

Summer 2018

Sequence-Based Simulation-Optimization Framework With Application to Port Operations at Multimodal Container Terminals

Mariam Aladdin Kotachi
Old Dominion University

Follow this and additional works at: https://digitalcommons.odu.edu/emse_etds

 Part of the [Operational Research Commons](#), and the [Systems Engineering Commons](#)

Recommended Citation

Kotachi, Mariam A.. "Sequence-Based Simulation-Optimization Framework With Application to Port Operations at Multimodal Container Terminals" (2018). Doctor of Philosophy (PhD), dissertation, Engineering Management, Old Dominion University, DOI: 10.25777/2j0v-0v84
https://digitalcommons.odu.edu/emse_etds/32

This Dissertation is brought to you for free and open access by the Engineering Management & Systems Engineering at ODU Digital Commons. It has been accepted for inclusion in Engineering Management & Systems Engineering Theses & Dissertations by an authorized administrator of ODU Digital Commons. For more information, please contact digitalcommons@odu.edu.

**SEQUENCE-BASED SIMULATION-OPTIMIZATION FRAMEWORK WITH
APPLICATION TO PORT OPERATIONS AT MULTIMODAL CONTAINER
TERMINALS**

by

Mariam Aladdin Kotachi
B.S. May 2011, University of Texas Arlington
M.S. May 2014, Old Dominion University

A Dissertation Submitted to the Faculty of
Old Dominion University in Partial Fulfillment of the
Requirements for the Degree of

DOCTOR OF PHILOSOPHY

ENGINEERING MANAGEMENT

OLD DOMINION UNIVERSITY

August 2018

Approved by:

Ghaith Rabadi (Director)

Mamadou Seck (Member)

Resit Unal (Member)

Dean Chatfield (Member)

ABSTRACT

SEQUENCE-BASED SIMULATION-OPTIMIZATION FRAMEWORK WITH APPLICATION TO PORT OPERATIONS AT MULTIMODAL CONTAINER TERMINALS

Mariam Aladdin Kotachi
Old Dominion University, 2018
Director: Dr. Ghaith Rabadi

It is evident in previous works that operations research and mathematical algorithms can provide optimal or near-optimal solutions, whereas simulation models can aid in predicting and studying the behavior of systems over time and monitor performance under stochastic and uncertain circumstances. Given the intensive computational effort that simulation optimization methods impose, especially for large and complex systems like container terminals, a favorable approach is to reduce the search space to decrease the amount of computation.

A maritime port can consist of multiple terminals with specific functionalities and specialized equipment. A container terminal is one of several facilities in a port that involves numerous resources and entities. It is also where containers are stored and transported, making the container terminal a complex system. Problems such as berth allocation, quay and yard crane scheduling and assignment, storage yard layout configuration, container re-handling, customs and security, and risk analysis become particularly challenging.

Discrete-event simulation (DES) models are typically developed for complex and stochastic systems such as container terminals to study their behavior under different scenarios and circumstances. Simulation-optimization methods have emerged as an approach to find optimal values for input variables that maximize certain output metric(s) of the simulation. Various traditional and nontraditional approaches of simulation-optimization continue to be used to aid in decision making.

In this dissertation, a novel framework for simulation-optimization is developed, implemented, and validated to study the influence of using a sequence (ordering) of decision variables (resource levels) for simulation-based optimization in resource allocation problems. This approach aims to reduce the computational effort of optimizing large simulations by breaking the simulation-optimization problem into stages.

Since container terminals are complex stochastic systems consisting of different areas with detailed and critical functions that may affect the output, a platform that accurately simulates such a system can be of significant analytical benefit. To implement and validate the developed framework, a large-scale complex container terminal discrete-event simulation model was developed and validated based on a real system and then used as a testing platform for various hypothesized algorithms studied in this work.

Copyright, 2018, by Mariam Aladdin Kotachi, All Rights Reserved.

This dissertation was only made possible by the guidance, strength and inspiration that Allah Almighty blessed me with, Alhamdulillah.

This effort is dedicated to:

My Parents, Mayada and Aladdin, for the sacrifices you made, for leaving your home in search for a better life and education for us, for the optimism regardless of the struggles and dead ends you encountered, for always putting us first and for every selfless decision. I am here because of everything you did, and I am eternally grateful.

To my companion in this journey since the very beginning My Husband Mohammad, for truly believing that I can achieve this and making me see it, for the positive energy, the daily pep talk, for knowing exactly what to say when I am down, and for always being there for me. You are solely responsible for this milestone in my life.

To my Sister Maisa and my Brother Ahmed, to my adorable Niece Jude, for the encouragement you always send from far away, for making me laugh, and for reminding me of the loving family I have.

To my second family that I have been blessed with, firstly, Khalto Suzan and Amo Fayez, for teaching me how to be humble and truthful and for all the concerned prayers and love you always have for me. To everyone in this cherished family that always cheered me up, prayed for me, and made life sweeter.

To my Grandma Huda and Aunt Alia, for all your help throughout my learning years and for showing me how to become a hard worker and an achiever regardless of life's challenges.

To the first person who inspired me to pursue this dream, my best friend Sawsan Abutabenjeh, thank you for the honest advice and for being a role model to follow.

ACKNOWLEDGMENTS

I would like to thank my advisor Dr. Ghaith Rabadi for the countless hours of assistance and guidance, for the daily meetings, and for the late hours of proofreading and editing. Thank you for believing in me and guiding me throughout this journey, I couldn't have asked for a better mentor.

I would like to thank the members of my dissertation committee: Dr. Mamadou Seck for coming to this research's aid when it hit a rough patch and for the optimism and faith; Dr. Resit Unal and Dr. Dean Chatfield for their valuable insights and criticisms.

I would also like to mention and thank the Engineering Management and Systems Engineering Department and all its professors, faculty, staff and research assistants for providing me with an encouraging and constructive environment that helped pave the way for this accomplishment.

Lastly, I would like to extend my thanks and acknowledgement to all my friends and family members who all directly or indirectly helped make this dream a reality.

This research was made possible by NPRP Grant No. NPRP 7-796-2-297 from the Qatar National Research Fund (a member of The Qatar Foundation).

TABLE OF CONTENTS

| | Page |
|--|------|
| LIST OF TABLES | ix |
| LIST OF FIGURES | x |
| Chapter | |
| 1. INTRODUCTION | 1 |
| 1.1 Operational Challenges at Container Terminals | 2 |
| 1.2 Motivation | 3 |
| 1.3 Problem Statement | 4 |
| 1.4 Proposed Solution and Research Objectives | 5 |
| 1.5 Contributions | 7 |
| 1.6 Dissertation Organization | 9 |
| 2. BACKGROUND | 10 |
| 2.1 Container Terminals and Ports | 10 |
| 2.2 Port Resources and Entities | 12 |
| 2.3 Defining Maritime Terminal Problems | 16 |
| 2.4 Discrete-Event Simulation and Arena | 18 |
| 2.5 Simulation-Based Optimization | 19 |
| 2.6 Evolutionary-Based Optimization | 20 |
| 2.7 Case Study: Hamad Port in Qatar | 20 |
| 3. RELATED WORK | 22 |
| 3.1 Operations Research Methods for Maritime Operations | 22 |
| 3.2 Simulation Methods for Maritime Operations | 42 |
| 3.3 Simulation Optimization Methods for Maritime Operations | 51 |
| 3.4 Search Space Reduction for Simulation Optimization | 61 |
| 4. SIMULATION MODEL DESIGN AND DEVELOPMENT | 63 |
| 4.1 Model Design and Construction | 63 |
| 4.2 Data Collection | 66 |
| 4.3 Modeling Highlights | 68 |
| 4.4 Input Analysis and Data Fitting | 73 |
| 4.5 Validation: Simulation vs. Historical Data | 76 |
| 4.6 Sensitivity Analysis | 77 |
| 5. SEQUENCE-BASED SIMULATION OPTIMIZATION FRAMEWORK | 82 |
| 5.1 Simulation-Optimization Infrastructure | 82 |
| 5.2 Evolutionary Algorithm-Based Optimization (Non-Sequence-Based) | 86 |
| 5.3 Sequence-Based Evolutionary Algorithm for Optimization | 89 |
| 5.4 Alternative Validation Scenario: Increased Vessel Arrivals | 93 |

| Chapter | Page |
|---|------|
| 6. COMPUTATIONAL EXPERIMENTS AND RESULTS..... | 94 |
| 6.1 Results: First Scenario | 94 |
| 6.2 Results: Second Scenario..... | 98 |
| 7. CONCLUSIONS AND FUTURE WORK..... | 101 |
| REFERENCES | 103 |
| VITA..... | 119 |

LIST OF TABLES

| Table | Page |
|--|------|
| 1. Simulation results vs. historical data | 77 |
| 2. Vessel arrival is at the same level as in 2014..... | 78 |
| 3. Vessel arrival level of 2014 is increased by 25% | 78 |
| 4. Terminal section, resources and levels | 87 |
| 5. The sequence of the last generation | 92 |
| 6. Population data after 20 births (generation 167) of the traditional (non-sequence-based) evolutionary optimization algorithm..... | 95 |
| 7. Scenario 1: Result of the traditional evolutionary method | 95 |
| 8. Scenario 1: Solution string for sequence-based method | 96 |
| 9. Scenario 1: Result of the sequence-based evolutionary method..... | 97 |
| 10. Scenario 1: Results comparison for the two methods | 97 |
| 11. Scenario 2: Result of the traditional evolutionary method | 98 |
| 12. Scenario 2: Solution string for sequence-based method | 99 |
| 13. Scenario 2: Result of the sequence-based evolutionary method..... | 99 |
| 14. Scenario 2: Results comparison for the two scenarios..... | 100 |

LIST OF FIGURES

| Figure | Page |
|---|------|
| 1. Port System with main subsystems | 2 |
| 2. Container terminal of the port of Houston [10]. | 12 |
| 3. Main resources and entities at a port: (a) containers, (b) quay cranes, (c) RTG, (d) yard trucks, and (e) cargo vessel [10]. | 15 |
| 4. Studies that used optimization methods vs. studies that used simulation methods for addressing container terminal problems. | 23 |
| 5. Previous work for integrating simulation and optimization for addressing problems in maritime operations. | 52 |
| 6. Conceptual model. | 65 |
| 7. Simulation Model Building Blocks | 69 |
| 8. Berth availability and seizing..... | 70 |
| 9. Container yard design. | 71 |
| 10. Vessel Interarrival times distribution fitting of 6-month historical data (data included in the model). | 75 |
| 11. Vessel Interarrival times distribution fitting of 6-month historical data (data used for testing)..... | 76 |
| 12. Regular and increased arrivals, TAT Vs. YTs..... | 79 |
| 13. Regular and increased arrivals, # Containers Vs. YTs. | 80 |
| 14. Regular and increased arrivals, TAT Vs. RTGs. | 80 |
| 15. Regular and increased arrivals, # Containers vs. RTG. | 81 |
| 16. Simulation Optimization Framework | 83 |
| 17. Variables' Relationship Graph..... | 92 |

CHAPTER 1

INTRODUCTION

Container terminals are considered complex systems since they consist of different functional areas, each with detailed and critical roles. Due to the challenges involved in managing container terminals, researchers rely on simulation and optimization methods to aid in decision making. One of the main challenges involved with simulation-optimization is the lengthy computational times. Finding optimal solutions to complex problems is, in general, time consuming even for objective functions that are easily evaluated. The computation becomes much more extensive when running a simulation for longer times and for many replications to evaluate viable solutions produced by an optimization algorithm.

This work presents a novel method for simulation-optimization that incorporates the sequence in which the decision variables (resource levels) are optimized. It is hypothesized that implementing such a sequence will reach a comparable solution in less computation time than the traditional method of optimizing simulations and will reduce the search space and improve the efficiency of the optimization process.

To illustrate the potential and inspiration of this work, this chapter presents, in its outset, a brief background of the study specifically focusing on operational challenges usually faced at container terminals. This is followed by the motivation behind this dissertation, the problem statement, as well as the research objectives. A summary of the contributions and an outline of the dissertation are also provided.

1.1 Operational Challenges at Container Terminals

Container terminals can be generally divided into several sections or subsystems where the main operations take place, and where delays and bottlenecks occur. Fig. 1 identifies the main subsystems, which are the vessel area, storage area and the terminal gate area. The vessel area is where the loading and unloading of containers takes place, the storage area is where containers are stored temporarily, and the gate area is the entrance to all external trucks. A more detailed description of the terminal operations and activities is provided in Chapter 2 of this dissertation.

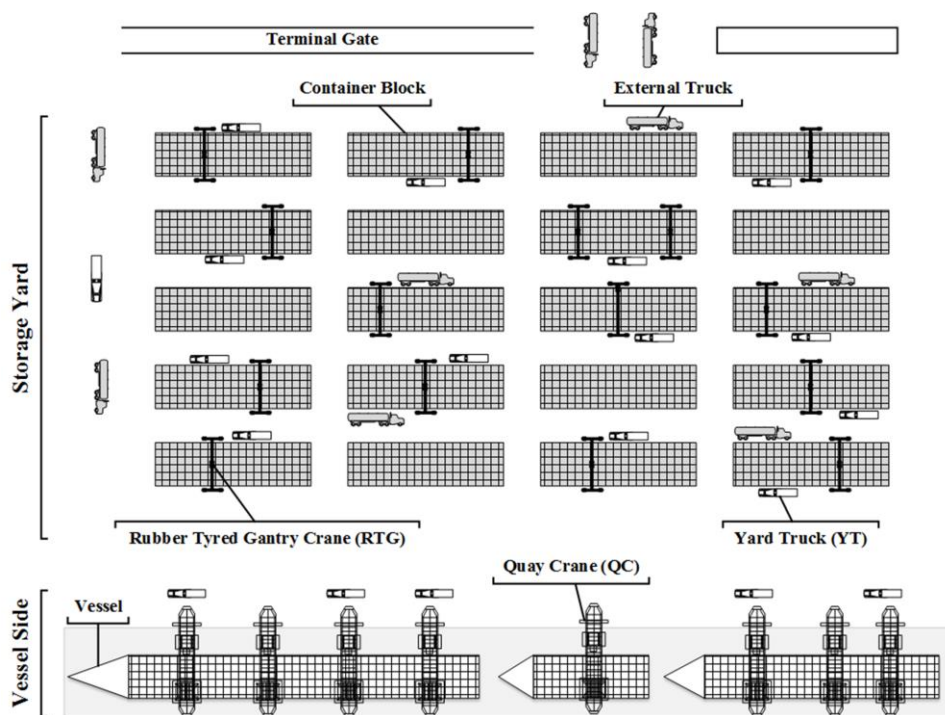


Fig. 1. Port System with main subsystems.

Every container terminal has three main resources that control a container, whether by loading and unloading it or transporting it inside the terminal. These resources are cranes that work on the vessel called STS Quay Cranes (QC), cranes that work in the storage yard called Yard

Cranes (YC) and finally transporters that transport containers from and to the storage yard, called Yard Trucks (YT).

To improve the terminal's performance, all these resources must be working efficiently; this is particularly true of both YCs and YTs, which must serve the QC effectively, given that the QCs are the main resource in container terminals. However, this process becomes complex and problematic over time since these resources work at different speeds and capacities, yet they must interact by exchanging containers.

Some of these differences are the handling speed of YCs which is around half of that of the QCs. YTs move great distances compared to both QCs and YCs. Their movement naturally differs based on the operator, traffic situations at the terminal, and human error. Both QCs and YCs are dependent on the availability of YTs to pick up and transfer a container. Finally, the workload is unevenly distributed over time since container vessels arrive at the terminal at different times during the day, which might congest the resources at some point and leave them idle at another [1].

1.2 Motivation

This dissertation was motivated by a research project on simulating the logistics of the first newly constructed container terminal system of Hamad's new port of Qatar, which has been under construction since 2010 and has recently started its operations at Terminal 1 in September 2017. The objective is to develop, test and validate simulation and simulation-optimization models for the container terminal, study the effectiveness of its operational policies and resource allocation, then generalize the applicability of the developed models.

1.3 Problem Statement

Much of the conducted research related to decision support systems for container terminals focused primarily on one section of the terminal, such as the container yard or the berths alone; isolating it from other influential resources such as Quay Cranes (QC), Yard Cranes (YC) and Yard Trucks (YT). Collectively, very little research focused on employing an integration of discrete-event simulation models with optimization to address allocation and assignment of involved resources as well as their interdependencies in a stochastic container flow system at a mega state-of-the-art container terminal. The literature gaps that this dissertation attempts to fulfill are described here:

- The first aspect that the current literature lacks is the inclusion of **uncertainty** and **stochasticity** in the model [2, 3]. Many optimization studies applied to container terminals are deterministic not stochastic. Using deterministic input data generally encapsulates several assumptions and can provide limited knowledge about the system's behavior. Container terminal systems naturally have a level of randomness that is captured much more realistically in a stochastic model.
- The second aspect is the fact that several research papers typically model one specific and isolated section of the terminal with limited consideration to other components, resources, and processes and eventually its effect on the optimized solution. There is a great deal of **interdependency** within the terminal processes and the advancements in one subsystem will not necessarily result in an overall improvement since the influence of the change on other components of the system should be considered. Traditionally, simulation-based optimization approaches treat the simulation model as a black box and consequently make no assumption about the internal structure of the search space,

treating decision variables in a non-differentiated way. Realistically, however, decision spaces for complex systems tend to have a complex internal structure such that the mutual influence of variables is not always symmetrical [4-6].

- Finally, studies in the existing literature did not develop a simulation model that considers all terminal activities and major resources, thus providing a **complete system** [7]. As in the second aspect mentioned earlier, most studies focus on solving one or two problems relating to one or two resources in one location of the container terminal.

In light of the above, a platform that allows exploration of how different container terminal resources allocation and assignment patterns affect the long-term performance of the container terminal is of great benefit; an elaborate and conclusive discrete-event simulation model can provide such a platform. More importantly, there is an apparent opportunity to supplement the current literature with *a framework that performs a thorough analysis of the components and resources of a given system, investigates the interactions and interdependencies among available resources, then implements a simulation-based optimization methodology to perform resource allocation governed by the level of interaction and influence that each resource exerts on the system*. This dissertation aims to address this need through developing and implementing such a framework on a container terminal system before extending it to a more generic level that is applicable to other systems and fields.

1.4 Proposed Solution and Research Objectives

To address the issues stated in the previous section, a container terminal discrete-event simulation model is proposed to provide a platform that can be used with various optimization

algorithms to improve performance in a container terminal. This simulation platform will allow studying and evaluating various scenarios introduced via optimization algorithms and changing different parameters to achieve improvements to the terminal performance. This model will provide solutions for modern container terminal problems like bottlenecks, resources allocation, and assignment.

The proposed simulation model will consider major terminal operations under uncertain and stochastic input and behavior. Additionally, it will capture interdependency between various components of the model and allow for studying the effects of optimizing these components in a sequence-based manner.

Optimization algorithms will be used to search the solution space of variables associated with resource allocation, whereas the simulation model will be used for solution evaluation by observing performance metrics such as container throughput and resources utilization. The combination of simulation and optimization approaches are anticipated to handle the stochastic and dynamic nature of this real-world complex system.

To implement and validate this simulation-optimization framework, this research will also consider a case study of an actual and newly constructed container terminal of Hamad port in Qatar. This system will be the basis for configuring the developed simulation-optimization framework and methodology. Results relevant to decision making in the port will be generated from the simulation and optimization for analysis. Furthermore, output analysis will be performed to create a platform for operational design, scenario analysis, as well as inferring relevant information and drawing conclusions.

More specifically, this dissertation will address the following objectives:

- **Objective 1:** Developing a discrete-event simulation model that replicates the operations of a mega seaport container terminal.
- **Objective 2:** Developing an optimization module that can communicate with and control the simulation model. The optimization method in this case is *Evolutionary Algorithms*, which can be replaced by another method.
- **Objective 3:** Investigating the influence of a sequence-based optimization approach to reduce the search space in a simulation-optimization problem.
- **Objective 4:** Implementing and validating the developed integrated simulation-optimization framework on a case study system with real historical data by obtaining and evaluating the sequence for optimization.

1.5 Contributions

In generic terms, the contribution of this work is twofold. The first is to develop a discrete-event simulation model for a modern container terminal, including operations, logistics, processes and resources to be able to study their impact on the overall performance of the terminal and not on a specific part of the terminal. The second is to introduce a sequence-based simulation-optimization methodology not addressed yet by the literature, utilizing the developed terminal simulation model. This will be evaluated by executing this methodology to address critical problems typically encountered in container terminals. The Hamad container terminal, a modern terminal located in Doha, Qatar will be considered as a case study to evaluate the conducted research.

More details of this dissertation's contributions are described in the following sections.

1.5.1 A platform for modeling and simulating container flow at mega seaports

The first contribution of this work is the design, development, and validation of an elaborate discrete-event simulation model that can demonstrate the effects of operative-level decisions and changes of a container terminal under uncertainty. This simulation model is designed to provide the following:

- An ability to evaluate various resource allocation scenarios.
- An ability to receive external input that specify internal parameters. This input can be obtained either from a user or through an external computer program.
- An ability to dynamically report relevant performance metrics intermittently during and at the conclusion of simulation runs.

1.5.2 A sequence-based framework for search space reduction for simulation-optimization

This dissertation proposes the development of a system that hypothesizes and demonstrates that the relative order (sequence) in which variables are optimized plays a significant role in identifying promising search regions to exploit, and in turn, reduces the amount of computations necessary to reach optimal or near-optimal solutions.

This work discusses optimizing large simulations by breaking the simulation-optimization problem into two stages: in the first stage, an evolutionary method is proposed to identify the most promising variable sequences, i.e., the order in which input variables should be optimized, and in the second stage a heuristic approach is used to exploit the search space determined by this identified sequence. This proposed method is implemented and evaluated on a container terminal discrete-event simulation model due to its high-level complexity.

1.6 Dissertation Organization

The work in this dissertation is organized in seven chapters. Chapter 2 introduces the terminology and technical background relating to container terminals, discrete-event simulation, and simulation-based optimization. Chapter 3 provides an analysis of existing research in the literature that addressed similar challenges in container terminals through employing simulation and/or optimization methods.

The design, development, and validation approaches used to construct a discrete-event simulation model for a container terminal is described in Chapter 4, addressing Objective 1 of Section 1.4 of this dissertation. Chapter 5 proposes and evaluates a sequence-based methodology for resource allocation in simulation models such as the one developed here for a container terminal. The content of this chapter corresponds to Objectives 2 and 3. Chapter 6 discusses the computational experiments and outcomes that demonstrate the viability of the developed sequence-based simulation-optimization methodology, addressing Objective 4 of the dissertation.

Finally, Chapter 7 provides a conclusion and summary of the work conducted in this dissertation and discusses how it can be generalized for other applications. Potential extensions of this effort and recommendations for future work are also described.

CHAPTER 2

BACKGROUND

To build a knowledge infrastructure for the succeeding chapters, and to understand the topics to be covered, the theoretical background is presented in this section. This chapter provides a detailed description of the events that occur in the container terminal and covers the main topics in the literature with which this dissertation is concerned.

2.1 Container Terminals and Ports

Ports act as gateways or hubs in geographically centralized locations that lie on shores or coasts. A port is usually an unavoidable transit location for transferring cargo among vessels and land transportation modes [8].

In the literature, a port is referred to as either a maritime or an inland container terminal. In a maritime container terminal, the involved transportation modes include both vessels and land vehicles. A maritime container terminal is a serialized facility that lies on a coast where vessels can dock for delivering and picking up containers. If the transportation modes include only land transportation, the terminal is referred to as an inland container terminal or a dry port in some cases. Inland container terminals are usually situated near major cities and are, in most cases, connected by railways to maritime container terminals. The main function of a port is transshipping or warehousing freight as well as the berthing, repairing and refueling of vessels.

A terminal (Fig. 2) is an amenity that exists in a central and an intermediate location within a port where freight and passengers are assembled, transferred, or interchanged within the same or different modes of transportation. Thus, there are different types of terminals depending on the transportation mode involved, the material transferred, and the equipment used.

Within a port on a shore, multiple specialty terminals can exist including, among others: freight terminals which include both containerized and break-bulk (non-containerized cargo) terminals, rail-road terminals, cruise and ferry terminals (passenger terminals) and mineral and energy terminals.

According to Rodrigue [8] a Maritime Terminal is an assigned region of a port that contains: wharves, warehouses, storage spaces, cold storage plants, grain elevators, bulk cargo handling structures, landing and receiving stations, among others. The activities that usually take place in this area are the transmission and interchange of passengers and different cargo types between land and water carriers or two water carriers.

According to Martin et al. [9] there are three types of container ports; however, one port may illustrate more than one functional category for different facilities:

- *Hub ports*: (load center ports) served by large container vessels that operate on main shipping lines, where containers can be shipped to other terminals by smaller vessels.
- *Feeder ports*: served by smaller vessels called feeder vessels that transport containers from hub ports within the same area.
- *Direct call ports*: the port receives containers by large vessels, then containers reach their destination by land, where these vessels are not used for transshipping other containers to other ports.

Another type of terminals is a *transshipment hub* (also known as intermediate hub, vessel-to-vessel terminal etc.), which is usually located in an intermediate location between Hub and Feeder ports. In this type of terminal, containers are stored in the terminal for a short time before getting transferred by another vessel to another location, usually a country different than the country of origin and the transient country [8].



Fig. 2. Container terminal of the port of Houston [10].

2.2 Port Resources and Entities

The resources and equipment that exist in most container terminals are introduced in this section and their functions will be described in detail.

Containers (Fig. 3 (a)) are standardized steel aluminum or fiberglass boxes, used for moving materials and cargo around the world, either through water by vessels or through land by trucks and railways. There are several types and sizes of containers with different specialties, including: general purpose, refrigerated, tank, bulk, platform, insulated and ventilated. The international standard sizes of containers are 20', 40', 45' 48' and 53'. A container has specially made corners that make it easy for the terminal resources to lift it or pick it up [11-13].

Storage Yard is where all the containers are stored and takes the greatest space in a container terminal. It is usually located close to the shore where vessels berth; to minimize the travel time for the transporting yard trucks. Containers stored in the yard are in sets stacked next to and on top of each other, which are called sections, zones, or stacks [11].

Quay Cranes (Fig. 3 (b)) are electric powered machines that lift and lower heavy objects and can also move horizontally along the length of the dock. In a container terminal, they transfer containers from and to the vessels [14]. Cranes should be separated by more than 50 ft. when working together, to prevent any crane confliction [11].

Rubber Tyred Gantry Cranes (RTG) (Fig. 3 (c)) are rubber tyred mobile gantry cranes that are able to lift containers from a container stack and transfer them to a yard truck. They are used for stacking containers within the container yard and are capable of transferring containers from rail or trucks to the stacking area and vice versa [15].

Yard Cranes are container-handling equipment that load/unload containers onto/from trucks at the storage yard and have similar functions as an RTG but are smaller in size and easier to navigate around the port [16].

Yard Trucks (trailers) (Fig. 3 (d)) are trucks that operate inside the container terminal facility only and are mainly responsible for transporting containers from incoming vessels and railways to the storage area and vice versa.

Vessels (Fig. 3 (e)) are large boats and are one of the main transportation modes in a maritime container terminal. They are responsible for transporting containers by water to and from the terminal. There are different container vessels with different capacities including: container vessel, bulk carrier, container vessel with cranes, small general freight carrier, heavy lift crane vessel, liquid natural gas carrier, RO-RO vehicle carrier and a tanker [13]. Vessels make several

stops to other terminals before and after berthing at one container terminal. The pier space where the vessel docks is called a berth. There may be more than one berth in one terminal, depending on the terminal size and the number of available cranes [12]. A vessel may also occupy more than one berth depending on its size.

Trucks (external trucks) are heavy automotive vehicles and are one of the inland transportation modes. Trucks arrive at the container terminal either empty for picking up an imported container or full for dropping off an exported container. Trucks must drive to the container storage area so that an RTG or a YT can transfer the container from or onto the truck [12].

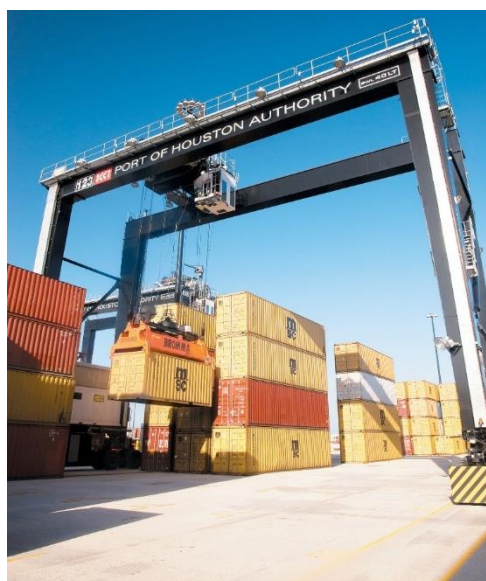
Railways are another mode of inland transportation. They arrive at the container terminal according to a schedule for delivering and/or picking up containers.



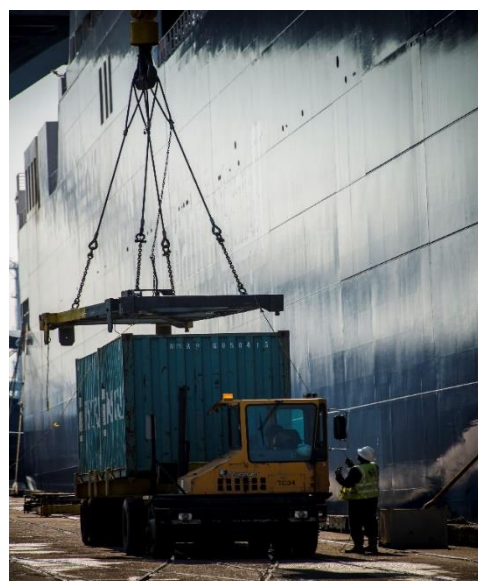
(a)



(b)



(c)



(d)



(e)

Fig. 3. Main resources and entities at a port: (a) containers, (b) quay cranes, (c) RTG, (d) yard trucks, and (e) cargo vessel [10].

2.3 Defining Maritime Terminal Problems

Challenges and problems facing decision makers at a maritime terminal can be classified into three categories: Seaside, Yard and Landside. This section describes the problems involved in each category.

2.3.1 Seaside

Berth Allocation and Scheduling Problem (BAP): Finding the optimal way to assign an arriving vessel to berth in a port, while taking into consideration the vessel size, dimensions and duration of stay and the berth capacity and layout, as well as accounting for any anticipated vessel arrival within the same timeframe [17].

Quay Crane Assignment Problem (QCAP) involves finding the optimal way to assign cranes to vessels to achieve the transshipment of required containers, while taking into consideration the crane constraints and restrictions. These restrictions include non-crossing cranes and maximum/minimum number of cranes allowed to serve a vessel. The *BAP* and the *QCAP* are sometimes considered interrelated since solving one problem will have a great impact on the other [17].

Quay Crane Scheduling Problem (QCSP) involves finding the optimal quay crane schedule that defines the starting time for every task on a crane while considering the minimization of vessel handling time (minimum makespan of the QC schedule), and other crane constraints [17].

2.3.2 Yard side

Storage Yard Related Problems: the container or storage yard is the busiest area in a container terminal where traffic congestion usually occurs, specifically when there are multiple

activities scheduled at the same time. Yard problems include but are not limited to:

- Assigning containers to different yard sub blocks to reduce traffic crowding.
- Determining yard layout (width of container blocks and number of containers in each stack) to reduce the number of yard cranes assigned.
- Reshuffling and restacking of containers (when the needed container is underneath one or more containers in a stack).
- Locating the storage yard to reduce terminal traffic.

Yard Crane utilization: the utilization of the yard cranes or other resources similar in function to reduce traffic congestion, reduce the number of cranes assigned, and optimize the yard crane schedule and deployment.

Transport Operations: reducing the number of yard trailers (trucks) assigned to QC and finding the optimal schedule for the trailers (trucks) assignment to container stacks.

2.3.3 Landside

Railway Layout Problem involves finding the optimal design and location for the railway terminal as well as scheduling and assigning a specific number of YC or RTG to process containers from the rail.

Security (Safety) deals with risk analysis, security breaches, and the recoverability of a terminal. Additionally, it involves applying different security measures/levels and analyzing their impact as well as finding the optimal configuration of checkpoints.

2.4 Discrete-Event Simulation and Arena

As the world became more digital and computers appeared in the 1950s and 1960s, people started using basic programming languages to write simulation models for complicated systems. However, it was monotonous and error prone since everything needed to be written and coded from scratch. Following that, simulation-specific programming languages (such as SIMAN and GPSS) appeared and aided simulation development and are still popular. Nonetheless, these languages posed a steep learning curve and necessitated an investment of time and effort to effectively master them. Therefore, numerous high-level simulation products were developed that operate with built-in graphical user interfaces.

A simulation model is an imitation or a copy of the real system used to study and better understand the real system. Simulation systems are categorized into several types, among which are discrete and continuous systems. A discrete system is one in which the contents of the system change instantaneously at different and separated points in time. A continuous system is one in which the system's contents change continuously with respect to time [18]. In this work, the focus will be mainly on discrete systems.

Discrete-event simulation is concerned with creating a copy of a real system in which the contents or the particles in that system act independently and in separate sets of points in time. Each of these points is called an event; an event is that point in time where the state of the system will change. An example of an event is the arrival of a particle to be processed that changes the state of the system from idle to busy. The particles or the contents that make up the system are called entities in the simulation language [18].

To this end, the discrete-event simulation software Arena 15 is used in this dissertation to model a mega size container terminal. Arena combines a user-friendly interface found in the high-

level simulators with the flexibility of the simulation specific programming, and it can also be used with general-purpose languages like Microsoft Visual Basic and C [19]. The core of Arena is the SIMAN simulation language. Arena is also compatible with Microsoft components and allows the user to import drawings, images, and 3D models.

Arena software also includes multiple helpful tools such as the Input Analyzer, the Output Analyzer, and the Process Analyzer. The Input Analyzer assists in fitting appropriate statistical distributions and defines the parameters of an existing dataset. The Output Analyzer compares multiple systems, determines confidence interval and warm up periods to reduce initial biases, and performs correlation analysis. The Process Analyzer aids with what-if scenario management and results analysis [20]. Arena provides simulation modeling animation on its workspace including simple graphics like the entity flow, queue lines, and resource status.

2.5 Simulation-Based Optimization

When simulating a complex system, a primary goal is to evaluate the effects of different values of input variables on the system. Yet, when attempting to conclusively find the optimal values for input variables in terms of the system outputs, all possible input variables will need to be evaluated, which requires numerous simulation runs and long computation times. This can quickly become impractical, particularly for complex systems. Therefore, it is increasingly beneficial to find the best values for input variables among all possibilities without explicitly evaluating each one. This process is called simulation optimization, which is the integration of optimization techniques into simulation analysis.

2.6 Evolutionary-Based Optimization

The discipline of evolutionary computation dates to the 1960s since the availability of digital computing technology. This made it easier to utilize computer simulations for analyzing complex systems that were difficult to evaluate mathematically. Three groups in particular were the pioneers of defining this field [21], Rechenberg and Schwefel at the Technical University of Berlin proposed using evolutionary processes to solve difficult optimization problems [22], Fogel projected utilizing artificial intelligence to solve problems by using evolutionary techniques at UCLA [23], while Holland at the university of Michigan anticipated using evolutionary adaptive processes as the solution to uncertain and altering environments [24, 25].

Evolutionary computation is a set of algorithms inspired by biological evolution to find optimal global solutions to complex and difficult problems. Evolutionary-based optimization starts with an initial set of candidate solutions that is then iteratively updated with every new generation by stochastically removing less fit solutions. This process results in a new generation with a population better in fitness.

2.7 Case Study: Hamad Port in Qatar

A case study is constructed where the work of this research is employed and implemented in a real-life example. The simulation-optimization model constructed was put into effect and the relevant results of the simulation-optimization were collected.

This case study is concerned specifically with the new Hamad port in Doha Qatar, which is in Western Asia on the coast of the Arabian Gulf and has been under construction since 2010 and recently started its operations at Terminal 1 in September 2017. Two more terminals are expected to launch in the succeeding years.

The Hamad Port is located at 25 kilometers south of Doha, the capital city of Qatar. This state-of-the-art mega project is anticipated to include, when at full capacity, three container terminals with a capacity of 6 million TEUs, general cargo terminal, multi-use terminal, off shore supply base, coastguard facility, port marine unit, port administration area, centralized custom inspection area, railway terminals, Qatari Emiri Naval forces base, Qatar Economic Zone 3 canal. This port will have worldwide connectivity and will be one of the world's deepest seaports in a strategic location where it will connect the internal Qatar railways and the Gulf Co-operation Council railways with state-of-the-art technology. The length of the Basin of the new port will be 3.8 kilometers and 700 meters wide, with an access channel of 10 kilometers long, 300 meters wide and 15 meters deep and a quay wall of 10 kilometers. Vessels arriving at Hamad port will enter through the access channel, constructed to accommodate the largest container vessels, bulk carriers and naval vessels. The width of the channel has been designed to facilitate two-way crossings.

CHAPTER 3

RELATED WORK

This chapter will discuss the related work of the topic of this dissertation, focusing mainly on recent publications in this area. Simulation and optimization methods are widely employed in the literature to address challenges posed in container terminals. Some studies use optimization or simulation methods only (Fig. 4), whereas others use an integration of simulation and optimization (Fig. 5) to achieve the goals of the study.

This chapter is divided into three sections corresponding to the methods used for solving container terminal problems; these sections are operations research, simulation methods, and simulation optimization.

3.1 Operations Research Methods for Maritime Operations

Articles considered using operations research methods to solve common container terminal problems will be reviewed in this section. These papers are classified according to the problems that were addressed in the container terminal.

3.1.1 Berth and Quay Crane Allocation and Scheduling

In 2009, Moghaddam et al. [26] created a mixed integer programming model for the Quay Crane scheduling and assignment problem by using genetic algorithms for real-world situations. They extended the mathematical model of Kim et al. [27] for the quay crane scheduling problem, and they applied their extended model in a container terminal located south of Iran.



Fig. 4. Studies that used optimization methods vs. studies that used simulation methods for addressing container terminal problems.

In their model they represented each job in the schedule as a gene in the chromosome; by the number of vessels and the number of jobs on that vessel, they used two crossover operators for two parts of the chromosome representation separately, one for the quay crane assignment to vessels and the other for the sequence of jobs on each vessel. They used arithmetic crossover for the first chromosome, to explore solution space and maintain feasibility, they also used extended patching crossover (uniform order-based crossover) for the second chromosome. Then they verified the performance of their proposed genetic algorithm by 30 numerical examples in different sizes. The small sized examples were solved by the branch and bound method by using the Lingo software. Then the same examples were solved using the proposed genetic algorithm and both results were compared.

They have concluded that their results have shown a reasonable gap of 1.9% and 3.5% between the optimal solutions obtained by the software Lingo and their proposed genetic algorithm, and that their algorithm is able to reach near-optimal solution in a reasonable time.

In 2011, Zhen et al. [28] focused on the berth and yard template planning, where they created a mixed integer nonlinear programming model to integrate the berth template and the yard template solution model. They created a local refinement stage that would refine the berth and yard related decision variables by an iterative process, where they were able to correct any problems with the yard template to optimize the berth template, and then the other way around. This process continues until no more improvements are achieved.

The aim of their research was to reduce the cost that usually applies when the vessel berth allocation schedule must be altered or changed to accommodate the actual vessel arrival time. They also created a heuristics algorithm to solve this problem on a large scale in a realistic environment.

In 2011, Zhen et al. [29] created a decision model to solve the berth allocation problem under uncertainty by using meta-heuristic for solving this problem in a larger scale. Their objective function considered two uncertainty factors; the first factor is the variation of the vessel's actual and scheduled arrival time, and the second factor is the difference between the scheduled and the actual operation's time of the vessel.

They created an objective function for minimizing the cost of creating the tactical plan for berth allocation while minimizing the cost of this plan's deviation, which is called "recovery cost". They used CPLEX to solve their model, but they had to reduce some of their decision variables to save some time since this is a large-scale problem. Then, for the improvement of objective function solution, they used simulated annealing. They conducted computational experiments to compare the meta-heuristic and the CPLEX approaches, in addition to studying the performance analysis on the proposed meta-heuristics and the numerical investigation on the proposed berth allocation problem model and validating the effectiveness of their proposed method.

In 2014, Trunfio [30] critiqued the work of Wang et al. [31] regarding the quay crane scheduling problem. Trunfio noted some mistakes in their work and proposed several adjustments to correctly implement the algorithm that Wang et al. had proposed. Wang et al. utilized the idea of Generalized External Optimization to solve the quay crane scheduling problem with respect to different interference constraints, by proposing a Modified Generalized External Optimization. In addition, they claimed that their proposed algorithm can find a best solution for seven out of ninety occurrences.

Trunfio was able to detect the errors of the latter by analyzing the Gantt charts of their solutions provided by their modified optimization. The author also concluded that the modified optimization cannot surpass other optimization methods known in the literature, because using a

randomized algorithm gives no guarantee of finding the optimum without proof of convergence. Finally, Trunfio suggested that an alternative heuristics approach that is more suitable for this kind of problem is an algorithm that alternates between global and local searches to balance the energy of considering the neighborhoods of viable solutions and staying within the feasible solution area.

In 2014 Simrin et al. [32] developed a heuristic model based on a genetic algorithm to solve the dynamic berth allocation problem, which is a Nondeterministic Polynomial Time type of problem. They proposed a new model for the problem, which considers the realistic dynamic arrival of vessels. They tested their algorithm by applying different values for the different parameters. They were able to test three different fitness functions, three types of crossover functions, different mutation rates and different population sizes and iterations. Their genetic algorithms provided very good results, reaching 0% in most -best- cases, while in the average case the gap was 1%. Their algorithm reached the optimal solution in less than a minute.

In 2014, Diabat et al. [32] developed a genetic algorithm formulation for integrating both the quay crane assignment and scheduling in order to solve this common problem in container terminals. Their objective was to minimize the makespan for all vessels and to consider a multi-vessel case, and an all-positioning condition for quay cranes. The objective function of their optimization model aims to minimize the handling makespan of the vessel that requires the greatest time while they created multiple constraints regarding the crane assignments to ensure optimization. Their problem had three different instances: small, medium and large. They were able to create problem specific chromosome representation and genetic operators to run an efficient algorithm where the genes represented the bays and they perform cross overs and swap mutation, while staying feasible when considering the constraints

They concluded that using the genetic algorithm to solve these integrated problems yields better results than attempting to solve each problem individually. Additionally, the formulation they have created remains simpler than the scheduling problem alone, instead of being complex because of the integration. Thus, genetic algorithms proved to be a successful approach for solving the quay crane assignment and scheduling problem, where their results returned near optimal solutions for small and medium-sized problems and performed extremely efficiently in terms of times for other sized problems.

In 2010 Bierwirth and Meisel [17] provided a summary of the research work conducted up to date related to the berth allocation and the quay crane scheduling and assignment problems in a container terminal. They described the operational planning problems against the limitations of different terminal properties and objectives. They categorized the berth allocation problem into three sections: *discrete*, *continuous* and *hybrid*; and surveyed the literature review accordingly.

Under the *discrete* type of BAP and QCAP, Hansen et al. [33] proposed compact Mixed Integer programming for the static problem of assigning and sequencing of vessels to berths while minimizing the vessel handling and waiting time. Imai et al. [34] minimized the waiting and handling time for the vessels and the deviation between the arrival order of vessels and the service order and used the Hungarian method to solve this problem. In 2008 the same authors [35] used genetic algorithms to minimize the weighted number of rejected vessels when the terminal could not serve these vessels. Hansen et al. [36] considered the discrete dynamic problem and considered minimizing the total cost of vessel waiting and handling keeping in consideration the earliness and tardiness for the vessels. Cordeau et al. [37] presented a tabu search method and formulated a discrete dynamic BAP with due dates. Mauri et al. [38] designed a column generation approach that provided better solutions in less time than the Tabu search proposed by Cordeau et al. [37].

Han et al. [39] and Zhou et al. [40] proposed a genetic algorithm to solve the problem of vessels restricting the berth assignment decisions; however, Zhou et al. [40] considered stochastic handling and arrival times for vessels in their model and a waiting time restriction that was considered as a due date. Lee et al. [41] formulated a bi-level programming model based on genetic algorithms to solve the problem of berth allocation and quay crane scheduling. Imai et al. [42] used genetic algorithms to develop a heuristic to find the efficient berth and crane allocation. Liang et al. [43] used genetic algorithms as well to find a solution for determining the berthing position and time of each vessel and the number of quay cranes to assign to each vessel. Giallombardo et al. [44] maximized crane utilization to be able to minimize the berthing position dependent on container flow between pairs of vessels.

Under the *continuous* type of BAP and QCAP, Li et al. [45] and Guan et al. [46] introduced the BAP in a continuous static method with fixed vessel handling times, where Li et al. formulated the problem as “multiple-job-on-one-processor” scheduling problem. The continuous dynamic BAP with fixed handling times, on the other hand, has been studied by multiple researchers, including Guan et al. [47], Moon et al. [48], Park et al. [49], Kim et al. [50] and Wang et al. [51] who proposed a new multiple stage search method (stochastic beam search algorithm) to solve this problem. Imai et al. [52] presented a heuristic for this problem and Chang et al. [53] extended their heuristic model by combining berth allocation and yard planning. Meisel et al. [54] proposed an improvement for crane utilization, then in [55] they presented a construction heuristic and two meta-heuristics procedures to solve the combined problem of BAP and CAP. Liu et al. [56] used Mixed Integer programming to formulate this problem as a large size problem, then proposed a heuristic decomposition approach to break the problem into smaller linked models. Zhu et al. [57] formulated this problem as an integer programming, then designed a branch and bound algorithm

to obtain optimal solutions, then simulated annealing was designed for large size instances. Hendriks et al. [58] proposed a mixed integer linear programming that minimizes the maximal crane capacity reservation. Legato et al. [59] suggested a two-way approach to solve this problem, the first phase using an Integer Programming model while the second phase using heuristics. Meisel et al. [60] used cut-and-run method to make sure that vessels meet their liner schedule, which is usually applied in packed terminals.

Under the *hybrid* type of BAP and QCAP with fixed handling times, Moorthy et al. [61] modeled this problem as a rectangle packing problem on a cylinder and they used a sequence pair based simulated annealing algorithm to solve it. They used the dynamic berth allocation package developed by Dai et al. [62] to evaluate the quality of their model. Imai et al [63] used genetic algorithms to solve this problem where both mega-container vessels and feeder vessels are served in a terminal. Lokuge et al. [64] proposed a hybrid Beliefs, Desires and Intention (BDI) framework with an intelligent module to solve this problem.

Under the QCSP, Lim et al. [65] presented approximation algorithms for the QCSP for the unidirectional type, then reformulated their problem later as a constraint programming model [66]. Then [67] they incorporated a non-crossing spatial constraint and studied a m-parallel machine scheduling that was inspired by crane scheduling in ports. Lee et al used genetic and greedy algorithms to solve this problem [68, 69], then they improved and updated their approach in a later publication [70]. Daganzo [71], Peterkofsky et al. [72] and Liu et al. [56] concluded that the sharing of bays among cranes can improve the makespan of a Quay Crane Scheduling Problem. Moccia et al. [73] formulated the QCSP as a vehicle routing problem with side constraints and solved the small instances by CPLEX while they developed a branch and cut algorithm for the larger instances. Sammarra et al. [74] proposed a tabu search heuristic for this problem. Ng et al. [75]

proposed a heuristic that decomposes this problem into a sub problem by sorting the vessel into a set of non-overlapping zones. Jung et al. [76] proposed a greedy randomized adaptive search procedure, which is a heuristic search algorithm, to solve this problem. Goodchild et al. [77] proposed that double cycling can reduce vessel turn-around time by using a greedy strategy.

In 2004 Steenken, Voß and Stahlbock, [78] presented a review paper that explored the major methods considered within the last four years for solving logistics and operational problems in container terminals. With regard to the BAPs, QCSPs and QCAPs, Bruzzone et al. [79] created an integrated simulation optimization model based on genetic algorithms for planning, scheduling and finding the optimal allocation for resources in container terminal. Legato et al. [80] adopted an approach to simulate the logistic activities of the arrival and departure of vessels in a container terminal. Lim [81] tackled the berth planning problem and showed that it is NP- Complete and used heuristics to propose a solution. Nishimura et al. [82] developed a heuristic procedure based on a genetic algorithm for the dynamic BAP with simultaneous service in a higher effectiveness container terminal in Japan. Similar to [82] Imai et al. [82] concentrated their research on developing a heuristics procedure based on the Lagrangian relaxation of the original dynamic berth allocation problem. Later on the same authors considered the nonlinear problem for the same issue [83], based on their paper published [34]. Kim et al. [50] formulated a mixed integer programming model for the berth-scheduling problem by using simulated annealing. Earlier the same authors used a heuristic procedure for finding a near optimal solution for the berth planning problem [48]. Guan et al. [47] developed a tree search procedure combined with pair-wise exchange heuristic for solving the vessel to berth allocating problem, they also developed a heuristic for the same problem using worst-case analysis in an earlier paper [46]. Park et al. [84] formulated an integer programming model by considering various practical constraints to find a method for the berth and

quay crane scheduling, the same authors also used sub-gradient optimization technique to solve the same problem in an earlier paper [49]. Gamberdella et al. [85] solved the resource allocation problem and the scheduling of loading and unloading problem hierarchically and verified their solution by using discrete event simulation. Bish [86] developed a heuristic algorithm based on transshipment problem to solve the container allocation problem, the vehicles to containers assignment problem and the crane assignment problem.

3.1.2 Storage Yard and Containers handling

In 2007, Lee et al. [87] were the first to address the yard allocation problem with consignment strategy and vicinity matrix. They formulated a mixed integer programming model and heuristic algorithm in order to create a solution for the yard storage allocation problem. They created a high workload rule and a vicinity matrix on the unloading/loading process, and they formulated the storage allocation problem as a mixed integer linear programming model. They were able to test their model using two set of input data for a simplified small-scale problem and a larger scale problem.

Their results for the small-scale problem show that the optimal solution can be obtained for low utilization, but it will take longer time for moderate utilization, and finally for high utilization the problem is infeasible. They were not able to solve the large-scale problem optimally in a reasonable time due to insufficient memory. They concluded that their proposed mixed integer programming is too complex to solve at a large-scale problem optimally, and therefore heuristic algorithms should be considered. They proposed two heuristic algorithms for finding a feasible solution: the sequential and the column generation methods. They concluded that the second method gives a better solution when compared to the first one, but only for those problems wherein

the first method can get a feasible solution, which means that the column generation method does not improve the solution quality effectively.

In 2008, Chew et al. [88] studied managing the storage yard template problem in a transshipment type of terminal, which is an intermediate destination, where most containers unloaded from one vessel will eventually be loaded onto another vessel in the same port. This study is an extension of Lee et al. 2007 [87]. The focus of this paper is the movement of export containers only. The first objective behind this study was to determine the yard template and then to find the smallest number of yard cranes to assign in order to reduce the operational cost. They proposed using a tabu search heuristic algorithm for obtaining an initial yard template, then they would improve their template through an improvement algorithm based on the information they obtained from the solution of the yard allocation problem which they have solved by the sequential method, this cycle will keep on repeating itself until an optimal or satisfactory solution is reached. Their goal was to determine the assignment of specific sub-blocks for each vessel where this assignment should be fixed at all times, and to determine the smallest number of cranes each sub-block needs, and the capacities of these sub-blocks. Finally, they validated their model by creating extreme cases to test their proposed method.

From the 2004 review paper by Steenken, Voß and Stahlbock, [78] under the *container stowage planning problem*, Sculli et al. [89] were one of the first few to create a simulation model to study the stacking and handling of identical size containers. Wilson et al. [12] developed a methodology for generating an automatic computerized solution for this problem using heuristic rules. In an earlier paper they applied principles of combinatorial optimization and tabu search to solve the same problem [90]. Debrovsky et al. [91] developed a genetic algorithm model for solving this problem, after testing a few other heuristics, including the suspensory heuristic

procedure, integer programming-based algorithm, simulated annealing and a branch and bound. They proposed that the GA model provided the most feasible and optimal results. Steenken et al. [92] proposed just-in-time scheduling formulation for the combined stowage and transport planning plus a consequent mixed integer model, exact and heuristics methods to solve this problem. Kang et al. [93] developed a greedy heuristic based on the transportation simplex method to solve the container assignment problem, and a tree search method to determine a loading pattern for the container assignment.

Under the *storage and stacking logistics problems*, Cao et al. [94] developed an algorithm based on tabu search to solve the transportation problems with nonlinear side constraints. Kim et al. [95] used dynamic programming to formulate the bay matching and the task sequencing problems, to convert the existing layout into a better layout by the minimum number of containers moved and shortest distance. Kim et al. [96] suggested a methodology based on the Lagrangian relaxation technique to find the optimal solution for the problem of minimizing the expected total number of container re-handles under certain conditions. Kim et al. [97] developed an analytic model to estimate cost components and to determine the number of storage spaces and transfer cranes required in an import container yard, this paper was an update to their previous publication to solve the same problem [98]. Kim et al. [99] formulated a dynamic programming model to determine the storage location of an arriving export container considering the weight constraint and develop a decision tree from the optimal solution to support real time decisions. Preston et al. [100] used genetic algorithms to minimize the turnaround time for container vessel and to determine the optimal storage strategy for different container handling schedules. Kim et al. [101] formulated a mixed-integer linear program and suggested two heuristic algorithms to allocate storage space for outbound container arriving at a storage yard, based on duration of stay and sub-

gradient optimization technique. Zhang et al. [102] used rolling-horizon approach to minimize the total distance to transport the containers between vessel berthing locations and storage yards, and to solve the storage space allocation problem. Kim et al. [103] formulated a routing problem as mixed integer program to minimize the total container handling time of a transfer crane including setup and traveling time.

Within the topic of *distributing empty containers to ports*, Crainic et al. [104] proposed a model and mathematical formulation for empty containers allocation and handling that addresses the uncertainty of supply and demand, the space and time dependency of events and equilibration flows. Shen et al. [105] presented a decision support system that uses network optimization, to solve the distribution of empty containers planning problem for a shipping company. Cheung et al. [106] developed a dynamic network model to solve the dynamic empty container allocation problem, by formulating a two-stage stochastic network, deterministic and random. Cao et al. [107] designed a heuristics branch and bound algorithm to search for a better solution for the capacitated multi-commodity p-median transportation problem, by applying a Lagrangian relaxation to the problem.

The literature review conducted by Steenken, Voß and Stahlbock [78] did not cover the *Loading Containers Problem*. However, among the groups that have looked into this problem are: Chen et al. [108], who created an analytical model where they formulated a zero-one mixed integer programming model to consider the problem on loading containers; Davies et al. [109], who proposed a new container loading heuristic that is capable of producing better lading arrangement when evaluated against other approaches and overcoming the limitations of other approaches; Scheithauer [110], who developed new bounds with new relaxation-layer for the container loading problem and the multi container loading problem that is based on the linear programming

relaxation; Eley [111], who presented a greedy algorithm approach with additional real world constraints for solving multiple container problem and the single container loading problem.

3.1.3 Shipping Routing, Scheduling and Liner Shipping Networks

In 2012 Wang et al. [112] proposed a way to optimize the sailing speed relation for container vessels and also investigated the optimal sailing speed of a container vessel and the optimal number of vessels to deploy on each ship route in a liner shipping network while considering container routing and transshipments. They formulated this problem as a mixed-integer nonlinear programming model and applied the model algorithms to a real case study for a global liner shipping company.

Qi et al. in 2012 [113] were one of the first to consider the stochastic nature of the port operational times when solving the problem of designing an optimal vessel schedule for a given shipping route to minimize the total expected fuel consumption and emissions. They used simulation based stochastic approximation methods to formulate the general optimization scheduling problem. They were able to validate their model based on a linear case study with multiple analysis and scenarios, and they have concluded that their proposed model will provide a significant fuel savings.

Plum et al. in 2014 [114] presented a novel compact model of the Liner Shipping Network Design Problem (LSNDP) that deals with handling multiple calls to the same port. Their goal was to create a shipping liner network that allow for container transport services and be able to maximize the profit of operating such network. They implemented their model as a Mixed Integer Programming and they solved it using a commercial solver. They were able to compare the results from this model, with the Benchmark Instance Algorithm of the study of Brouer et al. [115] in

2013. They have concluded that because of the many variables and constraints in their model, they were not able to achieve an optimal solution, and that they will address this issue in their future work.

Brouer et al. [116] in 2014 presented a metaheuristic for the Liner Shipping Network Design Problem where Metaheuristics are defined as methods exploiting the synergies of mathematical programming and metaheuristics. Their goal was to maximize cargo transport revenue while minimizing the network operations cost. The metaheuristic consisted of four main algorithms: construction, improvements reinsertion and perturbation heuristic. They were able to apply their mat heuristic as a decision support tool in a case study and concluded that it shows great improvements for a real-life network.

In 2014 Wang et al. [117] were one of the first to develop a mathematical model to simultaneously optimize ship route scheduling and cargo allocation scheme by taking into consideration the waiting time and the demurrage cost for the carrier with the objective of maximizing the total net profit for the container carrier. To guarantee a global optimal solution, they used an equivalent mixed-integer linear program to reformulate the schedule coordination and cargo allocation model. They were able to validate their model and test the effectiveness of the algorithms proposed by applying it to a case study and concluding the successfulness of their proposed model.

Agra et al. in 2015 [118] proposed a solution for the maritime inventory routing problem where sailing and waiting times are stochastic. They created a two-stage stochastic programming model with recourse (a feasible solution for the second stage can always be found as long as there is a feasible solution for the first stage), where the first stage contains the decisions of routing, loading and unloading and the second stage contains the decisions of scheduling and inventory.

Their solution combines the use of sample average approximation method with a decomposition procedure similar to an L-shaped algorithm, to check optimality for the complete model through an efficient separation method. They implemented their model based on real data from oil distribution at the Cape Verde islands, where they created ten instances. Based on their results they have concluded that the decomposition methods and stochastic programming is very effective for solving such problems and that the larger the demand and time horizon is the harder it becomes to solve the problem.

In 2015 Wang et al. [119] Evaluated the profitability of a shipping network and suggested improvements for the shipping services. They considered the problem of Profit-based maritime Container Assignment models (P-CA), where the container shipment demand is dependent on the freight-rate. They proposed two versions of the (P-CA) models, one at a tactical level and other at an operational level. They developed a nonlinear optimization model for the tactical-level (P-CA) with freight-rate dependent on container shipment demand. Then they designed a convergent trial-and-error approach to obtain the optimal freight-rate for an operational level (P-CA) with unknown but fixed demand functions. Finally, they proposed a practical trial-and-error method and were able to integrate the mathematical optimization and expert judgment for the operational level (P-CA) to maximize profit.

In another publication yet in a related topic Wang et al. in 2015 [120] developed a mixed integer programming model with a polynomial number of variables to propose an optimal alteration for the container liner shipping network problem, which is also called segment-based (a sequence of legs from a head port to a tail port that are visited by the same type of vessel more than once within the same shipping network). They were able to apply their proposed model to Asia-Europe-Oceania liner shipping network with a total of 46 ports and 11 shipping routes and

36 segments. Their results showed that their model gave improved results and that the optimized network has 8 ship routes, and the total cost of the initial network was reduced by 20%.

3.1.4 Port Resources and Transportation

From the review paper by Steenken, Voß and Stahlbock, [78] under the *container terminal systems* topic, Kozan [121] designed a network model to be used as a decision support system to analyze container transfer at a multimodal terminal. Meersmans et al. [122] discussed two operation research methods and classified some optimization scientific approaches for the container terminal problems. Murty et al. [123] designed a Decision Support System (DSS) to assist in decision making and solving the major bottle necks in container terminals. Kjetil [124] used CPLEX to assist in optimal routing decision, scheduling policy for vessels on operational levels, and determined a proper fleet size for strategic planning level, by evaluating the vessel scheduling problem. Dell'Olmo et al. [125] considered a container terminal as a service production system and presented it as a compels substructures of platforms to find resources allocation to minimize delay. Nam et al. [126] evaluated the difference between the conventional and the unmanned (automated) handling systems in container terminals. Liu et al. [127] designed analyzed and evaluated four different automated container terminal concepts, then each performance was evaluated by using microscopic simulation. Daganzo et al. [71] studied the crane scheduling problem by examining the static and dynamic crane work schedules, where he used integer programming to formulate the problem. Gambardella et al. [128] used simulation and optimization methods like genetic algorithms, mixed integer programming and job-shop scheduling to solve multiple container terminal scheduling, performance and allocation problems. Kozan et al. [129] used simulation and genetic algorithms to analyze the major factors that influence container

transfer efficiency in a container terminal. Peterkofsky et al. [72] used branch and bound methods to minimize delay cost and to solve the crane scheduling problem. Alicke [130] created a model based on Constraint Satisfaction Problem to model an intermodal terminal Mega Hub that was then transformed into a Constraint Optimization Problem, to configure long and short-term operations in a terminal.

Under the *transport optimization* topic, specifically automated guided vehicles (AGV), Evers et al. [131] used a new modeling technique to control the traffic of AGV by using a hierarchical system of so called semaphores. Bruno et al. [132] proposed two fast and effective heuristics that dynamically determine the home positions for AGV. Gademann et al. [133] addressed the problem of determining the home positions for the AGV that are in a loop layout. Wallace [134] presented an agent-based controller that is able to control the flow of other vehicles to reduce inefficient complex designs. Van der Heijden et al. [135] developed several rules and algorithms for empty vehicles management to reduce cargo waiting time, then they used simulation to evaluate the planning options on their performance. Leong [136] compared and analyzed the current performance of AGV in a container terminal with a new proposed efficient deployment algorithm scheme, by using discrete event simulation software AutoMode. Moorthy et al. [137] discussed the development of an efficient strategy of predicting and avoiding the deadlocks in a large scale AGVs in a terminal in Singapore. They implemented their solution using Automated simulation software. Grunow et al. [138] developed a flexible priority rule-based approach, by an alternative Mixed Integer Programming formulation. Hartmann [139] introduced a general model for scheduling container terminal resources by discussing priority rule-based heuristics and genetic algorithms in the port of Hamburg Germany. Lim et al. [140] used simulation to compare a dispatching method they proposed with other dispatching rules. The method they suggested by

using auction algorithm to dispatch AGV. Kim et al. [141] developed a beam search algorithm to route straddle carriers during the loading operation of export containers in a container terminal. They also formulated the routing problem as an integer programming in another publication [142] to minimize the total travel time for the straddle carrier. Bose et al. [143] studied the potential of evolutionary algorithms in improving the different dispatching strategies for straddle carriers to gantry cranes in order to reduce the time a vessel spends in the terminal. Li et al. [144] have developed an optimal and heuristic algorithm for the single quay crane dispatching problem and to optimize the loading and unloading of containers in a container terminal. Meersmans et al. [145] presented a branch and bound algorithm that uses various combinational lower bounds to solve the integrated scheduling of different types of handling equipment at an automated container terminal. Bish et al. [146] studied the container to yard position locating problem and developed a heuristic algorithm based on formulating the problem as an assignment problem. Lau et al. [147] formulated a mixed integer programming model that considers different limitations of operations of different handling equipment, to improve the productivity of an automated container terminal. Rebollo et al. [148] presented a multi agent system architecture for the automatic allocation problem in a port container terminal. Kim et al. [149] proposed using a beam search algorithm to solve the load-sequencing problem for export containers in port container terminals.

Within the *landside transport resources* topic, Kim et al. [150] suggested a dynamic programming model to solve the static sequencing problem in which all the arrival trucks are known in advance. They have also proposed a learning-based method and other heuristics and a simulation study to compare the impact of the different approaches. Koo et al. [151] proposed a new fleet management procedure based on a heuristic tabu search algorithm in a container transportation system.

Under the *crane transport optimization* topic, Kim et al. [142] formulated an integer programming and developed an optimization algorithm to minimize the total container handling time of the transfer crane. Cheung et al. [152] developed a new solution method called “the successive piecewise-linear approximation method” to solve the crane scheduling problem, by formulating a mixed integer linear program. Narasimhan et al. [153] developed a branch and bound based enumerated method to obtain an exact solution to the loading time minimization problem. Zhang et al. [154] formulated a mixed integer programming model and solved it using the Lagrangian relaxation model to solve the crane deployment problem. Linn et al. [155] presented a mixed integer program and a mathematical model for the optimal yard crane deployment.

3.1.5 Security Measures

In 2013, Yeo et al. [156] analyzed the relationship between seaport security levels and container volumes by building a system dynamics model. The use of system dynamics in their work demonstrates how this method helps in understanding the behavior of a complex system over time. They were able to test their findings by applying it to actual data collected from the seaport in Korea, which was used to build three interrelated models including a base model, an optimistic scenario model and a pessimistic scenario model. A different security level was applied to each model.

Yeo et al. concluded that applying high security measures will cause decreasing competitiveness resulting in significant loss of market share. On the other hand, applying low security measures will increase port attractiveness for stakeholders and increase the number of containers processed. However, if a security breach occurs, which is more likely in an optimistic scenario, it will have a significantly negative impact on the port and will cost the authorities time,

money and multiple delays. Consequently, it was concluded that these results should initiate more research interest in this area so that the different impacts can be further analyzed and studied.

3.2 Simulation Methods for Maritime Operations

In this section research applying simulation methods that address issues in berth area, storage yard, port operations and security measures will be discussed.

3.2.1 Berth and quay crane allocation and scheduling

In 2015, Ji et al. [157] considered a continuous and dynamic berth, where they integrated the berth allocation and the crane assignment problem. They created a continuous model using Monte Carlo simulation with different performance indicators. Their research is one of the few that considered the continuous nature of the berth system, with multiple different vessel sizes and random arrivals. They applied sensitivity analysis and double factors variance analysis to evaluate the operational efficiency of their model. Their results demonstrate the relationship between the crane assignment and the performance strategies considered at the terminal, and they suggested applying their optimal solutions to reduce waiting time and increase the resources utilization.

3.2.2 Storage yard

In 2009, Petering was the first to introduce the direct connection between the containers' block width and the long run performance at a container terminal [14]. A discrete event simulation model written and compiled in Microsoft Visual C++ 6.0 was designed to consider this study, where four different cases were studied: a small terminal and a large terminal, and two different container size configurations, less and more equipment. Nineteen different layout scenarios were

tested for each of the small terminal configurations, whereas fourteen were tested for the other configuration. In each of these different scenarios, the total yard storage capacity, the number of storage zones, as well as the number of containers in each zone was manipulated in order to introduce changes to the system. Ten simulation replications were performed for the small terminal configuration and six replications for the large terminal.

Petering concluded that the average quay crane work rate is concave with respect to block width and that the optimal block width configuration ranges between 6 to 12 rows, depending on the size and shape of the terminal. Additionally, wider blocks require less equipment whereas blocks with thinner width require more equipment optimally. Finally, he stated that the overall performance of the port improves as the shape of the terminal becomes more like a square.

In 2013, Petering [1] presented a system that determines real time container storage locations and investigates the effect on the overall long run of the container terminal. He had two objectives: to assess the importance of minimizing the container travel distance from quay to storage during unloading and from storage to quay during loading; to minimize yard truck congestion when containers are stored and when containers are retrieved. The second objective was to find specific real time storage locations that will maximize GCR (Gross Crane Rate).

The experiment considered two different terminal settings, a small terminal and a big terminal. He considered three main equipment sizes for each setting: scarce, less, and more yard trucks and yard cranes per quay crane. With these different settings, he proposed that many different and unique scenarios can be created from the setup above. For each scenario mentioned, the author set several container stacking restrictions and penalties in order to reach the best stacking method to achieve the goals of the study.

Petering concluded that maximum container dispersion and restrictive yard crane deployment systems will result in the highest GCR in the six different scenario terminals. He also concluded that a stacking strategy that is penalty based will improve GCR depending on the terminal by 1% to 7% and that the advantages of a penalty-based stacking strategy will increase as the terminal size gets larger or as the terminal equipment gets scarce. However, random storage systems are still considered a good system, especially with terminals that have more equipment, as the penalty-based experiments results showed an improvement of only 1% to 2%.

3.2.3 Port operations

In 1988, Chung et al. [11] developed a simulation model to increase the utilization of material handling equipment and reduce container loading time at the Port of Portland. Their research presents the idea of creating a buffer area located between the container storage area and the dock area, where containers can be stored temporarily while waiting to be loaded onto a vessel. The objective was to consider the effect of this area on the container terminal's operations and whether or not it will reduce bottlenecks caused by the transtainers (which in this case is an RTG or a yard crane).

The buffer area in their research can serve two scenarios. The first is one in which a transtainer is scheduled to pick up a container from the container stack; however, this container is located at the bottom of the stack and is to be loaded onto the vessel first, followed by the containers on top of it. In the second scenario, the transtainer is scheduled to leave the current section or stack of containers and work on another one; however, the transtainer will have to come back to this section later to pick up containers.

They were able to perform 96 simulation runs, observing a significant improvement of the flow of the port operations when their idea for the first scenario was applied, which resulted in a reduction of 4% in the total loading time. However, they concluded that using the buffer area in the second scenarios did not reduce the total time; because, while the transtainer will be moving the containers to the buffer area, the cranes will be idle because it will be waiting for the scheduled containers to arrive from the new sections.

Lee et al. [158] published their work in 2006 on using analytical calculations for creating a design for railways terminal facility, where the process of unloading containers from a train occur. This facility is located close to a container terminal. They used simulation for manipulating and evaluating the different available designs for such terminal. Their design parameters were the number of cranes to deploy and the number of transshipment rail lanes for the crane to be working on. They created seven different simulation testing scenarios, with two different operational times of rail terminal per day, one with 20 hours arrival window per day and another with 7 hours arrival window. Their simulation model allowed them to compare the different scenarios, and study the workload and the rail occupancy, and the containers flow, and also nominate the most efficient scenario.

In 2007, Cortes et al. [159] simulated the transportation of different types of cargo like containers, cements, steel, iron etc. that depart from and arrive at the Seville inland port in Spain. From data collected from the annual reports of the Port Authority of Seville, they were able to build a simulation model using Arena software that simulated vessels arrival and departure, dock assignment, truck arrival and departure, the different container terminals, cereal and cement facilities and some other docks. Two berths are assigned to receive container vessels involving

three different terminal companies, each with an average capacity of 700 TEUs. The cranes working on these berths have a 30 container per hour performance.

The simulation model was run for 90 days providing results from the arrival of 166 vessels to the port of Seville, and the total time the vessel spent in the dock and in the system was calculated. No details were provided regarding the simulation model as the focus was primarily on the different port operations. The results of their work concluded that the port resources can handle the current flow of freight and cargo, except for the rare and short-term situation when there are some difficulties in the port like down time and weather circumstances.

In 2010 Wan et al. [160] used the Arena software to build a conceptual and simulation model for a container terminal and were able to model all of the resources at this port, like quay cranes, container truck and yard crane. They were also able to utilize the Arena built in program “OptQuest” in order to find the optimal allocation of the port resources within a given constraints in order to minimize operation times and increase utilization. They have concluded that their results were realistic and that they were able to achieve practical values; however, since their model was created for teaching purposes, it still has some limitations and they intend to avoid these restrictions in their future work by building a model with more details and parameters.

In 2011, Petering [161], was the first to address nine different and independent studies in a vessel to vessel transshipment container terminal by creating a fully integrated discrete event simulation model that was designed to imitate the real system. He was able to analyze the various impacts of these studies on the long run of quay crane and the decisions made by the container terminal experts.

The first study focused on finding the optimal yard capacity for a terminal while all the other aspects remained unchanged. The second and third study investigated finding the ideal

number of Yard Cranes and Yard Trucks for a container terminal. The fourth study the impacts of substituting the Yard Cranes by Yard trucks. The fifth study the processing speeds of the Yard Cranes and Yard Trucks. The sixth study addresses the processing time variability of Yard Trucks and Yard Cranes on the Quay Crane Rate. The seventh addresses the minimum required yard crane separation distance. The eighth study addressed the substitutability of Yard Trucks traveling to the same location. Finally, the ninth study addresses the terminal scalability. The different impacts of all these studies on the Quay Crane Rate individually have been collected as an outcome measure.

In 2010, Yuan et al. [162] created a discrete event simulation model using Arena software in addition to optimization methods in order to analyze some issues in a raw material inland terminal. The issues they were dealing with are cargo transportation, vessel berthing and handling machinery performance. Since this is a special kind of terminal, it also faces some issues regarding raw material stock piling and production of material. After analyzing the current state of the port, they were able to improve performance in the terminal operations with their simulation and optimization measures.

in 2013, Kulak et al. [15] developed a simulation model using the Arena software in conjunction with a Visual Basic application at one of the biggest container terminals in Turkey. This study was to reconsider the terminal's operations, identify bottlenecks and optimize performance; as comparative empirical studies using Data Envelopment Analysis, classified this terminal as one of the most inefficient container terminals.

After analyzing the current port configuration, they identified yard cranes as the major bottleneck in the system and proposed to solve the problem by also improving resource allocation, i.e., storage yard allocation and truck allocation. They suggested increasing the yard cranes number, and after doubling the number of yard cranes, the total handling rate of containers

increased by 50%. For yard allocation, they assigned the outbound container's location to be close to the berth location where the assigned vessel is scheduled to berth in. A significant improvement was noticed in the total container handling rate when applying this strategy.

For the truck allocation, they noted that currently, the available yard trucks do not work under any specific assignment rule, so the trucks might have to travel long and unbalanced distances. In their experiment, they applied a dedication strategy in which each berth has a specific number of trucks, unless the berth is idle; then a priority rule is used to re-allocate the trucks. Their results showed that the total container handling rate can be increased with this allocation for a number of 30-yard trucks; however, if the number of yard trucks is 27, both strategies work well; but the lower the number of trucks, the lower the total container handling rate will be. Kulak et al.'s simulation model helped with analyzing the port operations and forecasting methods to resolve bottlenecks and emphasize on future developments and changes to the operation and the configuration of the operation system.

3.2.4 Security measures and risk

Rabadi et al. [163] presented a discrete event simulation model that represents US port activities on the east coast, in order to study the impact of risk activities and security breaches on the port activities and recoverability (the ability to go back to normal port activities). They were able to model the movement of full and empty containers, trucks, trains and vessels, and also terminal gates, straddle carriers, cranes and transtainers. Input data were based on a port historical data and they were also able to validate their model by comparing their outcomes with these data and by consulting subject matter experts. Then they implemented a theoretical risk scenario in their model and studied its impact and estimated the recovery time. They concluded that their

model can be applied to help with utilizing other port resources and studying the implementation of different scenarios.

In 2009, Caris et al. [164] created a discrete event simulation model in order to analyze and improve the complex system of intermodal freight flow and transport at the network of port Antwerp in Belgium. Intermodal transport refers to having at least two modes of transportation in a single transportation activity, where these modes are being transported by railways, inland or waterways. They succeeded in simulating numerous policy measures for the transportation of entities, toward facilitating the estimation of consequences and estimation before having to apply these policies in real life.

In 2009, Na et al. [165] developed an integrated simulation model using Monte Carlo technique and Arena software to mimics the port operations and activity in order to study the impact of earthquakes or any similar disaster on the operations of a container terminal and to evaluate the loss and damage caused to the economy of that area. Their main goal was to improve the decision-making process for the risk of earthquakes in container terminals and to assess the loss of throughput in that terminal upon the damage that has occurred.

In 2010, Longo [166] presented a simulation model that assists in applying better operational policies and practices on the flow of inspected containers in a container terminal. Longo created a simulation model, which describes the container terminal operations and contains most of the important resources and activities in a terminal, like vessels, forklifts, cranes, tractors, and trucks as well as the processes of loading/unloading and transferring containers, etc. The container cycle in this study follows any other cycle, when vessel arrives to the seaport; containers are moved to the storage area. Also, in this study, Longo applied an inspection area in the simulation model, where the model assigns a percentage of the incoming containers for inspection,

and this selection is based on container history information, container configuration information or any alert information. The simulation model was used to study the impact of the container inspection on the overall container terminal operations and productivity.

Longo concluded that the incorporation of container inspection process with the other container terminal operations is simply a matter of optimal trade-off between having more advanced technology and equipment that would speed up the inspection process and between finding a better organization of the internal container resources that aids with inspection like officers and yard trucks.

From the review paper by Steenken, Voß and Stahlbock, [78] on the topic of *simulation systems*, Nam et al. [126] performed computer simulation analysis representing various operational scenarios in a port in Korea to find the optimal port size and configurations of berths and quay cranes. Shabayek et al. [167] created a simulation model for the Kwai Chung container in Hong Kong, to study the simulation model's prediction capabilities and how accurate and relative the results will be to the real system. Kia et al. [168] created a simulation software for a container terminal to study the positive vessel to rail direct loading on the capacity of the terminal using real-time statistics. Their model also identified bottle necks, and other delay problems which helped save time and cost. Hartmann et al. [169] created a container terminal simulation software that is capable of generating various scenarios based on an algorithm. Their scenarios can be used to input data to the simulation model to solve and optimize the different port problems. Yun and Choi [170, 171] created an object-oriented model and simulated a general container terminal based on a terminal in Korea. Vis et al. [172] studied the effects of using Automated Guided Vehicles and Automated Lifting Vehicles on the unloading time of the vessel using simulation studies. Kozan et al. [173] developed a batch-arrival multi-server queueing system and compared it with another

analytical model and a simulation approach. Henesey et al. [174] and Meersmans et al. [145] were some of the first to present a multi-agent system approach with several agents, where quay cranes, berths, vessels and gantry cranes are considered agents.

3.3 Simulation Optimization Methods for Maritime Operations

According to Dragovic et al. [175], there were 21 journal papers that focus on the *integration of Simulation and Optimization in container terminal operations*, between 1998 and 2015. While the application is similar, these papers considered solving different problems in various areas of several types of terminals and systems, in addition to using different types of simulation and optimization tools as a solution method. This section will review all these papers in addition to other relevant publications. Fig. 5 depicts the most related studies that used an integration of simulation and optimization methods for various problems in container terminals.

3.3.1 Berth and quay crane allocation and scheduling

In 2005, Nishimura et al. [176] developed a new optimized dynamic routing strategy of the assignment of yard trailers to quay cranes to reduce the cost of container handling in a container terminal. They ran their solution on the C program where they considered a long quay terminal with four or six berths and 2 different types of container storage, one with a spread-out design while the second is located behind the berth.

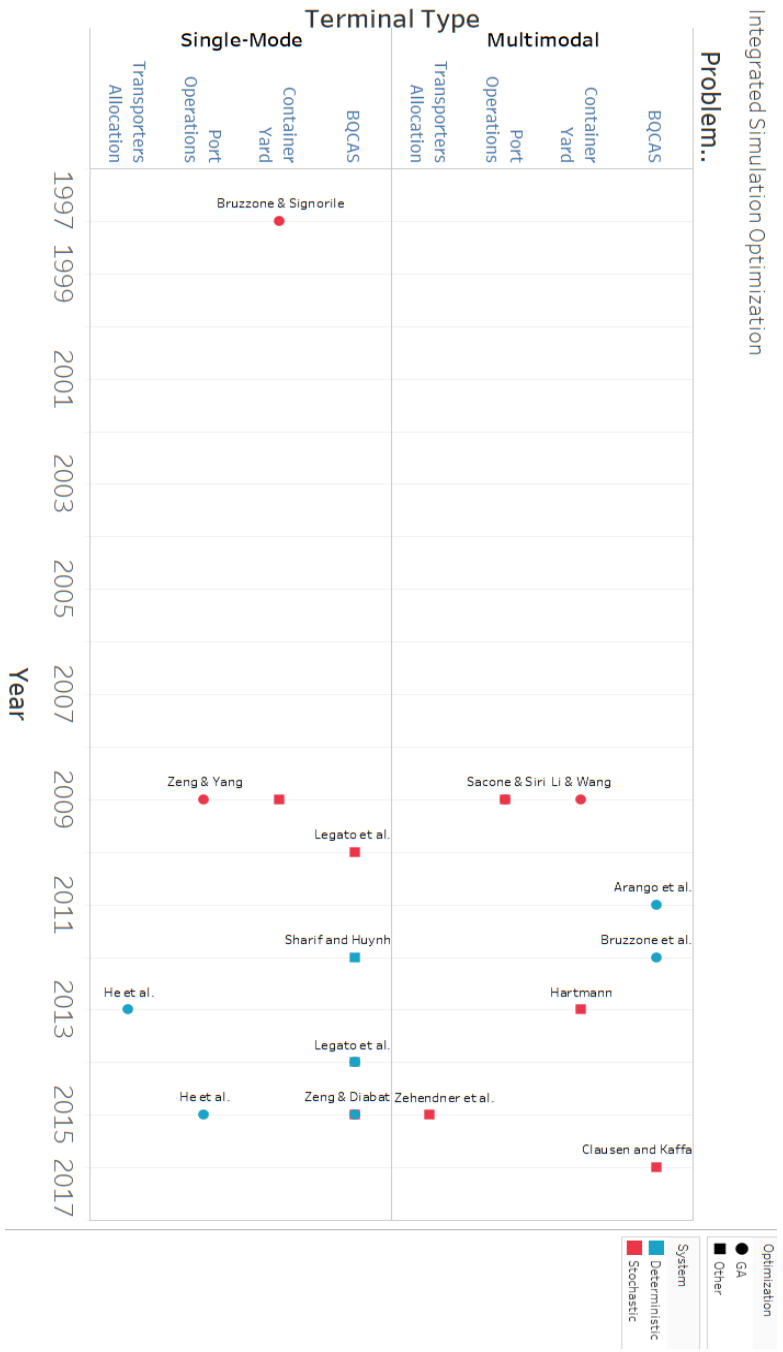


Fig. 5. Previous work for integrating simulation and optimization for addressing problems in maritime operations.

They generated ten different computational samples to include the random nature of stacking in a container terminal. After multiple computational experiments, they concluded that the dynamic trailers assignment to quay cranes are more improved than the static assignment that is usually employed in container terminals.

In 2012, Xu et al. [177] developed a hybrid heuristic algorithm to deal with the continuous berth allocation problem that usually deals with unreliable arrival and handling times for container vessels. their main goal was to prevent the adjustments that usually take place in the berth scheduling execution. They created a robust berth scheduling algorithm that integrates simulated annealing and branch and bound algorithm, and they were also able to test the reliability of their proposed algorithm by conducting computational studies where they used simulation to be able to validate the robustness of their model formulation.

Legato et al. [178] in 2014, proposed a simulation-optimization model to integrate the tactical-level of the berth allocation problem as a mathematical programming formulation with the operational-level model as a discrete-event simulator to solve the berth allocation problem. They created a framework by inserting a simulation engine in an optimization algorithm, where they created two separate two-level models. They integrated the first model as a mathematical programming formulation and the second model as a discrete event simulator. They also used simulated annealing for their optimization model. Their objective was to create a special model where it is possible to manipulate any tactical solution returned for the berth allocation problem, when in some cases things are not going according to plan in the operational level.

3.3.2 Storage yard

Tang et al. in 2014 [179] proposed an improvement to the static reshuffling and the dynamic restacking problem in a container terminal. They developed five effective heuristics to improve the reshuffling problem, and then they developed a discrete event simulation model with animation to mimic the stacking, reshuffling and retrieving operations and to be able to test and analyze their proposed algorithms. They have concluded that their improved model can return optimal or feasible solutions in less time than the current model used in the terminal, also the heuristics they proposed return better results and require less time than the current one for both the static and dynamic problems.

In 2015 Said et al. [102] presented an approach using discrete-event simulation modeling to optimize solution for the storage space location problem in a container terminal in Alexandria port in Egypt. They built a simulation model using the simulation software Flexsim to simulate the container terminal operations and to be able to optimize a solution for the storage space allocation problem, taking into consideration all the different container terminal handling activities. Their input data was collected from the port of Alexandria for one week of operations, with an arrival of 12 vessels and more than 9,000 different types and sizes of containers. Their results showed the effectiveness of their proposed optimization model with a significant decrease in the container handling time of 54%.

In another publication in 2014 Said et al. [180] proposed another optimization model based on their container terminal simulation software, where they analyzed vessel handling and berth allocation. They were able to reduce the vessel service time by 51% based on their optimization approach.

3.3.3 Port resources and transportation

In 2012, Nguyen et al. [181] conducted their research on solving the vehicle dispatching problem under uncertainty in container terminals in a dynamic environment, where they created a mixed integer programming algorithm that can adopt to the dynamic changes in the container terminal. Then they used Plant simulation model to evaluate their proposed heuristic algorithm under uncertainty. They compared the performance of their heuristic algorithm under uncertain travel times with a greedy algorithm for the deterministic traveling times, and concluded that under the uncertainty algorithm, the results performed better in vehicles' throughput and the total delay time of quay cranes.

In 2013, He et al. [182] proposed an integer programming model based on the rolling horizon approach to improve the traffic affectivity of a port. Where they proposed creating an approach where internal trucks are shared among multiple container terminals located close to each other. Their objective was to minimize the total transferring cost and the over-all flow of workloads and operations of all terminals during all times. They used a genetic algorithm to be able to search for solutions, whereas the simulation model will execute the rolling horizon approach, evaluate and repair solutions. They used random data to validate their proposed simulation optimization method, which provided good solutions in a good timeframe. They also concluded that the total cost from their proposed method was reduced.

Tao et al. [183] in 2015 suggested a multi-factor online dispatching strategy by combining evolution searching function and discrete event simulation to solve the vehicle dispatching problem in a container terminal in mainland China. They were able to build a realistic simulation model based on the layout of the port in Shenzhen, China, where they establish two berths with vessels arriving and departing at various times. They have concluded that this simulation study has

outperformed the traditional dedication vehicle dispatching method significantly in most of the cases they have studied.

In 2015 Cordeau et al. [184] presented a simulation-based optimization model for solving the Housekeeping problem in a transshipment container terminal. They were able to embed a simulation model in a local search heuristic to facilitate the routing of straddle carriers and multi trailer systems. They presented the term “Housekeeping” which represents the transportation of a container from a temporary location to a permanent location, closer to the berth location where it will be loaded on a vessel. They have concluded that using a simulation optimization model significantly improved results of vehicle routing and waiting times when compared to the standard scheduling policies.

3.3.4 Port operations

Nevins et al. in 1998 [185] utilized object-oriented programming to develop a discrete event simulation model that was implemented in PORTSIM evaluating complicated operations occurring in seaports. They intended to explore how object-oriented programming concepts such as data abstraction; data encapsulation as well as inheritance can be beneficiary in such a context. Such features allowed them to construct different types of cargo (different object classes) as well as creating shared attributes and functions across the model (by using inheritance and encapsulation techniques). The goal from their work was to study the complex operations in seaports in the context of military mobility in order to determine the throughput capability of the port and to be able to create a prototype of the port for the purpose of measuring the effectiveness of any plan changes in the seaport.

In 1998 Bruzzone et al. [79] created an integrated simulation optimization model based on genetic algorithms for planning, scheduling and finding the optimal allocation for resources in container terminal.

Bielli et al. [186] created a container terminal simulator in 2006 for the improvement of management decisions where they evaluated policies that were generated by optimization algorithms. They used distributed discrete event simulation in their model by applying multithreaded programming in Java. They analyzed the container terminal system based on an object-oriented paradigm where they identified different classes and diagrams to describe the system; they represented these diagrams using the Unified Modeling Language.

In 2009 Sacone and Siri [3] considered the operational planning of a container terminal by defining an integrated framework in which simulation and optimization interact. the simulation represents the dynamic environment of the terminal and the optimization is called (in this case the software Lindo) whenever the system faces a critical event condition. Their objective was to find the optimal handling rate among the different areas of the terminal to minimize the weighted sum of queue lengths.

In 2009 Li and Wang [187] used simulation-based optimization based on a genetic algorithm to solve the block planning and dynamic container truck configuration problem, by introducing the parallel computing technology which is realized by MPI to the solving process.

In 2009 Zeng and Yang [188] developed a simulation optimization method based on genetic algorithms and neural network for scheduling loading operations.

In 2009 Briano et al. [189] proposed modeling port operations through a systems dynamic approach integrated with ERP and simulation, where genetic algorithms and risk analysis tools help make better decisions.

In 2009 Zeng and Yang [190] proposed a method that integrates Q-learning algorithm and simulation techniques to optimize the operation scheduling in container terminals.

In 2010 Legato et al. [191] designed and implemented a simulation optimization framework based on simulated annealing algorithm, to address the quay crane scheduling problem.

In 2011 Arango et al. [192] created a discrete event simulation model using the Arena software for solving some of the berth allocation problems in the inland port in Seville, Spain. The port they studied is considered a multi-purpose terminal that contains different specialties terminals for handling and transferring these different cargos. They used genetic algorithms to solve the berth allocation problem based on first come first serve strategy, by aiming to reduce the service time for vessels. Their results confirm that simulation by optimization is indeed an appropriate solution strategy and it did improve performance in the port of Seville.

In 2011 Yu et al. [193] presented a simulation-based optimization model for job sequencing scheduling-optimization of a container terminal. They developed a mathematical model based on Hybrid Flow Shop Scheduling Problem, to optimize the operations of quay cranes, yard cranes and yard trailers. They also developed a simulation optimization model based on genetic algorithms to find a solution for this problem. They have concluded that their proposed method is successful in managing job sequence optimization.

In 2012 Bruzzone *et al.* [2] developed a simulation model to evaluate the fitness function of genetic algorithms to carry out a range allocation optimization on berth assignment and number of tractors to be assigned to each crane.

In 2012 Sharif et al. [194] considered a dynamic programming-based heuristic and agent-based simulation to solve the yard crane scheduling problem in both centralized and decentralized approaches.

In 2013 Hartmann et al. [195] discussed the scheduling of refer mechanics at a container terminal in a case study that integrates simulation optimization. Reefer containers are special temperature-controlled containers used to transport temperature-sensitive goods such as fresh fruits and vegetables. They are equipped with an integral refrigeration unit and require external power supply while on a vessel, truck or train and while being stacked on a container terminal. Reefer mechanics work in the stacking area of a container terminal, where they are responsible for unplugging departing containers and for plugging arriving containers. They have concluded that choosing the correct scheduling method, will have a crucial impact on the productivity of refer mechanics workforce and the completion of the job.

In 2013, He et al. [182] proposed an integer programming model based on the rolling horizon approach to improve the traffic affectivity of a port. Where they proposed creating an approach where internal trucks are shared among multiple container terminals located close to each other. Their objective was to minimize the total transferring cost and the over-all flow of workloads and operations of all terminals during all times. They used a genetic algorithm to be able to search for solutions, whereas the simulation model will execute the rolling horizon approach, evaluate and repair solutions. They used random data to validate their proposed simulation optimization method, which provided good solutions in good timeframe. They also concluded that the total cost of their proposed method is considerably lower than the obtained cost.

In 2014 Ilat at al. [7] developed a simulation-optimization method to address the integration of three resource allocation problem: berth, quay crane and tugboat assignment.

In 2015 He et al. [196] addressed the problem of integrating the scheduling of the main three terminal resources, Quay Cranes, Yard Cranes and Internal Trucks. They formulated the problem as a mixed integer programming model and their objective was to minimize the total

departure delay for vessels and the total transportation energy consumption of all tasks. To solve the problem, they developed a simulation-based heuristic method to solve it, where they integrated genetic and swarm optimization algorithms.

In 2015 He et al. [197] developed a modeling and optimization problem of multi-echelon container supply chain network, to minimize the total supply chain service cost. Since this is NP-hard problem, the authors proposed a simulation-based heuristic method to solve it, where they integrated genetic and swarm optimization algorithms.

In 2015 He et al. [198] considered the trade-off between energy-saving and time-saving in the Yard Crane scheduling problem, where integrated simulation optimization and the optimization algorithm integrates genetic algorithms and particle swarm optimization.

In 2015 Zehendner *et al.* [199] used optimization simulation for the allocation of straddle carriers at a tactical level at an intermodal container terminal.

In 2015 Zeng *et al.* [200] proposed a simulation optimization method to solve the Quay Crane Dual-Cycling scheduling problem based on a mixed integer programming model and a bi-level genetic algorithm heuristic method.

Said *et al.* in 2015 [201] developed a methodology using discrete event simulation to optimize resource utilization at El-Dekheila port in Egypt, with regard to the integration of equipment resources used in the container terminal. Data was collected from the port in Egypt, where they were also able to validate their proposed model and conclude its efficiency and effectivity where the quay crane utilization was increased by 41%.

In 2016, Clausen *et al.* [202] considered an inland terminal in Germany that serves as a hinterland hub for deep sea container terminals, where all areas are handled in an integrated terminal with a rail mounted gantry crane on one rail as the main handling equipment, where

these cranes execute all the handling in the operation areas in the terminal. The container they are considering is a multimodal terminal that serves vessels, trains and trucks.

3.3.5 Security measures

In 2008, Ding et al. [203] proposed a fuzzy simulation optimization model based on discrete event simulation and heuristic algorithm for the optimal configuration for check posts. They created three modules: simulation module, interface module and optimization module. The optimization module is responsible for generating data for the simulation module, whereas the interface module transfers the generated data from the optimization module to the simulation module. The results from the simulation module are then sent back in a feedback loop to the optimization module via the interface module similarly. OptQuest engine was adopted as a simulation optimizer integrating tabu search, scatter search, integer programming, and neural networks into one search algorithm. They validated their findings by applying it in the Free Trade Port Area in the north port in China. Their results showed that the best way to utilize the check post is by reducing the resource redundancy with the current traffic volume and concluded that more research should be conducted where other factors that might affect the port should be considered, such as road network structure and traffic operation modes.

3.4 Search Space Reduction for Simulation Optimization

This section identifies papers that may have considered sequence-based methods to reduce the search space and computation times in simulation-optimization.

Reducing search space in simulation-based optimization algorithms has motivated several studies [4, 5]. Particularly relevant to this work are studies that investigated novel ways to

efficiently evaluate decision variables. Dehghanimohammadabadi et al. [6] proposed an iterative optimization-based simulation framework, where in every optimization iteration, the number of decision variables and coefficients is updated in the traditional way and no predefined sequence is considered. In addition, their framework can call an optimizer upon the occurrence of predictable or unpredictable events within the simulation run, where a trigger event occurs upon which, the optimizer takes over to optimize the parameters of the system while the simulation model is halted.

Chang et al. [5] proposed a partitioning-based simulation optimization method to divide the feasible solution region into several sub regions. Then, their approach samples a few solutions from each region, evaluates this solution, and identifies the most promising region. Their convergence analysis showed significant reduction in computation time. Collectively, an investigation of the order/sequence in which optimization is executed and its effect on reducing the search space was not observed in the literature.

CHAPTER 4

SIMULATION MODEL DESIGN AND DEVELOPMENT

In Chapter 2, the complex and stochastic nature of container terminals as well as the various challenges involved were described. After establishing the understanding of the nature of the system and properly defining the problem, a discrete-event simulation model was constructed to mimic the system's behavior and capture its activities. Additionally, in this chapter, different methods are considered to validate the developed simulation model.

Data collected from an actual port as well as information from the literature are utilized to provide input parameters to run the simulation model. The simulation results are then compared to output data obtained from the same port. In addition to simulation results, this chapter will also discuss different scenarios considered and sensitivity analysis to aid with the model's validation.

4.1 Model Design and Construction

After studying the real system thoroughly and building an overall understanding of the on-going container terminal operations, a conceptual model can be inferred and constructed to assist with creating the model and make it a seamless, less complicated task. In this section, the design of the conceptual model and the building of the Arena simulation model will be described. Details regarding operations, activities and implementations of this work will be discussed in the succeeding sections.

4.1.1 Conceptual Model and System Architecture

In the context of terminal simulations, a conceptual model is a logical flow diagram that represents the flow of vessels, trucks, and containers between sea side and land side and identifies decision nodes and resources necessary for the movement of containers and transport systems.

The flow diagram shown in Fig. 6 depicts the conceptual model of the container terminal. This was developed after studying the container terminal system and the flow of its operations. In this abstract model, vessels arrive at the anchorage area at the port and wait for a berth. Once an available berth is allocated, the vessel moves to it via an access channel and docks at its assigned berth. Quay Cranes or Ship-to-Shore (STS) cranes are assigned to the vessel and start unloading containers to Yard Trucks that transport them to the Container Yard where containers are stacked. Yard Cranes, which in this case are rubber tyred gantry cranes (RTGs), pick up the containers from the Yard Trucks and place them in blocks of containers in the Container Yard. Yard Trucks then return to the Quay area to pick up more containers and bring them back to the Container Yard. In case of export containers, Yard Trucks take containers in the opposite direction for the STS cranes to load on the vessel. This process continues until the vessel is unloaded and export containers, if any, are loaded on the vessel. Multiple vessels can berth simultaneously and be serviced by STS cranes.

In the truck cycle, external trucks come from the land side and enter the port area through the security gates to the Container Yard to pick up containers and transport them inland outside the port. RTGs are used to load containers on the external trucks which then go through customs and exit the port area through the security gates. In the case of exporting containers or returning empty containers, external trucks bring these containers to the port and drop them off at the Container Yard.

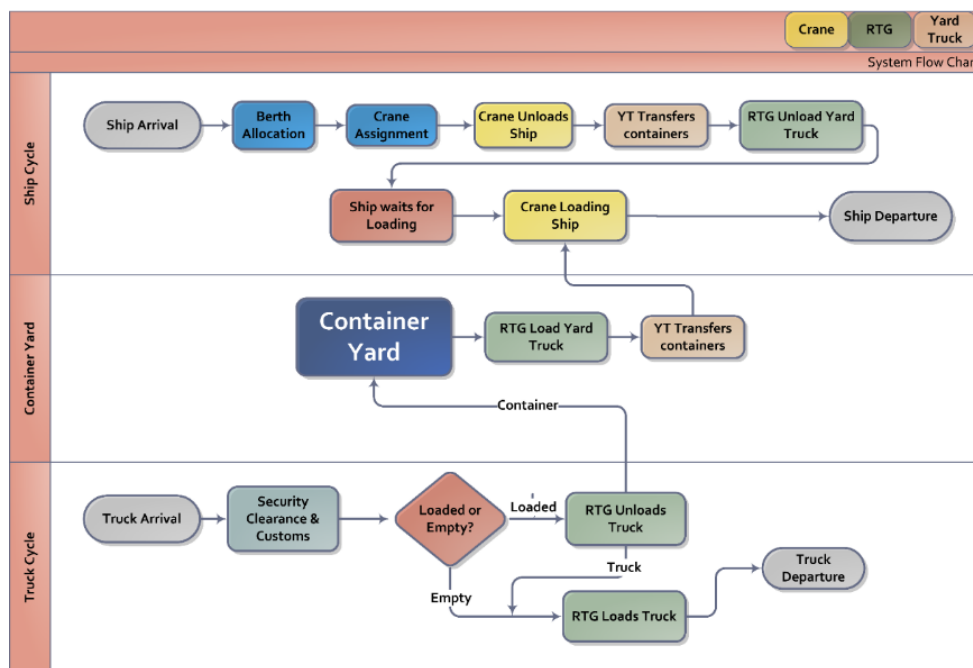


Fig. 6. Conceptual model.

4.1.2 Simulation Model

The design of the simulation model is intended to be at a macro level to study operations and policies such as the impact of resource allocation and scheduling on the terminal's performance. The model was developed using ARENA simulation modeling environment. The vessels (with different lengths), containers, and external trucks are modeled as dynamic entities that flow through and drive the simulation according to the conceptual design discussed earlier. The modeled resources include the vessel access channel, quay berth area, the STS cranes, and external truck gates. Yard Trucks and Yard Cranes are modeled as transporter resources to accurately simulate their functions. The external trucks were not, however, modeled as transporters since they do not belong to the port; they are instead treated as entities attached to the containers

as another entity type. It is assumed in this model that personnel are readily available to operate the modeled resources and the impact of their unavailability is captured within the data.

Many processes that take place at the terminal were modeled by seizing the necessary resources, delaying the entity by a certain time, and then releasing the resources. The berth area is considered continuous and vessels are allocated to berth areas based on the vessels' lengths.

4.2 Data Collection

To simulate the activities and operations taking place in the container terminal, real data and descriptive statistical distributions were collected. This data included the number of resources such as: Berth size, number of STS quay cranes, yard cranes, yard trucks; as well as storage capacities and dimensions. Data regarding the port layout and activities was also obtained from an open-source data repository, other similar ports, as well as previous studies and research conducted on port activities.

4.2.1 Data Categorization

The data collected can be categorized as follows:

- **Arrivals:** Inter-arrival times (the time between consecutive arrivals in a queuing system) of the main entities to the container terminal: vessels and external trucks.
- **Processing times:** The time it takes various transporters and resources to load/unload a container from one process to another: STS quay cranes, RTGs, external trucks and their gates and yard trucks.

- **Traveling times:** The time it takes various entities and transporters (yard trucks, vessels and external trucks) to travel in the port system based on measured distances and velocity estimates.
- **Availability:** The number of available berths, cranes, container yard size and yard trucks in the container terminal.
- **Capacity:** The number of available resources, such as STS quay cranes, yard cranes and yard trucks.

4.2.2 Defining Entities and Resources

Among the important steps for building a simulation model is defining its entities and resources to avoid misrepresenting the real system. The following describes the way entities and resources were defined:

Entities:

- **Containers:** They represent the entities that seize the resources to be loaded, unloaded or transferred. They arrive according to the same arrival distribution of the transportation mode that brought them in.
- **Vessels:** They represent the entities that bring/take the containers to/from the container terminal by sea.
- **External Trucks:** They represent the entities that bring/take the containers to/from the container terminal by land.

Resources:

- STS Quay Cranes: They perform the process of loading\unloading a vessel. A vessel seizes several cranes (multiple cranes) according to its (the vessel's) size.
- Berth Segments: The location where the vessel docks (parks) in the container terminal for a brief time until the unloading/loading process concludes. A vessel seizes several berth segments according to its (the vessel's) size.
- Gates for External Trucks: The checkpoints for all arriving external trucks, where check ins, paper work and some security scans take place.

Transporters:

- RTG (Rubber Tyred Gantry Crane): They perform the process of loading/unloading external and internal trucks. They usually work in the container yard, assigned to a specific area or stack of containers.
- Yard Trucks: They perform the process of transporting containers inside the container terminal only, among the different terminal resources.

4.3 Modeling Highlights

In this section, the primary areas in the simulation model in addition to the techniques followed to model various aspects of the system in a discrete-event simulation platform will be introduced. Fig. 7 below shows the outlines of the arena building blocks, where all entities arrive at the model through a Create module which is used to model the arrival of all vessels and trucks. While at the Dispose module, all the entities mentioned previously depart.

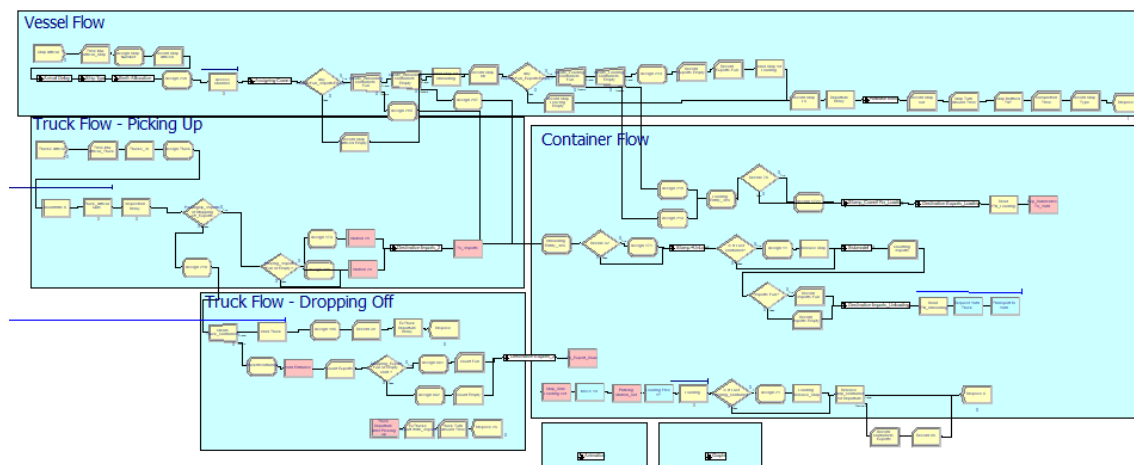


Fig. 7. Simulation Model Building Blocks

Vessel flow:

Vessels are generated in the simulation model through a create module based on an exponential distribution that represents the arrival distribution of the actual vessels. They are then sorted into thirteen different vessel types, and each vessel is assigned a size and a specific number of containers to load/unload. This number was based on historical data of the port under study.

Berth availability and allocation:

The berth area was divided into several segments where each segment represents a resource as shown in Fig. 8. A vessel requires several segments based on its size where larger vessels need more segments. Thus, after the vessel arrives in the simulation model, it must check if there are enough idle berth segments to seize based on its size. Once the vessel seizes the necessary number of segments (berth resources), it will be time stamped. If there was not a sufficient number of segments for the vessel to seize, or if all segments were busy, the vessel will be moved to a Hold module where it will wait until a condition is met and a signal is sent to this Hold module. After a vessel finishes the unloading/loading process it will release the berth segments and depart the

system, but before departing it will send a signal to the Hold module to release any vessel waiting in line.

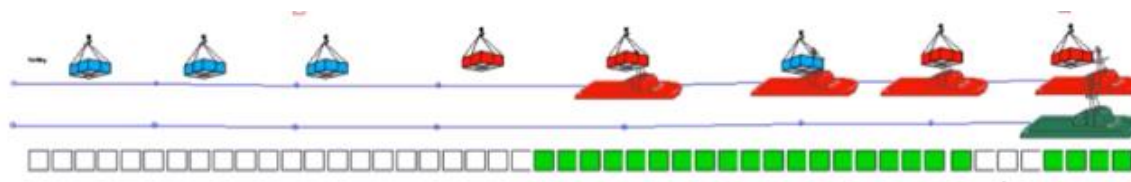


Fig. 8. Berth availability and seizing.

Crane allocation:

The STS cranes are modeled as a set of individual resources that are responsible for unloading/loading containers from/to the vessel. Based on the vessel size, a specific number of cranes is required to serve the vessel. Thus, cranes are assigned to several specific sections of the berth segments to prevent the cranes from crossing over. The cranes and berth assignment are modeled such that, depending on the predefined (user defined) number of berth segments and the number of cranes, the cranes are assigned to the corresponding berth segments. For example: If the berth size is at full capacity (48 segments) and there are 6 cranes available, then each crane will serve 8 segments. In another example, if the berth size is at 75% capacity (36 segments) and there are 4 cranes then each crane will serve 9 segments, and so on.

External Trucks flow and Gates:

External trucks are generated in the simulation model through a Create module based on a Weibull distribution that reflect the arrival distribution from the historical data of the port under study. They are then sorted into two different truck types, either a truck for picking up imported container or a truck that is dropping off exported container. Finally, both trucks will be sent to the

container yard to the designated container block (import or export). Once the job of picking up/dropping off is completed, the external truck departs the system.

Container yard:

The container yard is where the containers are stored in Hold modules and Queues, waiting to be picked up by an external or a yard truck. It is modeled as a set of container blocks, where four horizontal blocks make a row in the terminal, and there are three export rows and seven import rows. Each block has a set of queues to hold the containers which represents the process of storing the containers in the block (Fig. 9).

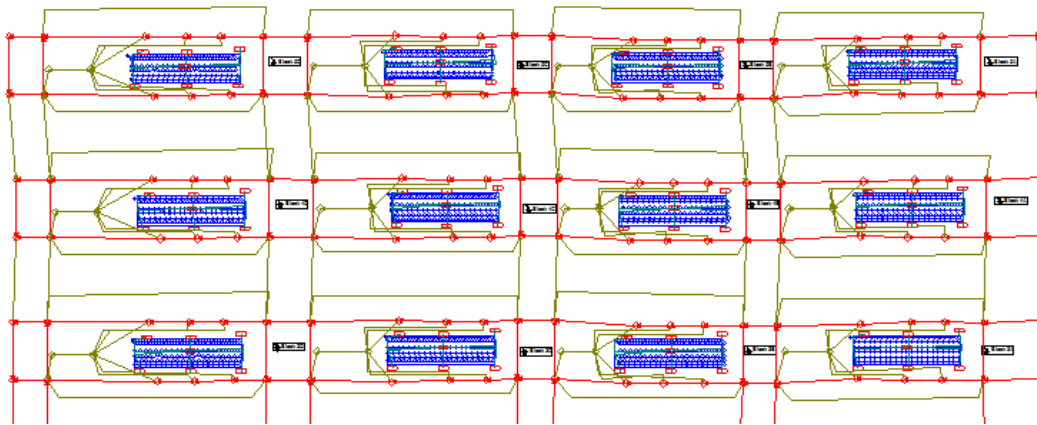


Fig. 9. Container yard design.

RTG (Rubber Tyred Gantry Cranes)

A Rubber Tyred Gantry Crane is a yard crane that serves the external and internal trucks in the container yard, specifically to load and unload containers to/from the trucks and from/to the yard. The RTGs are modeled as a set of guided transporters in ARENA.

Yard Truck:

Yard trucks are responsible for transporting containers from/to the vessel and to/from the yard. It is served either by the RTG if working in the Yard or by STS Crane if working near the vessel.

Containers Flow:

A *container* entity can arrive at the terminal in two different ways, either on a vessel, or on a truck. The first process the containers go through is the unloading process, where they are removed by an STS quay crane if they arrived on a vessel or by RTG if they arrived on an external truck. After they are unloaded, a yard truck will drive to the unloading area for vessels to transfer the unloaded containers to the container yard. The loading process follows identical but reversed steps, where a yard truck will transfer a container from the container yard to the departing vessel.

Read/Write features:

When constructing the discrete-event simulation model in Arena, some of its *read/write* features were utilized, to make the information incoming to and outgoing from the model available externally. The direct *read* feature allows for reading data directly into variables and attributes in Arena, this *read* process takes place at the beginning of the simulation; replication or even during the simulation run. The input data is created using a text file which the optimizer can access and write the solution string to. The simulation model then accesses that file and performs a *read* process to access the provided solution string values and run the simulation model accordingly.

4.4 Input Analysis and Data Fitting

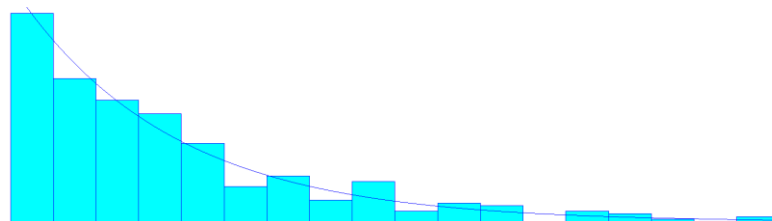
A case study was considered, of the newly constructed container terminal system of Hamad's new port of Qatar, which has been under construction for the past several years and has recently started its operations at Terminal 1. Data from the existing Doha port has been collected on the number of vessels as well as the number of import and export containers over a year.

Vessel historical data is expected to reflect a comparable demand levels and material flow that the new terminal will undergo in the first years of operations. One-year worth of historical data was analyzed, and appropriate distributions were fitted to it. This included vessel lengths, vessel interarrival times, number of containers (full, empty, import and export), delay times from the arrival to the start of operations, loading/unloading times, and delay times from the end of operations to the vessel's departure. It is recognized that the resource types and terminal design differ between the new and current port; however, for validating the simulation model, the model was populated with data based on the current port to be able to validate against the historical data. For example, the STS crane types in the new terminal can move more containers per hour, and hence, are expected to service a vessel faster; however, crane movement rates that resemble those of the current crane types were used to validate the simulation output. Another example is the access channel, which does not exist for the current port; therefore, the time for the vessel to pass through it (in the model) was set to zero. Similar assumptions have been made to accommodate the differences between both systems for the purpose of validation.

One-year worth of historical data was analyzed. To conduct proper validation analysis, data only for six months were used to fit the distribution and the remaining six months were used for testing purposes. To further remove any biases due to possible trends and seasonality in the data, the historical data of the full year was randomized, and then half of the dataset was drawn randomly

for input analysis, while the other half was used for validation after running the model and comparing its outcome. After the data is fitted to theoretical statistical distributions, both Chi Square and Kolmogorov-Smirnov (K-S) tests were used to assess the goodness of fit. The null hypothesis in this case is that the fitted distribution and the theoretical one are the same, and when we reject the null hypothesis ($p\text{-value} > 5\%$) which means that there is no difference between the distributions (i.e., the fit is acceptable).

Fig. 10 is an example of fitting the vessel interarrival times over six months of randomized data using ARENA's Input Analyzer, which shows that the best fit is an Exponential distribution with a mean of 13.7 hours and shows that this distribution passes both tests as the p-values are large (> 0.05). This means that the null hypothesis that both distributions are the same cannot be rejected and therefore the fit is acceptable. Other simulation inputs were conducted similarly and empirical distributions (based on six-month worth of data) were used whenever fitting to a theoretical distribution failed the hypothesis testing.

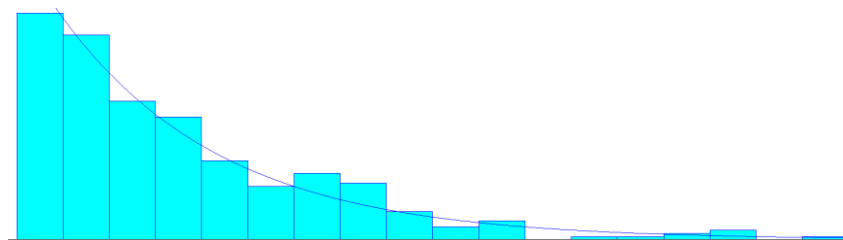


Vessel Interarrival times Distribution Summary

| | | |
|-----------------------|-------------|--------------------------------|
| Distribution: | Exponential | Chi Square Test |
| Expression: | EXPO(13.7) | Number of intervals = 10 |
| Square Error: | 0.001987 | Degrees of freedom = 8 |
| | | Test Statistic = 7.28 |
| | | Corresponding p-value = 0.507 |
| | | Kolmogorov-Smirnov Test |
| | | Test Statistic = 0.0438 |
| | | Corresponding p-value > 0.15 |
| Data Summary | | |
| Number of Data Points | = 324 | |
| Min Data Value | = 0 | |
| Max Data Value | = 66.4 | |
| Sample Mean | = 13.7 | |
| Sample Std Dev | = 12.9 | |

Fig. 10. Vessel Interarrival times distribution fitting of 6-month historical data (data included in the model).

To validate the distributions inputted into the simulation, the historical data of the remaining six months are fitted into a distribution and both distributions are compared. Fig. 11, for example, shows the interarrival time distribution fitting of the six-month data that was used for testing. The result shows that an exponential distribution with an average interarrival time of 13.1 hours is the best fit which is very close to the result shown in Fig. 10.



Vessel Interarrival times Distribution Summary

| | | |
|-----------------------|-------------|--------------------------------|
| Distribution: | Exponential | Chi Square Test |
| Expression: | EXPO(13.1) | |
| Square Error: | 0.002313 | |
| Data Summary | | Number of intervals = 10 |
| Number of Data Points | = 328 | Degrees of freedom = 8 |
| Min Data Value | = 0.03 | Test Statistic = 11.5 |
| Max Data Value | = 66.5 | Corresponding p-value = 0.19 |
| Sample Mean | = 13.1 | Kolmogorov-Smirnov Test |
| Sample Std Dev | = 11.7 | |
| | | |
| | | Corresponding p-value > 0.15 |

Fig. 11. Vessel Interarrival times distribution fitting of 6-month historical data (data used for testing).

4.5 Validation: Simulation vs. Historical Data

As was mentioned previously, the data set was randomized to remove any trends or seasonality and was split into two six-month data sets (training and testing). The training data set was used for input analysis and distribution fitting, while the testing data set was used for comparisons with the simulation output. A summary of the simulation results for some of the measures compared to the historical data are shown in Table 1. Note that the Historical data column does not correspond to six consecutive months of the year but rather randomly selected days of the year that add up to six months.

The simulation was run for 25 replicates where each replicate is six months. It was observed that with this number of replicates, the half width of the 95% confidence interval was about 5% of the mean for most measures. Hence, it was concluded that this is a good number of replicates. Furthermore, it was noted that the model goes into steady state after about 14 days; therefore, a simulation warm-up period of 14 days was used. A Summary of the simulation results for some of the measures compared to the historical data are shown in Table 1.

Table 1: Simulation results vs. historical data

| Measure | Historical Data | | Simulation Output | | |
|---------------------------------------|-----------------|--------------|-------------------|----------|--------|
| | Mean | 95% C.I. | Mean | 95% C.I. | |
| <i>Vessel Turn Around Time (hrs.)</i> | 36.5 | [33.9, 39.1] | 33.04 | 28.88 | 37.20 |
| <i>Vessel in</i> | 327 | - | 318 | 311.12 | 324.88 |
| <i>Vessel out</i> | 327 | - | 317 | 310.13 | 323.87 |
| <i>Imports Full</i> | 99,951 | - | 90,854 | 88,693 | 93,015 |
| <i>Exports Empty</i> | 42,109 | - | 47,990 | 46,861 | 49,119 |

4.6 Sensitivity Analysis

Two cases were considered, with different vessel arrivals in the simulation model, while changing the number of resources, to test the simulation model under extreme conditions. In the first case, the same level of the vessel arrival from the historical data was considered, with changes to the resources of the terminal. The resources manipulated are the Yard Trucks, Yard Cranes and the STS Quay Cranes. The performance measures that were considered are the Vessel Turn Around time, the number of vessels in and out, the number of exported and imported containers (throughput), in addition to resources utilization. Results of the simulation scenarios are provided in Table 2.

Table 2: Vessel arrival is at the same level as in 2014

| Measures | Baseline | 5 Yts | 10 Yts | 20 Yts | 25 Yts | 30 Yts | 40 Yts | 5 RTGs | 10 RTGs | 15 RTGs | 20 RTGs | 25 RTGs | 30 RTGs | 3 Cranes (18 Berths) | 4 Cranes (24 Berths) | 6 Cranes (36 Berths) | 8 Cranes (48 Berths) |
|-----------------------------|----------|---------|---------|---------|---------|---------|---------|--------|---------|---------|---------|---------|---------|----------------------|----------------------|----------------------|----------------------|
| Ship TAT | 35.38 | 53.16 | 40.92 | 40.56 | 31.89 | 35.38 | 31.60 | 0.00 | 13.97 | 31.10 | 28.94 | 28.77 | 35.38 | 16.60 | 30.43 | 31.61 | 35.38 |
| Ship in | 316 | 319 | 319 | 317 | 313 | 316 | 284 | 0 | 12 | 277 | 304 | 325 | 316 | 35 | 249 | 291 | 316 |
| Ship out | 315 | 318 | 317 | 316 | 312 | 315 | 283 | 0 | 9 | 276 | 305 | 326 | 315 | 35 | 248 | 290 | 315 |
| Imports | 91,026 | 90,679 | 90,361 | 90,359 | 89,480 | 91,026 | 80,754 | - | 3,485 | 79,568 | 86,322 | 92,708 | 91,026 | 9,616 | 69,877 | 82,248 | 91,026 |
| Exports | 48,514 | 47,919 | 50,016 | 49,111 | 48,652 | 48,514 | 42,293 | - | 1,527 | 41,420 | 45,926 | 48,618 | 48,514 | 4,817 | 38,338 | 44,877 | 48,514 |
| All Containers | 139,540 | 138,598 | 140,377 | 139,470 | 138,132 | 139,540 | 123,047 | - | 5,012 | 120,988 | 132,249 | 141,326 | 139,540 | 14,433 | 108,215 | 127,125 | 139,540 |
| Resource Utilization | | | | | | | | | | | | | | | | | |
| Berth | 0.29 | 0.37 | 0.31 | 0.33 | 0.27 | 0.29 | 0.26 | 0.93 | 0.89 | 0.35 | 0.24 | 0.26 | 0.29 | 0.44 | 0.54 | 0.36 | 0.29 |
| Crane | 0.29 | 0.29 | 0.29 | 0.29 | 0.28 | 0.29 | 0.25 | 0 | 0.01 | 0.25 | 0.27 | 0.29 | 0.29 | 0.08 | 0.45 | 0.35 | 0.29 |
| RTG | 0.40 | 0.07 | 0.19 | 0.34 | 0.37 | 0.40 | 0.53 | 0.3 | 0.23 | 0.45 | 0.41 | 0.42 | 0.40 | 0.06 | 0.32 | 0.37 | 0.40 |
| Yard Truck | 0.92 | 0.72 | 0.83 | 0.89 | 0.91 | 0.92 | 0.93 | 1.00 | 0.99 | 0.94 | 0.93 | 0.92 | 0.92 | 0.94 | 0.92 | 0.92 | 0.92 |

In the second case, the vessel arrival was increased by 25%, the same changes to the resources were applied, and the same performance measures were captured as shown in Table 3.

Table 3: Vessel arrival level of 2014 is increased by 25%

| Measures | Baseline | 5 Yts | 10 Yts | 20 Yts | 25 Yts | 30 Yts | 40 Yts | 5 RTGs | 10 RTGs | 15 RTGs | 20 RTGs | 25 RTGs | 30 RTGs | 3 Cranes (18 Berths) | 4 Cranes (24 Berths) | 6 Cranes (36 Berths) | 8 Cranes (48 Berths) |
|-----------------------------|----------|---------|---------|---------|---------|---------|---------|--------|---------|---------|---------|---------|---------|----------------------|----------------------|----------------------|----------------------|
| Ship TAT | 83.0839 | 81.50 | 81.79 | 79.73 | 71.75 | 83.08 | 83.23 | 0.00 | 8.00 | 67.05 | 83.53 | 82.86 | 83.08 | 12.37 | 34.56 | 63.06 | 83.08 |
| Ship in | 356 | 354 | 363 | 363 | 368 | 356 | 353 | 0 | 17 | 235 | 354 | 355 | 356 | 37 | 117 | 326 | 356 |
| Ship out | 354 | 350 | 361 | 361 | 365 | 354 | 352 | 0 | 16 | 232 | 351 | 353 | 354 | 36 | 116 | 324 | 354 |
| Imports | 102,175 | 102,489 | 104,618 | 103,925 | 104,002 | 102,175 | 100,830 | - | 2,432 | 66,275 | 101,573 | 102,099 | 102,175 | 10,423 | 33,654 | 91,477 | 102,175 |
| Exports | 55,791 | 56,529 | 55,839 | 55,819 | 55,673 | 55,791 | 54,917 | - | 4,414 | 35,631 | 54,382 | 56,116 | 55,791 | 5,584 | 18,108 | 49,338 | 55,791 |
| All Containers | 157,966 | 159,018 | 160,457 | 159,744 | 159,675 | 157,966 | 155,746 | - | 6,846 | 101,907 | 155,955 | 158,215 | 157,966 | 16,008 | 51,762 | 140,815 | 157,966 |
| Resource Utilization | | | | | | | | | | | | | | | | | |
| Berth | 0.81 | 0.80 | 0.82 | 0.80 | 0.72 | 0.81 | 0.81 | 0.93 | 0.88 | 0.79 | 0.81 | 0.81 | 0.81 | 0.52 | 0.71 | 0.82 | 0.81 |
| Crane | 0.33 | 0.33 | 0.33 | 0.33 | 0.33 | 0.33 | 0.32 | 0 | 0.01 | 0.21 | 0.32 | 0.33 | 0.33 | 0.09 | 0.21 | 0.39 | 0.33 |
| RTG | 0.39 | 0.07 | 0.18 | 0.35 | 0.37 | 0.39 | 0.48 | 0.49 | 0.24 | 0.36 | 0.41 | 0.4 | 0.39 | 0.11 | 0.14 | 0.35 | 0.39 |
| Yard Truck | 0.90 | 0.69 | 0.78 | 0.87 | 0.89 | 0.90 | 0.92 | 1.00 | 0.99 | 0.94 | 0.90 | 0.90 | 0.90 | 0.93 | 0.90 | 0.90 | 0.90 |

The outputs from the previous scenarios are represented in graphs to better understand the behavior of the system under these extreme conditions. Both the Vessel turn-around time and the container throughput were depicted against the number of the manipulated resource and compared with both the regular and the increase vessel arrival.

Fig. 12 shows the comparison of the vessel turn-around time with both the regular and increased arrival when the number of yard trucks was changed. With the regular arrival, it can be noted that the vessel turn-around time improves as the number of yard trucks increases; however, that is not the case with the increased arrival.

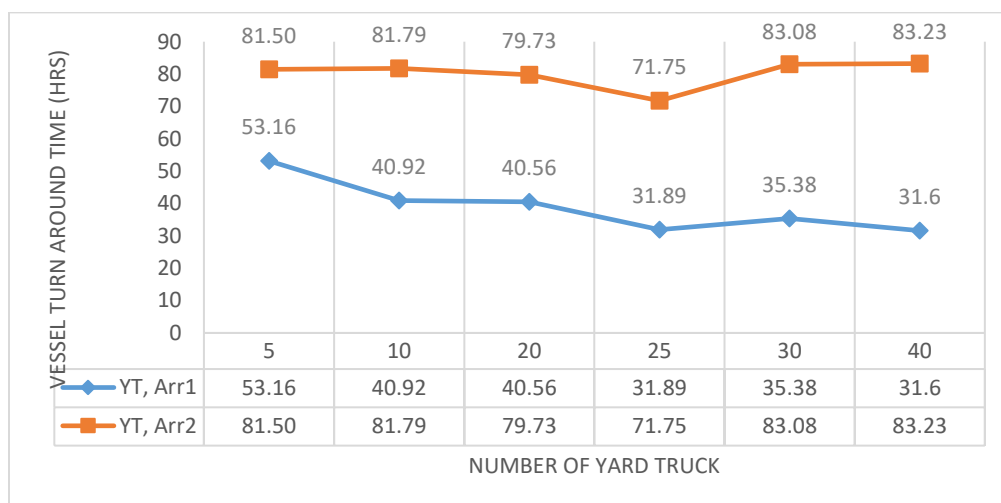


Fig. 12. Regular and increased arrivals, TAT Vs. YTs.

When tracking the total containers processed for this same case as shown in Fig. 13, it seems that the number of containers decreases when there are 40 yard trucks, which means that having more yard trucks in the system, introduces congestion, and this also explains the decrease in the vessel turn-around time for that same instance (Fig. 12), with fewer containers to process, the vessels will spend less time in the system.

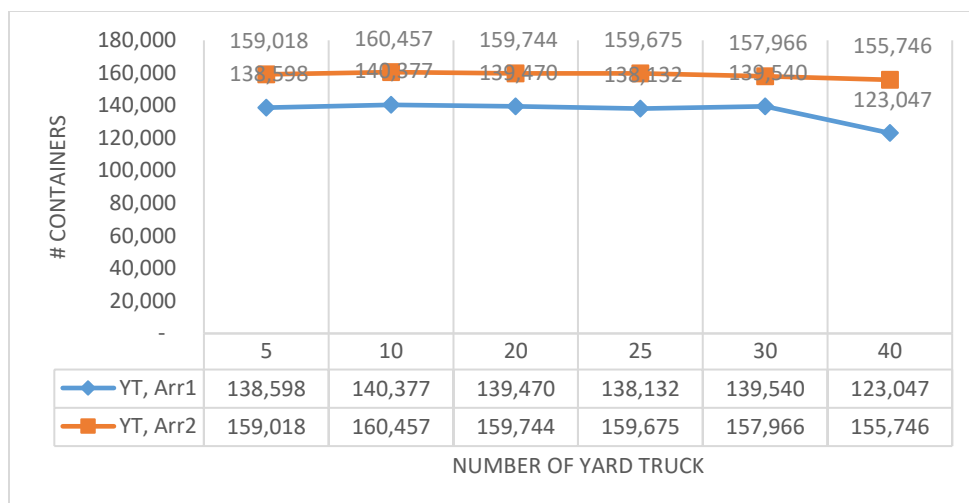


Fig. 13. Regular and increased arrivals, # Containers Vs. YTs.

Fig. 14 shows the comparison of the vessel turn-around time with both the regular and increased arrival when the number of RTGs was changed. It can be concluded that, with lower number of RTGs the system gets congested as well, due to the lack of resources to process the containers, thus creating a bottle neck, which explains the low value of the time spent by the vessel in the system.

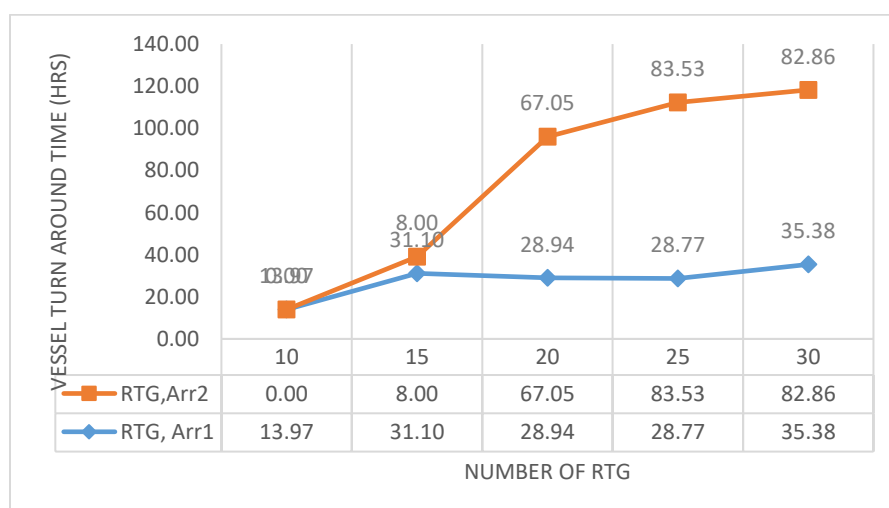


Fig. 14. Regular and increased arrivals, TAT Vs. RTGs.

This can also be noted when considering the number of containers processed, as shown in Fig. 15. Since the system's congestion is caused by fewer resources, eventually fewer containers will be processed.

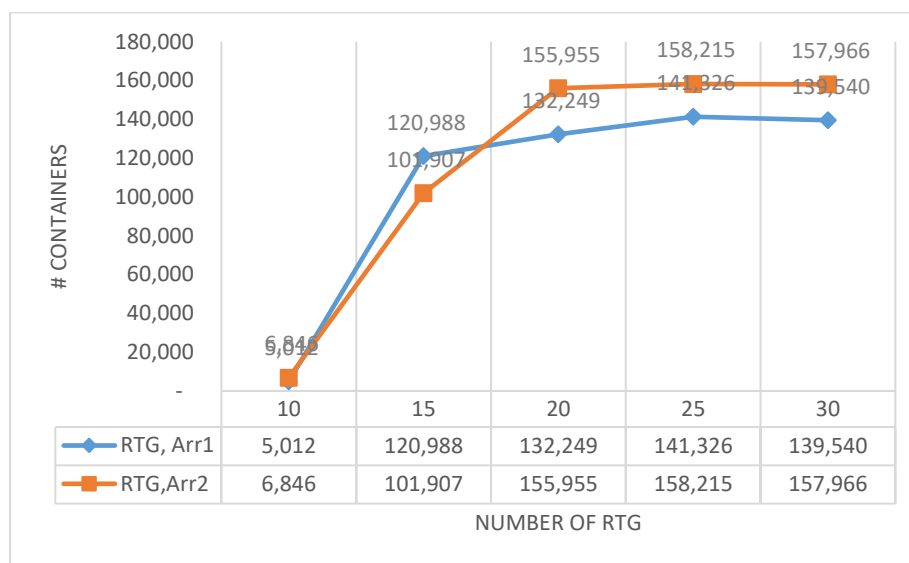


Fig. 15. Regular and increased arrivals, # Containers vs. RTG.

The sensitivity analysis confirms that the system can process containers under extreme conditions, whether an increased arrival or a combination of increase and decrease in the number of resources. However, an increase in resources does not always cause an improvement in the system's behavior since this increase might create congestion that leads to a bottleneck that would affect the system's overall throughput and behavior.

CHAPTER 5

SEQUENCE-BASED SIMULATION OPTIMIZATION FRAMEWORK

This work investigates the influence of incorporating a sequence that governs the search space for a simulation-based optimization. In this chapter, the framework developed, methods used, and approaches taken to construct such a platform are discussed.

The design and infrastructure are introduced, followed by a thorough discussion of the necessary building blocks constructed to allow for seamless communication and function within the system. Lastly, the implementation of the simulation-optimization model is discussed as well as relevant modifications made to incorporate a governing optimization sequence.

5.1 Simulation-Optimization Infrastructure

The simulation-optimization framework has a two-stage iterative process. As shown in Fig. 16, the process starts with an Optimizer Control that determines an initial solution and queries the Simulation Model to execute the simulation given this solution. In the second stage, the Simulation Model returns the performance measures of interest to the Optimizer Control using a feedback loop which allows the Optimizer to reevaluate the initial parameters and use a heuristic approach to reach the best solution iteratively.

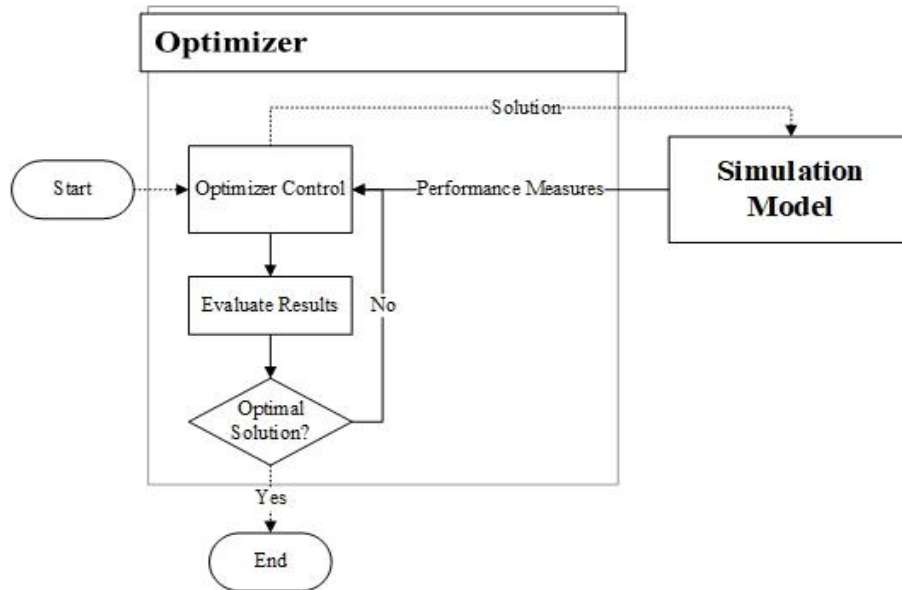


Fig. 16. Simulation Optimization Framework

5.1.1 Running the Simulation Model Externally

As mentioned in Chapter 2, the simulation platform considered in this dissertation is Arena 15.0. Arena software's underlying language is SIMAN. When SIMAN was first designed, it was meant to be able to operate on a personal computer with 256 K of memory. To be able to have some memory for a practical size simulation model, the simulation functions were divided into five separate programs (executables) [204]:

- *Siman.exe* is the simulation engine.
- *Model.exe* is where the simulation network model is read, compiled, and reported.
- *Expmt.exe* is where the simulation experiment frame is processed.
- *Linker.exe* is where the resulting data files were reconciled and combined.
- *Output.exe* is a batch processor of simulation data files, which is the final program.

This divided structure still exists in current Arena models, specifically under the installed ARENA software file in addition to some necessary direct link library (DLL) files. For purposes

of this dissertation, these files are essential for running the Arena program externally, meaning there will not be a need for opening the Arena file to manually run the simulation model.

Thus, in practice, once the simulation model in Arena is verified and validated, both the *.mod* and *.exp* files can be automatically created by writing them from within Arena (RUN> SIMAN> Write). These files are then utilized to run the simulation model externally using the mentioned executables, following the steps:

- Open “expmt.exe”, print “Arenafilename.exp”, print “Arenafilename.e”.
- Open “model.exe”, print “Arenafilename.mod”, print “Arenafilename.m”.
- Open “linker.exe”, print “Arenafilename.m”, print “Arenafilename.e”, print “Arenafilename.p”.
- Open “siman.exe”, print “Arenafilename.p”.

This last step will run the Arena model externally. These steps can be implemented in any programming platform to be automated, as explained in the next section. It is crucial to locate all the SIMAN running *.dll* files in addition to all the *.exe* files in the same file directory (same folder) as the programming platform, in addition to the *.doe* Arena program file of the desired simulation model.

5.1.2 Simulation-Optimization: Input, Output and Automation

The programming platform utilized for this research work is C++. This program was developed to run and control the Arena model externally, by successfully implementing and automating the previous steps and the various building blocks within the introduced framework (see 5.1). To automate the simulation-optimization platform, a link must be established to connect

the Simulator with the Optimizer to manage input and output, information and data flow, and command communication.

As discussed in Chapter 4, when constructing the discrete-event simulation model in Arena, some of its *read/write* features were utilized to successfully make incoming and outgoing information available externally. In this manner, the optimizer program runs the simulation model externally by calling the necessary Arena *.exe* files using the Windows Command Processor. The simulation model will then read the text file provided by the optimizer (solution string) and use the values as its running parameters. Before Arena terminates the simulation run, it will utilize a similar direct read/write feature, but in this case to write data directly from variables and attributes to report the behavior and performance of the simulation. This writing function can take place any time during the simulation run, thus the output data is exported into a text file which can be accessed by the optimizer.

As shown in Fig. 16, the optimizer will be responsible for sending solutions to the simulation model, running the simulation model, and evaluating the performance measures. This feedback loop will continue until a termination criterion is met, at which time the process will terminate this loop. The established connections between the optimization and the simulation modules will achieve the following steps:

1. Initial solution
 - 1.1. Start with a candidate initial feasible solution
 - 1.2. Write solution (input) into text file
 - 1.3. Signal the simulation model to run with the current solution
2. Simulation
 - 2.1. Evaluate candidate solution by executing the simulation model

- 2.2. Write fitness (output) into text file
3. Optimization
 - 3.1. Evaluate fitness (Simulation results)
 - 3.2. If terminating criteria is met (or convergence) then terminate loop
 - 3.3. If not optimal, then improve: create a new solution then go to 1.2

5.2 Evolutionary Algorithm-Based Optimization (Non-Sequence-Based)

In this section the optimization part of the Simulation-Optimization infrastructure is discussed. Evolutionary algorithm-based optimization is the optimization method considered in this dissertation.

5.2.1 Solution Representation, Initial Population and Generations

The container terminal resources in the simulation model are represented by a solution string (chromosome) and each element (gene) in the string refers to a changeable value (decision variables) of a resource being considered for optimization in the simulation model.

At the first generation, the Optimizer randomly generates an initial population of strings in which each string consists of a set of variables whose values are randomly selected from the predefined list of values given in Table 4. A string is used as an input to the simulation model to evaluate and produce a fitness value that corresponds to one or more of the simulation outputs. Every string in the initial population is evaluated in this way before evolving it to the next generation. Every generation that follows has the same population size.

Table 4: Terminal section, resources and levels

| Solution String Index | Resources | Possible Values |
|------------------------------|---------------------------|--------------------------------------|
| 1 | <i>Berth Size</i> | (25%, 50%, 75%, 100%) |
| 2 | <i>STS Cranes</i> | (3, 4, 5, 6, 7, 8) |
| 3 | <i>Gates</i> | (6, 9, 12, 15) |
| 4 | <i>Yard Trucks</i> | (15, 20, 25, 30, 35, 40, 45, 50, 55) |
| 5 | <i>Exports rows</i> | (1, 2, 3) |
| 6 | <i>Imports rows</i> | (4, 5, 6, 7, 8, 9, 10) |
| 7 | <i>Yard Crane per row</i> | (1, 2, 3, 4) |

5.2.2 Mutation, evolution, and termination

In evolutionary algorithms, a string is randomly selected, mutated and competed with another randomly selected string from the population. The fitness of the offspring (i.e., the mutated string) is evaluated and is compared to the fitness of the competing string. The winner (i.e., the string with the better fitness) makes it to the second generation and the loser is disposed. This process is repeated for n times where n is the population size. As such, a new generation with better average fitness will be created. Proceeding in this manner, the population of strings evolve from one generation to the next with improved average fitness values. Note that using this approach, the goodness of fit of future generations will never decline as only individuals with the best fitness values will make it to new generations. The process converges when all strings in the population have the same fitness value. If convergence is deemed to be too time consuming to reach, the process can be stopped after a predefined number of generations as defined by the analyst. In this method where the solution string consists of the number of resources to allocate, the mutation is to randomly pick one of the seven resources of the parent string and change its value up or down by one level.

5.2.3 Fitness Function

The Fitness function can mainly be one or more of the simulation outputs to reflect and evaluate the behavior of the simulation model and how it is influenced by each change in the inputs. Such outputs can be throughput, resource utilization, waiting times, delay times, etc. A meaningful fitness function in this application is the throughput, which is the total number of exported and imported containers via the container terminal. However, setting the fitness function in this manner will naturally set the levels for all resources to the highest possible value since more resources typically produce more throughput. To overcome this, and since cost is not captured in this simulation, the weighted average resource utilization was incorporated into the fitness function to challenge the Optimizer to find a balance between the two, as follows:

$$\text{Fitness} = \text{throughput} \times \text{weighted average resource utilization} \quad (1)$$

As such, the optimal fitness will be driven by a search to find the configuration that produces the most throughput with the least resources since resource utilization increases with fewer resources used.

However, since resources vary in terms of number and influence, this effect was accounted for when computing the fitness function. For instance, the utilization of a truck should not be weighed as that of the berth. A utilization equation was, therefore, formulated where appropriate coefficients were used to represent these weights based on the resource market price. Namely, the research yielded that one STS Crane costs around 5 M dollars, whereas a yard truck costs 0.1 M, etc. Accordingly, the weighted average resource utilization equation is:

$$0.63 \text{ Berths} + 0.32 \text{ Cranes} + 0.031 \text{ RTGs} + 0.013 \text{ Gates} + 0.006 \text{ Trucks} \quad (2)$$

the utilization equation considers actual resources with utilization only; thus, the number of imported and exported container rows are excluded due to the lack of an actual usage value.

The simulation model is executed once every iteration using the number of resources given by the solution string. It will then output the total containers throughput and the average utilization per resource to the Optimizer, which will calculate the fitness based on the fitness function given in (1).

5.3 Sequence-Based Evolutionary Algorithm for Optimization

This section will discuss the necessary modifications implemented to incorporate the optimization algorithm described in the preceding section with a sequence that governs the order in which resources are optimized. The hypothesis is that implementing such a sequence will reach a comparable solution in less computation time. This approach is anticipated to reduce the search space and improve the efficiency of the optimization process.

5.3.1 Solution representation, initial population and generations

In this method, the solution string represents the order (sequence) in which the resources are optimized, with the aim of finding the order that results in the best fitness value. Each resource is represented by its location number in the solution string (Table 4). For example, in the string 1-2-7-5-6-4-3, the algorithm starts by optimizing the berth first and fix its best value, it then optimizes and fixes the number of STS cranes (2nd variable) before moving to the 7th variable which is the number of RTGs or yard cranes whose value is then fixed and is followed by the 5th

variable which is the export rows and so on until we finally optimize and fix the 3rd variable which is the Gates (see Table 4).

The initial population consists of a set of randomly generated sequences whose fitness is obtained by running the simulation with optimizing and fixing variable values in the order they appear in the string as explained earlier.

5.3.2 Mutation, evolution, and termination

Unlike traditional evolutionary algorithms that mutate the value of the variables themselves, the mutation in this algorithm swaps the sequence of two variables, by randomly selecting two resources and swapping their location in the sequence, thus creating a new mutated sequence. The evolution process in this method does not differ from the previous one. The termination condition, however, is pre-defined at the beginning of the simulation-optimization by placing a limit on the number of generations for the sequence-based algorithm since it is computationally expensive to reach convergence due to the large number of simulation runs required per solution string.

5.3.3 Fitness Function

The fitness in this method is calculated similar to the previous non-sequence-based method using equation (2), but it is more complex as it requires more than one simulation run for each sequence of variables to be evaluated. Since a sequence consists of seven variables in this case, the simulation model must run several times for each variable which is dependent on the number of variable levels as given in Table 4. Once the best value for a certain variable is reached, the resources for that variable are fixed to the value that produced the best fitness, then it proceeds to find the best value for the next variable in the sequence and so on until all variables are optimized

sequentially. The resulting fitness after computing all simulation runs required to evaluate a single sequence-based solution string is reported to the Optimizer as the fitness value for this sequence.

A Similar process is followed for each sequence or string in the population.

5.3.4 Stage 1: Partial ordering for obtaining the sequence

Once the last generation for the sequence-based optimization is reached, the n best sequences are obtained. Each sequence corresponds to an ordering of the variables that led to a good fitness. To determine the best overall sequence, a partial ordering is created by determining c_{ij} for each pair of variables (v_i and v_j), where c_{ij} counts the number of times that v_i precedes v_j in the n sequences of the generation. If c_{ij} exceeds a value $T.n$, then v_i is considered to precede v_j in the overall optimization sequence. $T \in [0.5, 1]$ is a threshold value where the higher the threshold T , the higher the stringency of the partial ordering relation. A threshold of $T = 1$ would mean that v_i must precede v_j in all n sequences of the generation to be considered a legitimate ordering relation.

The partial ordering relation can be represented as a directed graph $G = (N, E)$, such that:

- N_i is the representative node for the variable i .
- E is the set of edges c_{ij} between N_i and N_j , such that $c_{ij} > T.n$
- G is a legitimate graph for the ordering relation between variables $v_1 \dots v_n$ if and only if G is a directed acyclical graph.

Fig. 17 shows the variables' partial ordering graph based on the sequences presented in

Table 5. For example, variable 7 (represented by node 7) is preceded by variables 2, 1, 4, and 3. Also, node 1 has [7] as successors; node 2 has [3, 6, 7] as successors; node 3 has [7] as

successors; node 4 has [7] as successors; node 6 has [1] as successors; and nodes 5 and 7 have no successors. In this case, for a threshold $T = 0.6$, the “optimal” sequence is: $2 \rightarrow (3,4) \rightarrow 6 \rightarrow 1 \rightarrow (5 \rightarrow 7)$.

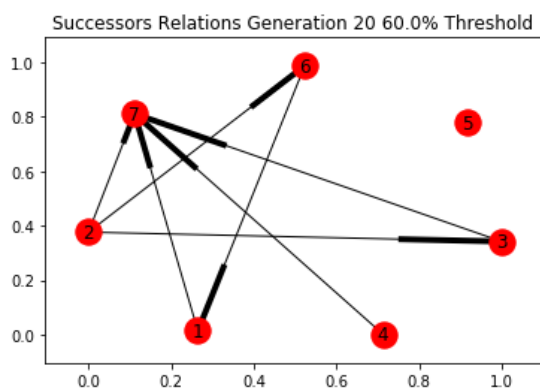


Fig. 17. Variables' Relationship Graph.

Table 5: The sequence of the last generation

| Seq # | |
|-------|---------------|
| 1 | 4 7 6 3 2 1 5 |
| 2 | 5 2 4 3 6 1 7 |
| 3 | 6 7 5 1 3 4 2 |
| 4 | 5 2 6 3 4 1 7 |
| 5 | 2 3 5 4 1 7 6 |
| 6 | 4 1 2 3 7 6 5 |
| 7 | 5 2 6 3 1 4 7 |
| 8 | 4 2 6 3 7 1 5 |
| 9 | 6 1 2 3 7 4 5 |
| 10 | 1 3 5 4 2 7 6 |

If there is a cycle in G , it means that the generation has not sufficiently converged to determine the ordering relation. Two solutions can be envisaged in this case; more generations can be created for the sequence optimization to obtain a more cohesive set of best sequences, or the threshold T can be raised to achieve a more stringent ordering. The best sequence S is given by the longest path in the directed acyclic graph. When T is raised too high, or when the sequence optimization has not yet converged, the best sequence might not involve all variables.

5.3.5 Stage 2: optimizing variables in the sequence

After obtaining the sequence in Stage 1 as discussed in the previous section, the variables are optimized in that order via stochastic sampling, with a decreasing sample size. That is, the simulation is run for many runs for the first variable, and the number of runs is reduced for the subsequent variables as more variables are fixed. When optimizing one variable at a time, every possible value (level) is considered for that variable, while the levels of the remaining variables are determined based on a random selection rule for a predefined number of simulation runs and the fitness is calculated accordingly. The rationale for using this approach is to alleviate the effect of randomness by conducting a substantial number of iterations to expose this variable value to a wide mix of combination values of the other unfixed variables.

5.4 Alternative Validation Scenario: Increased Vessel Arrivals

To validate the proposed method, another scenario is tested by increasing the vessel arrival frequency by 25% and performing similar experiments as discussed in 5.2 and 5.3. Both scenarios were executed for the same container terminal simulation model with similar run parameters for comparison purposes.

CHAPTER 6

COMPUTATIONAL EXPERIMENTS AND RESULTS

The previous chapter discussed the two evolutionary approaches developed. To demonstrate the effectiveness and influence of the proposed sequence-based approach, its performance is compared to the traditional evolutionary approach. Both methods were executed for the same container terminal simulation described earlier with similar run parameters for comparison purposes. Both methods were executed on the same computer running Windows 10 pro with a i7-7820 GHz 2.9 processor, 32 GB of RAM, and a Quadro M1200 GPU. Additionally, the same version and type of the simulation software was used (Arena 15.0 64-bit).

In this chapter the computational experiments conducted in this work are presented. Two scenarios were considered within a container terminal system, the first scenario includes regular vessel arrival and the second scenario with increased vessel arrival.

6.1 Results: First Scenario

In this section the results for the regular vessel arrival will be discussed and analyzed.

6.1.1 Traditional evolutionary optimization algorithm (Non-sequence-based)

In this method, an initial population of size 20 is used. Each generation required an average of 25 minutes to compute. The algorithm was terminated at generation 200, as no improvement was observed since generation 167 (Table 6). It was concluded that convergence occurred at generation 167 after 75 hours of computation time, where 18 out of the 20 members of the population carried the same fitness value.

Table 6: Population data after 20 births (generation 167) of the traditional (non-sequence-based) evolutionary optimization algorithm

| Individual | Birthdate | Utilization | Through-put | Fitness | Solution String |
|------------|-----------|-------------|-------------|---------|-----------------------|
| 1 | 1200 | 0.825427 | 55392 | 45722 | (12 3 9 55 3 10 1) |
| 2 | 3056 | 0.825427 | 55392 | 45722 | (12 3 9 55 3 10 1) |
| 3 | 2780 | 0.825427 | 55392 | 45722 | (12 3 9 55 3 10 1) |
| 4 | 1940 | 0.825427 | 55392 | 45722 | (12 3 9 55 3 10 1) |
| 5 | 456 | 0.825427 | 55392 | 45722 | (12 3 9 55 3 10 1) |
| 6 | 1407 | 0.825427 | 55392 | 45722 | (12 3 9 55 3 10 1) |
| 7 | 1919 | 0.825427 | 55392 | 45722 | (12 3 9 55 3 10 1) |
| 8 | 1790 | 0.825427 | 55392 | 45722 | (12 3 9 55 3 10 1) |
| 9 | 2132 | 0.823941 | 55392 | 45639 | (12 3 12 55 3 10 1) |
| 10 | 2106 | 0.825427 | 55392 | 45722 | (12 3 9 55 3 10 1) |
| 11 | 2268 | 0.825427 | 55392 | 45722 | (12 3 9 55 3 10 1) |
| 12 | 2497 | 0.823941 | 55392 | 45639 | (12 3 12 55 3 10 1) |
| 13 | 1529 | 0.825427 | 55392 | 45722 | (12 3 9 55 3 10 1) |
| 14 | 1531 | 0.825427 | 55392 | 45722 | (12 3 9 55 3 10 1) |
| 15 | 1853 | 0.825427 | 55392 | 45722 | (12 3 9 55 3 10 1) |
| 16 | 3294 | 0.825427 | 55392 | 45722 | (12 3 9 55 3 10 1) |
| 17 | 2062 | 0.825427 | 55392 | 45722 | (12 3 9 55 3 10 1) |
| 18 | 3338 | 0.825427 | 55392 | 45722 | (12 3 9 55 3 10 1) |
| 19 | 1604 | 0.825427 | 55392 | 45722 | (12 3 9 55 3 10 1) |
| 20 | 2274 | 0.825427 | 55392 | 45722 | (12 3 9 55 3 10 1) |

The optimal solution string in (Table 4) indicates the optimal levels for all the resources to produce the highest possible fitness. Recall that the solution string presented represents the resource levels presented in Table 7.

Table 7: Scenario 1: Result of the traditional evolutionary method

| Utilization | Throughput | Fitness | Solution String |
|-------------|------------|---------|---------------------|
| 83% | 55,392 | 45,722 | (50% 3 9 55 3 10 1) |

6.1.2 Sequence-based evolutionary method

In this method, an initial population of size 10 was used instead of 20 as the process of finding the fitness per sequence is much more computationally intensive. Each generation required an average

of 110 minutes to compute. The algorithm was terminated at generation 27, as no significant improvement was observed since generation 20. It was concluded that convergence occurred at generation 20 after 37 hours of computation time. Analyzing the sequences resulting at generation 20 and exploring the sequence pair dependencies produced the following sequence: 2 3 4 6 1 5 7.

This sequence indicates that

the algorithm starts by optimizing the STS cranes first and fix its best value, it then optimizes and fixes the number of gates (3rd variable) before moving to the 4th variable which is the number of yard trucks whose value is then fixed and is followed by the 6th variable which is the import rows and so on until we finally optimize and fix the 7th variable which is the RTGs or yard cranes (see Table 9).

This sequence was then used in Stage 2 to control the order in which variables are optimized as discussed in section 5.3.5. This evaluation process required 13 hours of simulation run time. The optimal solution string from this method is shown in (Table 8) which shows the optimal sequence of the variables in the solution string in addition to the optimal level for that variable or resource, to achieve the best possible fitness. In total, the two-stage sequence-based approach required $37 + 13 = 50$ hours of computation time. To compare with the previous method, Table 9 shows the utilization, throughput and fitness of the optimal solution string.

Table 8: Scenario 1: Solution string for sequence-based method

| | | | | | | | |
|-----------------------|---|---|----|---|------|---|---|
| Order/Sequence | 2 | 3 | 4 | 6 | 1 | 5 | 7 |
| Resource Value | 3 | 9 | 30 | 9 | 100% | 2 | 1 |

Table 9: Scenario 1: Result of the sequence-based evolutionary method

| Utilization | Throughput | Fitness | Solution String |
|-------------|------------|---------|---------------------|
| 75% | 57,027 | 42,461 | (100% 3 9 30 2 9 1) |

6.1.3 Discussion: First Scenario

Following the proposed steps of finding the variable sequence first then optimizing them in that order, it is evident that the proposed sequence-based method reached a comparable solution, a fitness of 42,461, in 66% of the time, when compared to the traditional non-sequenced evolutionary approach.

In the traditional evolutionary method, the algorithm is required to consider the entire search space as it searches for input variable candidates, whereas when optimizing in a sequence, a predefined path is enforced to limit and reduce the search space to one section at a time, thus significantly reducing the total computational time.

When comparing results from both methods in (Table 10), it can be concluded that that the non-sequence-based method favored a smaller berth size (50%) with a larger yard and more resources; thus, it achieved a higher utilization of the berth but negatively impacted the throughput. The sequence-based method favored a larger berth (100%) with a smaller yard, resulting in lower berth utilization and higher throughput. Furthermore, the solution itself in terms of resource allocation is quite similar to the traditional and more exhaustive method.

Table 10: Scenario 1: Results comparison for the two methods

| | Utilization | Throughput | Fitness | Solution String |
|---------------------------|-------------|------------|---------|---------------------|
| <i>Non-sequence-based</i> | 83% | 55,392 | 45,722 | (50% 3 9 55 3 10 1) |
| <i>Sequence-based</i> | 75% | 57,027 | 42,461 | (100% 3 9 30 2 9 1) |

6.2 Results: Second Scenario

In this section the results for the increased vessel arrival will be discussed and analyzed.

6.2.1 Traditional evolutionary optimization algorithm (Non-sequence-based)

In this method, an initial population of size 20 is used. Each generation required an average of 74 minutes to compute. The algorithm was terminated at generation 173, as no improvement was observed since generation 134. It was concluded that convergence occurred at generation 134 after 166 hours of computation time, where 6 out of the 20 members of the population carried the highest and same fitness value. The optimal solution string in (Table 11) indicates the optimal levels for all the resources to produce the highest possible fitness. Recall that the solution string presented represents the resource levels presented in Table 4.

Table 11: Scenario 2: Result of the traditional evolutionary method

| Utilization | Throughput | Fitness | Solution String |
|-------------|------------|---------|--------------------|
| 80% | 70,159 | 56,069 | (50% 4 6 20 1 8 4) |

6.2.2 Sequence-Based Evolutionary Method

In this method, an initial population of size 10 was used instead of 20 as the process of finding the fitness per sequence is much more computationally intensive. Each generation required an average of 8 hours to compute. The algorithm was terminated at generation 9 since it is computationally expensive to consider more generations. It was concluded that convergence occurred at generation 9 after 75 hours of computation time. Analyzing the sequences and exploring the sequence pair dependencies as discussed in 5.3.4 produced the following sequence:

4 3 7 1 2 6 5. This sequence was then used in Stage 2 to govern the order in which variables are optimized as discussed in 5.3.5. This evaluation process required 23 hours of simulation run time. The optimal solution string from this method is shown in (Table 12) which shows the optimal sequence of the variables in the solution string in addition to the optimal level for that variable or resource, to achieve the best possible fitness. In total, the two-stage sequence-based approach required $75 + 23 = 98$ hours of computation time. To compare with the previous method, Table 13 shows the utilization, throughput and fitness of the optimal solution string.

Table 12: Scenario 2: Solution string for sequence-based method

| | | | | | | | |
|-----------------------|----|---|---|------|---|---|---|
| Order/Sequence | 4 | 3 | 7 | 1 | 2 | 6 | 5 |
| Resource Value | 50 | 9 | 2 | 100% | 3 | 7 | 2 |

Table 13: Scenario 2: Result of the sequence-based evolutionary method

| Utilization | Throughput | Fitness | Solution String |
|--------------------|-------------------|----------------|------------------------|
| 86% | 60,274 | 52,002 | (100% 3 9 50 2 7 2) |

6.2.3 Discussion: Second Scenario

Following the proposed steps of finding the variable sequence first and then optimizing them in that order, it is evident that the proposed sequence-based method reached a comparable solution, a fitness of 52,002, in 60% of the time, when compared to the traditional non-sequenced evolutionary approach.

When comparing results from both methods in (Table 14), it can be concluded that both methods were able to achieve comparable results and favored a similar container terminal size. However, it can be noted that since this scenario considered an increased vessel arrival, meaning

the system will process more vessels carrying more containers, the system is generally congested. Therefore, whatever resources are allocated in this case, they will be working at full capacity, which produces higher throughput and resource utilization. This explains why, in the non-sequenced-based method, higher throughput was coupled with a lower number of resources, while the sequenced-based method had more resources and lower throughput.

Table 14: Scenario 2: Results comparison for the two scenarios

| | Utilization | Throughput | Fitness | Solution String |
|---------------------------|--------------------|-------------------|----------------|------------------------|
| <i>Non-sequence-based</i> | 90% | 60,585 | 54,978 | (100% 3 6 20 1 5 2) |
| <i>Sequence-based</i> | 86% | 60,274 | 52,002 | (100% 3 9 50 2 7 2) |

CHAPTER 7

CONCLUSIONS AND FUTURE WORK

Operations research and mathematical algorithms can provide near-optimal solutions for problems that are usually encountered in a container terminal. Such optimization algorithms can be utilized to search solution space to support decisions for resource allocation and assignment. Additionally, simulation models can aid in predicting and studying the behavior of the system over time and monitoring its performance under stochastic and unforeseen circumstances.

In this dissertation, a search-space reduction framework was developed and validated to reduce the computational effort of simulation-optimization algorithms through decreasing the number of possible solutions to consider and evaluate. The method was applied to a complex container terminal system and is compared to a traditional evolutionary method that optimizes the variables directly without considering the sequence by which these variables are optimized.

The *first aim* of this work is to develop a large-scale discrete-event simulation model for the newly constructed container terminal of Hamad's new port of Qatar. The model captures the flow of vessels, containers, and external trucks as well as important resources including STS cranes, yard trucks, and yard cranes. The model was validated by comparing its output to historical data obtained from the current port of Doha. Two scenarios were executed in which resources were reduced to reflect a policy change. The results indicate that since the port is equipped with more resources than is necessary for the current demand levels and in anticipation of increase in future demands, the moderate reduction in resources did not impact the vessel turnaround time. However, significantly reducing the number of yard trucks will have a more profound impact on vessel turnaround time.

After establishing the accuracy of the developed simulation model, the *second aim* of this work is to develop an optimization technique not addressed in the literature and investigate its utility using the simulation platform. For the resource allocation problem in the container terminal, the relative order (sequence) in which variables (resources) are optimized was investigated. This has played a significant role in identifying promising search regions to consider, and in turn, reduced the amount of computations necessary to reach optimal or near-optimal solutions. Results show that optimization of decision variables sequentially can indeed reduce the total computational time with solution quality that is comparable to traditional simulation-optimization methods.

Future research includes investigating the effectiveness of the developed sequence-based simulation optimization framework: (i) for different scenarios that reflect changes in the layout or the policies in the container terminal, (ii) for different optimization techniques (other than an evolutionary-based algorithm), and (iii) for a different complex real-world system (other than container terminals).

REFERENCES

- [1] M. H. Petering, "Real-time container storage location assignment at an RTG-based seaport container transshipment terminal: problem description, control system, simulation model, and penalty scheme experimentation," (in English), *Flexible Services and Manufacturing Journal*, pp. 1-31, 2013/10/30 2013.
- [2] A. Bruzzone, F. Longo, L. Nicoletti, and R. Montanari, "Simulation, analysis and optimization of container terminals processes," *International Journal of Modeling, Simulation, and Scientific Computing*, vol. 3, no. 04, p. 1240006, 2012.
- [3] S. Sacone and S. Siri, "An integrated simulation-optimization framework for the operational planning of seaport container terminals," *Mathematical and Computer Modelling of Dynamical Systems*, vol. 15, no. 3, pp. 275-293, 2009.
- [4] R. de Carvalho Miranda, J. A. B. Montevechi, A. F. da Silva, and F. A. S. Marins, "Increasing the efficiency in integer simulation optimization: Reducing the search space through data envelopment analysis and orthogonal arrays," *European Journal of Operational Research*, vol. 262, no. 2, pp. 673-681, 2017.
- [5] K.-H. Chang and P.-Y. Kuo, "An efficient simulation optimization method for the generalized redundancy allocation problem," *European Journal of Operational Research*, vol. 265, no. 3, pp. 1094-1101, 2018.
- [6] M. Dehghanimohammadabadi, T. K. Keyser, and S. H. Cheraghi, "A novel Iterative Optimization-based Simulation (IOS) framework: An effective tool to optimize system's performance," *Computers & Industrial Engineering*, vol. 111, pp. 1-17, 2017.
- [7] G. Ilati, A. Sheikholeslami, and E. Hassannayebi, "A simulation-based optimization approach for integrated port resource allocation problem," *PROMET-Traffic&Transportation*, vol. 26, no. 3, pp. 243-255, 2014.
- [8] J.-P. Rodrigue, C. Comtois, and B. Slack, *The geography of transport systems*. Routledge, 2013.
- [9] J. Martin and B. J. Thomas, "The container terminal community," *Maritime Policy & Management*, vol. 28, no. 3, pp. 279-292, 2001.
- [10] Port of Houston Authority. *Photo Gallery*. Available: <http://www.portofhouston.com/inside-the-port-authority/communications/photo-gallery/>
- [11] Y.-G. Chung, S. U. Randhawa, and E. D. McDowell, "A simulation analysis for a transtainer-based container handling facility," *Computers & Industrial Engineering*, vol. 14, no. 2, pp. 113-125, // 1988.
- [12] I. Wilson and P. Roach, "Container stowage planning: a methodology for generating computerised solutions," *Journal of the Operational Research Society*, vol. 51, no. 11, pp. 1248-1255, 2000.

- [13] P. O. Oakland, "2013–2014 MARITIME HANDBOOK," P. O. Oakland, Ed., ed, 2014.
- [14] M. E. H. Petering, "Effect of block width and storage yard layout on marine container terminal performance," *Transportation Research Part E: Logistics and Transportation Review*, vol. 45, no. 4, pp. 591-610, 7// 2009.
- [15] O. Kulak, O. Polat, R. Gujjula, and H.-O. Günther, "Strategies for improving a long-established terminal's performance: a simulation study of a Turkish container terminal," (in English), *Flexible Services and Manufacturing Journal*, vol. 25, no. 4, pp. 503-527, 2013/12/01 2013.
- [16] W. C. Ng and K. L. Mak, "Yard crane scheduling in port container terminals," *Applied Mathematical Modelling*, vol. 29, no. 3, pp. 263-276, 3// 2005.
- [17] C. Bierwirth and F. Meisel, "A survey of berth allocation and quay crane scheduling problems in container terminals," *European Journal of Operational Research*, vol. 202, no. 3, pp. 615-627, 2010.
- [18] A. Law, *Simulation Modeling and Analysis*, 4th ed. McGraw-Hill Science/Engineering/Math, 2006.
- [19] W. D. Kelton, R. P. Sadowski, and N. B. Swets, *Simulation with Arena*, 5th ed. McGraw-Hill Science/Engineering/Math, 2009.
- [20] J. Banks, J. S. Carson II, B. L. Nelson, and D. M. Nicol, *Discrete-Event System Simulation*, 5th ed. Prentice Hall, 2009.
- [21] K. A. De Jong, *Evolutionary computation: a unified approach*. MIT press, 2006.
- [22] I. Rechenberg, "Cybernetic solution path of an experimental problem," *Library Translation*, vol. 1122, no. Farnborough: Royal Aircraft Establishment., 1965.
- [23] L. Fogel, A. Owens, and M. Walsh, *Artificial Intelligence through Simulated Evolution*. New York: John Wiley & Sons, 1966.
- [24] J. H. Holland, "Outline for a logical theory of adaptive systems," *Journal of the ACM (JACM)*, vol. 9, no. 3, pp. 297-314, 1962.
- [25] J. Holland, "Nonlinear environments permitting efficient adaptation " *Computer and Information Sciences*, vol. II, no. Academic Press, 1967.
- [26] R. Tavakkoli-Moghaddam, A. Makui, S. Salahi, M. Bazzazi, and F. Taheri, "An efficient algorithm for solving a new mathematical model for a quay crane scheduling problem in container ports," *Computers & Industrial Engineering*, vol. 56, no. 1, pp. 241-248, 2// 2009.
- [27] K. H. Kim and Y.-M. Park, "A crane scheduling method for port container terminals," *European Journal of Operational Research*, vol. 156, no. 3, pp. 752-768, 8/1/ 2004.

- [28] L. Zhen, E. P. Chew, and L. H. Lee, "An Integrated Model for Berth Template and Yard Template Planning in Transshipment Hubs," *Transportation Science*, vol. 45, no. 4, pp. 483-504, 2011.
- [29] L. Zhen, L. H. Lee, and E. P. Chew, "A decision model for berth allocation under uncertainty," *European Journal of Operational Research*, vol. 212, no. 1, pp. 54-68, 7/1/2011.
- [30] R. Trunfio, "A note on: A modified generalized extremal optimization algorithm for the quay crane scheduling problem with interference constraints," *Engineering Optimization*, no. ahead-of-print, pp. 1-17, 2014.
- [31] P. Guo, W. Cheng, and Y. Wang, "A modified generalized extremal optimization algorithm for the quay crane scheduling problem with interference constraints," *Engineering Optimization*, vol. 46, no. 10, pp. 1411-1429, 2014.
- [32] A. Simrin and A. Diabat, "The dynamic berth allocation problem: a linearized formulation," *RAIRO-Oper. Res. doi*, vol. 10, p. 1051, 2014.
- [33] P. Hansen and C. Oguz, *A note on formulations of the static and dynamic berth allocation problems*. Groupe d'études et de recherche en analyse des décisions, HEC Montréal, 2003.
- [34] A. Imai, K. I. Nagaiwa, and C. W. Tat, "Efficient planning of berth allocation for container terminals in Asia," *Journal of Advanced Transportation*, vol. 31, no. 1, pp. 75-94, 1997.
- [35] A. Imai, E. Nishimura, and S. Papadimitriou, "Berthing ships at a multi-user container terminal with a limited quay capacity," *Transportation Research Part E: Logistics and Transportation Review*, vol. 44, no. 1, pp. 136-151, 2008.
- [36] P. Hansen, C. Oğuz, and N. Mladenović, "Variable neighborhood search for minimum cost berth allocation," *European Journal of Operational Research*, vol. 191, no. 3, pp. 636-649, 2008.
- [37] J.-F. Cordeau, G. Laporte, P. Legato, and L. Moccia, "Models and tabu search heuristics for the berth-allocation problem," *Transportation science*, vol. 39, no. 4, pp. 526-538, 2005.
- [38] G. R. Mauri, A. C. Oliveira, and L. A. N. Lorena, "A hybrid column generation approach for the berth allocation problem," in *Evolutionary Computation in Combinatorial Optimization*: Springer, 2008, pp. 110-122.
- [39] M. Han, P. Li, and J. Sun, "The algorithm for berth scheduling problem by the hybrid optimization strategy GASA," in *Control, Automation, Robotics and Vision, 2006. ICARCV'06. 9th International Conference on*, 2006, pp. 1-4: IEEE.
- [40] P. Zhou, H. Kang, and L. Lin, "A dynamic berth allocation model based on stochastic consideration," in *Intelligent Control and Automation, 2006. WCICA 2006. The Sixth World Congress on*, 2006, vol. 2, pp. 7297-7301: IEEE.

- [41] D.-H. Lee, L. Song, and H. Wang, "Bilevel programming model and solutions of berth allocation and quay crane scheduling," in *Transportation Research Board 85th Annual Meeting*, 2006, no. 06-1386.
- [42] A. Imai, H. C. Chen, E. Nishimura, and S. Papadimitriou, "The simultaneous berth and quay crane allocation problem," *Transportation Research Part E: Logistics and Transportation Review*, vol. 44, no. 5, pp. 900-920, 2008.
- [43] C. Liang, Y. Huang, and Y. Yang, "A quay crane dynamic scheduling problem by hybrid evolutionary algorithm for berth allocation planning," *Computers & Industrial Engineering*, vol. 56, no. 3, pp. 1021-1028, 2009.
- [44] G. Giallombardo, L. Moccia, M. Salani, and I. Vacca, "The Tactical Berth Allocation Problem (TBAP) with quay crane assignment and transshipment-related quadratic yard costs," in *European Transport Conference (ETC)*, 2008, no. TRANSP-OR-TALK-2008-015.
- [45] C.-L. Li, X. Cai, and C.-y. Lee, "Scheduling with multiple-job-on-one-processor pattern," *IIE transactions*, vol. 30, no. 5, pp. 433-445, 1998.
- [46] Y. Guan, W.-Q. Xiao, R. K. Cheung, and C.-L. Li, "A multiprocessor task scheduling model for berth allocation: heuristic and worst-case analysis," *Operations Research Letters*, vol. 30, no. 5, pp. 343-350, 2002.
- [47] Y. Guan and R. K. Cheung, "The berth allocation problem: models and solution methods," *Or Spectrum*, vol. 26, no. 1, pp. 75-92, 2004.
- [48] K. Moon, "A mathematical model and a heuristic algorithm for berth planning," *Brain Korea*, vol. 21, 2000.
- [49] K. Park and K. H. Kim, "Berth scheduling for container terminals by using a sub-gradient optimization technique," *Journal of the operational research society*, vol. 53, no. 9, pp. 1054-1062, 2002.
- [50] K. H. Kim and K. C. Moon, "Berth scheduling by simulated annealing," *Transportation Research Part B: Methodological*, vol. 37, no. 6, pp. 541-560, 2003.
- [51] F. Wang and A. Lim, "A stochastic beam search for the berth allocation problem," *Decision Support Systems*, vol. 42, no. 4, pp. 2186-2196, 2007.
- [52] A. Imai, X. Sun, E. Nishimura, and S. Papadimitriou, "Berth allocation in a container port: using a continuous location space approach," *Transportation Research Part B: Methodological*, vol. 39, no. 3, pp. 199-221, 2005.
- [53] D. Chang, W. Yan, C.-H. Chen, and Z. Jiang, "A berth allocation strategy using heuristics algorithm and simulation optimisation," *International Journal of Computer Applications in Technology*, vol. 32, no. 4, pp. 272-281, 2008.

- [54] F. Meisel and C. Bierwirth, "Integration of berth allocation and crane assignment to improve the resource utilization at a seaport container terminal," in *Operations Research Proceedings 2005*: Springer, 2006, pp. 105-110.
- [55] F. Meisel and C. Bierwirth, "Heuristics for the integration of crane productivity in the berth allocation problem," *Transportation Research Part E: Logistics and Transportation Review*, vol. 45, no. 1, pp. 196-209, 2009.
- [56] J. Liu, Y. w. Wan, and L. Wang, "Quay crane scheduling at container terminals to minimize the maximum relative tardiness of vessel departures," *Naval Research Logistics (NRL)*, vol. 53, no. 1, pp. 60-74, 2006.
- [57] Y. Zhu and A. Lim, "Crane scheduling with non-crossing constraint," *Journal of the Operational Research Society*, vol. 57, no. 12, pp. 1464-1471, 2006.
- [58] M. Hendriks, M. Laumanns, E. Lefebvre, and J. T. Udding, "Robust periodic berth planning of container vessels," in *Proceedings of the Third German Korean Workshop on Container Terminal Management: IT-based Planning and Control of Seaport Container Terminals and Transportation Systems*, 2008, pp. 1-13.
- [59] P. Legato, D. Gullì, and R. Trunfio, "The quay crane deployment problem at a maritime container terminal," in *Submitted to the 22th European Conference on Modelling and Simulation*, 2008.
- [60] F. Meisel and C. Bierwirth, "The berth allocation problem with a cut-and-run option," in *Operations Research Proceedings 2008*: Springer, 2009, pp. 283-288.
- [61] R. Moorthy and C.-P. Teo, "Berth management in container terminal: the template design problem," *OR spectrum*, vol. 28, no. 4, pp. 495-518, 2006.
- [62] J. Dai, W. Lin, R. Moorthy, and C.-P. Teo, "Berth allocation planning optimization in container terminals," in *Supply chain analysis*: Springer, 2008, pp. 69-104.
- [63] A. Imai, E. Nishimura, M. Hattori, and S. Papadimitriou, "Berth allocation at indented berths for mega-containerships," *European Journal of Operational Research*, vol. 179, no. 2, pp. 579-593, 2007.
- [64] P. Lokuge and D. Alahakoon, "Improving the adaptability in automated vessel scheduling in container ports using intelligent software agents," *European Journal of Operational Research*, vol. 177, no. 3, pp. 1985-2015, 2007.
- [65] A. Lim, B. Rodrigues, and Z. Xu, "Approximation schemes for the crane scheduling problem," in *Algorithm theory-swat 2004*: Springer, 2004, pp. 323-335.
- [66] A. Lim, B. Rodrigues, and Z. Xu, "Solving the crane scheduling problem using intelligent search schemes," in *Principles and Practice of Constraint Programming—CP 2004*: Springer, 2004, pp. 747-751.

- [67] A. Lim, B. Rodrigues, and Z. Xu, "A m-parallel crane scheduling problem with a non-crossing constraint," *Naval Research Logistics (NRL)*, vol. 54, no. 2, pp. 115-127, 2007.
- [68] D.-H. Lee, H. Q. Wang, and L. Miao, "An approximation algorithm for quay crane scheduling with non-interference constraints in port container terminals," *Tristan VI, Phuket, June*, pp. 10-15, 2007.
- [69] D.-H. Lee, H. Q. Wang, and L. Miao, "Quay crane scheduling with non-interference constraints in port container terminals," *Transportation Research Part E: Logistics and Transportation Review*, vol. 44, no. 1, pp. 124-135, 2008.
- [70] D.-H. Lee, H. Qiu Wang, and L. Miao, "Quay crane scheduling with handling priority in port container terminals," *Engineering Optimization*, vol. 40, no. 2, pp. 179-189, 2008.
- [71] C. F. Daganzo, "The crane scheduling problem," *Transportation Research Part B: Methodological*, vol. 23, no. 3, pp. 159-175, 1989.
- [72] R. I. Peterkofsky and C. F. Daganzo, "A branch and bound solution method for the crane scheduling problem," *Transportation Research Part B: Methodological*, vol. 24, no. 3, pp. 159-172, 1990.
- [73] L. Moccia, J. F. Cordeau, M. Gaudioso, and G. Laporte, "A branch-and-cut algorithm for the quay crane scheduling problem in a container terminal," *Naval Research Logistics (NRL)*, vol. 53, no. 1, pp. 45-59, 2006.
- [74] M. Sammarra, J.-F. Cordeau, G. Laporte, and M. F. Monaco, "A tabu search heuristic for the quay crane scheduling problem," *Journal of Scheduling*, vol. 10, no. 4-5, pp. 327-336, 2007.
- [75] W. Ng and K. Mak, "Quay crane scheduling in container terminals," *Engineering Optimization*, vol. 38, no. 6, pp. 723-737, 2006.
- [76] D. H. Jung, Y.-M. Park, B. K. Lee, K. H. Kim, and K. R. Ryu, "A quay crane scheduling method considering interference of yard cranes in container terminals," in *MICAI 2006: Advances in Artificial Intelligence*: Springer, 2006, pp. 461-471.
- [77] A. Goodchild and C. Daganzo, "Reducing ship turn-around time using double-cycling," *Institute of Transportation Studies*, 2004.
- [78] D. Steenken, S. Voß, and R. Stahlbock, "Container terminal operation and operations research-a classification and literature review," *OR spectrum*, vol. 26, no. 1, pp. 3-49, 2004.
- [79] A. Bruzzone and R. Signorile, "Simulation and genetic algorithms for ship planning and shipyard layout," *Simulation*, vol. 71, no. 2, pp. 74-83, 1998.
- [80] P. Legato and R. M. Mazza, "Berth planning and resources optimisation at a container terminal via discrete event simulation," *European Journal of Operational Research*, vol. 133, no. 3, pp. 537-547, 2001.

- [81] A. Lim, "The berth planning problem," *Operations research letters*, vol. 22, no. 2, pp. 105-110, 1998.
- [82] E. Nishimura, A. Imai, and S. Papadimitriou, "Berth allocation planning in the public berth system by genetic algorithms," *European Journal of Operational Research*, vol. 131, no. 2, pp. 282-292, 6/1/ 2001.
- [83] A. Imai, E. Nishimura, and S. Papadimitriou, "Berth allocation with service priority," *Transportation Research Part B: Methodological*, vol. 37, no. 5, pp. 437-457, 2003.
- [84] Y.-M. Park and K. H. Kim, "A scheduling method for berth and quay cranes," in *Container Terminals and Automated Transport Systems*: Springer, 2005, pp. 159-181.
- [85] L. M. Gambardella, M. Mastrolilli, A. E. Rizzoli, and M. Zaffalon, "An optimization methodology for intermodal terminal management," (in English), *Journal of Intelligent Manufacturing*, vol. 12, no. 5-6, pp. 521-534, 2001/10/01 2001.
- [86] E. K. Bish, "A multiple-crane-constrained scheduling problem in a container terminal," *European Journal of Operational Research*, vol. 144, no. 1, pp. 83-107, 1/1/ 2003.
- [87] L. Lee, E. Chew, K. Tan, and Y. Han, "An optimization model for storage yard management in transshipment hubs," in *Container Terminals and Cargo Systems*, K. Kim and H.-O. Günther, Eds.: Springer Berlin Heidelberg, 2007, pp. 107-129.
- [88] Y. Han, L. Lee, E. Chew, and K. Tan, "A yard storage strategy for minimizing traffic congestion in a marine container transshipment hub," (in English), *OR Spectrum*, vol. 30, no. 4, pp. 697-720, 2008/10/01 2008.
- [89] D. Sculli and C. Hui, "Three dimensional stacking of containers," *Omega*, vol. 16, no. 6, pp. 585-594, 1988.
- [90] I. D. Wilson and P. A. Roach, "Principles of combinatorial optimization applied to container-ship stowage planning," *Journal of Heuristics*, vol. 5, no. 4, pp. 403-418, 1999.
- [91] O. Dubrovsky, G. Levitin, and M. Penn, "A genetic algorithm with a compact solution encoding for the container ship stowage problem," *Journal of Heuristics*, vol. 8, no. 6, pp. 585-599, 2002.
- [92] D. Steenken, T. Winter, and U. T. Zimmermann, "Stowage and transport optimization in ship planning," in *Online optimization of large scale systems*: Springer, 2001, pp. 731-745.
- [93] J.-G. Kang and Y.-D. Kim, "Stowage planning in maritime container transportation," *Journal of the Operational Research Society*, pp. 415-426, 2002.
- [94] B. Cao and G. Uebe, "Solving transportation problems with nonlinear side constraints with tabu search," *Computers & Operations Research*, vol. 22, no. 6, pp. 593-603, 1995.

- [95] K. H. Kim and J. W. Bae, "Re-marshaling export containers in port container terminals," *Computers & Industrial Engineering*, vol. 35, no. 3, pp. 655-658, 1998.
- [96] K. H. Kim and H. B. Kim, "Segregating space allocation models for container inventories in port container terminals," *International Journal of Production Economics*, vol. 59, no. 1, pp. 415-423, 1999.
- [97] K. H. Kim and H. B. Kim, "The optimal sizing of the storage space and handling facilities for import containers," *Transportation Research Part B: Methodological*, vol. 36, no. 9, pp. 821-835, 2002.
- [98] K. H. Kim and H. B. Kim, "The optimal determination of the space requirement and the number of transfer cranes for import containers," *Computers & Industrial Engineering*, vol. 35, no. 3, pp. 427-430, 1998.
- [99] K. H. Kim, Y. M. Park, and K.-R. Ryu, "Deriving decision rules to locate export containers in container yards," *European Journal of Operational Research*, vol. 124, no. 1, pp. 89-101, 2000.
- [100] P. Preston and E. Kozan, "An approach to determine storage locations of containers at seaport terminals," *Computers & Operations Research*, vol. 28, no. 10, pp. 983-995, 2001.
- [101] K. H. Kim and K. T. Park, "A note on a dynamic space-allocation method for outbound containers," *European Journal of Operational Research*, vol. 148, no. 1, pp. 92-101, 2003.
- [102] G. A. E.-N. A. Said and E.-S. M. El-Horbaty, "A Simulation Modeling Approach for Optimization of Storage Space Allocation in Container Terminal," *arXiv preprint arXiv:1501.06802*, 2015.
- [103] K. H. Kim and K. Y. Kim, "An optimal routing algorithm for a transfer crane in port container terminals," *Transportation Science*, vol. 33, no. 1, pp. 17-33, 1999.
- [104] T. G. Crainic, M. Gendreau, and P. Dejax, "Dynamic and stochastic models for the allocation of empty containers," *Operations research*, vol. 41, no. 1, pp. 102-126, 1993.
- [105] W. Shen and C. Khoong, "A DSS for empty container distribution planning," *Decision Support Systems*, vol. 15, no. 1, pp. 75-82, 1995.
- [106] R. K. Cheung and C.-Y. Chen, "A two-stage stochastic network model and solution methods for the dynamic empty container allocation problem," *Transportation science*, vol. 32, no. 2, pp. 142-162, 1998.
- [107] B. Cao and G. Uebe, "An algorithm for solving capacitated multicommodity p-median transportation problems," *Journal of the Operational Research Society*, pp. 259-269, 1993.
- [108] C. Chen, S.-M. Lee, and Q. Shen, "An analytical model for the container loading problem," *European Journal of Operational Research*, vol. 80, no. 1, pp. 68-76, 1995.

- [109] A. P. Davies and E. E. Bischoff, "Weight distribution considerations in container loading," *European Journal of Operational Research*, vol. 114, no. 3, pp. 509-527, 1999.
- [110] G. Scheithauer, "LP-based bounds for the container and multi-container loading problem," *International Transactions in Operational Research*, vol. 6, no. 2, pp. 199-213, 1999.
- [111] M. Eley, "A bottleneck assignment approach to the multiple container loading problem," *OR Spectrum*, vol. 25, no. 1, pp. 45-60, 2003.
- [112] S. Wang and Q. Meng, "Sailing speed optimization for container ships in a liner shipping network," *Transportation Research Part E: Logistics and Transportation Review*, vol. 48, no. 3, pp. 701-714, 2012.
- [113] X. Qi and D.-P. Song, "Minimizing fuel emissions by optimizing vessel schedules in liner shipping with uncertain port times," *Transportation Research Part E: Logistics and Transportation Review*, vol. 48, no. 4, pp. 863-880, 2012.
- [114] C. E. M. Plum, D. Pisinger, and M. M. Sigurd, "A service flow model for the liner shipping network design problem," *European Journal of Operational Research*, vol. 235, no. 2, pp. 378-386, 6/1/ 2014.
- [115] B. D. Brouer, J. F. Alvarez, C. E. Plum, D. Pisinger, and M. M. Sigurd, "A base integer programming model and benchmark suite for liner-shipping network design," *Transportation Science*, vol. 48, no. 2, pp. 281-312, 2013.
- [116] B. D. Brouer, G. Desaulniers, and D. Pisinger, "A matheuristic for the liner shipping network design problem," *Transportation Research Part E: Logistics and Transportation Review*, vol. 72, pp. 42-59, 2014.
- [117] H. Wang, S. Wang, and Q. Meng, "Simultaneous optimization of schedule coordination and cargo allocation for liner container shipping networks," *Transportation Research Part E: Logistics and Transportation Review*, vol. 70, pp. 261-273, 2014.
- [118] A. Agra, M. Christiansen, A. Delgado, and L. M. Hvattum, "A maritime inventory routing problem with stochastic sailing and port times," *Computers & Operations Research*, vol. 61, pp. 18-30, 2015.
- [119] S. Wang, Z. Liu, and M. G. Bell, "Profit-based maritime container assignment models for liner shipping networks," *Transportation Research Part B: Methodological*, vol. 72, pp. 59-76, 2015.
- [120] S. Wang, Z. Liu, and Q. Meng, "Segment-based alteration for container liner shipping network design," *Transportation Research Part B: Methodological*, vol. 72, pp. 128-145, 2015.
- [121] E. Kozan, "Optimising container transfers at multimodal terminals," *Mathematical and Computer Modelling*, vol. 31, no. 10, pp. 235-243, 2000.

- [122] P. J. Meersmans and R. Dekker, "Operations research supports container handling," *Econometric Institute Research Papers* 2001.
- [123] K. G. Murty, J. Liu, Y.-w. Wan, and R. Linn, "A decision support system for operations in a container terminal," *Decision Support Systems*, vol. 39, no. 3, pp. 309-332, 2005.
- [124] K. Fagerholt, "Evaluating the trade-off between the level of customer service and transportation costs in a ship scheduling problem," *Maritime Policy & Management*, vol. 27, no. 2, pp. 145-153, 2000.
- [125] P. Dell'Olmo and G. Lulli, "Planning activities in a network of logistic platforms with shared resources," *Annals of operations Research*, vol. 129, no. 1-4, pp. 155-169, 2004.
- [126] K.-C. Nam and W.-I. Ha, "Evaluation of handling systems for container terminals," *Journal of waterway, port, coastal, and ocean engineering*, vol. 127, no. 3, pp. 171-175, 2001.
- [127] C.-I. Liu, H. Jula, and P. A. Ioannou, "Design, simulation, and evaluation of automated container terminals," *Intelligent Transportation Systems, IEEE Transactions on*, vol. 3, no. 1, pp. 12-26, 2002.
- [128] L. M. Gambardella, A. E. Rizzoli, and M. Zaffalon, "Simulation and planning of an intermodal container terminal," *Simulation*, vol. 71, no. 2, pp. 107-116, 1998.
- [129] E. Kozan and P. Preston, "Genetic algorithms to schedule container transfers at multimodal terminals," *International Transactions in Operational Research*, vol. 6, no. 3, pp. 311-329, 1999.
- [130] K. Aliche, "Modeling and optimization of the intermodal terminal Mega Hub," in *Container Terminals and Automated Transport Systems*: Springer, 2005, pp. 307-323.
- [131] J. J. Evers and S. A. Koppers, "Automated guided vehicle traffic control at a container terminal," *Transportation Research Part A: Policy and Practice*, vol. 30, no. 1, pp. 21-34, 1996.
- [132] G. Bruno, G. Ghiani, and G. Improta, "Dynamic positioning of idle automated guided vehicles," *Journal of Intelligent Manufacturing*, vol. 11, no. 2, pp. 209-215, 2000.
- [133] A. J. R. M. Gademann and S. L. van de Velde, "Positioning automated guided vehicles in a loop layout," *European Journal of Operational Research*, vol. 127, no. 3, pp. 565-573, 2000.
- [134] A. Wallace, "Application of AI to AGV control- agent control of AGVs," *International Journal of Production Research*, vol. 39, no. 4, pp. 709-726, 2001.
- [135] M. van der Heijden, M. Ebben, N. Gademann, and A. van Harten, "Scheduling vehicles in automated transportation systems Algorithms and case study," *OR spectrum*, vol. 24, no. 1, pp. 31-58, 2002.

- [136] C. Y. Leong, "Simulation Study of a Dynamic AGV-container Job Deployment Scheme," SINGAPORE-MIT ALLIANCE, 2001.
- [137] R. L. Moorthy, W. Hock-Guan, N. Wing-Cheong, and T. Chung-Piaw, "Cyclic deadlock prediction and avoidance for zone-controlled AGV system," *International Journal of Production Economics*, vol. 83, no. 3, pp. 309-324, 2003.
- [138] M. Grunow, H.-O. Günther, and M. Lehmann, *Dispatching multi-load AGVs in highly automated seaport container terminals*. Springer, 2005.
- [139] S. Hartmann, *A general framework for scheduling equipment and manpower at container terminals*. Springer, 2005.
- [140] J. K. Lim, K. H. Kim, K. Yoshimoto, J. H. Lee, and T. Takahashi, "A dispatching method for automated guided vehicles by using a bidding concept," in *Container Terminals and Automated Transport Systems*: Springer, 2005, pp. 325-344.
- [141] K. H. Kim and K. Y. Kim, "Routing straddle carriers for the loading operation of containers using a beam search algorithm," *Computers & Industrial Engineering*, vol. 36, no. 1, pp. 109-136, 1999.
- [142] K. Y. Kim and K. H. Kim, "A routing algorithm for a single straddle carrier to load export containers onto a containership," *International Journal of Production Economics*, vol. 59, no. 1, pp. 425-433, 1999.
- [143] J. Bose, T. Reiners, D. Steenken, and S. Voß, "Vehicle dispatching at seaport container terminals using evolutionary algorithms," in *System Sciences, 2000. Proceedings of the 33rd Annual Hawaii International Conference on*, 2000, p. 10 pp. vol. 2: IEEE.
- [144] C.-L. Li and G. L. Vairaktarakis, "Loading and unloading operations in container terminals," *IIE Transactions*, vol. 36, no. 4, pp. 287-297, 2004.
- [145] P. J. Meersmans and A. P. Wagelmans, "Effective algorithms for integrated scheduling of handling equipment at automated container terminals," 2001.
- [146] E. K. Bish, T. Y. Leong, C. L. Li, J. W. Ng, and D. Simchi-Levi, "Analysis of a new vehicle scheduling and location problem," *Naval Research Logistics (NRL)*, vol. 48, no. 5, pp. 363-385, 2001.
- [147] H. Y. Lau and Y. Zhao, "Integrated scheduling of handling equipment at automated container terminals," *International Journal of Production Economics*, vol. 112, no. 2, pp. 665-682, 2008.
- [148] M. Rebollo, J. Vicente, C. Carrascosa, and V. Botti, "A MAS approach for port container terminal management," in *3rd Iberoamerican workshop on DAI-MAS*, 2001, pp. 83-94.

- [149] K. H. Kim, J. S. Kang, and K. R. Ryu, "A beam search algorithm for the load sequencing of outbound containers in port container terminals," *OR spectrum*, vol. 26, no. 1, pp. 93-116, 2004.
- [150] K. H. Kim, K. M. Lee, and H. Hwang, "Sequencing delivery and receiving operations for yard cranes in port container terminals," *International Journal of Production Economics*, vol. 84, no. 3, pp. 283-292, 2003.
- [151] P. H. Koo, W. S. Lee, and D. W. Jang, "Fleet sizing and vehicle routing for container transportation in a static environment," in *Container Terminals and Automated Transport Systems*: Springer, 2005, pp. 123-139.
- [152] R. K. Cheung, C.-L. Li, and W. Lin, "Interblock crane deployment in container terminals," *Transportation Science*, vol. 36, no. 1, pp. 79-93, 2002.
- [153] A. Narasimhan and U. S. Palekar, "Analysis and algorithms for the transtainer routing problem in container port operations," *Transportation science*, vol. 36, no. 1, pp. 63-78, 2002.
- [154] C. Zhang, Y.-w. Wan, J. Liu, and R. J. Linn, "Dynamic crane deployment in container storage yards," *Transportation Research Part B: Methodological*, vol. 36, no. 6, pp. 537-555, 2002.
- [155] R. Linn, J.-y. Liu, Y.-w. Wan, C. Zhang, and K. G. Murty, "Rubber tired gantry crane deployment for container yard operation," *Computers & Industrial Engineering*, vol. 45, no. 3, pp. 429-442, 2003.
- [156] G.-T. Yeo, J.-Y. Pak, and Z. Yang, "Analysis of dynamic effects on seaports adopting port security policy," *Transportation Research Part A: Policy and Practice*, vol. 49, no. 0, pp. 285-301, 3// 2013.
- [157] M. Ji, H. Zhu, Q. Wang, R. Zhao, and Y. Yang, "Integrated strategy for berth allocation and crane assignment on a continuous berth using Monte Carlo simulation," *SIMULATION*, vol. 91, no. 1, pp. 26-42, January 1, 2015 2015.
- [158] L. Byung Kwon, J. Bong Joo, K. Kap Hwan, P. Soon Oh, and S. Jeong Hoon, "A Simulation Study for Designing a Rail Terminal in a Container Port," in *Simulation Conference, 2006. WSC 06. Proceedings of the Winter, 2006*, pp. 1388-1397.
- [159] P. Cortés, J. Muñuzuri, J. Nicolás Ibáñez, and J. Guadix, "Simulation of freight traffic in the Seville inland port," *Simulation Modelling Practice and Theory*, vol. 15, no. 3, pp. 256-271, 3// 2007.
- [160] W. Jie, M. Qing-yu, and W. Wen, "Optimization for container Handling equipments allocation based on ARENA," in *Industrial Engineering and Engineering Management (IE&EM), 2010 IEEE 17Th International Conference on*, 2010, pp. 1194-1197.

- [161] M. E. H. Petering, "Decision support for yard capacity, fleet composition, truck substitutability, and scalability issues at seaport container terminals," *Transportation Research Part E: Logistics and Transportation Review*, vol. 47, no. 1, pp. 85-103, 1// 2011.
- [162] Y. Ziwen, Z. Peilin, and Y. Chuanbo, "Research on the simulation of industry port raw material terminal," in *Computer Design and Applications (ICCD A), 2010 International Conference on*, 2010, vol. 5, pp. V5-133-V5-136.
- [163] G. Rabadi, C. A. Pinto, W. Talley, and J.-P. Arnaout, "Port recovery from security incidents: a simulation approach," in *Bichou, Bell & Evans: Risk Management in Port Operations, Logistics and Supply Chain Security*, vol. Chapter 5, 2007, pp. 83-94.
- [164] A. Caris, G. Janssens, and C. Macharis, "Modelling Complex Intermodal Freight Flows," in *From System Complexity to Emergent Properties*, M. A. Aziz-Alaoui and C. Bertelle, Eds. (Understanding Complex Systems: Springer Berlin Heidelberg, 2009, pp. 291-300.
- [165] U. J. Na and M. Shinozuka, "Simulation-based seismic loss estimation of seaport transportation system," *Reliability Engineering & System Safety*, vol. 94, no. 3, pp. 722-731, 3// 2009.
- [166] F. Longo, "Design and integration of the containers inspection activities in the container terminal operations," *International Journal of Production Economics*, vol. 125, no. 2, pp. 272-283, 6// 2010.
- [167] A. Shabayek and W. Yeung, "A simulation model for the Kwai Chung container terminals in Hong Kong," *European Journal of Operational Research*, vol. 140, no. 1, pp. 1-11, 2002.
- [168] M. Kia, E. Shayan, and F. Ghotb, "Investigation of port capacity under a new approach by computer simulation," *Computers & Industrial Engineering*, vol. 42, no. 2, pp. 533-540, 2002.
- [169] S. Hartmann, "Generating scenarios for simulation and optimization of container terminal logistics," *OR Spectrum*, vol. 26, no. 2, pp. 171-192, 2004.
- [170] W. Y. Yun and Y. S. Choi, "A simulation model for container-terminal operation analysis using an object-oriented approach," *International Journal of Production Economics*, vol. 59, no. 1, pp. 221-230, 1999.
- [171] Y. Choi and W. Yun, "Simulator for Port Container Terminal Using An Object-Oriented Approach," *Report Brain Korea*, vol. 21, 2000.
- [172] I. F. Vis and I. Harika, "Comparison of vehicle types at an automated container terminal," *OR Spectrum*, vol. 26, no. 1, pp. 117-143, 2004.
- [173] E. Kozan, "Comparison of analytical and simulation planning models of seaport container terminals," *Transportation Planning and Technology*, vol. 20, no. 3, pp. 235-248, 1997.

- [174] L. Henesey and J. Törnquist, "Enemy at the gates: introduction of multi-agents in a terminal information community," *Proceedings of Ports and Marinas, Rhodos, Greece*, <http://www.assert.bth.se/publications/HeneseyTornquist.pdf>—last check of address: July, vol. 30, p. 2003, 2002.
- [175] B. Dragović, E. Tzannatos, and N. K. Park, "Simulation modelling in ports and container terminals: literature overview and analysis by research field, application area and tool," *Flexible Services and Manufacturing Journal*, pp. 1-31, 2016.
- [176] E. Nishimura, A. Imai, and S. Papadimitriou, "Yard trailer routing at a maritime container terminal," *Transportation Research Part E: Logistics and Transportation Review*, vol. 41, no. 1, pp. 53-76, 1// 2005.
- [177] Y. Xu, Q. Chen, and X. Quan, "Robust berth scheduling with uncertain vessel delay and handling time," (in English), *Annals of Operations Research*, vol. 192, no. 1, pp. 123-140, 2012/01/01 2012.
- [178] P. Legato, R. M. Mazza, and D. Gullì, "Integrating tactical and operational berth allocation decisions via Simulation–Optimization," *Computers & Industrial Engineering*, vol. 78, no. 0, pp. 84-94, 12// 2014.
- [179] L. Tang, W. Jiang, J. Liu, and Y. Dong, "Research into container reshuffling and stacking problems in container terminal yards," *IIE Transactions*, no. just-accepted, pp. 00-00, 2014.
- [180] G. A. E.-N. A. Said, A. M. Mahmoud, and E.-S. M. El-Horbaty, "Simulation and optimization of container terminal operations: a case study," *arXiv preprint arXiv:1407.6257*, 2014.
- [181] V. D. Nguyen and K. H. Kim, "Dispatching vehicles considering uncertain handling times at port container terminals," *Progress in Material Handling Research, 11th International Material Handling Research Colloquium, Material Handling Industry of America, Milwaukee, USA*, pp. 210-226, 2012.
- [182] J. He, W. Zhang, Y. Huang, and W. Yan, "A simulation optimization method for internal trucks sharing assignment among multiple container terminals," *Advanced Engineering Informatics*, vol. 27, no. 4, pp. 598-614, 10// 2013.
- [183] J. Tao and Y. Qiu, "A simulation optimization method for vehicles dispatching among multiple container terminals," *Expert Systems with Applications*, vol. 42, no. 7, pp. 3742-3750, 5/1/ 2015.
- [184] J.-F. Cordeau, P. Legato, R. M. Mazza, and R. Trunfio, "Simulation-based optimization for housekeeping in a container transshipment terminal," *Computers & Operations Research*, vol. 53, no. 0, pp. 81-95, 1// 2015.
- [185] Michael R. Nevins, Charles M. Macal, and J. C. Joines, "A Discrete-Event Simulation Model for Seaport Operations," *Simulation Councils*, vol. 70, no. 4, pp. 213-223, 1998.

- [186] M. Bielli, A. Boulmakoul, and M. Rida, "Object oriented model for container terminal distributed simulation," *European Journal of Operational Research*, vol. 175, no. 3, pp. 1731-1751, 12/16/ 2006.
- [187] H. Li and D. Wang, "Parallel Simulation-based Optimization on Block Planning and Dynamic Truck Configuration of Container Terminals," *International Journal of Information*, vol. 4, no. 2, pp. 1-8, 2009.
- [188] Q. Zeng and Z. Yang, "Integrating simulation and optimization to schedule loading operations in container terminals," *Computers & Operations Research*, vol. 36, no. 6, pp. 1935-1944, 2009.
- [189] E. Briano, C. Caballini, M. Mosca, and R. Revetria, "A system dynamics decision cockpit for a container terminal: The case of Voltri terminal Europe," *International journal of mathematics and computers in simulation*, vol. 3, no. 2, pp. 55-64, 2009.
- [190] Q. Zeng and Z. Yang, "An approach integrating simulation and Q-learning algorithm for operation scheduling in container terminals," in *Proceedings of the Eastern Asia Society for Transportation Studies Vol. 7 (The 8th International Conference of Eastern Asia Society for Transportation Studies, 2009)*, 2009, pp. 419-419: Eastern Asia Society for Transportation Studies.
- [191] P. Legato, R. M. Mazza, and R. Trunfio, "Simulation-based optimization for discharge/loading operations at a maritime container terminal," *OR spectrum*, vol. 32, no. 3, pp. 543-567, 2010.
- [192] C. Arango, P. Cortés, J. Muñuzuri, and L. Onieva, "Berth allocation planning in Seville inland port by simulation and optimisation," *Advanced Engineering Informatics*, vol. 25, no. 3, pp. 452-461, 8// 2011.
- [193] Q. Yu, C. Zhang, and J. Wang, "The Research of Simulation and Optimization of Multimodal Transport System Based on Intelligent Materials on Container Terminal Job Scheduling," in *Advances in Computer Science, Intelligent System and Environment*, vol. 104, D. Jin and S. Lin, Eds. (Advances in Intelligent and Soft Computing: Springer Berlin Heidelberg, 2011, pp. 479-483.
- [194] O. Sharif and N. Huynh, "Yard crane scheduling at container terminals: A comparative study of centralized and decentralized approaches," *Maritime economics & logistics*, vol. 14, no. 2, pp. 139-161, 2012.
- [195] S. Hartmann, "Scheduling reefer mechanics at container terminals," *Transportation Research Part E: Logistics and Transportation Review*, vol. 51, pp. 17-27, 2013.
- [196] J. He, Y. Huang, W. Yan, and S. Wang, "Integrated internal truck, yard crane and quay crane scheduling in a container terminal considering energy consumption," *Expert Systems with Applications*, vol. 42, no. 5, pp. 2464-2487, 2015.

- [197] J. He, Y. Huang, and D. Chang, "Simulation-based heuristic method for container supply chain network optimization," *Advanced Engineering Informatics*, vol. 29, no. 3, pp. 339-354, 2015.
- [198] J. He, Y. Huang, and W. Yan, "Yard crane scheduling in a container terminal for the trade-off between efficiency and energy consumption," *Advanced Engineering Informatics*, vol. 29, no. 1, pp. 59-75, 2015.
- [199] E. Zehendner, G. Rodriguez-Verjan, N. Absi, S. Dauzère-Pérès, and D. Feillet, "Optimized allocation of straddle carriers to reduce overall delays at multimodal container terminals," *Flexible Services and Manufacturing Journal*, vol. 27, no. 2-3, pp. 300-330, 2015.
- [200] Q. Zeng, A. Diabat, and Q. Zhang, "A simulation optimization approach for solving the dual-cycling problem in container terminals," *Maritime Policy & Management*, vol. 42, no. 8, pp. 806-826, 2015.
- [201] G. A. E.-N. A. Said, A. M. Mahmoud, and E.-S. M. El-Horbaty, "An Integrated Approach for Optimization of Resource Utilization at Seaports," *International Journal of Computer Science Issues*, vol. 12, no. 2, pp. 1694-0784 2015.
- [202] U. Clausen and J. Kaffka, "Development of priority rules for handlings in inland port container terminals with simulation," *Journal of Simulation*, vol. 10, no. 2, pp. 95-102, 2016.
- [203] D. Jiangbo, J. Chun, and G. Peng, "Optimal configuration of checkpoints on free trade port area based on fuzzy simulation optimization," in *Service Operations and Logistics, and Informatics, 2008. IEEE/SOLI 2008. IEEE International Conference on*, 2008, vol. 2, pp. 1520-1525.
- [204] M. S. Seppanen, "Building flexible data-driven models," ed, 2016.

VITA

Mariam Aladdin Kotachi

Department of Engineering Management & Systems Engineering

Old Dominion University

Norfolk, VA 23529

Educational Background

- Ph.D. : August 2018, Old Dominion University, Norfolk, VA, USA
 Major : Engineering Management
 Thesis : “*Sequence-Based Simulation-Optimization Framework with Application to Port Operations at Multimodal Container Terminals*”
- M.Sc. : May 2014, Old Dominion University, Norfolk, VA, USA
 Major : Engineering Management
 Thesis : “*Simulation Modeling and Analysis of Customs-Regulated Container Terminal Operations with Multimodal Transportation*”
- B.Sc. : May 2011, University of Texas at Arlington, Arlington, TX, USA
 Major : Biology

Selected Publications

Journal Articles

- Msakni, M. K.; Diabat, A.; Rabadi, G.; Al-Salem, M.; **Kotachi, M.** (2018). “Exact methods for the quay crane scheduling problem when tasks are modeled at the single container level”. *Computers & Operations Research*.
- **Kotachi, M.**; Rabadi, G.; Obeid, M.F. “Simulation Modeling and Analysis of Complex Port Operations with Multimodal Transportation,” In *Procedia Computer Science*, vol. 20, pp. 229-234, 2013.

Conference Proceedings

- Msakni, M.; Al-Salem, M.; Rabadi, G.; **Kotachi, M.**, Diabat, A., “Quay Crane Scheduling with Vessel Stability.” *emc-ftl-2018 EURO Mini Conference on “Advances in Freight Transportation and Logistics,”* Padova, Italy, March 7th, 2018. (Submitted).
- **Kotachi, M.**; Rabadi, G.; Seck, M.; Msakni, M.; Al-Salem, M.; Diabat, A., “Sequence-Based Simulation Optimization: An Application to Container Terminals.” *2018 IEEE Technology Management Conference (TEMSCON 2018)*, Evanston, IL, USA, June 27th, 2018.