 Generating container blocks further and defining secondary yard area for RMG (Author's illustration)

The result of the design tool for several shapes of the container terminal using RMG is shown in Figure 73.

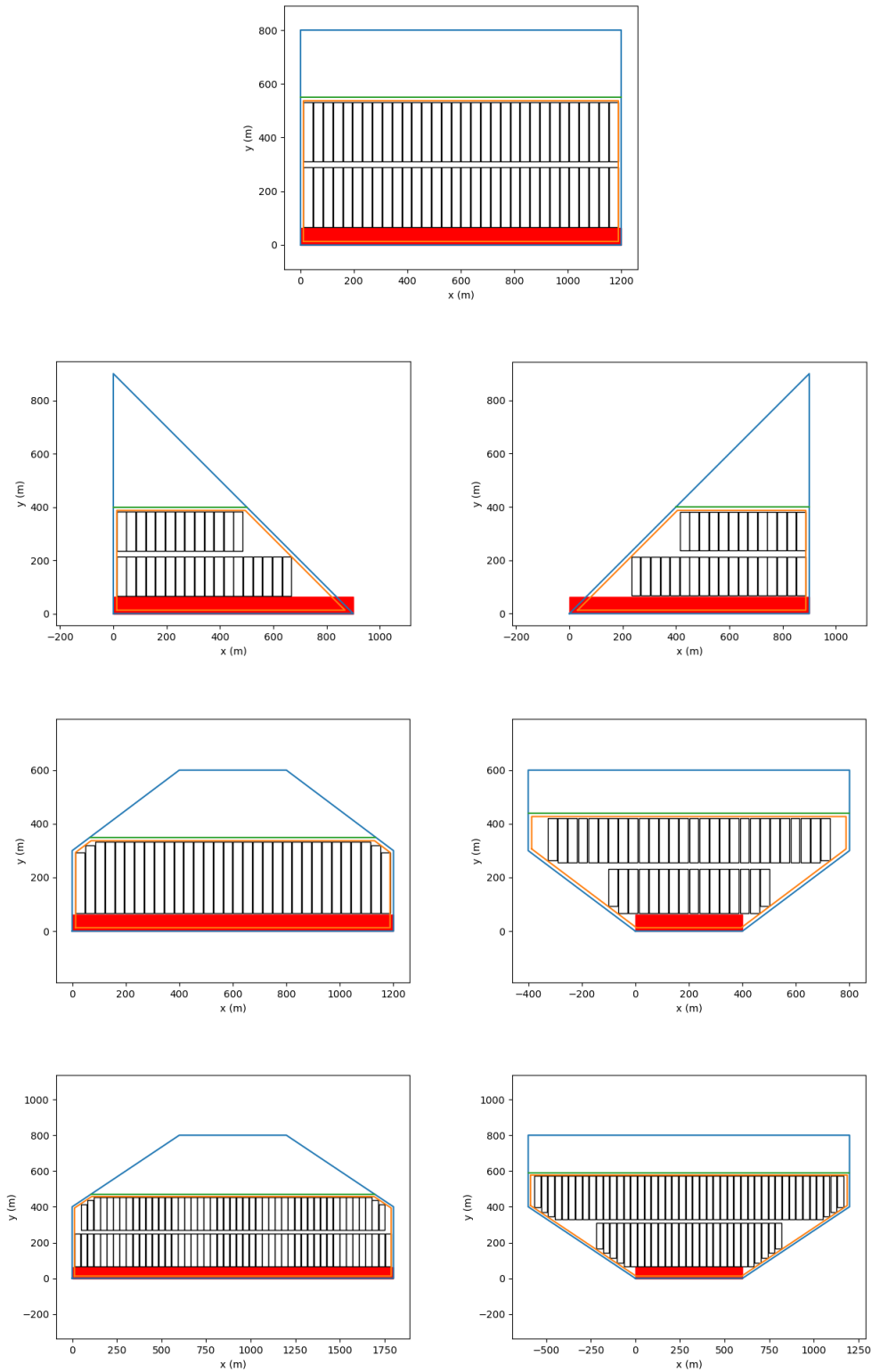


Figure 73 The result of the design tool for RMG for several shapes of container terminals (Author's Illustration)

(c) Straddle Carrier (SC)

For SC, first, not only the block length is determined, but also block width because the block width is not fixed to a certain number of TGS. The block length and block width are determined based on the design rules of the container terminal for each type of equipment. For SC, block length can be calculated using the equations below.

$$n_{\text{block-length}} \approx \frac{\text{terminal depth} - \text{apron width} - \text{margin at stack head}}{\text{max. block width} + \text{traffic lane}}$$

$$\text{block length} = \frac{\text{terminal depth} - \text{apron width} - \text{margin stack head} - \text{traffic lane}}{n_{\text{block}}} - \text{traffic lane}$$

And, block width can be calculated using the equations below.

$$n_{\text{block-width}} \approx \frac{\text{quay length} - \text{traffic lane}}{\text{max. block length} + \text{traffic lane}}$$

$$\text{block width} = \frac{\text{quay length} - \text{traffic lane}}{n_{\text{block}}} - \text{traffic lane}$$

For SC, the algorithm is more or less combining the algorithm from RTG and RMG system, since it can adapt based on its block length and block width. After determining the primary yard (same as RTG and RMG system), the generation of the container blocks is started. First, the design tool will define the point (in coordinate) that represents the 'first point' for each container block, which the design tool called the 'current_origin'. And then, based on that point, the design tool defines the points (in coordinates) that will represent each edge location of a container block. There will be 5 points, where for RTG, point 0 and point 4 are the 'current_origin', point 1 is the 'current_origin' y-coordinate plus the block width, point 2 is the 'current_origin' x-coordinate plus the block length and y-coordinate plus the block width, point 3 is the 'current_origin' x-coordinate plus the block length. For illustration, the points on the container blocks can be shown in Figure 74. Then, the design tool checks whether the points of the container blocks are inside the terminal or not. Adopting from RTG-system algorithm, if the container block is inside the terminal, the design tool checks whether it is possible to add either 1 TGS or 1 full container block in quay-parallel direction if it is possible to add either 1 TGS or 1 full container block, the design tool does the process until it reaches the condition when it is not possible anymore and defines the new block configuration, and If it is not possible to add even 1 TGS, the design tool defines the initial block configuration. In addition, adopting from the RMG-system algorithm, the design tool will check whether there will be only 1 container block in the quay-perpendicular area. If there will be only 1 container block in the quay-perpendicular area, and if it is possible to add either 1 TGS or 1 full container block, the design tool does the process until it reaches the condition when it is not possible anymore and defines the new block configuration. If there will be more than 1 container block in quay-perpendicular direction, the design tool defines the initial block configuration. After the block configuration is defined, the design tool generates a container block in the form of a rectangle based on the block coordinate points. In generating the container block, a set of design rules regarding maximum and minimum container block length is applied. A container block will be generated if only the block length is less than the maximum block length, and more than the minimum block length. For illustration, this process can be shown in Figure 75.

Equipment type : Straddle Carrier (SC)

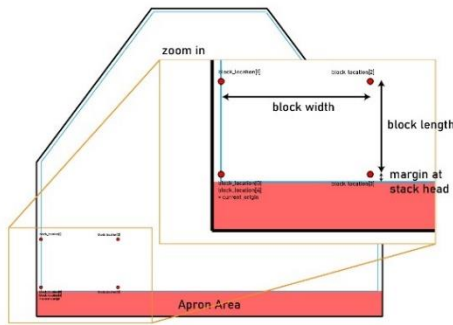


Figure 74 The current_origin and container_block points for SC (Author's illustration)

Equipment type : Straddle Carrier (SC)

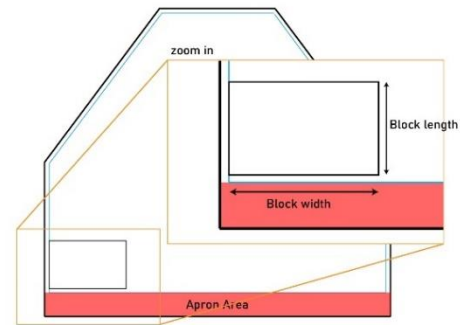


Figure 75 The process of generation of a container block for SC (Author's illustration)

Next, the design tool checks whether there are available spaces at the quay-parallel direction, and if there are available spaces at the quay-parallel direction, the design tool moves the container block coordinate points to the next container block location in quay-parallel direction, and the design tool does the container block generation iteratively until it reaches the condition where there are no available spaces to generate a container block in quay-parallel direction. For illustration, this process can be shown in Figure 76. If there are no available spaces at the quay-parallel direction, the design tool checks whether there are available spaces at the quay-perpendicular direction, and if there are available spaces at the quay-perpendicular direction, the design tool moves the container block coordinate points to the next container block in quay-perpendicular direction. Design rules regarding the required space between the container block are applied in this algorithm. For illustration, this process can be shown in Figure 77.

Equipment type : Straddle Carrier (SC)

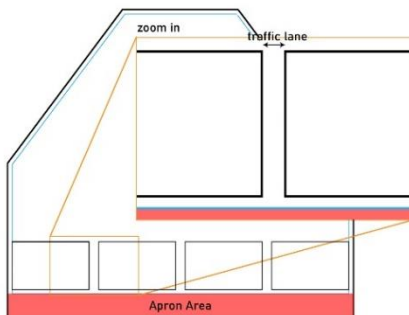


Figure 76 Container block generation in quay-parallel direction for SC (Author's illustration)

Equipment type : Straddle Carrier (SC)

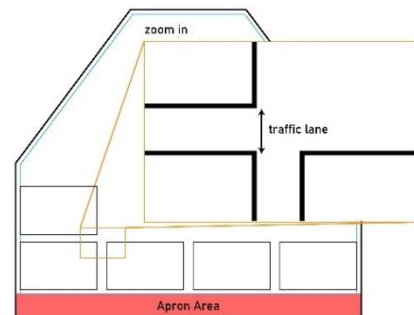
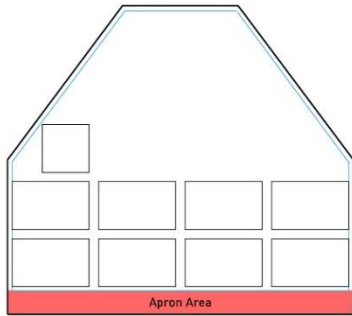


Figure 77 Container block generation in quay-perpendicular direction for SC (Author's illustration)

Then, the design tool checks whether the points of the container blocks are inside the terminal or not iteratively, and if the container block is outside the terminal, meaning that one of the container block coordinate points is outside the stacking area, the design tool checks whether it is possible to subtract 1 TGS block, and if it is possible to subtract 1 TGS, the design tool does the process until it reaches the condition when it is not possible anymore and defines the new block configuration. A container block will be generated if only the block length is more than the minimum block length, and if the container block length is less than the minimum block length, the design tool

moves the container block coordinate points to the next container block location in quay-parallel direction. For illustration, this process can be shown in Figure 78. Subsequently, the design tool continues the container block generation until it reaches the boundary of the stacking area, where there are no available spaces to generate container blocks in the quay-perpendicular direction. For illustration, this process can be shown in Figure 79.

Equipment type : Straddle Carrier (SC)



Equipment type : Straddle Carrier (SC)

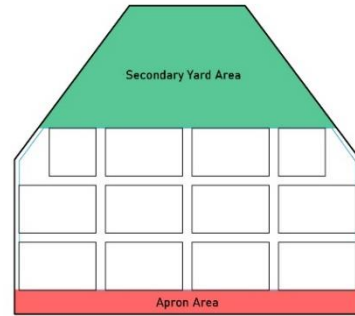


Figure 78 Generating container block and process of decreasing container block length for SC (Author's illustration)

Figure 79 Generating container blocks further and defining secondary yard area for SC (Author's illustration)

The result of the design tool for several shapes of the container terminal for Straddle Carrier is shown in Figure 81 below.

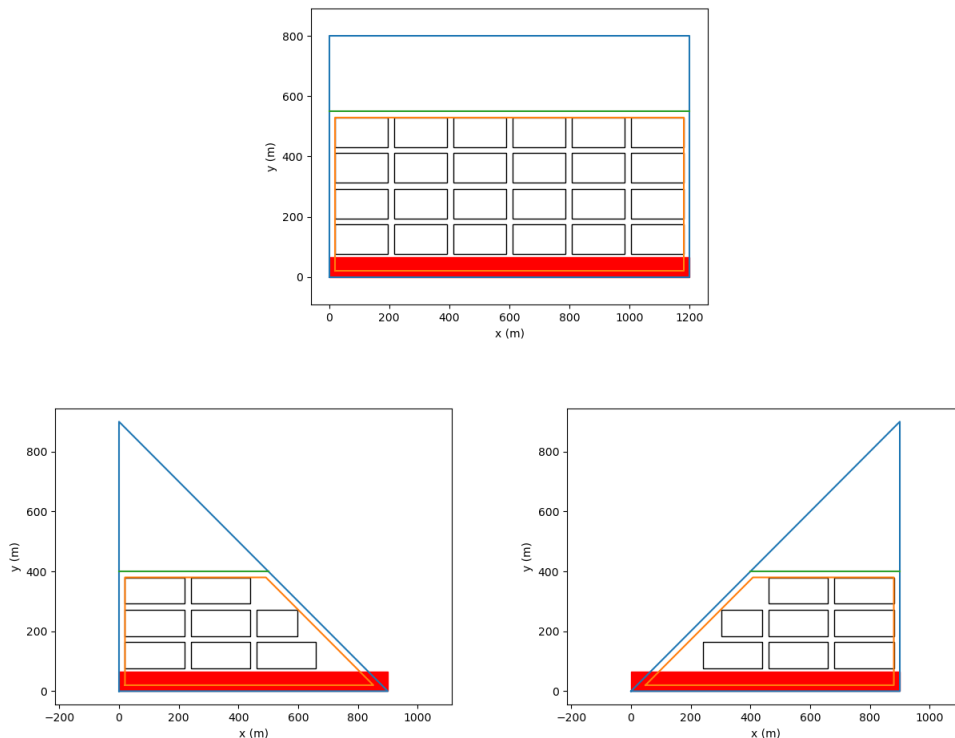


Figure 80 The result of the design tool for Straddle Carrier for several shapes of container terminals (Author's Illustration)

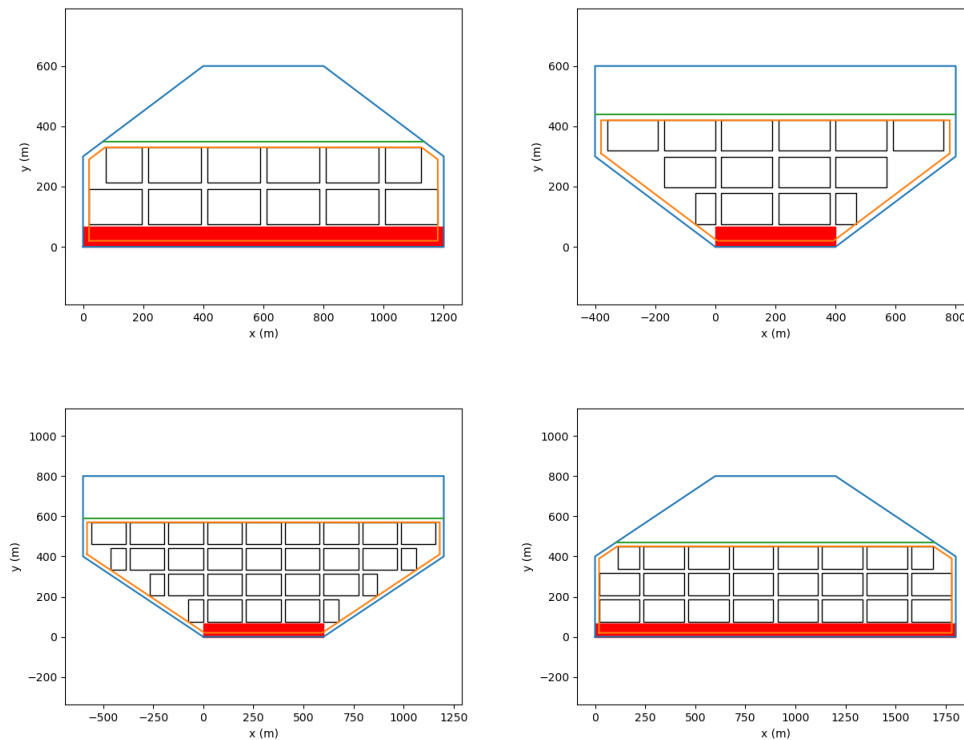


Figure 81 The result of the design tool for Straddle Carrier for several shapes of container terminals [cont.]
(Author's Illustration)

(d) Reach Stacker (RS)

For RS, the block length can be calculated using the equations below.

$$n_{\text{block}} \approx \frac{\text{quay length} - \text{traffic lane}}{\text{max. block length} + \text{traffic lane} + 2 * \text{margin at stack head}}$$

$$\text{block length} = \frac{\text{quay length} - \text{traffic lane}}{n_{\text{block}}} - \text{traffic lane} - 2 * \text{margin at stack head}$$

The algorithm of container block generation for RS-system is very similar to RTG-system; thus, this section only presents the main algorithm, and the detail can be seen in the RTG-system part. After determining the primary yard, the design tool will define the point (in coordinate) that represents the 'first point' for each container block, which the design tool called the 'current_origin', and then the design tool defines the points (in coordinates) that will represent each edge location of a container block, and for illustration, this process can be shown in Figure 82. Then, the design tool checks whether the points of the container block are inside the terminal or not if the container block is inside the terminal the design tool checks whether it is possible to add either 1 TGS or 1 full container block. If it is possible to add either 1 TGS or 1 full container block, the design tool does the process until it reaches the condition when it is not possible anymore and defines the new block configuration, and if it is not possible to add even 1 TGS, the design tool defines the initial block configuration. After that, the design tool generates a container block. For illustration, this process can be shown in Figure 83.

Equipment type : Reach Stacker (RS)

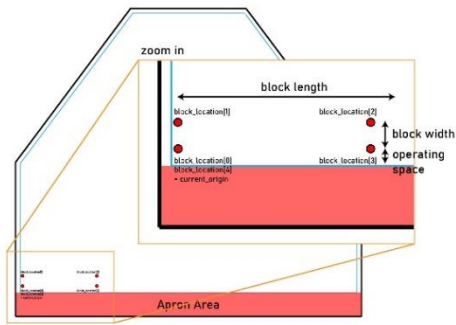


Figure 82 The current_origin and container_block points for RS (Author's illustration)

Equipment type : Reach Stacker (RS)

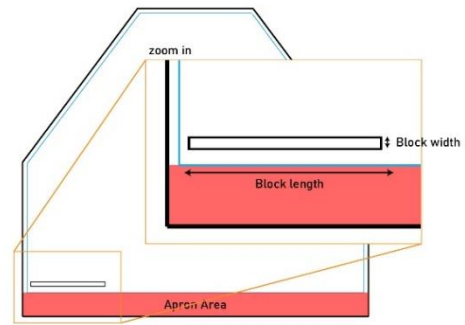


Figure 83 The process of generation of a container block for RS (Author's illustration)

Next, the design tool checks whether there are available spaces at the quay-parallel direction, and if there are available spaces at the quay-parallel direction, the design tool moves the container block coordinate points to the next container block location in the quay-parallel direction. The design tool does the container block generation iteratively until it reaches the condition where there are no available spaces to generate a container block in quay-parallel direction. For illustration, this process can be shown in Figure 84. If there are no available spaces in quay-parallel direction, the design tool checks whether there are available spaces in the quay-perpendicular direction. If there are available spaces in the quay-perpendicular direction, the design tool moves the container block coordinate points to the next container block in the quay-perpendicular direction. For illustration, this process can be shown in Figure 85.

Equipment type : RTG

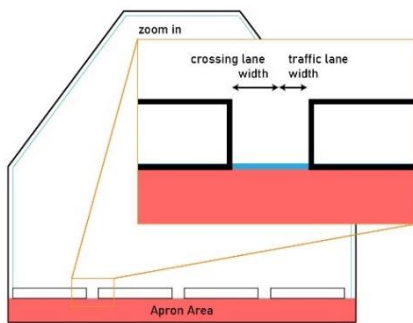


Figure 84 Container block generation in quay-parallel direction for RS (Author's illustration)

Equipment type : RTG

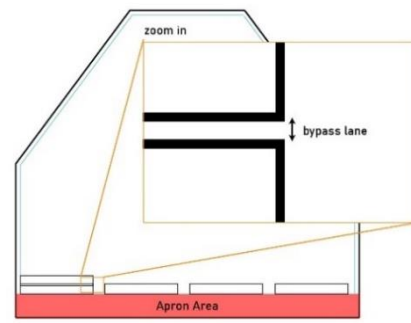
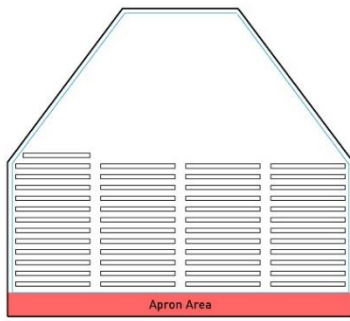


Figure 85 Container block generation in quay-perpendicular direction for RS (Author's illustration)

Then, the design tool checks whether the points of the container blocks are inside the terminal or not iteratively. However, if the container block is outside the terminal, meaning that one of the container block coordinate points is outside the stacking area, the design tool checks whether it is possible to subtract 1 TGS block, and if it is possible to subtract 1 TGS, the design tool does the process until it reaches the condition when it is not possible anymore and defines the new block configuration. For illustration, this process can be shown in Figure 86. Subsequently, the design tool continues the container block generation until it reaches the boundary of the stacking area, where there are no available spaces to generate container blocks in the quay-perpendicular direction. For illustration, this process can be shown in Figure 87.

Equipment type : Reach Stacker (RS)



Equipment type : Reach Stacker (RS)

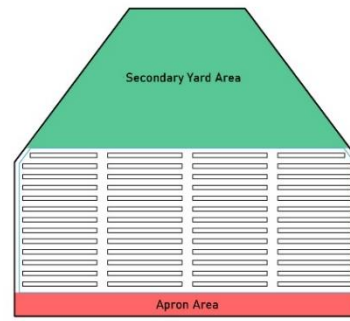


Figure 86 Generating container block and process of decreasing container block length for RS (Author's illustration)

Figure 87 Generating container blocks further and defining secondary yard area for RS (Author's illustration)

The result of the design tool for several shapes of the container terminal for Reach Stacker is shown in Figure 89 below.

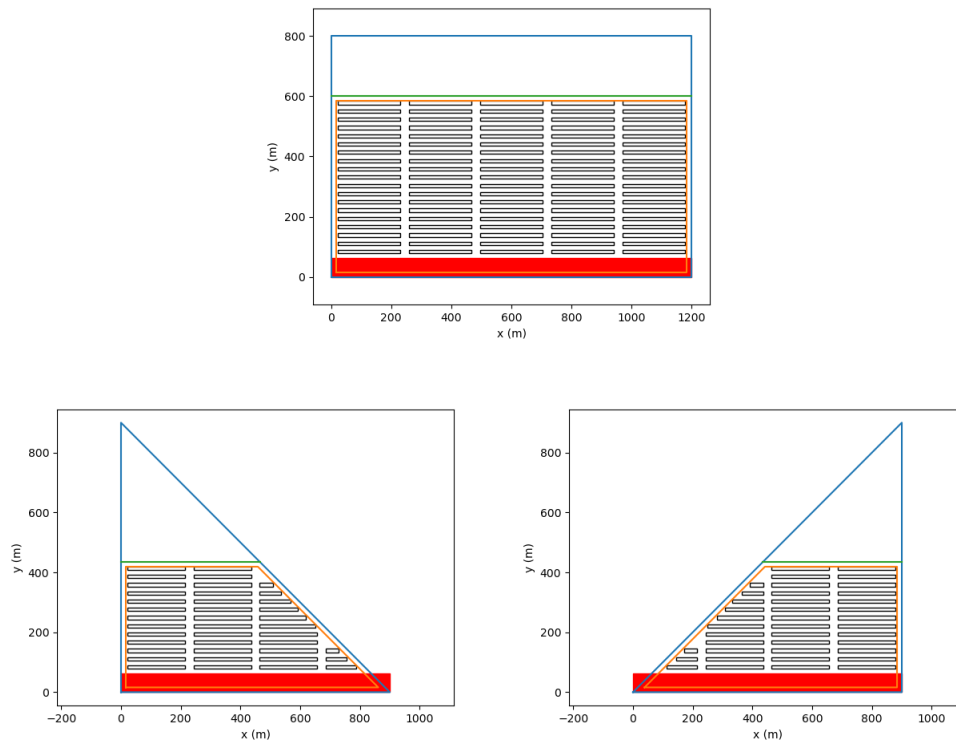


Figure 88 The result of the design tool for Reach Stacker for several shapes of container terminals (Author's Illustration)

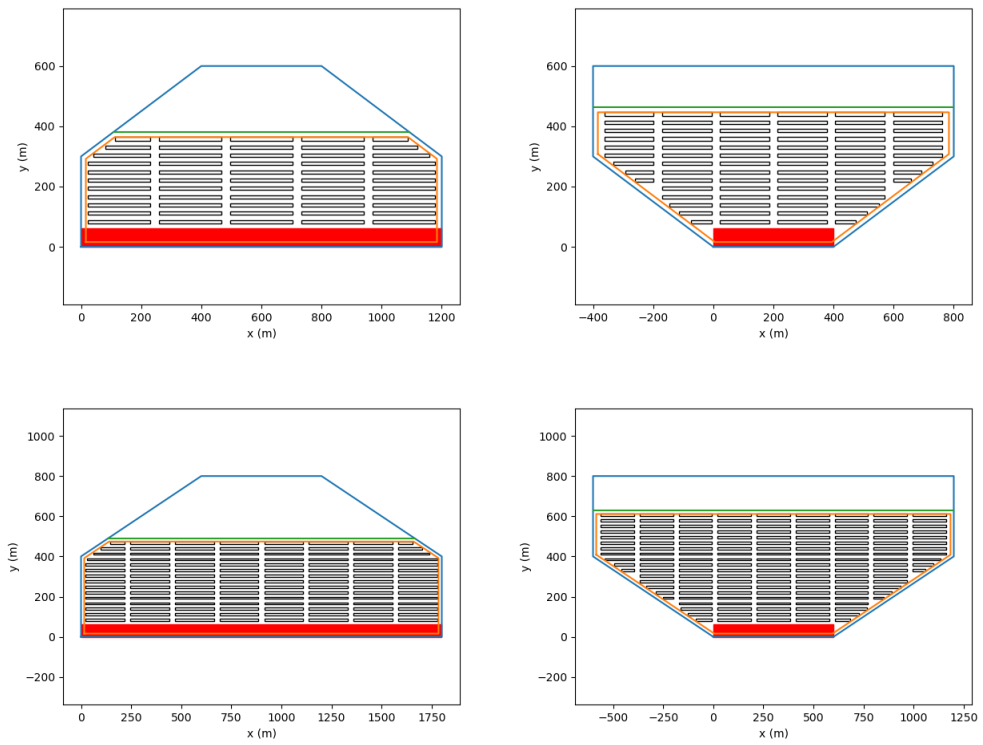


Figure 89 The result of the design tool for Reach Stacker for several shapes of container terminals [cont.]
(Author's illustration)

Appendix D

Design tool validation details

The container terminal layout is generated based on several inputs, including the design rules as the basis of the design tool. In this case, because it is validating the design tool using an existing project, design rules are implemented based on the case.

(a) APM Terminal Rotterdam – Port of Rotterdam (SC Terminal)

Table 16 Inputs and defaults for the design tool validation APMT-R case study

Parameters	Value	Units	Remarks
General			
Throughput	3.250.000	TEU	https://www.apmterminals.com/en/rotterdam/about/our-terminal
Dwell time (laden)	7.5	day	
Dwell time (reefer)	6.5	day	
Stacking height ratio (laden)	0.9	-	
Stacking occupancy (laden)	0.8	-	
Stacking height ratio (reefer)	0.8		
Stacking occupancy (reefer)	0.7		
Peak factor	1.2	-	
Terminal Dimension			
Quay Length	1600	m	(Google Earth, 2018b)
Quay Width	82	m	(Google Earth, 2018b)
Terminal Width	500	m	(Google Earth, 2018b)
Design Rules			
Net TGS	2.44 x 6.10	m	(Koster, 2019)
Gross TGS	2.90 x 6.70	m	(Koster, 2019)
Width traffic lane	10	m	(Google Earth, 2018b)
Margin at stack head	2	m	(Google Earth, 2018b)
Maximum stack length	126	m	Koster (2019)
Maximum stack width	210	m	Koster (2019)
Minimum stack width	0.5 x minimum stack width	m	Author's assumption
Minimum stack length	0.5 x maximum stack length	m	Author's assumption
Stacking orientation	Perpendicular to quay	-	(Google Earth, 2018b)

For the design tool validation, the design tool will use an input for the 'default' set up of container block dimension, as a basis of the container block that is being generated using the design tool. This scenario chooses the following container block dimension : Width = 45 TEU, Length = 20 TEU, Height = 4 TEU, as a default container block dimension. It is chosen based on the container block dimension that is mostly used on the existing terminal, which for the illustration, it can be shown with the green box, and can be shown from the Figure 90.

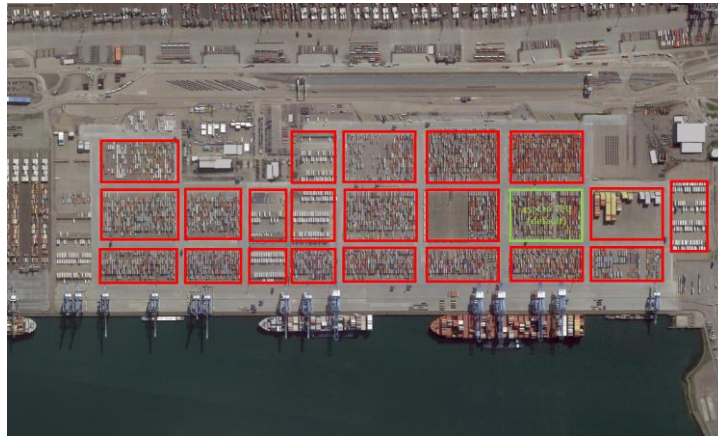


Figure 90 Default Container Block for the first scenario of the APMT-R case study (Author illustration, adapted from (Google Earth, 2018b))

(b) Container Terminal 2 – Port of Jebel Ali (RTG Terminal)

Table 17 Inputs and defaults for the design tool validation CT2 Jebel Ali case study

Parameters	Value	Units	Remarks
General			
Throughput	5.500.000	TEU	(Wang, 2008)
Dwell time (laden)	6.5	day	
Dwell time (reefer)	6.5	day	
Stacking height ratio (laden)	0.8	-	(Ligteringen, 2017)
Stacking occupancy (laden)	0.8	-	(Ligteringen, 2017)
Stacking height ratio (reefer)	0.7		(Ligteringen, 2017)
Stacking occupancy (reefer)	0.7		(Ligteringen, 2017)
Peak factor	1.2	-	(Ligteringen, 2017)
Terminal Dimension			
Quay Length	3000	m	(Google Earth, 2019)
Quay Width	95	m	(Google Earth, 2019)
Terminal Width	520	m	(Google Earth, 2019)
Design Rules			
Net TGS	2.44 x 6.10	m	(Koster, 2019)
Gross TGS	3.15 x 7.05	m	(Google Earth, 2019)
Width traffic lane	10	m	(Google Earth, 2019)

Equipment track	3	m	(Google Earth, 2019)
Vehicle track	10	m	(Google Earth, 2019)
Bypass lane	10	m	(Google Earth, 2019)
Margin at stack head	10	m	(Google Earth, 2019)
Maximum stack length	90	TEU	(Petering & Murty, 2009)
Minimum stack length	18	TEU	(Petering & Murty, 2009)
Stacking orientation	Parallel to quay	-	(Kemme, 2013)

For the design tool validation, the design tool will use an input for the ‘default’ set up of container block dimension, as a basis of the container block that is being generated using the design tool. This scenario chooses the following container block dimension : Width = 45 TEU, Length = 10 TEU, Height = 5 TEU, as a default container block dimension. It is chosen based on the container block dimension that is mostly used on the existing terminal, which for the illustration, it can be shown with the green box, and can be shown in Figure 91.

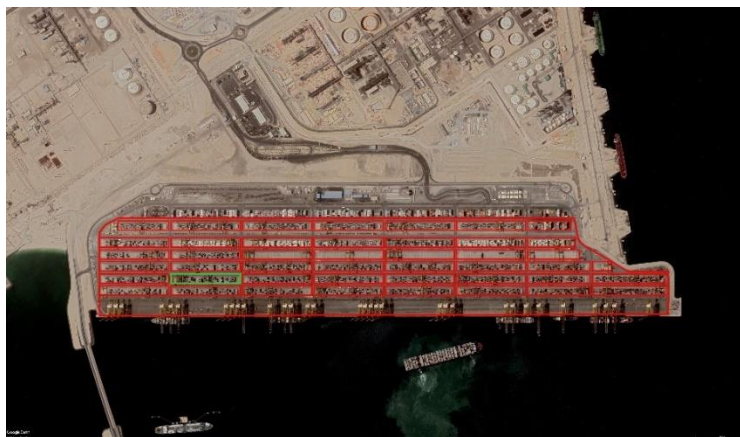


Figure 91 Default Container Block for CT2 Jebel Ali case study (Author illustration, adapted from (Google Earth, 2019))

(c) Rotterdam World Gateway – Port of Rotterdam (RMG Terminal)

To validate the output from the design tool for an RMG system terminal, a case study of Rotterdam World Gateway at Port of Rotterdam, the Netherlands (RWG) is done. In implementing the first scenario, the following block configuration is used as the default block configuration: width = 10 TEU, length = 40 TEU, height = 5 TEU. It is chosen based on the container block dimension that is mostly used on the existing terminal. In detail, input for the design tool and selection of block configuration can be shown in Table 18.

Table 18 Inputs and defaults for the design tool validation RWG Terminal case study

Parameters	Value	Units	Remarks
General			
Throughput	2.350.000	TEU	https://www.rwg.nl/en/terminal/container-s-journey
Dwell time (laden)	9	day	(Ligteringen, 2017)
Dwell time (reefer)	9	day	(Ligteringen, 2017)

Parameters	Value	Units	Remarks
Stacking height ratio (laden)	0.8	-	(Ligteringen, 2017)
Stacking occupancy (laden)	0.7	-	(Ligteringen, 2017)
Stacking height ratio (reefer)	0.8		(Ligteringen, 2017)
Stacking occupancy (reefer)	0.7		(Ligteringen, 2017)
Peak factor	1.2	-	(Ligteringen, 2017)
Terminal Dimension			
Quay Length	1200	m	(Google Earth, 2019)
Quay Width	90	m	(Google Earth, 2019)
Terminal Width	480	m	(Google Earth, 2019)
Design Rules			
Net TGS	2.44 x 6.10	m	(Koster, 2019)
Gross TGS	3.15 x 6.70	m	(Google Earth, 2019)
Width traffic lane	15	m	(Google Earth, 2019)
Equipment track	6	m	(Google Earth, 2019)
Length buffer	36	m	(Google Earth, 2019)
Width buffer	2.5	m	(Google Earth, 2019)
Margin at stack head	5	m	(Google Earth, 2019)
Maximum stack length	48	TEU	(Google Earth, 2019)
Minimum stack length	28	TEU	(Google Earth, 2019)
Stacking orientation	Perpendicular to quay	-	(Kempe, 2013)

RWG container terminal layout, and the generated container terminal layout from the design tool for scenario 1 and scenario 2 can be shown in Figure 92 and Figure 93.

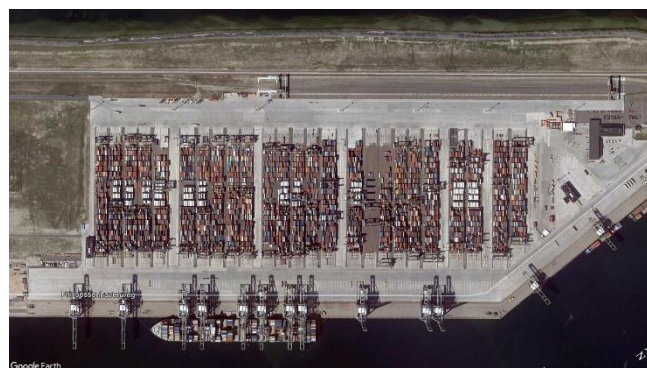


Figure 92 RWG Terminal at Port of Rotterdam
(Google Earth, 2018b)

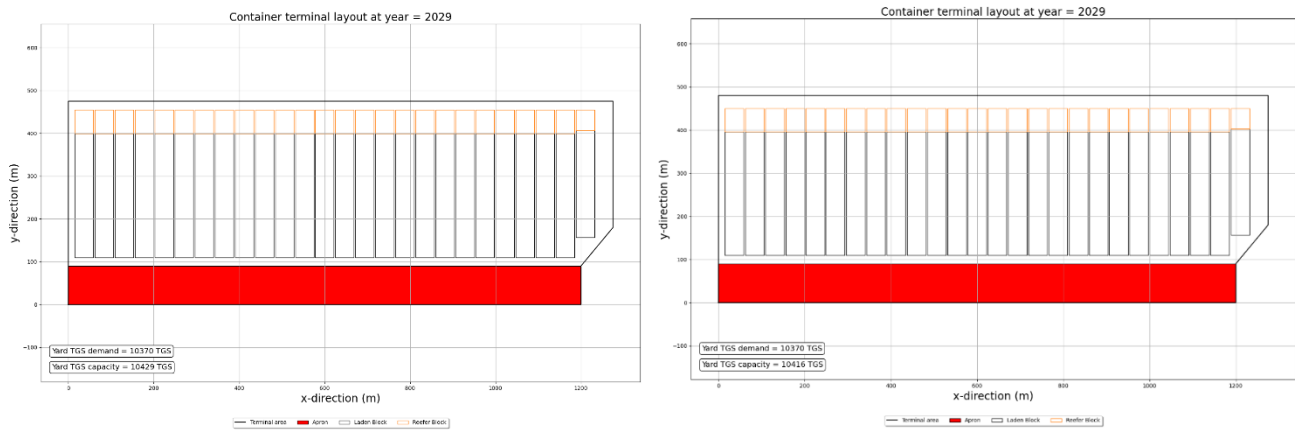


Figure 93 Results of the design tool for the RWG case study: scenario 1 (left) and scenario 2 (right) (Author's illustration)

As we can see from Figure 43 and Figure 44, qualitatively, we can see the layout generated from the design tool with the actual project is very similar, starting from the number of container blocks, the dimension of the container block, ground slots capacity, etc. Furthermore, we can see in detail the comparison of the actual project, and the result from the design tool, which can be shown in Table 19.

Table 19 Comparison of the RWG Terminal with the design tool

Parameters	RWG	Scenario 1	Scenario 2	Unit	Source*
Capacity (TEU)	2.350.000	2.354.155	2.354.155	TEU	(RWG Terminal, 2020)
TEU Ground Slots/TGS	10.099	10.416	10.416	TEU	
Number of Blocks	25	26	26	Blocks	(Google Earth, 2018e)
Block length	40	40	40	TEU	(Google Earth, 2018e)
	340	340	340	m	
Block width	10	10	10	TEU	(Google Earth, 2018e)
	41	43	43	m	
Block height	5	5	5	TEU	(Google Earth, 2018e)
	9,71	9,71	9,71	m	
Stacking density	1000 - 1200	1.055	1.054	TEU/ha	(Saanen & van Valkengoed, 2005), (Brinkmann, 2011), (Kemme, 2013)
	200 - 240	211	211	TGS/ha	
Stacking orientation	Perpendicular to quay	Perpendicular to quay	Perpendicular to quay	ha	

* for the actual project

Based on Table 19, we can see that the terminal capacity resulted from the design tool is slightly larger than the actual project. This means that the design tool can determine container terminal design that satisfies the terminal capacity from the actual project. Other than that, we can see that the number of blocks resulted from the design tool is higher by one container block compared to the actual project. This is because even though the design tool uses the same default container block dimension, the existence of the additional traffic lane in the actual projects

causes the differences in the number of blocks. On its principle, the design tool is governed by the design rules. In the design rules, each container blocks is only separated by one specific margin. This is based on the references from several existing RMG-system terminals, such as such as ECT Delta Terminal in Rotterdam (Netherlands) and the CTA in Hamburg (Germany), and DP World Gateway Antwerp (Belgium). On the one hand, there are several larger margin that separates container blocks on the case of RWG Terminal. This condition makes the design tool produces one additional block because there is available space for the container block. However, it is still considered as a minor difference (<10%) and considered acceptable (Agerschou, et al., 2004). Next, same as the previous case study, the stacking density is validated using three benchmark references, where for RMG-system, the container handling operations usually achieve up to 1200 TEU/ha (Saanen & van Valkengoed, 2005), 1000 TEU/ha for 4 to 5 stacking height (Brinkmann, 2011), and 1,200 TEUs per hectare for 1-over-6 stacking (Kemme, 2013). As can be seen from Table 19, the stacking density from the design tool is 1.055 TEU/ha, and it is considered accurate because it is well within the predetermined bandwidth from the literature, which around 1000-1200 TEU/ha. This concludes that the design tool can determine an accurate container terminal design for an SC-system.

Appendix E

Existing tool detail information: terminal element calculation

1. Quay and STS Crane

For the berth (quay and STS crane) calculation, first, the throughput is translated from TEU to the number of boxes. Then, total loading and unloading time can be calculated based on the crane characteristics and the number of cranes per berth input. The number of berths can be calculated using the total loading and unloading time, the number of expected vessels, the berth occupancy, and the operational hours. Then, this value is rounded up, to ensure that the ships are handled in time. Furthermore, the total number of required berths can now be combined with the predetermined berth length to produce the quay length.

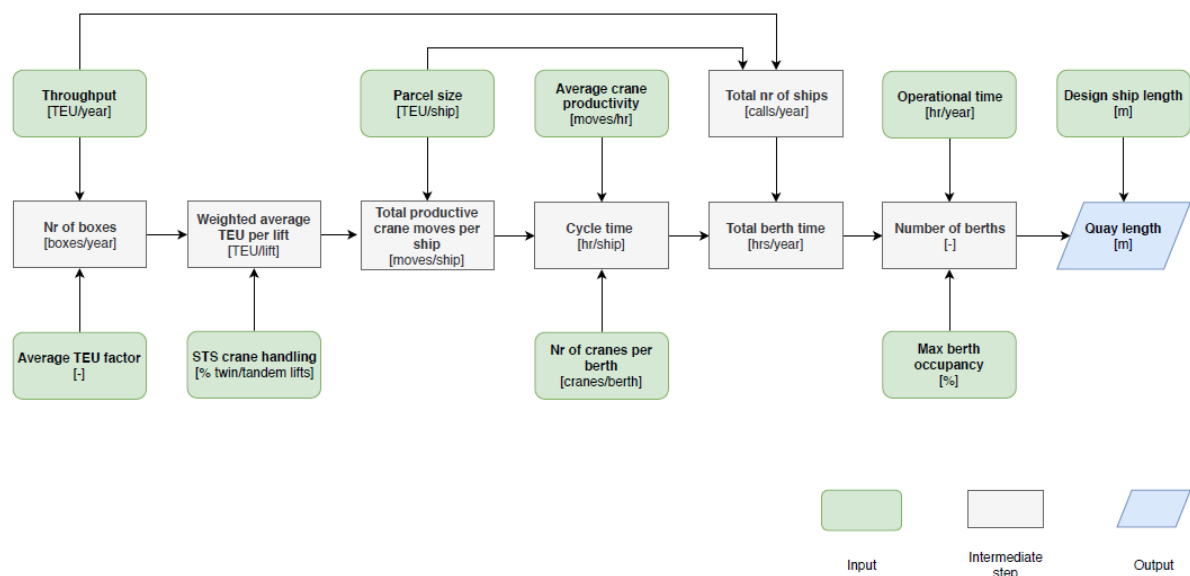


Figure 94 Schematic overview required berth calculation (Koster, 2019)

2. Stack Equipment

The number of stacking equipment is based on the typical ratios from (PIANC, 2014a), where experience has shown that the following ratios of stacking equipment may be appropriate for typical terminals. These ratios are on the basis that the landside movements should always be sufficient to keep pace with ship-to-shore handling operations, that's why it uses the number of equipment in the quay as the basis of the ratios. The ratios shown below exclude the additional numbers of equipment needed to handle containers between the stacks and inland transport facilities. For each combination of the equipment shown, thy typical ratios are as follows:

a. STS + TT (Tractor-Trailer)

For the quay-to-stack transfer, 1 STS required 3 to 6 TTs.

For stacking, 1 STS requires 2 to 3 RTGs/RMGs/or RSs – in general, transshipment terminals require lower ratios than gateway terminals

b. STS + SC

For the quay-to-stack transfer and stacking, 1 STS requires 3 to 5 SCs

3. Gate Calculations

The number of gates limits the number of containers that can enter or exit the terminal per hour. The total number of containers is divided using the modal split into import, export and transshipment. The latter does not leave the terminal by the gate and is therefore excluded from the gate calculations. Furthermore, the inspection time for entering the terminal differs from the inspection time when leaving the terminal. This distinction leads to separate calculations for the number of entry lanes and the number of exit lanes. Moreover, the number of container moves is corrected with a factor due to the fact that some trucks leave or enter the terminal with two container boxes and some leave or enter without container boxes. This ratio is accounted for in this factor. Also, to avoid costly congestion, the number of required lanes is calculated for the truck moves during a peak hour on a peak day.

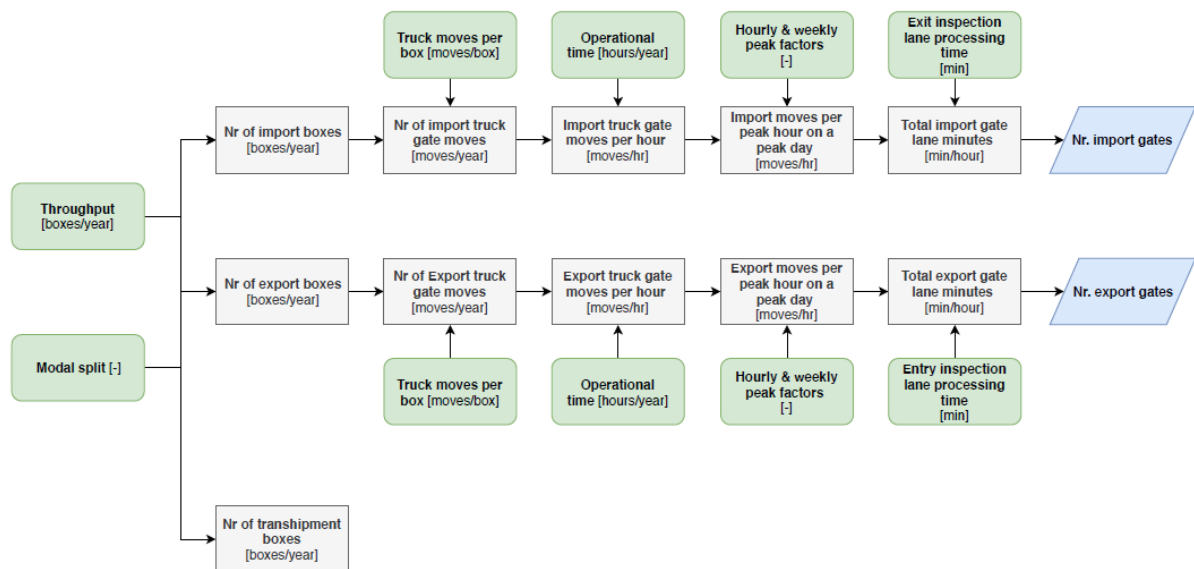


Figure 95 Schematic overview gate calculation (Koster, 2019)

Existing tool detail information: terminal element cost calculation

The tool can assess the concept terminal on several cost items, e.g., the CAPEX, OPEX or NPV. These three outputs allow the terminal planner to evaluate the various design options, based on their costs. The NPV consists of the design's CAPEX and OPEX over a period of time, corrected for the discount rate. The CAPEX is estimated based on the calculated terminal elements from the layout and the unit costs. The unit costs for the CAPEX calculation can be shown in Table 20.

Table 20 Default unit rate of CAPEX calculation
(Koster, 2019)

Parameters	Value	Unit	Unit
Pavement Apron	125	USD/m ²	Pavement underneath the apron area
Pavement Stacking Area	100	USD/m ²	Pavement underneath the stacks area
Drainage	15	USD/m ²	Takes the complete terminal area into account
STS Crane equipment	10.000.000	USD	Price per STS crane equipment unit
RTG equipment	1.500.000	USD	Price per RTG equipment unit
RMG equipment	2.500.000	USD	Price per RMG equipment unit
SC equipment	2.000.000	USD	Price per SC equipment unit
RS equipment	500.000	USD	Price per RS equipment unit
Tractor	85.000	USD	Price per tractor equipment unit
Trailer	30.000	USD	Price per trailer equipment unit
Mobilization	5	%	Percentage of acquisition costs
Maintenance tool	10.000.000	USD	A fixed price for medium to large-sized terminal
Workshop and repair costs	1.000	USD/m ²	Including the buildings and required services
Workshop and repair area	650	m ²	For every 500.000 TEU/year throughput an additional 650m ² required
Lighting mast	30.000	USD	40 meter high
Reefer rack	3.500	USD	For a regular 4-high reefer rack, including the costs for the steel structure, the foundation, power connection, and mobilization costs

On the other hand, The tool's calculated OPEX consists of the following cost items: (1) Labour; (2) Electrical power; (3) Fuel; (4) Maintenance. The OPEX calculations require the unit rate of these four elements (e.g. fuel price [USD/l]) and the consumption rate (e.g. fuel consumption per year). The unit rate is considered fixed, but the consumption rate is different per terminal and depends on the design choices. For example, the fuel costs can be calculated by multiplying the number of box moves with the fuel consumption per box move and the unit rate of fuel. The complete overview of unit rates and cost parameters can be found in Appendix F. The unit costs for the OPEX calculation can be shown in Figure

Table 21 Default unit rate of OPEX calculation

Parameters	Value	Unit	Unit
Household moves	0.1	move/box	Additional moves per box over the quay
Digout moves	20	%	Additional digout moves as a percentage of box move over the quay and household moves
Fuel consumption	1.1	l/box move	
Maintenance	1.3	%	Maintenance costs as a percentage of acquisition costs
Lighting	1.0	W/m ²	Total stack area
Reefer	4.0	kW/reefer slot	Electrical power only required when reefer present at slot
Salary	35.000	USD/FTE/year	Salary for all blue-collar personnel at the terminal
Required RTG drivers	1.3	FTE/shift	Full-time operations, accounting for leaving days
Required planning personnel	8	FTE/shift	Full-time operations, accounting for leaving days

Appendix F

Overview of Python module: Shapely

Shapely is a Python package for the manipulation and analysis of two-dimensional geospatial geometries. Shapely is based on the GEOS library, which implements a wide range of geospatial data manipulations in C++. GEOS is itself based on a library called the Java Topology Suite, which provides the same functionality for Java programmers. Shapely provides a Pythonic interface to GEOS, which makes it easy to use these manipulations directly from your Python programs.

The Shapely library is organized, as can be shown in Figure 96:

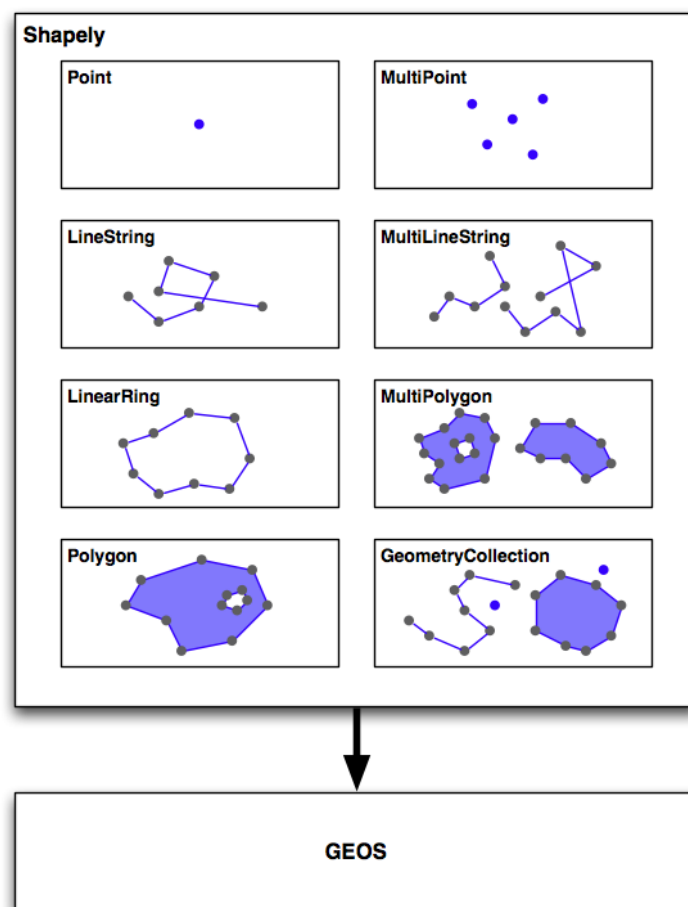


Figure 96 Organization of the shapely library (Westra, 2013)

Shapely itself consists of eight major classes, representing different types of geometrical shapes:

- The **Point** class represents a single point in space. Points can be two-dimensional (x, y), or three-dimensional (x, y, z).
- The **LineString** class represents a sequence of points joined together to form a line. LineStrings can be simple (no crossing line segments) or complex (where two line segments within the LineString cross).
- The **LinearRing** class represents a line string that finishes at the starting point. The line segments within a LinearRing cannot cross or touch.
- The **Polygon** class represents a filled area, optionally with one or more "holes" inside it.
- The **MultiPoint** class represents a collection of Points.
- The **MultiLineString** class represents a collection of line strings.
- The **MultiPolygon** class represents a collection of Polygons.
- The **GeometryCollection** class represents a collection of any combination of Points, LineStrings, LinearRings, and Polygons.

As well as being able to represent these various types of geometries, Shapely provides a number of methods and attributes for manipulating and analyzing these geometries. For example, the LineString class provides a *length* attribute that equals the length of all the line segments that make up the LineString, and a *crosses()* method that returns true if two LineStrings cross. Other methods allow you to calculate the intersection of two polygons, dilate or erode geometries, simplify a geometry, calculate the distance between two geometries, and build a polygon that encloses all the points within a given list of geometries (called the *convex_hull* attribute).

Note that Shapely is a spatial manipulation library rather than a geospatial manipulation library. It has no concept of geographical coordinates. Instead, it assumes that the geospatial data has been projected onto a two-dimensional Cartesian plane before it is manipulated, and the results can then be converted back into geographic coordinates if desired.

The following program creates two Shapely geometry objects, a circle, and a square, and calculates their intersection.

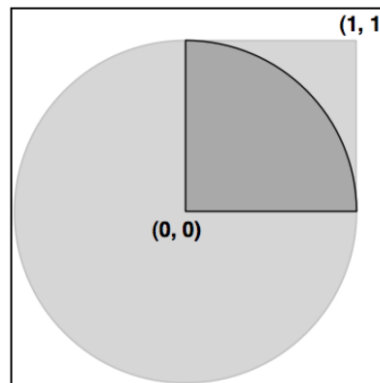


Figure 97 Examples for shapely code
(Westra, 2013)

The intersection will be a polygon in the shape of a quarter circle, as indicated by the dark grey portion of the Figure 97:

```
import shapely.geometry
pt = shapely.geometry.Point(0, 0)
circle = pt.buffer(1.0)
square = shapely.geometry.Polygon([(0, 0), (1, 0),
(1, 1), (0, 1),
(0, 0)])
intersect = circle.intersection(square)
for x,y in intersect.exterior.coords:
    print x,y
```

Notice how the circle is constructed by taking a Point geometry and using the *buffer()* method to create a Polygon representing the outline of a circle.

Appendix G

Code archive

This section provides links to the Python code developed within this study. The code has been published using OpenTISim. The code is also available through GitHub. These two can be attained by following the QR codes below.

OpenTISIM



Figure 98 OpenTISim QR code

GitHub



Figure 99 GitHub QR code