A study on the practical use of operations research and vessels big data in benefit of efficient ports utilization in Panama

Gabriel Fuentes Lezcano

Follow this and additional works at: https://commons.wmu.se/all_dissertations

Part of the Databases and Information Systems Commons

Recommended Citation

This Dissertation is brought to you courtesy of Maritime Commons. Open Access items may be downloaded for non-commercial, fair use academic purposes. No items may be hosted on another server or web site without express written permission from the World Maritime University. For more information, please contact library@wmu.se.
A STUDY ON THE PRACTICAL USE OF OPERATIONS RESEARCH AND VESSELS BIG DATA IN BENEFIT OF EFFICIENT MARITIME PORTS UTILIZATION IN PANAMA

By

GABRIEL FUENTES LEZCANO
Panama

A dissertation submitted to the World Maritime University in partial Fulfilment of the requirements for the award of the degree of

MASTER OF SCIENCE
In
MARITIME AFFAIRS
(SHIPPING MANAGEMENT AND LOGISTICS)

2017
DECLARATION

I certify that all the material in this dissertation that is not my own work has been identified, and that no material is included for which a degree has previously been conferred on me.

The contents of this dissertation reflect my own personal views, and are not necessarily endorsed by the University.

(Signature)
(Date): 13th September 2017

Supervised by: Dr. Daniel Seong-Hyeok Moon
World Maritime University

Internal Assessor: Dr. Dong-Wook Song
World Maritime University

External Assessor: Dr. Jan Hoffmann
United Nations Conference on Trade and Development
ACKNOWLEDGEMENTS

First and foremost, I have to thank my wife Rosario Hernandez for her continuous support, love and inspiration that encouraged me in giving that extra mile needed for this project.

My deepest appreciation to my supervisor Dr. Daniel Moon for his constructive advice, dedication, time and availability to address my questions and guide me through this dissertation.

I gratefully acknowledge the funding sources given by ITF Seafarers Trust so I could undertake studies at World Maritime University.

I will like to thank Mr. Juan Carlos Croston and Mrs. Larissa Barrios from Manzanillo International Terminal for their openness, availability and feedback such as per the supplied information very much needed as central part of this dissertation.

Particularly, I will like to thank my parents and sister for their emotional support and encouragement throughout my life.

Last but not least, I will like to thank my colleagues, professors and staff of WMU for their support and all the professional and life lessons received from them.
ABSTRACT

Title of dissertation: A study of the practical use of operations research and vessels big data in benefit of efficient maritime ports utilization in Panama

Degree: MSc

This dissertation assesses the impact of operations research tools and big data analytics on vessels and ports operations using Manzanillo International Terminal (MIT) as an example.

Results for the Berth Allocation Problem (BAP) and the Berth and Quay Cranes Allocation and Scheduling Problem (BAQCASP), which are operation research problems, on several occasions lack robustness due to the uncertainty of quay crane performance and vessels unattachment to static optimal berth allocation schedules.

With the modeling of MIT port using Python programming language and its linear programming modeler PuLP, solutions for the optimal scheduling of vessels are tested against dynamic adjustments from a database that simulates vessel information refined from big data analytics.

The modeling of MIT is done through a multi-step optimization problem using BAP and BAQCASP and solved by GUROBI Mixed Integer Linear Programming (MILP) solver. The findings from the research show how through a collaborative feed of information between vessels and terminals, a better port utilization, vessel consumption and overall reduction of emissions, could be reached.

The creation of a programming code for this dissertation has the capability of plotting the results on a graph and in an excel .csv file that could then be utilized to generate simulations for additional experiments. An identified additional potential of the code is used as a business decision tool based on its capability of adapting to different scenarios in a very short time.

Based on the results, it can be concluded that an integration of big data analytics to operation research tools has a great potential to be used in practice. However, additional research is recommended in order to identify its financial feasibility.

TABLE OF CONTENTS

Declaration ........................................................................................................... Error! Bookmark not defined.
Acknowledgements ............................................................................................. ii
Abstract .............................................................................................................. iv
Table of Contents ............................................................................................... v
List of Tables ....................................................................................................... vii
List of Figures ..................................................................................................... viii
List of Abbreviations ........................................................................................... ix

1. Introduction .................................................................................................. 11
   1.1 Background ............................................................................................... 11
   1.2 Overview of MIT ....................................................................................... 12
   1.3 Objectives ................................................................................................ 14
   1.4 Methodology ............................................................................................. 14
   1.5 Research Contribution ............................................................................ 16

2. Literature review ........................................................................................... 18
   2.1 Maritime Big Data .................................................................................... 18
   2.2 Maritime and port inefficiencies ............................................................... 19
   2.3 BAP and BAQCASTP ................................................................................ 20

3. BAP and BAQCASTP modeling of MIT ....................................................... 23
   3.1 BAP Modeling .......................................................................................... 23
      3.1.1 Problem formulation ......................................................................... 24
         3.1.1.1 BAP Input elements ................................................................... 24
         3.1.1.2 BAP Assumptions .................................................................... 29
         3.1.1.3 BAP Variables, Objective Function and Constraints ............... 30
      3.1.2 BAP expected solution ....................................................................... 32
   3.2 BAQCASTP Modeling .............................................................................. 33
3.2.1 Problem formulation ......................................................................................... 34
3.2.1.1 BAQCASP additional input elements ......................................................... 35
3.2.1.2 BAQCASP assumptions ................................................................................. 36
3.2.1.3 BAQCASP Variables, Objective Function and Constraints ...................... 37
3.2.2 BAQCASP expected solution ............................................................................. 39
3.3 BAP & BAQCASP rolling horizon problem ......................................................... 41
3.3.1 Rolling horizon problem proposed solution .................................................... 42
3.3.2 Expected results from proposed solution ....................................................... 44
4. Dynamic Adjustment of BAP and BAQCASP using real time vessels and port data ........................................................................................................ 46
4.1 Feasible Big Data values for BAP/BAQCASP adjustment ............................... 47
4.1.1 Terminal input .................................................................................................... 48
4.1.2 Vessels input ...................................................................................................... 49
4.2 Dynamic BAP/ BAQCASP development ............................................................ 51
4.2.1 Vessels real-time data random simulation ....................................................... 51
4.2.2 Vessel evaluation to consider for shifting to a new slot .................................. 53
4.2.3 Dynamic BAP/BAQCASP simulated scenario ............................................... 54
5. Conclusions ............................................................................................................. 63
References .................................................................................................................. 65
Appendices ................................................................................................................... 70
Appendix A. Liner services including MIT ................................................................. 70
Appendix B. Python code for BAP/BAQCASP MILP solution, database manipulation, graph and schedule .csv file creation ........................................ 77
LIST OF TABLES

Table 1. Layout description of container berths in MIT. ........................................... 13
Table 2. A calculation example of bi, average feasible processing time of vessel i. 25
Table 3. Result for BAP, utilizing MIT 10/05/2016 schedule .................................... 32
Table 4. BAP input values repeated as input for BAQCAE ......................................... 34
Table 5. QC assignment based on vessels length.......................................................... 35
Table 6. Vessels information after big data analysis/calculation for 4/5/2016........... 55
Table 7. Savings caused by speed reduction to arrive just in time for berth allocation ................................................................................................................................. 58
Table 8. Vessels information after big data analysis/calculation for 5/5/2016........... 59
Table 9. Arrival at 04:00 scenario. Consumption difference for next destination ..... 61
Table 10. Arrival at 07:00 scenario. Consumption difference for MIT arrival. ......... 62
LIST OF FIGURES

Figure 1. Aerial image of the port of MIT. ................................................................. 13
Figure 2. Proposed methodology ............................................................................. 16
Figure 3. Spatial/temporal graph with 2 vessels assigned to berth 2 .................. 24
Figure 4. Graphical representation of BAP solution for MIT 10/05/2016 schedule 32
Figure 5. BAQCASP solution graph of Berth 1 with 4 vessels and their assigned 34
  cranes .....................................................................................................................
Figure 6. BAQCASP graph plot returned by python code after solving multi step MILP 39
  for MIT 10/05/2016 schedule ..............................................................................
Figure 7. Excel .csv file with BAQCASP values of vessels assigned to Berth 1 at MIT 40
  to the 10/05/2016 schedule .................................................................................
Figure 8. Dialogue Box generated by python code with final multistep MILP for MIT 40
  10/05/2016 schedule ............................................................................................
Figure 9. Rolling horizon problem on MILP BAP/BAQCASP ............................. 41
Figure 10. Rolling horizon violation of 3 vessels overlapping space and same cranes 42
  utilized by 3 vessels ..............................................................................................
Figure 11. Schedule for the day following Figure 9 enabled with rolling horizon 45
  problem solution ....................................................................................................
Figure 12. Available idle slots after BAP/BAQCASP solution ............................... 48
Figure 13. Random geodetic positioning of vessels bound to MIT ..................... 52
Figure 14. Scenario timeline .................................................................................. 54
Figure 15. Schedule for vessels served on 3/5/2016 ............................................ 54
Figure 16. Static BAP/BAQCASP for 4/5/2016 (Day 2) ........................................ 55
Figure 17. Day 2 dynamically adjusted and original schedule after BAP/BAQCASP 57
  ..............................................................................................................................
Figure 18. Fuel consumption depending on ship speed ........................................ 58
Figure 19. Static BAP/BAQCASP solution for 5/5/2016 (Day 3) ......................... 59
Figure 20. Day 3 dynamically adjusted and original schedule after BAP/BAQCASP 61
  ..............................................................................................................................
LIST OF ABBREVIATIONS

AIS Automatic Identification System
BAQCAASP Berth and Quay Cranes Allocation and Scheduling problem
BAP Berth Allocation Problem
CITOS Computer Integrated Terminal Operations System
DNV-GL Det Norske Veritas- Germanischer Lloyd
DO Diesel Oil
ECDIS Electronic Chart Display and Information System
ECLAC Economic Commission for Latin America and the Caribbean
FAL Facilitation Committee
GPS Global Positioning System
HDFS Hadoop Distributed File System
HFO Heavy Fuel Oil
IFO Intermediate Fuel Oil
IMO International Maritime Organization
IoT Internet of Things
JIT Just in Time
LOA Length Overall
MGO Marine Gas Oil
MILP Mixed Integer Linear Programming
MIT Manzanillo International Terminal
MT Metric ton
NM Nautical miles
RPM Revolutions per minute
S-AIS Satellite- Automatic Identification System
SOG Speed Over Ground
SOLAS International Convention for the Safety of Life at Sea
SQL Structured Query Language
<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>STW</td>
<td>Speed Through Water</td>
</tr>
<tr>
<td>TEU</td>
<td>Twenty-foot Equivalent Unit</td>
</tr>
<tr>
<td>VHF</td>
<td>Very High Frequency</td>
</tr>
</tbody>
</table>
1. INTRODUCTION

1.1 Background

In this challenging time for the maritime industry, use of information will be paramount for the better utilization of assets.

As the industry is struggling to meet higher profits while facing overcapacity problems, the use of operations research tools will definitely reduce the gap between inefficiencies.

Certainly, in this extremely competitive environment, the difference will be at the costs and service levels. Some of the maritime industry needs have been deeply assessed through extensive research while others require additional assessment (Lee & Song, 2017). In Lee and Song efforts to classify the existing research done to provide value to industry needs, the study referred to 5 important segments and the research attached to these needs:

- Shipment routeing and capacity procurement
- Container fleet and repositioning
- Vessel fleet and operations
- Terminal operations and container handling
- Inland transport vehicle and container handling

In addition to these necessities and the use of optimal or heuristic tools, the next natural step needed to approach dynamic and real-time solutions is the use of vessel’s and port available data in real time.

As stated in “DNV-GL Strategic Research & Innovation Position Paper 4-2014. Big Data. The new data reality and industry impact” (Løvoll & Kadal, 2014), the use of big data is authentic and could create an important impact on the following areas:

- Technical operation and maintenance
- Energy efficiency
- Safety Performance
- Management and monitoring of accident and environmental risks from shipping traffic
- Commercial operation (as part of a logistics chain)
- Automation of ship operations
According to (Drewry Shipping Consultants, 2012) the schedule reliability of container shipping achieved 73%. This uncertainty directly leads to underutilization of resources for both the port and the liner shipping vessels. Not following a Just in Time (JIT) process caused by both port and vessel schedule unpunctuality leads to increased cost caused by vessel fuel consumption and consequently increased CO2 emissions (Moon & Woo, 2014).

In a win-win situation, terminals would like to have regular arrival patterns of ships, in line with what was agreed or announced by shipping companies and in return shipping companies want to minimize their vessels waiting time, even if they are delayed (Lang & Veenstra, 2010).

Expecting this to be a dissertation that could help in identifying the benefits of a business model driven by a collaborative approach between stakeholders. The goal is on concentrating efforts to demonstrate how big data could impact on commercial operations. Specifically, in port utilization and vessels performance enhanced by schedule punctuality and proper berth allocation planning, utilizing MIT as an example.

1.2 Overview of MIT

MIT is a container and Ro-Ro terminal located close to the north entrance of the Panama Canal in the city of Colon. Operated in a joint venture called MIT- Panama. S.A by Carrix, Inc. which is a parent company of SSA Marine, Manzanillo together with the port of Cristobal operated by Panama Ports Company and Colon Container Terminal operated by Evergreen Group complements the Colon Port region.

As per the Economic Commission for Latin America and the Caribbean (ECLAC) regarding the last values, Colon stands 2nd in container throughput in Latin America and the Caribbean just after Santos, Brazil with a general throughput of 3.2 M TEU in 2016 (Economic Commission for Latin America and the Caribbean, 2017).

MIT is considered an important node for several liner services (see Appendix A) that utilizes this port mostly for transshipment, which accounts for almost 85% of its total operations. The port has 2,500 meters of berth from which 3 Ro-Ro berths and 6 container berths forms part of the layout. For the purpose of this research, the container terminal is divided in 3 berths, i.e. 1 berth of 1,200 m containing berths
1,2,3,4 and the remaining berths of 400 m each classified as individual berths (see Table 1).

Table 1. Layout description of container berths in MIT.

<table>
<thead>
<tr>
<th>New layout description</th>
<th>Existing description</th>
<th>Length</th>
<th>Cranes</th>
<th>Cranes reach</th>
<th>Cntrs. moves/hour/crane</th>
</tr>
</thead>
<tbody>
<tr>
<td>Berth 1</td>
<td>Berth 1,2,3,4</td>
<td>1,200 m</td>
<td>1,2,3,4,5,6,7,8,9,10,11</td>
<td>17 cntrs. [1-6]</td>
<td>18 cntrs. [7-11]</td>
</tr>
<tr>
<td>Berth 2</td>
<td>Berth 5</td>
<td>400 m</td>
<td>12,13,14,15</td>
<td>22 cntrs.</td>
<td></td>
</tr>
<tr>
<td>Berth 3</td>
<td>Berth 6</td>
<td>400 m</td>
<td>16,17,18,19</td>
<td>25 cntrs.</td>
<td></td>
</tr>
</tbody>
</table>

Figure 1. Aerial image of the port of MIT. (Google Maps, 2017)

All container berths have a minimum depth of 14 m with a very small average tide variation of 0.3 m.
The use of 19 cranes with a reach from 17 containers up to 25 containers allows MIT to provide services including New Panamax vessels whose volume of transit has increased due to the opening of the expanded Panama Canal.

The increase of more than 10 additional liner services after Panama Canal expansion (Panama Canal Authority, 2017) together with important logistic services and its proximity to the transisthmian railway and the biggest free trade zone of the region make MIT an important node of transit for the new services.

1.3 Objectives
Main Objective
- Identify the feasibility of operation research techniques together with real time data exchange and demonstrate their impact on vessels and port operations at the port of MIT.

Specific Objectives
- Adjust the use of the BAP and BAQCASP given the real conditions of the port of MIT.
- Demonstrate the impact of proper planning and dwell time reduction at port on vessels operation.
- Compare static models of BAP and BAQCASP with a dynamic scenario where decisions of vessels scheduling are taken based on a combination of the previously mentioned models and real-time data received from vessels.
- Provide with a decision-making tool to the port of MIT based on BAP, BAQCASP and vessels real time data exchange.

1.4 Methodology
The BAP and BAQCASP will be modeled given the actual conditions of MIT with the use of Python programming language and its module PuLP which is a MILP modeler. GUROBI will be utilized as the solver for the optimal results modeled through PuLP.

---

1 New Panamax Vessel: Container vessel of 366m length, 49m beam, 15.2 m draft and capacity of 13,000 TEU.
Data utilized as input for these models was provided by the port of MIT with some few exceptions given their sensitive nature. Assumptions taken to replace these exceptions will be properly pointed out throughout this dissertation.

Data Base manipulation and creation of vessel values were done utilizing Python module Pandas. Pandas used as a replacement of SQL language has the same operationality with the powerful platform provided by Python.

In some scenarios, the approach is done through a comparison of the results on a vessel following a fixed schedule against others with more flexibility, accounting factors like speed reduction, weather routeing and slot exchange causing speed adjustment.

Vessels’ real-time data will be simulated based on assumptions described throughout this document. The use of this database will then be taken into account to perform adjustments to BAP and BAQCASP results.

The first stage of the project is to model the port into a BAP utilizing the database supplied by MIT with the arrival and departure time of real vessels calling the port in the past year.

The second step will create an individual BAQCASP model that could return optimal results provided by the crane description for each berth. Berths assigned to a vessel will dynamically determine the processing time of the vessels to be attended based on their per hour movement and the containers to be moved from/to the vessel.

The integration of both models will be achieved by merging them into a multistep optimization approach that will utilize the results of step 1 as input for step 2, meaning that the selection of a berth will be done in step 1 and that result will determine in what of the three BAQCASP optimization problems a vessel will be part of, later on. The sum of the three BAQCASP will return the minimum feasible combination of position and berthing time considering available space in the three available berths.

Finally, an evaluation of the optimal results with a created scenario simulating real time data feed will help in adjusting the model and re running it through the BAP and BAQCASP considering the real time of arrival of a vessel and if it could cause a speed saving to arrive on time at the next destination.
Figure 2. Proposed methodology

1.5 Research Contribution

Given the high capital investment required for vessels and ports to remain competitive, the use of operation research techniques such as BAP and BAQCASP and the utilization of technological trends such as Big Data has now proven practical when utilized individually and has the potential to create a powerful combination for better utilization of assets which is aimed to be proved in this research.

The outcomes of this research will serve to demonstrate the power of MILP with the manipulation of databases utilizing Pandas, which in a future research could be replaced by the use of the most significant Big Data Hadoop Distributed File Systems (HDFS) feature. This is the MapReduce: that enhances significantly mining through huge patches of data. A Python module named mrjob has already proven a good adjustment for Python of the JAVA written MapReduce.

The creation of a multi-step MILP model exclusively for the layout and quay distribution of MIT could be used for the formulation of, for example, additional linear

---

2 Hadoop MapReduce: is the original framework for writing applications that processes large amounts of structured and unstructured data stored in Hadoop Distributed File System. Source: (Horton Works, n.d.)
programming models, integer programming models, heuristics models and genetic algorithms. at the port of MIT.

The creation of a multi-step optimization model with BAP and BAQCASP could be segregated to measure the performance of the port considering each model separately.

This research is expected to prove that the first step towards the dynamic combination of big data is feasible and could be used to compare the actual port performance against the recommended scheduling of the optimization model. The next step is now open towards the integration of a dynamic merge of big data algorithms that could recognize the best fit for empty spaces based on constraints such as the speed needed to arrive and if it is worth increasing or reducing speed based on next destinations.

Future research including a collaborative approach between different multimodal stakeholders including trucks, ports, vessels and the Panama Canal will help in enhancing the model and introducing new constraints that will aim in finding the objective of minimum cost of a whole logistic platform. For the time being, the model could be adapted to change the objective of the minimum cost for the minimum level of CO2 emissions, considering the distance of trucks to suboptimal vessel locations, the speed performance of the vessel before arriving MIT and the savings that results in being served promptly and consequently being able to steam slower to the next destination.
2. LITERATURE REVIEW

2.1 Maritime Big Data

The use of big data has generated a great enthusiasm among researchers as per the powerful capabilities offered in a constant retrieval of information from vessel conditions. Big Data is defined as data sets whose size is beyond the ability of typical database software tools to capture, store, manage and analyze (Mckinsey Global Institute, 2011). The difficulties arising from big data created the opportunity and the inherent need of human skills of insight and pattern recognition to find genuine meaning in the data. (Lloyds Register Foundation, 2014)

Based on the more common usage of “big data” in the maritime industry, an analysis done in “Major challenges and solutions for utilizing big data in the maritime industry” (Koga, 2015) correctly assessed the potential of the practical use of big data in line with the enhancement of maritime areas such as energy efficiency, safety performance, technical operations and maintenance and commercial operations. Koga’s dissertation has related this information from very important sources of research such as “DNV-GL Strategic Research & Innovation Position Paper 4-2014 Big Data” (Løvoll & Kadal, 2014) and “Foresight review of big data: Towards data-centric engineering”. (Lloyds Register Foundation, 2014). These sources validate Koga’s assertions in the potential of the areas of development, additional to the challenges that big data could face from where privacy issues, data governance, availability of powerful tools, flexibility and near monopoly providers, are mentioned. A valuable fact from Koga’s dissertation and papers of DNV-GL and LRF is that for the data to be recognized as big data it should be extraordinary in one or multiple of the following dimensions (volume, velocity, variety and veracity).

Koga’s dissertation creates a good foundation for the next natural step that is the utilization of big data as a game changer in practice and the more efficient utilization of maritime and logistics assets through a dynamic decision-making process driven by operations research techniques explained in simulated scenarios utilizing vessels and port data as input.

The enhancer of big data utilization in the maritime field follows a comparison done between the traditional Deming’s Cycle and the Information Centric cycle done by Thomas Vitsounis. The traditional Deming’s cycle which is classified as a static
environment, performs by having long cycles decision-making processes based on past events and the optimization based on old data. The counterpart which is the information centric data also known as real time data, bases the decision-making process making use of current events and the optimization process based on actual data (Vitsounis, 2014). The complexity of this approach as expressed by Vitsounis is that supply chains are dynamic, unpredictable and diverse by nature. The low visibility, repetitive information and high fragmentation of the supply chain stakeholders add additional pressure to the best performance a supply chain could have. A central organism such as a port community system that could work as a central, open and neutral infrastructure for the whole community, will help in enhancing strategies to optimize supply chains through the identification of bottle necks and the constant feed of information to sophisticated algorithms through big data. From this analysis, it is clear that big data on integrated logistics platforms such as where ports and vessels merge requires a highly collaborative approach.

2.2 Maritime and port inefficiencies

One of the factors that are of great concern by liner shipping companies is the schedule reliability and their implications in their port dwell time that affects their bottom line profits through additional cost and the increased required speed to keep up with their next destinations in a full liner service. An analysis of the factors causing liner schedule reliability and the implications of decreasing schedule integrity for seaports and their actors was carried out by Vernimmen, Dullaert and Engelen in 2007. It has been argued by Notteboom, in 2006 that delays at port caused by late arrivals might be a competitive factor for mature markets where competing lines are not differentiated and transit times are comparable. Notteboom described the sources of liner delays considering ports and vessels and the measures and planning tools container services utilizes to maximize schedule reliability.

An analysis of Drewry Shipping Consultants in 2012 mentions that there is a group of carriers that is almost never on time and may easily have delays of 2 or 3 days. A complete analysis on carriers’ punctuality was also carried in this published document.

A quantitative analysis was carried by Lang and Veenstra (2010) with a comparison between optimized approach speed of vessels and non-optimized speed. In the
scenarios delivered by Lang and Veenstra, the introduction of a central Arrival Planning Strategy (APS) enhances the coordination of shipping lines and terminals. Hence, generating monetary and operational benefits for both parties, which were quantified as part of the results of this publication.

A research carried by Moon and Woo (2014) assesses the impact of port operations on vessel speed performance and the effect it has on CO\textsuperscript{2} emissions and fuel consumption. The study focuses on how the waiting and berthing time at port influences the vessel operations costs and at the same time affects the terminal competitiveness measured most of the time through the ship time at a port which is a key indicator to measure the efficiency of port operations.

A delay caused by the terminal could have its origin in factors such as port congestion, quay cranes breakdowns, strikes, natural disasters and bottlenecks in access channels, which are factors assessed by Maloni and Jackson (2005), Lun, Lai and Cheng (2010) and Notteboom (2006) as cited by (Moon & Woo, 2014).

It is important to mention that there is a gap of research related to the utilization of centralized data from the integration of ports and vessels’ information to create an optimal schedule less sensitive to the vessels’ arrival punctuality.

2.3 BAP and BAQCASP

BAP and BAQCASP have been extensively assessed in the existing literature as they are two important issues in the operation of container terminals.

The use of the widely known operations research problem, the Multiple Machine Scheduling Problem and the Resource Allocation and Scheduling Problem mostly utilized in manufacturing and their introduction into port and vessels operations is what is now known as BAP and Quay Cranes Allocation and Scheduling Problem (QCASP). The similarity is based on the fact that tasks have to be assigned by machines utilizing resources and vessels (tasks) must be served at quays (machines) by additional resources (quay cranes).

BAP is classified in two different branches, the discrete BAP and the Continuous BAP. Literature describing BAP should fall in one or another section.

The discrete BAP considers an entire wharf as a set of partitions where a vessel could be moored within the allocated berth. A description of this problem was published by
Lai and Shih (1992) while assessing the characteristics of a major container terminal in Hong Kong and a comparison of their allocation procedures which were usually following a First Come First Served approach in relation with a heuristic and simulation solution. They could demonstrate improvements in comparison with the terminal allocation practices. An approach taken by Imai, Nagaiwa and Tat (1997) addressed the BAP with the assumption that every vessel is ready to be served at the beginning of a planning horizon. This assumption was trimmed by Imai, Nishimura and Stratos (2001) where the consideration of vessels arriving after the beginning of the planning horizon was introduced. An introduction of a weighting for giving a service priority to selected vessels was then assessed by Imai, Nishimura and Papadimitriou (2003).

The continuous BAP considers the quay as an undivided space where vessels can berth at any position in a single continuous line of a given length (Froja, Correcher, Valdes, Koulouris, & Tamarit, 2015). One of the first to introduce the concept were Li, Cai and Lee (1998) as a machine sequencing problem not adapted to address berths. A MILP relaxed by a Lagrangian relaxation with discretized space of berth and time to reduce computational load was developed by Park and Kim (2002).

The addition of multiple quays to the problem of continuous BAP was introduced by Froja, Correcher, Valdes, Koulouris and Tamarit (2015), where the variables of assigning vessels to a specific quay, berthing times and positions are mixed with the regular BAP constraint of time and space while minimizing the penalty cost of being deviated from the desired position, delay penalty, different quay assignment cost and the cost of waiting for a berth.

Some of the research on BAP and BAQCASP weakest link is the uncertainty caused by unknown information, breakdowns, changes or incidences, making the initial optimal plans infeasible. Studies realized by Umang, Bierlaire and Erera (2013) has introduced stochastic approaches to the arrival and handling times of vessels which are uncertain.

The introduction of time buffers for dealing with the uncertainty of vessels arrival and vessels handling time at BAP was studied by Xu, Chen and Quan (2012). A proper real-time monitoring of vessels aims to create a follow-up and convert arrival times in deterministic for the correct application of the BAP and BAQCASP results in real life scenarios.
The BAQCASP have been usually assessed separately from the BAP. Few studies have addressed these problems together. This problem aims within the minimum optimal cost caused by delays and positioning of the vessel, to locate every expected vessel in a position where it could be served with less than the maximum number of available quay cranes. This problem considers the use of continuous BAP’S of a single berth. One of the first to address this problem were Park and Kim (2003) through an integer programming model formulated by a two-step solution. The publication of Imai, Chen, Nishimura and Papadimitriou (2008), tackles the formulation of simultaneous quays and crane allocation problem through a heuristic solution. The introduction of the constraint of the cranes range of motion was part of the formulation of a MILP model proposed by Zhang, Zheng, Shi and Armstrong (2010). Following an integrated approach, Correcher, Alvare-Valdes and Tamarit (2017) delivered a continuous BAQCASP that does not require berth discretization, allowing vessels to be moored at any position of the berth and being served with sets of cranes that could not surpass the area of the next crane.

For the creation of a model that could simulate the characteristics of MIT, no literature has created what should be considered as a Continuous BAQCASP with multiple quays. An additional challenge for this dissertation is the creation of an optimal solution where vessels are assigned to berths and position together with the cranes and the number to be used and based on that number and crane type, a dynamically adjusted processing time.
3. BAP AND BAQCASp MODELING OF MIT

3.1 BAP Modeling

BAP refers to the optimal location of a vessel in a berth over a planning horizon considering an objective function that could be selected based on the delay and position cost of placing a vessel in a berth position at a specific time. Modifications to the objective function could be related with finding the minimum makespan\(^3\) or to find the minimum emission level caused by the position and time of berthing a vessel.

The main constraints related with this problem deals with spatial and temporal restrictions. The spatial restrictions have to do with the berth dimensions, including length. The temporal restrictions need to acquaint for the arrival and berthing date of the vessel (Rodriguez, Rosa, Gomes, & Ribeiro, 2016).

As mentioned previously, according to the spatial restriction, BAP could be divided into discrete BAP, continuous BAP and hybrid BAP. The discrete BAP uses set of quays that divides a berth, each of which can harbor one vessel at a time (Lin, Ying, & Wan, 2014). In continuous BAP, the quay is not divided and vessels can berth at any position along the pier (Bierwith & Meisel, 2010) as cited by (Rodriguez, Rosa, Gomes, & Ribeiro, 2016). The description of hybrid BAP as a partition area of a quay with the possibility of having larger vessels occupying more than one quay was given by Rodriguez, Rosa, Gomes and Ribeiro (2016).

Based on the temporal approach, a BAP can be divided into static or dynamic. The static problem has to do with a set of vessels ready to be served at a quay at the beginning of a planning horizon while a dynamic problem accepts to receive vessels through the planning horizon, once the arrival time is known in advance (Gkolias, 2007).

Results for BAP are expected as a two-dimensional temporal/spatial graph with the x axis representing the time from 0 up to the planned horizon and the y axis representing the length of the berth (see Figure 3).

\(^3\) Makespan: The total length of schedule when all task had finished their processing time.
Figure 3. Spatial/temporal graph with 2 vessels assigned to berth 2

The configuration of MIT layout requires of a BAP of a continuous type with dynamic arrivals but with the addition that a selection of a single among multiple quays (3) could be done.

The main goal of BAP, in this case, will be assigning vessels to feasible quays taking into consideration the minimum cost caused by the waiting time, suboptimal position and delay after planned departure time. The result will also be provided with the optimal position in the selected berth, together with the berthing time. However, at this stage, the main interest is of knowing at what berth a vessel is assigned.

3.1.1 Problem formulation

Information utilized as input for the model is absorbed from a database given by the terminal but trimmed and adjusted for the purpose of this dissertation. Some columns were created following some assumptions. The description of the needed values for the BAP model will be provided with the detail of how they were acquired and if it is from an assumption or from the original data. This model was created and adapted from, “The continuous Berth Allocation Problem in a container terminal with multiple berths” published by Froja, Correcher, Valdes, Koulouris and Tamarit (2015).

3.1.1.1 BAP Input elements

Berths

$q \in Q$. A set of berths, elements of total berths. ($q=1,2,3$)

$L_q$. $\forall q \in Q$. Length of each berth. ($q_1=1240m$, $q_2 = 400m$, $q_3= 400 m$). Information acquired from (Georgia Tech Panama. Logistics and Innovation & Research Center, 2017).
Vessels

\( i \in V \) Set of vessels elements of Vessels. \((i=1,2,3, \ldots, n)\). Each planning day has a number of vessels to be served. The filtering criteria of this section is the date in the database when the vessel was served.

\( l_i \), \( \forall i \in V \). Length of each vessel. For vessels of more than 130 m a 10% safety margin is added as part of the length, otherwise, 10m is added to the vessel original length.

\( b_i \), \( \forall i \in V \). Estimated handling time (loading/unloading). Value was calculated considering the number of containers to be moved from/to the vessel divided by several combinations of the feasible crane assignment each vessel could have. i.e. a vessel with a beam of 49 meters for being served up to the last container requires a crane with a reach at least of 20 containers, based on the width of a container (49m/2.43m + allowance). This means that it could be served by cranes from berth 2 and berth 3 exclusively (see Table 1). Given that the performance of the cranes is provided and the minimum and maximum per vessel crane requirement is provided, a set of values having all the handling times for all combinations of containers to be handled and cranes servicing was calculated. This set of values provides an average of the vessel handling time before setting the amount and type of containers to be used.

e.g. The case of vessel \( i \) with 49 m beam and 350 containers to be moved at an agreed min/max cranes allocation of 2 and 4 respectively:

Processing of vessel \( i \) with 2 quay cranes combination at berth 2 = 350 cntrs. / 2x32 cntrs. per hour = 5.5 hours.

Table 2. A calculation example of \( b_i \), average feasible processing time of vessel \( i \).

<table>
<thead>
<tr>
<th>Berth</th>
<th>Cranes</th>
<th>Reach</th>
<th>Processing crane/hour</th>
<th>Feasible comb (2-4 cranes)</th>
<th>Processing of vessel i</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>12,13,14,15</td>
<td>22 cntrs.</td>
<td>32</td>
<td>(12,13)/ (13,14)/ (14,15)</td>
<td>5.5 hours</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>(12,13,14)/ (13,14,15)</td>
<td>3.6 hours</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>(12,13,14,15)</td>
<td>2.7 hours</td>
</tr>
</tbody>
</table>
\( a_i \forall i \in V. \) **Arrival time.** Time when a vessel is ready to be processed. 2 hours after confirmed and monitored Estimated Time of Arrival (ETA). Available from raw database.

\( s_i \forall i \in V. \) **Desired departure time.** Time when a vessel is scheduled to depart due to next destination planning or Panama Canal transit scheduled window. Actual Time of Departure (ATD) value at the raw database was used as input.

\( d_{iq} \forall i \in V, \forall q \in Q. \) **The ideal position of vessel \( i \) at berth \( q \).** The best position to moor the vessel at each berth, based on the following considerations in order of importance:

If a vessel’s length is more or equal than 250 m then vessel \( i \) ideal position is \( \{q_1=(1240m-l_i), \ q_2=1, \ q_3=1\} \). This means that considering the maneuvering capabilities of bigger vessels, it is desirable to have them served on the northerly border of berth 1. No constraints are expected for a big vessel in the wide areas of berth 2 and 3, so any position is expected.

If a vessel’s reefer containers are more or equal than 100 units, then \( \{q_1=500, \ q_2=1, \ q_3=1\} \). MIT moves 85% of the containers for transshipment purposes. This consideration allows the port to utilize most of their yard for transshipment services, the increased movement of reefer cargo that will reach values of 120 million tons in 2020 (Drewry, 2016) will attach a level of importance to the position at which vessels are moored. The assumption of having vessels with an important amount of reefer containers moored at the middle section of berth 1, follows the logic of moving reefer containers out of electrical plugs as less as possible hence the distance to the quay side should be the lowest. Plugs are assumed to be located in the middle section of berth 1. Berth 2 and 3 as transversal berths does not follow this consideration; hence, any position is ideal for these berths.

| 3 | 16,17,18,19 | 25 cntrs. | 31 | (16,17)/ (17,18)/ (18,19) | 5.6 hours | (16,17,18)/ (17,18,19) | 3.7 hours | (16,17,18,19) | 2.8 hours | \( \text{Avg.}(b_i) = 4.0\text{hours} \) |
If a vessel’s dangerous cargo containers are more or equal than 100 units, then \( \{ q_1 = 1240m - li, q_2 = 1, q_3 = 1 \} \). The consideration of this assumption is based on safety. Based on the terminal layout, in case of a fire or dangerous leak, the vessel must ideally be isolated as far as possible from the port offices and gate of the free trade zone at the south border.

If a vessel’s transshipment/import cargo is more or equal than a 2:1 ratio, then \( \{ q_1 = 500m, q_2 = 1, q_3 = 1 \} \). The assumption following this assignment is that since transshipment cargo is expected to be a short time at the yard. A connection to the next vessel should be done from the closest location possible. Imports are expected to be stored close to customs check points.

Otherwise \( \{ q_1 = 1, q_2 = 1, q_3 = 1 \} \). If any of the previous considerations are not met, the vessel is not strictly forced to have an ideal position.

**Cost**

\( c_{iw}, \forall i \in V \). **Cost per hour waiting for a berth.** Cost of anchorage not included as anchorage out of Panama Canal waters is utilized. Bunker consumption cost for Diesel Oil (DO) electric generators calculated per hour, takes the assumption that a vessel that is more than 3000 TEU could be considered as a deep-sea vessel with 7.9 MT of Intermediate Fuel Oil (IFO) 380 consumption per day at a price of $315.52 /MT for and hourly consumption of $103.80/hr. A vessel that is less than 3000 TEU could be considered as a feeder vessel with 1.2 MT of Marine Gas Oil (MGO) consumption per day at a price of $606.05 /MT for an hourly consumption of $30.30/hr. The values of consumption were supplied by The Port of Rotterdam Authority (2006) and the bunker cost was obtained from an average of values for the last year obtained from Clarksons Research Limited Services (2017).

\( cid, \forall i \in V \). **Cost per unit time delay with respect to the desired departure time.** This value includes the cost of bunker utilized at a port, having the same considerations as the previous cost description. If the vessel has a reserved slot for a southbound transit of the Panama Canal via the Pacific Ocean, the cost of cancelling the reservation with short notice (less than 36 hours) is 100% of the reserve fee (Panama Canal Authority, 2017) and is also added as cost of time delay with respect to desired departure. The vessels transiting the Panama Canal were filtered by merging the list
of vessels per liner service and an analysis of all the services transiting from MIT to the Pacific Ocean for a next destination.

The price consideration for every case takes into account the following filter:

If a vessel’s length is less than 150m and transits the Panama Canal right after completing operations at the terminal, then \( c_{id_i} = \$30.30/\text{hr.} \) (feeder vessel DO hourly consumption) + \$2,500 (Panama Canal reservation fee for vessels up to 92 m of length overall (LOA)). Note that the vessel length is fixed in 150m, taking the assumption that both vessels 92m and 150m are considered feeder vessels.

If a vessel’s beam is less than 24m, the length is more or equal than 150m and the vessel is expected to transit the Panama Canal, then \( c_{id_i} = \$103.80/\text{hr.} \) (deep sea vessel Heavy Fuel Oil (HFO) hourly consumption) + \$5,500 (Panama Canal reservation fee for vessels less than 24 m beam and more than 92m LOA).

If a vessel’s beam is more or equal than 24m but less than 27m, the length is more or equal to 150m and the vessel is expected to transit the Panama Canal, then \( c_{id_i} = \$103.80/\text{hr.} \) (deep sea vessel HFO hourly consumption) + \$10,500 (Panama Canal reservation fee for vessels more than 24m beam but less than 27m and more than 92m LOA).

If a vessel’s beam is more or equal than 27m but less than 31m, the length is more or equal than 150m and the vessel is expected to transit the Panama Canal, then \( c_{id_i} = \$103.80/\text{hr.} \) (deep sea vessel HFO hourly consumption) + \$18,500 (Panama Canal reservation fee for vessels more or equal to 27m beam but less than 31m and more than 92m LOA).

If a vessel’s beam is more or equal than 31m, the length is more or equal than 150m and the vessel is expected to transit the Panama Canal, then \( c_{id_i} = \$103.80/\text{hr.} \) (deep sea vessel HFO hourly consumption) + \$35,500 (Panama Canal reservation fee for vessels more than 24m beam but less than 27m and more than 92m LOA).

If a vessel not transiting the Canal has a length more or equal than 150m, then \( c_{id_i} = \$103.80/\text{hr.} \) (deep sea vessel HFO hourly consumption)

If vessel not transiting the Canal has a length less than 150m, then \( c_{id_i} = \$30.30/\text{hr.} \) (feeder vessel DO hourly consumption).
\( ca_i q_{i,q} \forall i \in V, \forall q \in Q. \) Cost of assigning a vessel \( i \) to a berth \( q \). The default values for this cost is 1 for every vessel in any berth. The adjustment to these values is generated based on the case of a vessel being served after the 24\textsuperscript{th} hour of the planning horizon of a single day given the result of the BAQCASP. In case a vessel is required to be processed in a time period exceeding the 24 hours planning horizon limit, it is virtually introduced once again in the BAP problem as starting from time 0 at the specific same spot where it was served in the previous day at time 24. To force the model to allocate the vessel in the exact same berth it was served, a dummy value is created for a vessel, forcing it to have an extremely high cost of \( ca_i q_{i,q} \) in this case will be of $50,000 for the quays the vessel should avoid and $1 for the quay expected to keep serving the vessel.

\( cpi \forall i \in V. \) Cost of unit deviation of vessel \( i \) from its ideal position at berth \( q \). Every vessel has a desired position based on the quantity and the characteristics of the containers to be handled at every call as described in \( diq_{i,q} \) (desired vessel \( i \) position at berth \( q \)). Failing to be moored at the ideal position will represent an additional cost to the trucks handling containers for more distance as planned. The cost takes the consideration of a loaded Volvo truck with a trailer and the capacity to load up to 60 tons including chassis weight. A truck with these characteristics is expected to burn 43 lt. of Diesel Oil in 100kms (Volvo Trucks, 2014), tear and wear are not considered in this assumption. For the purpose of this calculation it is assumed that a truck has to be positioned below a quay crane and to the assigned area of the yard; hence, a roundtrip is to be assigned as a cost for a vessel being moored away from the ideal position (close to the assigned stacking yard area). Given the previous assumptions, the cost of moving a truck away from it desired position is calculated as follows:

\[
0.00043 \text{lts/meter} \times \$1.50 \text{ (DO price per liter)} = \$0.000645 \times 2 \text{ (Roundtrip)} = \$0.001 \\
\text{per container moved away from ideal position.}
\]

A vessel expected to be served beyond the 24\textsuperscript{th} hour of the planning horizon will create a dummy value that enables the vessel to be served in the exact same position in the next day where it was served at the end of the previous day.

3.1.1.2 BAP Assumptions

- Positions at the berth can only be assigned to one vessel at a time.
• Vessels will be processed as soon as they are moored; hence, maneuvering time and mooring time is included in the arrival time ai, ∀ i ∈ V.

• Once a process starts it cannot be interrupted. Vessels are not allowed to shift position in a berth.

• A vessel that could not complete her processing in the 24th hour of the planning horizon will be forced to be assigned at the same position and berth where it was processed the previous day.

• Vessels are all assumed to have less or equal than 13.30m draft at arrival that is calculated from 14 m – 5% safe UKC margin alongside.

• Handling times for the vessels are considered fixed and independent of the berthing position. This assumption is reasonable if the quay has enough machinery and workers for the loading/unloading task at any moment. (Lim, 1998) as cited by (Froja, Correcher, Valdes, Koulouris, & Tamarit, 2015).

3.1.1.3 BAP Variables, Objective Function and Constraints
The chosen variables for this problem could be described as follows:
Decision variables:
• ti, ∀ i ∈ V = berthing time of vessel i
• pi, ∀ i ∈ V = position of vessel i at assigned berth
• hi, ∀ i ∈ V = delay of vessel i in units (hours)
• ei, ∀ i ∈ V = deviation of vessel i from ideal position at assigned berth in meters

Binary variables:
• mi,q ∀ i ∈ V, ∀ q ∈ Q. 1 if vessel i is moored at berth q, 0 otherwise
• σij ∀ i, j ∈ V. 1 if vessel i is completely processed before vessel j, 0 otherwise
• δij ∀ i, j ∈ V. 1 if vessel i is positioned completely below vessel j, 0 otherwise

The BAP goal is to find the optimal position and berth allocation of every served vessel that could generate the minimum cost. However, it is of great interest to solely find the feasible optimal combination of vessels in each berth. Based on this, the objective function for this problem is:

\[
\text{Min} \left[ \sum_{i \in V} (ciw_i (t_i - ai_i) + cir_i h_i + \sum_{q \in Q} \sum_{i \in V} (cai_{i,q} m_{i,q} + cpi_i e_i)) \right]
\]

The model is subject to the following constraints:
- To avoid overlapping in time and space of vessels.
  \[ \delta_{i,j} + \delta_{j,i} + \sigma_{i,j} + \sigma_{j,i} \geq m_{i,q} + m_{j,q} - 1 \quad \forall \ i, j \in V, i \neq j, \forall \ q \in Q \]
  - Time overlapping avoidance.
  \[ t_j - (t_i + b_i) - (\sigma_{i,j} - 1)H \geq 0 \quad \forall \ i, j \in V, i \neq j \]
  - Space overlapping avoidance.
  \[ p_j - (p_i + l_i) - (\delta_{i,j} - 1)L_{max} \geq 0 \quad \forall \ i, j \in V, i \neq j \]

- The length of vessel \( i \) has to be less than the maximum length of selected berth.
  \[ p_i + l_i \leq \sum_{q \in Q} m_{i,q}L_q \quad \forall \ i \in V, \forall \ q \in Q \]

- Every vessel has to be assigned only to one berth.
  \[ \sum_{q \in Q} m_{i,q} = 1 \quad \forall \ i \in V \]

- Vessel must be moored after arrival.
  \[ t_i \geq a_i \quad \forall \ i \in V \]

- A positive number is given to the delay time if it is later than desired departure time.
  \[ h_i \geq t_i + b_i - s_i \quad \forall \ i \in V \]

- Deviation of vessel from desired position at the assigned berth
  - If the vessel is positioned above of her ideal position (graph perspective).
    \[ e_i \geq p_i - \sum_{q \in Q} d_{i,q}m_{i,q} \quad \forall \ i \in V \]
  - If the vessel is positioned below of her ideal position (graph perspective).
    \[ e_i \geq \sum_{q \in Q} d_{i,q}m_{i,q} - p_i \quad \forall \ i \in V \]

- Decision variables must all be non-negative.
  \[ t_i, h_i, e_i \geq 0 \quad \forall \ i \in V \]

- Vessel \( i \) should not be moored directly at the borders of the berth.
  \[ p_i \geq 1 \]
3.1.2 BAP expected solution

This problem solution will be provided with values to the decision variables of where to position the vessels based on the previous constraints while trying to provide the optimal minimum cost of vessels allocation (see Table 3 and Figure 4). These values will be disregarded with the exception of \( m_{ij} \) which is a binary variable stating in what berth does the vessel have a reserved spot and will be utilized as input for choosing in what of the BAQCASP a vessel will be processed. i.e. in Table 3, Vessel_1 will be assigned to BAQCASP for berth 3.

Table 3. Result for BAP, utilizing MIT 10/05/2016 schedule

<table>
<thead>
<tr>
<th>Vessel</th>
<th>Assigned_Berth</th>
<th>Berth_Position</th>
<th>Berth_Time</th>
<th>Processing_Time</th>
<th>Length</th>
<th>Vessel_Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vessel_1</td>
<td>3</td>
<td>1</td>
<td>7</td>
<td>6</td>
<td>231</td>
<td>ANTWERP TRADER</td>
</tr>
<tr>
<td>Vessel_2</td>
<td>1</td>
<td>1</td>
<td>17</td>
<td>6</td>
<td>225</td>
<td>CONTI ARABELLA</td>
</tr>
<tr>
<td>Vessel_3</td>
<td>3</td>
<td>1</td>
<td>20</td>
<td>6</td>
<td>296</td>
<td>DUBLIN EXPRESS</td>
</tr>
<tr>
<td>Vessel_4</td>
<td>1</td>
<td>916</td>
<td>8</td>
<td>6</td>
<td>324</td>
<td>FLEUR N</td>
</tr>
<tr>
<td>Vessel_5</td>
<td>2</td>
<td>1</td>
<td>6</td>
<td>6</td>
<td>324</td>
<td>MAERSK BATUR</td>
</tr>
<tr>
<td>Vessel_6</td>
<td>1</td>
<td>620</td>
<td>6</td>
<td>6</td>
<td>231</td>
<td>MAERSK NUJMEGEN</td>
</tr>
<tr>
<td>Vessel_7</td>
<td>1</td>
<td>620</td>
<td>20</td>
<td>13</td>
<td>185</td>
<td>MAERSK WILMINGTON</td>
</tr>
<tr>
<td>Vessel_8</td>
<td>1</td>
<td>915</td>
<td>18</td>
<td>7</td>
<td>325</td>
<td>MOL EMPIRE</td>
</tr>
<tr>
<td>Vessel_9</td>
<td>2</td>
<td>1</td>
<td>17</td>
<td>6</td>
<td>286</td>
<td>SEASPAN DALIAN</td>
</tr>
<tr>
<td>Vessel_10</td>
<td>1</td>
<td>157</td>
<td>7</td>
<td>4</td>
<td>185</td>
<td>VIKING MERLIN</td>
</tr>
<tr>
<td>Vessel_11</td>
<td>1</td>
<td>1</td>
<td>5</td>
<td>6</td>
<td>156</td>
<td>X-PRESS TAJUMULCO</td>
</tr>
<tr>
<td>Vessel_12</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>2</td>
<td>225</td>
<td>LOUISIANA TRADER</td>
</tr>
</tbody>
</table>

Figure 4. Graphical representation of BAP solution for MIT 10/05/2016 schedule
3.2 BAQCASP Modeling

The BAQCASP is an optimization problem that aims to find the least cost from the combination of vessels assignment at a continuous berth while allocating available quay cranes at that berth.

Quay cranes are assigned considering the minimum and maximum allowance for every vessel, which most of the time is agreed contractually by the liner shipping company and the terminal. For the purpose of this model, a series of assumptions are taken based on the size of the vessel to assign a minimum and maximum number of cranes.

The processing time $b_i$ dynamically changes based on a number of cranes used to serve the vessel. Hence, the model requires of additional constraints to help it overcome the increased computational load. The same temporal and spatial constraints affecting BAP affects BAQCASP. Vessels cannot be moored at a utilized space.

An important difference with BAP is the fact that multiple quays are not enabled at BAQCASP, which is the reason why it was chosen to process each berth with their available cranes via individual BAQCASP. The inclusion of quay cranes adds difficulty to the model, considering that all vessels/cranes combinations have to satisfy the physical constraint of the vessel beam and crane reach constraint.

MIT quays follow a traditional construction where quay cranes move along the berth in a rail designed for that purpose. The constraint of quay cranes unavailability to transpose with other cranes is adapted in this model with the assumption that once a vessel is assigned to a quay crane, it is kept fixed until the end of the vessel processing time.

Results for BAQCASP, similar to BAP, are expected as a two-dimensional temporal/spatial graph with the x axis representing the time from 0 up to the planned horizon and the y axis representing the length of the berth. The main difference is that the inclusion of non-overlapping quay cranes so as the name of the vessel is part of the result (see Figure 5).
3.2.1 Problem formulation

The use of the initial values utilized for BAP will remain unchanged for BAQCASP and the description of how they were obtained could be reviewed in section 3.1.1.1. From the BAP input values, those required as input for BAQCASP are the arrival time, desired departure time, vessel beam, vessel length, desired vessel position, cost of delay prior berthing, cost of delay from desired departure time and cost of not being located in the desired position (see Table 4). It should be noted the use of cost of berth assignment (caiq,iq) and the fix processing time (bi) are no longer needed for BAQCASP. This model was created adapted from, *A new mixed integer linear model for the berth allocation and quay crane assignment problem* published by Correcher, Alvare-Valdes and Tamarit (2017).

Table 4. BAP input values repeated as input for BAQCASP

<table>
<thead>
<tr>
<th>Description</th>
<th>Notation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Arrival Time</td>
<td>ai, ∀ i ∈ BACASP_q ≤ V</td>
</tr>
<tr>
<td>Desired departure time</td>
<td>si, ∀ i ∈ BACASP_q ≤ V</td>
</tr>
<tr>
<td>Vessel’s Beam</td>
<td>bi, ∀ i ∈ BACASP_q ≤ V</td>
</tr>
<tr>
<td>Vessel’s Length</td>
<td>li, ∀ i ∈ BACASP_q ≤ V</td>
</tr>
</tbody>
</table>
### Ideal vessel position

\[ d_iq \forall i \in BACASP_q \subseteq V, \forall q \in Q \]

### Cost of delay prior berthing

\[ ciw_i \forall i \in BACASP_q \subseteq V \]

### Cost of delay from desired departure time

\[ cid_i \forall i \in BACASP_q \subseteq V \]

### Cost of not being positioned at the ideal position

\[ cpi_i \forall i \in BACASP_q \subseteq V \]

### 3.2.1.1 BAQCASp additional input elements

Additional to the above mentioned, a set of values is needed for the creation of this model and this will be described as follows:

#### Quay cranes

\[ g \in QG \]. **Set of crane groups elements of Quay Cranes Groups.** Obtained from the terminal description at (Georgia Tech Panama. Logistics and Innovation & Research Center, 2017). Each existing combination from 1 to 4 consecutive cranes are assigned to a specific group. i.e. \( g=1=\text{group of cranes (10,11)} \) or \( g=39=\text{group of cranes (8,9,10)} \)

\[ n_g \forall g \in QG \]. **Number of cranes in group g.** The total number of cranes in group \( g \)

\[ f_g \forall g \in QG \]. **Name of the first crane in the group g.** i.e. crane 15

\[ z_g \forall g \in QG \]. **Name of the last crane in the group g.**

#### Vessels

\[ g_i^{\text{min}}, g_i^{\text{max}} \forall i \in V \]. Minimum and maximum quantity of quay cranes QC agreed to serve a specific vessel \( i \). This was created for the sole motivation of this dissertation as it was not available from the supplied database. The assumption taken for the assigned values is based on the vessels length which is shown in Table 5.

### Table 5. QC assignment based on vessels length

<table>
<thead>
<tr>
<th>Vessel LOA</th>
<th>( g_i^{\text{min}} )</th>
<th>( g_i^{\text{max}} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Less than 100 m</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>100m-150m</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>150m-250m</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>More than 250m</td>
<td>2</td>
<td>4</td>
</tr>
</tbody>
</table>
\[ u_i^g \quad \forall i \in V, g \in QG : g \in [g_i^{\min}, g_i^{\max}] \]. Estimated processing time of vessel \( i \) if using cranes from \( g \) group from where it is feasible. The value is calculated considering the number of containers to be handled \( c_i \) and the containers handled per hour of each feasible combination of cranes that could serve vessel \( i \) based on her beam and the quay cranes reach.

**QC**\( _q \). Total Number of quay cranes at berth \( q \) processed in BAQCASP\( _q \).

**Berths**

As every berth is processed separately, the value of each \( q \) will be the same as the berth processed in the respective BAQCASP.

**L**\( _q \). Length of processed berth \( q \) in BAQCASP\( _q \).

**Time**

\( T \in [0, H] \). Set of time periods from 0 to planning horizon.

3.2.1.2 **BAQCASP assumptions**

This model will adopt all the assumptions described in section 3.1.1.2 BAP assumptions, together with some additions created exclusively for the introduction of quay cranes to the model.

- Each berth \( q \) has a set of quay cranes QC assigned exclusively for that berth.
- All quay cranes can move across the berth without transposing other quay cranes in service.
- Each vessel should be served by the same number of quay cranes for the whole duration of her processing time.
- Assigned cranes change the processing time of vessels based on the per hour movement of its type.
- Cranes can only be assigned with other cranes from the same type (same movements per hour).
- Each quay crane can be assigned at most to one vessel in each time period.
- Each available crane combination, feasible with the previous assumptions, will be assigned a specific group.
3.2.1.3 BAQCASP Variables, Objective Function and Constraints

With the use of BAP decision variables and binary variables described in section 3.1.1.3, the only change is that the \( maq_i \) will not be part of this model and the introduction of \( r_{igm} \) as a binary variable will be very much needed.

\[ r_{igm} \forall i \in BAQCASP_q \subseteq V, \forall g \in QG, tm \in [0, H]. \]

A value of 1 is assigned if vessel \( i \) is handled with cranes from group \( g \) at time \( tm \), 0 otherwise. Note that \( H \) stands for maximum time of the planning horizon.

This step has the final goal of providing with the values of the decision variables that will position the vessels in optimal combinations that will return the minimum cost of allocating those vessels in each berth. An objective function for BAQCASP is generated for each berth following the vessels filtered according to the results of the BAP. It is necessary to recognize that the constraints are adjusted depending on the selected vessels for each berth/objective function. The overall minimum cost will then be calculated as the sum of all the BAQCASP objective functions. Considering the following, the BAQCASP objective function that has to be replicated for every berth is:

\[
\min \left[ \sum_{i \in BAQCASP_q \subseteq V} (ciw_i(t_i - ai_i) + cid_i + cpi_i) \right]
\]

The model is subject to the following constraints:

- Only one time of handling commencement could be assigned to a vessel \( i \).

\[
\sum_{tm=a_i}^H \sum_{g \in QG_i} r_{igm} = 1 \quad \forall i \in BAQCASP_q \subseteq V.
\]

- Processing time \( t_i \) will be equal to the assigned handling started time.

\[
t_i = \sum_{tm=a_i}^H \sum_{g \in QG_i} r_{igm} \times tm \quad \forall i \in BAQCASP_q \subseteq V.
\]

- To avoid overlapping in time and space of vessels.

\[
\delta_i, j + \delta_j, i + \sigma_i, j + \sigma_j, i \geq 1 \quad \forall i, j \in BAQCASP_q \subseteq V, i \neq j
\]

\( \circ \) Time overlapping avoidance.

\[
t_j - t_i - \sum_{tm=a_i}^H \sum_{g \in QG_i} (r_{igm} \times u_{ig}^0) - (\sigma_i, j - 1)H \geq 0 \quad \forall i, j \in BAQCASP_q \subseteq V, i \neq j
\]
- Space overlapping avoidance.
  \[ p_j - (p_i + l_i) - (\delta_{i,j} - 1)L_q \geq 1 \quad \forall \, i,j \in BAQCAP_q \subseteq V, \, i \neq j \]
- The length of vessel \( i \) has to be less than the maximum length of the selected berth.
  \[ p_i + l_i \leq L_q + 1 \quad \forall \, i \in BAQCAP_q \subseteq V \]
- Utilized cranes can never be more than total number of available cranes at any period.
  \[ \sum_{v : i \in BAQCAP_q} \sum_{t = \max(a_i,tm - u^q_i + 1)} r_{g,t}n_g \leq QC_q \quad \forall \, tm \in T \]
- In each period \( tm \) a crane group \( g \) can be assigned at most to one vessel.
  \[ \sum_{i : g \in QG} \sum_{t = \max(a_i,tm - u^q_i + 1)} r_{g,t} \leq 1 \quad \forall \, tm \in T, \forall \, g \in QG \]
- If a vessel is above other vessel in the graphic, the highest vessel should have at least the number of cranes assigned to the lower vessel.
  \[ \sum_{i,j \in BAQCAP_q} f_{g,t} \delta_{i,j} \geq 1 - QC_q (\delta_{i,j} + \sigma_{i,j} + \sigma_{j,i}) \]
  \[ \forall \, i,j \in BAQCAP_q \subseteq V, \, i \neq j \]
- A positive number is given to the delay time if it is later than desired departure time.
  \[ h_i \geq t_i - s_i + \sum_{t = \max(a_i,tm - u^q_i + 1)} r_{g,t} * u^q_i - 1 \quad \forall \, i \in BAQCAP_q \subseteq V \]
- Deviation of vessel \( i \) from the desired position at the assigned berth.
  - If the vessel \( i \) is positioned above of her ideal position (graph perspective).
    \[ e_i \geq p_i - d_i q_i \quad \forall \, i \in BAQCAP_q \subseteq V \]
  - If the vessel \( i \) is positioned below of her ideal position (graph perspective).
    \[ e_i \geq d_i q_i - p_i \quad \forall \, i \in BAQCAP_q \subseteq V \]
- Decision variables must all be non-negative.
  \[ t_i, h_i, e_i \geq 0 \quad \forall \, i \in V \]
- Vessel $i$ should not be moored directly at the borders of the berth. $p_i \geq 1$

3.2.2 BAQCASP expected solution

This final step will provide the multistep optimization problem with a solution for each vessel. From the amount of information handled in this problem the values to be used are: the assigned berth, berth position, berthing time, dynamically created processing time based on the selected quay cranes, quay cranes assigned, name of assigned quay cranes group and the finishing time considering the processing time.

After solving the multistep MILP problem and adding together the minimum cost of every BAQCASP, the python built code has the capability of creating individual graphs for each BAQCASP/berths (see Figure 6) and an easy to distribute individual excel .csv files for each berth (see Figure 7). The final solution is displayed in a dialogue box with the minimum cost of the optimization problem including a satellite image of the terminal for easy comparison (see Figure 8). The full python code could be found in Appendix B.

![Figure 6](image1.png)

Figure 6. BAQCASP graph plot returned by python code after solving multi step MILP for MIT 10/05/2016 schedule. (Quay 3 not included).
Figure 7. Excel .csv file with BAQCAP values of vessels assigned to Berth 1 at MIT to the 10/05/2016 schedule

<table>
<thead>
<tr>
<th>Vessel</th>
<th>Berth_Position</th>
<th>Berth_Time</th>
<th>Length</th>
<th>Processing_Time</th>
<th>Crane_Group</th>
<th>Cranes_in_use</th>
<th>Vessel_Name</th>
<th>Arrival</th>
<th>Departure</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vessel_2</td>
<td>1</td>
<td>17</td>
<td>225</td>
<td>3</td>
<td>31 (1, 2, 3)</td>
<td>CONRI ARABELLA</td>
<td>FLEUR N</td>
<td>17</td>
<td>28</td>
</tr>
<tr>
<td>Vessel_4</td>
<td>916</td>
<td>8</td>
<td>324</td>
<td>7</td>
<td>32 (9, 10)</td>
<td>MAERSK NIUMGEN</td>
<td>MAERSK WILMING</td>
<td>6</td>
<td>18</td>
</tr>
<tr>
<td>Vessel_6</td>
<td>620</td>
<td>6</td>
<td>231</td>
<td>9</td>
<td>46 (6)</td>
<td>CONTI ARABELLA</td>
<td>FLEUR N</td>
<td>8</td>
<td>18</td>
</tr>
<tr>
<td>Vessel_7</td>
<td>620</td>
<td>20</td>
<td>185</td>
<td>11</td>
<td>12 (1, 2)</td>
<td>CONTI ARABELLA</td>
<td>X-PRESS TAJUMULCO</td>
<td>20</td>
<td>42</td>
</tr>
<tr>
<td>Vessel_8</td>
<td>915</td>
<td>18</td>
<td>325</td>
<td>5</td>
<td>26 (8, 9, 10, 11)</td>
<td>MAERSK NIUMGEN</td>
<td>FLEUR N</td>
<td>18</td>
<td>28</td>
</tr>
<tr>
<td>Vessel_10</td>
<td>157</td>
<td>7</td>
<td>185</td>
<td>6</td>
<td>44 (2)</td>
<td>X-PRESS TAJUMULCO</td>
<td>MAERSK WILMING</td>
<td>7</td>
<td>14</td>
</tr>
<tr>
<td>Vessel_11</td>
<td>1</td>
<td>5</td>
<td>156</td>
<td>8</td>
<td>14 (1)</td>
<td>X-PRESS TAJUMULCO</td>
<td>X-PRESS TAJUMULCO</td>
<td>5</td>
<td>15</td>
</tr>
<tr>
<td>Vessel_12</td>
<td>1</td>
<td>0</td>
<td>225</td>
<td>2</td>
<td>14 (1)</td>
<td>LOUISIANA TRADER</td>
<td>MAERSK NIUMGEN</td>
<td>0</td>
<td>2</td>
</tr>
</tbody>
</table>

Figure 8. Dialogue Box generated by python code with final multistep MILP for MIT 10/05/2016 schedule

The vessels schedule of 10th of May of 2016 used as an example provides with a scenario where berth 1 has $25 as the assigned cost when the minimum desirable is $0. This event triggers an analysis of the root cause that based on the graphs, schedule values and root database shows as a conclusion that the appointed position of the vessel Viking Merlin at berth 1 was not her desired position (1m) and was
instead moored at position 157m given that the position 1m was in service at the same time of need. The difference caused serving trucks to travel more distance as expected from the yard to the vessel and vice versa causing the following cost:

\[(157_m \text{ (given position)} - 1_m \text{ (desired position)}) \times 165 \text{ containers to be handled by M/V Viking Merlin} \times $0.001 \text{ (D.O cost per meter away of the desired position)} = $25.7\]

### 3.3 BAP & BAQCASTP rolling horizon problem

One of the biggest problems following the BAP/BAQCASTP multi-objective problem is the scenario of vessels being served after the considered planning horizon. That means that a vessel completion time will be fulfilled the next day (see Figure 9).

![Figure 9. Rolling horizon problem on MILP BAP/BAQCASTP](image)

The initial approach of the model is to process all the vessels scheduled for a specific day from 00:00 to 24:00. If a vessel being served from the previous day is not correctly included in the next day as part of the next day MILP problem, then the solution could have the chance of being not feasible as two vessels could be assigned to the same space/time section (see Figure 10). Additionally, most probably is that in case of running the model with the vessel not being forced to be processed at the same
position utilized at 24:00 for 00:01, a new assignment will be translated in a vessel requested to be shifted in less than 1 minute. Therefore, the assumption of a vessel being served continuously at the same position up to its completion will be violated. Another problem shown in Figure 10 is the infeasibility of one crane serving two vessels at the same time.

Figure 10. Rolling horizon violation of 3 vessels overlapping space and same cranes utilized by 3 vessels.

3.3.1 Rolling horizon problem proposed solution

The solution for this problem was approached by identifying a way that the MILP could force the vessels served from previous days to be assigned to the same quays, same positions and handled with the same or more cranes than initially assigned. In case of assigning a new crane(s), the only request is that the new combination should serve the vessel in the same or less time than the previously assigned crane(s). The assumption of being served by the same crane combination is relaxed just in this scenario.

Vessel next day assignment
By the use of the python module Pandas and some data base comprehension, a duplicate of a vessel could be generated with the next day time stamp and written in the very same database opened from a .csv file.

The criteria for choosing what vessel to duplicate in the database is selected by creating an “if” clause in python where the assigned BAQCASP berthing time of a vessel is then summed to the assigned processing time and if the value sums up to more than 24, then the vessel with all its related information is duplicated the next day.

After the vessel is processed the very next day, the code assures that a fresh copy of the database is generated from the original database, erasing the written duplicate and avoiding the introduction of errors caused by the creations of rows with empty columns.

**Same berth assignment**

The value of $m_{i,q}$ given as result of the BAP will help in generating a value in a newly created column in the original database referred from now on as the Preferred Berth column. This column will have the value 0 as a default and will generate a value pointing out if a vessel has a berth where it prefers to be allocated. This value is generated only in the case of a vessel surpassing the 24-hour barrier. Any value not being 0 in this column will trigger what is identified as the rolling horizon structure of the code.

Once the next day is processed through the code, the cost of being assigned to other berths different from the one pointed out at the Preferred Berth column will generate an extremely high cost of $50,000. As the BAP model aims to minimize the cost of berth assignment $c_{aiq_{i,q}}$, at all cost it will avoid assigning a vessel to a position other than the preferred.

**Same berth position engagement**

A new column called Berth Position is created at the original database with a default value of 0 for every existing vessel. In case a vessel enters into the rolling horizon condition, the berth position assigned as result of BAQCASP in the previous day is then added as part of the information of the duplicated vessel with the next day time stamp.
To force the BAP/BAQCASP to situate the vessel to the same position of the same berth it was served the previous day, the code will identify if the vessel has a preferred berth. Then it will provide a value of desired berth position $d_{ij}$ with the same values assigned as a solution of the previous day.

**Cranes assignment**

Given that the vessel has already processed part of the containers to be handled, the remaining containers are then computed based on the type of cranes and the assigned quantity of cranes together with the processing time up to the 24th-hour barrier. The result of this calculation is then added to the introduced vessel duplicated information as part of the existing column called containers to be handled.

To assign the duplicated vessel new desired time of departure, berthing time and processing time of the BAQCASP solution for that vessel are summed up to then be subtracted from 24. This will tell the BAP/BAQCASP that the vessel is requesting to finish operations at the desired departure time or before that. The problem is then forced to assign the necessary number of cranes required to fulfill the task of the remaining containers to be handled. By default, the cranes allocated to a vessel affected by the rolling horizon should be more or equal than the crane assignment for the previous day.

**Processing time assignment**

Any vessel affected by the rolling horizon problem will then be assigned to a value of 0 as the required processing time commencement; therefore, copied as part of the information of the duplicated vessel.

In order to force the model to serve the vessel at time 00:00, a value of $5,000 is added to the regular consideration of the cost of berth delay $c_{iw}$ for that vessel. Once again, this condition will be triggered with the Berth Preference column values. Any value of 0 will process the problem as usual, while any other number not being 0 will assign an extremely high cost to that vessel, hence avoiding violating her requirements.

**3.3.2 Expected results from proposed solution**

Using the same example of Figure 9 and Figure 10, now enabled with the proposed solution described in the previous section gives what is observed in Figure 11.
The forced introduction of two vessels from the previous day solution at berth 1, completely changes the assignment of the vessel *Erato* which is now positioned at berth 3 as compared with Figure 10 when it was assigned to berth 1. The crane violation from the previous example has now been solved with the small adjustment of having cranes being changed but keeping the same number as assigned the previous day.

As soon as the Figure 11 model was processed, the introduced new vessel duplicates were erased from the original database.

![Figure 11](image_url)

**Figure 11.** Schedule for the day following Figure 9 enabled with rolling horizon problem solution
4. DYNAMIC ADJUSTMENT OF BAP AND BAQCASp USING REAL TIME VESSELS AND PORT DATA

The key concept for the fulfillment of optimal dynamically adjusted port assignment problems is a collaborative approach. Any proposed solution in this section must be backed up by the consideration that shipping companies and the port terminal involved in the study should be convinced of exchanging real time key information via a third-party data analysis center, which at this time could be recognized as a synchromodal control center.

Synchromodality allows goods to be dynamically routed through the supply chain, continuously optimizing the transport modes and routes chosen, taking a balanced view of cost vs reliability vs sustainability vs lead time (MJC2, 2015).

Considering that the concept of synchromodality comprises a higher level of integration with additional modes and the final customer, the concept is over simplified by the sole use of ships and ports scheduling. However, this simple approach will help in identifying the benefits of a higher level of cooperation.

Now with the introduction of customized door to door services, in many cases the strategy behind the fulfillment of this level of customer service has to be with the effective network integration of port terminal operators with transport and third-party logistics providers (Rodrigue & Notteboom, 2012) as referred by (Song & Panayides, 2015).

The main idea behind the introduction of big data analytics on berth assignment is related to the fact that in some occasions vessels steam at a higher speed than needed to then wait in an anchorage for berth availability. Also, delays caused by a not optimal berth assignment to a vessel engaged in a liner service will develop in increased adjusted speed to reach planned schedule at next ports. On the other hand, port planning is compromised with vessels not arriving as reported with a worrying 27% chance (Drewry Shipping Consultants, 2012). It is well known that some ports plan their resources such as stevedores in advance so that a change in plans represent an additional cost of idle labor waiting to be reassigned.

In this dissertation, the information intended to be used to identify feasible combinations via big data analytics is exclusively from what the terminal and the vessels could actually offer based on the available technology.
4.1 Feasible Big Data values for BAP/BAQCASP adjustment

The idea of improving optimal BAP/BAQCASP problem via big data analytics starts with the opportunity observed from the idle slots in the presented solutions (see Figure 12). The goal is not to fill all the empty slots, but to identify, by offering better arrival times to vessels monitored in real time, if planning could be improved at the terminal and if vessels could reduce their consumption based on average speed.

The perceptiveness of this optimal solutions indicates that if a vessel is allowed to be allocated in an empty slot without obstructing the original vessels, the introduced vessel will be granted with their desired position without an extra cost and certainly will be released earlier than expected. Therefore, port utilization is improved and vessels are given the option to steam slower to the next destination.

The feed of information is powered by the Internet of Things (IoT) enhanced products. This concept was raised from the capacity of connecting every physical device to the Internet with the functionality of having that device exchanging information with people and other devices (Burrus, 2014). This trend has become the milestone for the third age of internet and has high expectations of connecting 50 billion objects by 2020 (World Economic Forum, 2015). The use of mobile sensors together with improved internet satellite connectivity and advances in cloud connectivity provides with the right environment to have vessels and terminal assets supplying and receiving information.

The easiness attached to enhancing existing equipment’s for being IoT ready opens the door of having real time information to become a reality in a short term.
4.1.1 Terminal input

Ports are like engines with several gears moving at the same time. The interaction of terminal dynamic parts such as trucks, yard cranes, quay cranes, gates and yard layout will give as a result the performance of the terminal.

Considering that just the waterborne interface is taken into account for this BAP/BAQCASP problem, which is a way to enhance the model could be by using real-time information exchanged by the quay cranes and their hourly container movement. It is important to remember that for calculating the processing times of vessels at the BAP, an average is taken based on the performance of all feasible cranes for the specific vessel. A more realistic average will provide with more robust solutions for BAP. In the same way, as the quay cranes per hour movement is updated, the same will happen to the BAQCASP and the processing time of vessels thereafter, based on the real characteristics of the quay cranes instead of quay cranes specifications. Quay cranes breakdown probability could also be a factor to improve
the model. A next stage of this research could be by enhancing the robustness of the model by including additional processing time-based on a stochastic analysis of the vessels/quay cranes probabilities of breakdown; the necessary values could be supplied by quay cranes performance real-time information.

A dynamically adjusted model requires a proactive and a reactive response. The proactive feed offered by ports is the information supplied by quay cranes, while the reactive response of ports, which is going to be their more important task, is the prompt reply for having vessels cleared from customs and assigned with a new stowage plan and yard layout plan generating instructions based on the mix of real time-shared information of the vessel and their own information.

A system well proved that has the availability to perform based on these requirements is the Computer Integrated Terminal Operations System (CITOS), developed by the port of Singapore in 1988. CITOS has among other developments, the capability of integrating assets for coordinating berthing system, ship planning system, yard planning systems, resource allocation, flow through gate and reefer monitoring. The perfect match to CITOS and which principle is required for applying a dynamic model in Panama is the introduction of a system such as PortNet, developed by PSA Corporation. The platform created by PortNet allows vessels and the port to make critical business decisions through the use of business intelligence while integrating regular port clearance platforms such as Haulier Community System and a document portal for vessels and port exchange in Cargo D2D. The single window concept being implemented by the FAL committee in IMO will further provide a regulatory background to the real-time exchange of documentation, which is very much needed in the reactive port approach.

4.1.2 Vessels input

Vessels have a more proactive approach because they provide a continuous feed that could help in improving the planning given by the BAP/BAQCASP problem. Their reactive action follows the instructions provided by the data analysis center given a new layout, schedule, and stowage plan.

Vessels had been traditionally recognized as disconnected from the technological global enhancements. In previous years this was a very realistic statement considering the few satellite communications offers given to vessels. However, at
present this is no longer an excuse as an extensive offer is provided by companies such as INMARSAT or IRIDIUM, which assures that real time data exchange is not just possible but also cheap.

From the available information on board that could enhance a BAP/BAQCASP model, the following could be used:

- **GPS positions**: Available via Satellite-Automatic Identification System (S-AIS) utilizing satellite connectivity and Automatic Identification System (AIS+) enhanced by Very High Frequency (VHF). A regular frequency of data sharing of 15 seconds will provide with the opportunity of calculating real speed over ground (SOG) and the average SOG for the whole voyage.

- **Passage plan**: The main interest of having a valid vessel passage plan is to have a distance to go (DTG) at the same time frequency as the Global Positioning System (GPS) stamps. Together with the SOG calculated from the GPS feed, a real ETA could also be calculated to confirm if a vessel could reach their assigned schedule window or if it could increase speed to arrive earlier based on open slots at the port. Electronic Chart Display and Information System (ECDIS) loaded passage plan could serve as a valid passage plan.

- **Speed Through Water (STW)**: Information generated by a speed doppler sensor and shared through S-AIS/AIS. The main function of this value, also generated every 15 seconds, is to determine if the vessel speed is affected by natural forces such as the wind or sea currents if different than SOG.

- **Weather information**: Information shared by an anemometer regarding wind speed and direction together by barographic information regarding atmospheric pressure will help in identify if a vessel is affected by weather or is just underperforming.

- **Vessel Revolutions per minute (RPM's)**: Together with the speed log information, the vessel SOG and saved vessel engine specifications, the performance of a vessel together with her speed will determine if the vessel is

---

4 AIS: Automatic Identification System. An automated tracking system that displays other vessels in the vicinity and supply with information via VHF to other stations. Required for vessels above 300 GT and engaged in international voyages as per SOLAS Regulation V/19.2.4.
capable of arriving on time. In other words, a container vessel steaming at 17 knots SOG with 90 RPM's with regular 1012mb/hPa and STW of 17 knots, while the vessel specifications indicate that the vessel at 90 RPM's should be steaming at 23 knots, is a clear indicative that the vessel is underperforming and that most probably if asked to increase speed to 23 knots to arrive earlier, they will not be able to perform.

4.2 Dynamic BAP/ BAQCASP development

So far in this dissertation, the development of the BAP followed by the BAQCASP, in addition to the rolling horizon problem had followed a static condition. In this kind of models, the required input values are received in advance to generate a schedule based on the available resources at the terminal and the promised arrival time of vessels.

This stage presents the adjustment of the same static models via information of the expected vessels that based on their real position, speed, next destination, and characteristics could be selected to adjust the delivered static BAP/BAQCASP solutions.

In order to have an idea of how a model can improve the berth utilization, a comparison will be done between a regular static model solution and how it evolves based on a dynamic adjustment.

4.2.1 Vessels real-time data random simulation

The values used for the dynamic solutions have been randomly created adapted from the original data base based on the assumption that each vessel belongs to a liner service in where all the origins before MIT together with all the destinations after MIT are combined.

A scenario was created and the schedule of vessels to arrive for the next 24 and 48 to 72 hours after the selected day is used to generate the simulated data.

A random geodetic coordinate was created for every vessel and was constrained by the approximate area where the vessel should sail to arrive in Panama, so a vessel whose origin is the Dominican Republic, should have a random position created in a valid passage between the Dominican Republic and Panama (see Figure 13). It is
possible to find cases where the vessel is still in the origin port prior departing to Panama.

Distances from and to MIT are calculated individually and introduced in the database, so every origin/destination going to/from Panama should be attached to its distance. Given the geodetic coordinate, a DTG calculation is carried and generated having Manzanillo Bay pilot station as the destination. The creation of random generated SOG from 15 to 25 knots for every vessel is now assumed as the average speed calculated for the sailed distance of a vessel.

RPM’s are also randomly created given the chosen vessel speed. The considerations took into account the following not related to any type of engine assumption: Any vessel steaming from 13 kts to 16 knots should have any number between 80 and 85 RPM, vessels between 16 to 19 kts between 85 to 89 RPM, vessels from 19 to 22 kts from 89 to 94 RPM and from 22 to 25 knots the range is between 94 and 100 RPM.

![Random geodetic positioning of vessels bound to MIT. Next day arrivals in orange, 48 hours or more arrivals in blue](image)

Values such as time to destination using average SOG, the difference between assigned ETA and real ETA, proposed ETA based on berth slots, speed needed to reach proposed ETA will be generated based on the values randomly created as per the previous description.
4.2.2 Vessel evaluation to consider for shifting to a new slot

Every vessel planned to arrive from 24 to 72 hours after time 0 of day 1 (initial day processed in static BAP/BAQCASP) will be part of an evaluation that considers several factors to determine their suitability for being shifted into a proposed empty slot. The considerations to be taken are:

1. The database with all the details needed as explained from the previous section has to be absorbed in 12:00 of day 1 for vessels arriving from 24 to 72 hours after.
2. Vessels arriving from 18:00 of day 1 to 06:00 of day 2 are candidates for being processed after 18:00 of day 1 if space is suitable for the vessel to be processed without any additional cost.
3. If a vessel is confirmed 24 or fewer hours in advance and more than 25 knots are needed to be on time, the vessel schedule will be modified with real ETA based on voyage average SOG.
4. Vessels with an expected south bound transit through the Panama Canal are completely disregarded as it is assumed that departure time could not be modified based on the reservation system of the Panama Canal. However, the real ETA is verified to identify if arriving as scheduled.
5. Vessels that are still in the origin port which is less than 300 NM from MIT will have their information analyzed 12 hours prior arrival.
6. A shift will be considered valid if the departure time allows the vessel to depart earlier and reach their next destination with a speed higher than 15 knots but less than 25 knots.
7. If a vessel finishes in a time where for its next destination and needs less than 15 knots, then it will be required to reduce speed prior arrival of MIT and arrive at its proposed time or at the next time slot where it could fulfill this condition. A case by case analysis will help in determining the best combination in this scenario.
8. Vessels assigned to a schedule as part of a BAP/BAQCASP solution cannot have their position adjusted unless optimal position could not be found in their assigned processing time.
4.2.3 Dynamic BAP/BAQCASP simulated scenario

The creation of this scenario takes into account the time period from the 3rd of May 2016 and the subsequent 24 to 72 hours following day 1 (see Figure 14). Day 1 is fixed up to 12:00 so no change is allowed for vessels assigned that time. However, vessels expected from 18:00 of day 1 to 06:00 of next day are prospects to adjust the last 6 hours of day 1.

![Figure 14. Scenario timeline](image)

The assignment of 12 vessels in a day makes more difficult the task of allocating vessels from a different schedule optimally in the few available spaces (see Figure 15). Additionally, the fact that no vessel of day 2 can be forced to steam at more than 25 knots to reach on time for any period on day 1.

![Figure 15. Schedule for vessels served on 3/5/2016](image)

**Day 2**
The schedule for day 2 gives the model the opportunity of having some adjustments based on its available slots (see Figure 16). The information of all the vessels is analyzed at 12:00 of day 1 and shown in Table 6.

Figure 16. Static BAP/BAQCASP for 4/5/2016 (Day 2)

Table 6. Vessels information after big data analysis/calculation for 4/5/2016

<table>
<thead>
<tr>
<th>Name</th>
<th>APL OMAN</th>
<th>DOMINGO</th>
<th>EMS TRADER</th>
<th>MAERSK WAKAYAMA</th>
<th>ROSSINI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Origin</td>
<td>Puerto Limon</td>
<td>La Guaira</td>
<td>Kingston</td>
<td>Kingston</td>
<td>Kingston</td>
</tr>
<tr>
<td>Origin_NM</td>
<td>192</td>
<td>834</td>
<td>555</td>
<td>555</td>
<td>555</td>
</tr>
<tr>
<td>Destination</td>
<td>Kingston</td>
<td>Cartagena</td>
<td>Guanta</td>
<td>Guanta</td>
<td>Guanta</td>
</tr>
<tr>
<td>Destination_NM</td>
<td>555</td>
<td>276</td>
<td>627</td>
<td>627</td>
<td>627</td>
</tr>
<tr>
<td>Latitude</td>
<td>9.75</td>
<td>13.35</td>
<td>18.20</td>
<td>17.98</td>
<td>18.33</td>
</tr>
<tr>
<td>Longitude</td>
<td>-83.00</td>
<td>-69.24</td>
<td>-78.97</td>
<td>-79.00</td>
<td>-78.96</td>
</tr>
<tr>
<td>DTG</td>
<td>184.79</td>
<td>671.25</td>
<td>532.41</td>
<td>519.42</td>
<td>540.40</td>
</tr>
<tr>
<td>SOG</td>
<td>15.5</td>
<td>24.4</td>
<td>18.7</td>
<td>16.5</td>
<td>21.2</td>
</tr>
<tr>
<td>RPM</td>
<td>81.3</td>
<td>95.9</td>
<td>88.5</td>
<td>86.7</td>
<td>91.2</td>
</tr>
<tr>
<td>ETA_proposed</td>
<td>4/5/2016 03:00</td>
<td>4/5/2016 15:00</td>
<td>4/5/2016 18:30</td>
<td>4/5/2016 19:00</td>
<td>4/5/2016 17:00</td>
</tr>
<tr>
<td>Speed Needed</td>
<td>12.3</td>
<td>24.9</td>
<td>17.5</td>
<td>16.5</td>
<td>19.3</td>
</tr>
<tr>
<td>New Departure</td>
<td>4/5/2016 10:00</td>
<td>5/5/2016 00:00</td>
<td>5/5/2016 03:36</td>
<td>5/5/2016 01:00</td>
<td>4/5/2016 21:00</td>
</tr>
<tr>
<td>Speed for next ETA</td>
<td>8.6</td>
<td>16.7</td>
<td>6.6</td>
<td>6.8</td>
<td>8.0</td>
</tr>
</tbody>
</table>

A case by case analysis is then done on every vessel and their actual conditions. The goal is to improve the existing schedule with day 2 slot availability and the feasibility of vessels to adjust speed to reach a better slot. The analysis of all the prospects (vessels) is described as detailed below:
• APL Oman. They are requested to reduce speed to super slow steaming of 12.3 knots for the last 185 NM in order to arrive at 4/5/2016 03:00 and be served at 05:00. It is not recommended for them to arrive at 23:00 of the 3rd as even after completing their service as they might need to steam less than 12 knots (not recommended for container vessels that consider 12 knots as super slow steaming and might cause engine underperformance) or drift at open sea to burn time to arrive JIT to their next destination.

As there is no incentive in arriving earlier than proposed, the recommendation is to burn less fuel in this leg (pre-MIT arrival) and still be JIT for an available slot.

• Domingo. As per the database analysis, it was discovered that they are going to be late to their proposed ETA and no notice was given to the terminal. As they are probably performing at their maximum capacity (24.4 knots in 96 RPM’s), the speed they could gain by increasing speed might be minimal.

It is recommended for them to maintain speed and arrive at real ETA of 4/5/2016 15:00. Originally, they were assigned to be served at 08:00, so this assignment will be adjusted by 17:00. If they arrive at this time, the speed needed to arrive at their next destination on time will be of 21.2 knots, a reason why is very important for them to maintain their speed.

• EMS Trader. With their actual speed, they should be arriving 1.5 hours before scheduled. An analysis of the benefit showed that for the next ETA they will need 7.6 knots; hence, not recommended. It is recommended instead to reduce their speed to 17.5 knots and arrive as initially scheduled at 18:30.

• Maersk Wakayama. As there are no vessels affected after their service if they delay for 3 hours, they are allowed to maintain speed and arrive 3 hours late at 19:00 instead of 16:06 as planned.

• Rossini. No vessel is affected for Rossini being late until 16:00. Considering the number of containers to be handled by them, their processing is expected to be fast. Therefore, they are asked to reduce speed to 19.3 knots and arrive at 16:00.

With no additional cost involved, the modifications processed through BAP/BACAPS returned the following:
Figure 17. Day 2 dynamically adjusted (above) and original schedule after BAP/BAQ/CASP (below)

The adjustment done to vessels enhanced a still optimal schedule. The result from this adjustment allowed vessels to arrive JIT for their new schedules based on their new distances to go and avoiding waiting for a berth. Waiting for a berth impacts directly on cost generation and the increment of local emissions in Panama. By reducing speed, savings could be generated. A good example is the one observed with the 1.9 knots saving for 540 NM in the case of Rossini and a shift from the normal speed section with 21.2 knots to the slow steaming section with 19.3 knots as referenced in Figure 18.
Figure 18. Fuel consumption depending on ship speed. Source: (Notteboom & Carriou, Fuel surcharge practices of container shipping lines: Is it about cost recovery or revenue making), 2009) as referenced by (Hofstra University, 2009)

Using the actual IFO 380 price of $270/ton in Panama (Clarksons Research Limited Services., 2017) and given that Rossini is a 3,000 TEU vessel, the savings for the speed reduction could be presented as follows:

Table 7. Savings caused by speed reduction to arrive just in time for berth allocation

<table>
<thead>
<tr>
<th>Speed</th>
<th>Consumption per day</th>
<th>Consumption per hour</th>
<th>Hours to go (540NM)</th>
<th>Tons consumed</th>
<th>Cost (Actual price)</th>
</tr>
</thead>
<tbody>
<tr>
<td>21.3 kts.</td>
<td>80 tons</td>
<td>3.33 tons</td>
<td>25.3</td>
<td>84.2</td>
<td>$22,734</td>
</tr>
<tr>
<td>19.2 kts.</td>
<td>65 tons</td>
<td>2.70 tons</td>
<td>28.1</td>
<td>75.9</td>
<td>$20,493</td>
</tr>
</tbody>
</table>

Rossini saving: $2,241

A case analysis with the prices of September 2013 for IFO in Panama was $624.00 (Clarksons Research Limited Services., 2017) shows a non-despicable saving of $5,054.00 in one day.
Day 3

Four vessels are expected to be processed on day 3 together with the two vessels overpassing the rolling horizon barrier from day 2 (see Figure 19). The information of all the vessels is analyzed at 12:00 of day 1 and shown in Table 8.

Figure 19. Static BAP/BAQCASP solution for 5/5/2016 (Day 3)

Table 8. Vessels information after big data analysis/calculation for 5/5/2016

<table>
<thead>
<tr>
<th>Name</th>
<th>ANTWERP TRADER</th>
<th>LIVERPOOL EXPRESS</th>
<th>LUTETIA</th>
<th>SAN ALESSIO</th>
</tr>
</thead>
<tbody>
<tr>
<td>Origin</td>
<td>Mariel</td>
<td>La Guaira</td>
<td>Puerto Cabello</td>
<td>Caucedo</td>
</tr>
<tr>
<td>Origin_NM</td>
<td>970</td>
<td>834</td>
<td>793</td>
<td>824</td>
</tr>
<tr>
<td>Destination</td>
<td>Cartagena</td>
<td>Cartagena</td>
<td>Cartagena</td>
<td>Caucedo</td>
</tr>
<tr>
<td>Destination_NM</td>
<td>276</td>
<td>276</td>
<td>276</td>
<td>824</td>
</tr>
<tr>
<td>Latitude</td>
<td>25.3</td>
<td>10.0</td>
<td>10.0</td>
<td>18.8</td>
</tr>
<tr>
<td>Longitude</td>
<td>-79.0</td>
<td>-67.3</td>
<td>-68.4</td>
<td>-69.9</td>
</tr>
<tr>
<td>DTG</td>
<td>955.9</td>
<td>744.2</td>
<td>680.2</td>
<td>813.1</td>
</tr>
<tr>
<td>SOG</td>
<td>23.7</td>
<td>20.2</td>
<td>22</td>
<td>15.7</td>
</tr>
<tr>
<td>RPM</td>
<td>97.1</td>
<td>93.1</td>
<td>101.6</td>
<td>83.9</td>
</tr>
<tr>
<td>ETA_real</td>
<td>5/5/2016 04:00</td>
<td>5/5/2016 00:00</td>
<td>4/5/2016 18:00</td>
<td>5/5/2016 15:00</td>
</tr>
<tr>
<td>ETA_assigned</td>
<td>5/5/2016 08:06</td>
<td>5/5/2016 00:06</td>
<td>5/5/2016 14:06</td>
<td>5/5/2016 00:30</td>
</tr>
<tr>
<td>ETA_proposed</td>
<td>5/5/2016 07:00</td>
<td>5/5/2016 00:06</td>
<td>5/5/2016 08:00</td>
<td>5/5/2016 15:00</td>
</tr>
<tr>
<td>Speed Needed</td>
<td>22.2</td>
<td>20.6</td>
<td>15.5</td>
<td>15.9</td>
</tr>
<tr>
<td>Speed for next ETA</td>
<td>24.9</td>
<td>15.0</td>
<td>11.5</td>
<td>13.0</td>
</tr>
</tbody>
</table>
In the case by case analysis of these four vessels, some adjustment changed the original schedule for day 3. However, no additional cost is added for doing the rearrangement. Additional, one full berth is not utilized. Hence, maintenance could be done to the quay cranes and berth. In order to reach the results for the schedule on day 3 the following considerations are taken into account:

- Antwerp Trader. Is requested to reduce their speed to 22.2 knots from 23.7 knots, to arrive at 07:00 of the 5th. An adjustment that represents arriving one hour earlier than planned, but very much needed to avoid being requested to steam at more than 25 knots to next destination.
- Liverpool Express. Should keep steaming at the same speed to arrive at planned ETA.
- Lutetia. Speed reduction of 15.4 knots from 22 knots is instructed for them to arrive at 08:00 of the 5th. With an arrival 6 hours prior the initial schedule, they will be served and still have spare time to steam at super slow steaming to the next destination.
- San Alessio. As they are already steaming at super slow steaming. Their instruction is to maintain speed an arrive 15 hours later than assigned. They will have the opportunity to steam at super slow steaming, even after their delay.

The new results based on the adjustments delivered tighter vessel schedules. Compared to the static BAP/BAQCASP a better utilization is fulfilled with berth 2 and 3 (see Figure 20).
A good example of savings caused by the dynamic solution is the case of the vessel Antwerp Trader where a fuel consumption reduction is expected. As part of day 3 schedule and based on the position and SOG delivered by Antwerp Trader, their assignment needs to be selected based on the DTG and the distance to the next destination which is Cartagena at 276 NM. This case presents a tradeoff between increasing speed and arriving earlier so less speed could be used for next destination or maintain speed and arrive later at MIT then be pushed to steam faster to the next destination. The advantages of both scenarios which are presented next will help in understanding how the decision was taken to assign this vessel.

Table 9. Arrival at 04:00 scenario. Consumption difference for next destination

<table>
<thead>
<tr>
<th>Speed (knots)</th>
<th>Consumption per day (tons)</th>
<th>Consumption per hour (tons)</th>
<th>Hours to next dest (276NM)</th>
<th>Tons consumed</th>
<th>Cost ($270/ton)</th>
</tr>
</thead>
<tbody>
<tr>
<td>19.6</td>
<td>60</td>
<td>2.5</td>
<td>14.1</td>
<td>35.3</td>
<td>$9,531</td>
</tr>
<tr>
<td>25.1</td>
<td>150</td>
<td>6.3</td>
<td>11.0</td>
<td>69.3</td>
<td>$18,711</td>
</tr>
</tbody>
</table>

Savings: $9,180
Table 10. Arrival at 07:00 scenario. Consumption difference for MIT arrival.

<table>
<thead>
<tr>
<th>Speed (knots)</th>
<th>Consumption per day (tons)</th>
<th>Consumption per hour (tons)</th>
<th>Hours to go (970NM)</th>
<th>Tons consumed</th>
<th>Cost ($270/ton)</th>
</tr>
</thead>
<tbody>
<tr>
<td>22.2</td>
<td>100</td>
<td>4.2</td>
<td>43.7</td>
<td>183.5</td>
<td>$49,545</td>
</tr>
<tr>
<td>23.7</td>
<td>140</td>
<td>5.8</td>
<td>40.9</td>
<td>237.2</td>
<td>$64,044</td>
</tr>
</tbody>
</table>

Savings: $14,499

07:00 Scenario − 04:00 Scenario = $14,499 − $9,180 = $5,319

04:00 Scenario − 07:00 Scenario = $9,180 − $14,499 = ($5,319)

Making the comparison of both situations, it is clear that based on the fact that the next destination is closer from MIT than the distance needed to arrive at MIT, the selection of 07:00 scenario could generate an overall saving of $5,319 in fuel consumption. The difference is more significant if a depressed market is faced with low freight rates and higher bunker prices. Still, a saving generated in two days is very much welcomed by liner shipping companies.
5. CONCLUSIONS

The use of the BAP and BAQCASP problem which are operation research problems adapted to the environment of MIT has proven in this dissertation to be beneficial for the optimal scheduling of vessels based on the specific characteristics of the port. This MILP has been extensively applied by ports throughout the world, so an innovative approach required the implementation of a different concept that could enhance the weak points faced by their users. The introduction of big data as an operations researcher enhancer represents a whole new area of additional research.

The BAP provides an initial adjustment where vessels could be categorized by berth, so they then could be processed by individual BACAPS problems. The input for BAP is highly modifiable in the generated code for this dissertation. Fixed information, such as quay cranes performance, berth length and vessels to be served are easily adapted for any port condition.

BAQCASP introduction of quay cranes into the scene, proved demanding in terms of computation power, which is a reason why additional constraints and pre-input adjustment are needed and generated by the code before processing the MILP. The result as a sum of the results of the BAQCASPs involved provides with an overall minimum cost of the terminal scheduling for a planned day.

The introduction of a collaborative model between ports and liner shipping services through live information sharing has a high potential of being beneficial. This logic is proven with the integration of Big Data and the BAP/BAQCASP model in this dissertation. From the results of the scenarios following a dynamic adjustment of BAP/BAQCASP based on Big Data information some clear benefits were found and could be classified as:

- Precise stevedores’ assignment
- Optimal berth and quay cranes allocation
- Savings in fuel consumption based on efficient collaborative ETA control
- Savings in electricity demand based on optimal allocation of cranes
- Local emission reduction caused by D.O generators of vessels being anchored less than needed in Panamanian waters.
- Port emissions reduction based on optimal yard trucks traveling distances
• Global emission reduction based on efficient control of speed in liner service legs.
• An adaptable maintenance window for berth and quay cranes given by optimal allocation of vessels forced to be optimally allocated in 2 berths.
• A highly adaptable algorithm to vessels’ ETA, quay cranes performance, quay cranes breakdowns.

The adaptability of traditional logistics equipment to share big data by being enabled to be IoT ready requires an additional investment whose analysis could be a source of additional research.

The maritime industry seems to be targeted towards the implementation of technologies that could enhance the live interaction of vessels with other logistics stakeholders. The introduction of the single window concept by IMO FAL committee is advancing by leaps and bounds and will provide a solid background towards the easy exchange of information from vessels. ISO electronic record books standard is being developed for marine purposes as an initiative to ease the manufacturing and testing of equipment being used to exchange information from vessels. The ever-evolving maritime satellite communications industry is already offering services that could facilitate the introduction of big data exchange.

The implementation of a collaborative model between vessels and ports is no longer constrained by the cost and availability of technology but for the disposition of ship owners and logistics stakeholders towards the better utilization of common assets.
REFERENCES


Google Maps. (2017, August 13). Aerial image of the port of MIT. Retrieved from https://www.google.se/maps/place/Manzanillo+International,+Terminal+Panam%C3%A1/@9.365064,-79.881483,17z/data=!3m1!4b1!4m5!3m4!1s0x8fab6d0481e9891d:0xfe963a91ebe8d010!8m2!3d9.365064!4d-79.881483?hl=en


## APPENDICES

Appendix A. Liner services including MIT

<table>
<thead>
<tr>
<th>Service</th>
<th>Ports</th>
</tr>
</thead>
<tbody>
<tr>
<td>Crowley/Zim - Jamaica/Costa Rica Service</td>
<td>Jacksonville, Port Everglades, Kingston, Puerto Limón, Manzanillo-Panama, Puerto Limón, Port Everglades, Jacksonville</td>
</tr>
<tr>
<td>Hamburg Sud/Maersk Line - Alianca/CMA CGM/Hapag/MSC/SeaLand - Oceania Service</td>
<td>Philadelphia, Charleston, Cartagena, Balboa, Auckland, Sydney, Melbourne, Port Chalmers, Napier, Tauranga, Auckland, Manzanillo-Panama, Cristobal, Cartagena, Philadelphia</td>
</tr>
<tr>
<td>Maersk Line/SeaLand – ECUBEX</td>
<td>St Petersburg, Bremerhaven, Rotterdam, Antwerp, Cartagena, Manzanillo-Panama, Balboa, Guayaquil, Balboa, Manzanillo-Panama, Rotterdam, Hamburg, Bremerhaven, St Petersburg</td>
</tr>
<tr>
<td>Hamburg Sud/Hapag Lloyd/Alianca/CMA CGM/ANL - SW1/Eurosal Sling 1</td>
<td>Rotterdam, London Gateway, Hamburg, Antwerp, Le Havre, Caucedo, Cartagena, Manzanillo-Panama, Buenaventura, Callao, Mejillones, Valparaiso, Callao, Buenaventura, Manzanillo-Panama, Cartagena, Caucedo, Rotterdam</td>
</tr>
<tr>
<td>Maersk Line/Safmarine/SeaLand – ECUMED</td>
<td>Guayaquil, Balboa, Manzanillo-Panama, Algeciras, Marsaxlokk, Izmit, Ambarli side, Estambul, Odessa, Novorossiysk, Ambarli side, Estambul, Izmit, Izmir, Algeciras, Caucedo,</td>
</tr>
<tr>
<td>Line</td>
<td>Destinations</td>
</tr>
<tr>
<td>------------------------------------------------</td>
<td>------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Manzanillo-Panama, Buenaventura, Guayaquil</td>
<td></td>
</tr>
<tr>
<td>NYK RoRo - US Gulf Express PCTC Loop 2</td>
<td>Veracruz, Port Canaveral, FL, Jacksonville, Baltimore, Port of NY &amp; NJ, San Juan, Santo Domingo, Santa Marta, Cartagena, Manzanillo-Panama, Puerto Limón, Puerto Cortes, Santo Tomas de Castilla, Veracruz</td>
</tr>
<tr>
<td>Wallenius Wilhelmsen - South America/North America Service PCTC Loop 1</td>
<td>Rio Grande, Paranaquia, Santos, Cartagena, Manzanillo-Panama, Veracruz, Galveston, Cartagena, Manzanillo-Panama, Callao, San Antonio, Rio Grande</td>
</tr>
<tr>
<td>Wallenius Wilhelmsen - RTW Service one way PCTC</td>
<td>Bremerhaven, Zeebrugge, Southampton, Manzanillo-Panama, Port Hueneme, Tacoma, Yokohama, Bremerhaven</td>
</tr>
<tr>
<td>CMA CGM/Alianca/ANL/Hamburg Sud/Maersk/Melfi/SeaLand - Brazil Express</td>
<td>Vitoria, Navegantes, Paranaagua, Santos, Rio de Janeiro, Salvador, Cartagena, Kingston, New Orleans, Houston, Altamira, Veracruz, Kingston, Cartagena, Manzanillo-Panama, Manaus, Vitoria</td>
</tr>
<tr>
<td>Shipping Lines</td>
<td>Routes</td>
</tr>
<tr>
<td>----------------</td>
<td>--------</td>
</tr>
<tr>
<td>Hamburg Sud/Maersk Line - Alianca/SeaLand - Central America/Caribbean Loop 1</td>
<td>Manzanillo-Panama, Cristobal, Cartagena, La Guaira, Manzanillo-Panama</td>
</tr>
<tr>
<td>CMA CGM/Marfret/ANL - RTWPAN</td>
<td>Manzanillo-Panama, Mariel, Santiago de Cuba, Cartagena, Manzanillo-Panama</td>
</tr>
<tr>
<td>Wallenius Wilhelmsen - North America/Europe/South America One Way RoRo Service</td>
<td>Bremerhaven, Zeebrugge, Southampton, Santander, Baltimore, Galveston, Veracruz, Manzanillo-Panama, Cartagena, Santa Marta</td>
</tr>
<tr>
<td>Melfi Marine/Zim - Panama/Cuba Express-PANCUX</td>
<td>Manzanillo-Panama, Barranquilla, Cartagena, Santa Marta, Point Lisas, Georgetown, Paramaribo, Point Lisas, Cartagena, Barranquilla, Santa Marta, Manzanillo-Panama</td>
</tr>
<tr>
<td>Maersk Line/MSC/SeaLand/APL - Caribbean Feeder Service</td>
<td>Puerto Limón, Manzanillo-Panama, Caucedo, San Juan, Rio Haina, Caucedo, Puerto Limón</td>
</tr>
<tr>
<td>Maersk Line/Hapag-Lloyd/Seaboard/SeaLand - Calypso Feeder Service</td>
<td>Manzanillo-Panama, Barranquilla, Cartagena, Santa Marta, Point Lisas, Georgetown, Paramaribo, Point Lisas, Cartagena, Barranquilla, Santa Marta, Manzanillo-Panama</td>
</tr>
<tr>
<td>Caribbean Feeder/Hapag-Lloyd/Maersk Line/SeaLand/Zim - MIT/MAR</td>
<td>Kingston, El Guamache, Guaranao, Maracaibo, Manzanillo-Panama, Kingston</td>
</tr>
<tr>
<td>Hamburg Sud/Alianca/Maersk/SeaLand - Central America/Caribbean Loop 2</td>
<td>Manzanillo-Panama, Cartagena, La Guaira, Manzanillo-Panama</td>
</tr>
<tr>
<td>Line</td>
<td>Ports</td>
</tr>
<tr>
<td>--------------</td>
<td>--------------------------------------</td>
</tr>
<tr>
<td>CMA CGM/COSCO/HamSud/Hapag-Lloyd/Alianca/ANL/APL-PEX2/JCS</td>
<td>Singapore, Chiwan, Hong Kong, Kaohsiung, Ningbo, Shanghai, Qingdao, Busan, Ensenada, Manzanillo-Mexico, <strong>Manzanillo-Panama</strong>, Cartagena, Kingston, Caucedo, Singapore</td>
</tr>
<tr>
<td>Caribbean Feeder/Maersk/Hapag-Lloyd/Zim - Manzanillo/Guanta Service</td>
<td>Kingston, <strong>Manzanillo-Panama</strong>, Guanta, Kingston</td>
</tr>
<tr>
<td>Hapag Lloyd/Hamburg Sud-Alianca/Zim - Med Pacific Services MPS</td>
<td>Cagliari, Leghorn, Genoa, Marseilles, Fos, Barcelona, Valencia, Tanger, Cartagena, Puerto Quetzal, <strong>Manzanillo-Mexico</strong>, Long Beach, Oakland, Seattle, Vancouver, Oakland, Long Beach, <strong>Manzanillo-Panama</strong>, Cartagena, Caucedo, Lisbon, Tanger, Valencia, Cagliari</td>
</tr>
<tr>
<td>NYK RoRo - USCX PCTC one way</td>
<td>Sagunto, Bremerhaven, Zeebrugge, Halifax, Port of NY &amp; NJ, Baltimore, Jacksonville, Houston, Veracruz, Cartagena, <strong>Manzanillo-Panama</strong>, Guayaquil, Callao, Iquique, San Antonio</td>
</tr>
<tr>
<td>Wallenius Wilhelmsen - RTW RoRo Service</td>
<td>Zeebrugge, Bremerhaven, Southampton, Baltimore, Savannah, <strong>Manzanillo-Panama</strong>, Auckland, Brisbane, Port Kembla, Melbourne, Fremantle, Singapore, Incheon, Gunsan, Masan, Kobe, Nagoya, Hitachinaka, Yokohama, Tacoma, Long Beach, Lazaro Cardenas</td>
</tr>
<tr>
<td>Line</td>
<td></td>
</tr>
<tr>
<td>---------------------------------------------------------------------</td>
<td></td>
</tr>
<tr>
<td>Hoegh Autoliners/SC Lines - Caribbean Short Sea Service</td>
<td></td>
</tr>
<tr>
<td>Port Everglades, Nassau, Kingston, Port au Prince, Santo Domingo,</td>
<td></td>
</tr>
<tr>
<td>Cartagena, <strong>Manzanillo-Panama</strong>, Altamira, Veracruz, Kingston,</td>
<td></td>
</tr>
<tr>
<td>Santa Marta, San Juan, Port Everglades</td>
<td></td>
</tr>
<tr>
<td>CMA CGM/Marfret - Mediterranean Caribbean Service</td>
<td></td>
</tr>
<tr>
<td>Algeciras, Leghorn, Genoa, Marseilles Fos, Barcelona, Valencia,</td>
<td></td>
</tr>
<tr>
<td>Tanger, Pointe a Pitre (Port de Jarry), Fort de France, Cartagena,</td>
<td></td>
</tr>
<tr>
<td><strong>Manzanillo-Panama</strong>, Puerto Limón, Turbo, Pointe a Pitre (Port</td>
<td></td>
</tr>
<tr>
<td>de Jarry), Algeciras</td>
<td></td>
</tr>
<tr>
<td>X-Press/CMA CGM/APL/Evergreen Line/King Ocean/MOL/Seaboard Ocean/</td>
<td></td>
</tr>
<tr>
<td><strong>Manzanillo-Panama</strong>, Cartagena, Barranquilla, Santa Marta,</td>
<td></td>
</tr>
<tr>
<td>Cartagena, <strong>Manzanillo-Panama</strong>, Puerto Limón, <strong>Manzanillo-Panama</strong></td>
<td></td>
</tr>
<tr>
<td>NYK RoRo/Wallenius Wilhelmsen - South America/North America Service</td>
<td></td>
</tr>
<tr>
<td>Zarate, Paranaqua, Santos, Cartagena, <strong>Manzanillo-Panama</strong>,</td>
<td></td>
</tr>
<tr>
<td>Veracruz, Houston, Tampa, <strong>Manzanillo-Panama</strong>, Oranjestad,</td>
<td></td>
</tr>
<tr>
<td>Willemstad, Santos, Paranaqua, Zarate</td>
<td></td>
</tr>
<tr>
<td>Maersk/SeaLand/APL - North Atlantic Express NAE/ACX</td>
<td></td>
</tr>
<tr>
<td>Cartagena, <strong>Manzanillo-Panama</strong>, Port Everglades, Philadelphia,</td>
<td></td>
</tr>
<tr>
<td>New York, Savannah, Port Everglades, Cartagena</td>
<td></td>
</tr>
<tr>
<td>Hapag Lloyd/Maersk Line/SeaLand - Gulf Express GCS</td>
<td></td>
</tr>
<tr>
<td>Houston, Altamira, Veracruz, Santo Tomas de Castilla, Puerto Cortes,</td>
<td></td>
</tr>
<tr>
<td>Puerto Limón, <strong>Manzanillo-Panama</strong></td>
<td></td>
</tr>
<tr>
<td>Shipping Line/Feeder</td>
<td>Ports</td>
</tr>
<tr>
<td>----------------------</td>
<td>-------</td>
</tr>
<tr>
<td>Maersk Line/Safrmarine/SeaLand - South Atlantic Express</td>
<td>Cartagena, Caucedo, San Juan, Cartagena, <strong>Manzanillo-Panama</strong>, Puerto Limón, Santo Tomas de Castilla, Puerto Cortes, Houston</td>
</tr>
<tr>
<td>Caribbean Feeder/Hapag-Lloyd/Zim-ACPL</td>
<td>Norfolk, Philadelphia, Wilmington, Savannah, Santo Tomas de Castilla, Puerto Cortes, Moin Bay, <strong>Manzanillo-Panama</strong>, Colon Container Terminal S.A., Puerto Cortes, Santo Tomas de Castilla, Wilmington, Norfolk</td>
</tr>
<tr>
<td>Seaboard/King Ocean - South Central America</td>
<td>Kingston, Montego Bay, <strong>Manzanillo-Panama</strong>, Colon Container Terminal S.A., Willemstad, Barcadera, Kingston</td>
</tr>
<tr>
<td>The Alliance - North America East Coast-EC1</td>
<td>Miami, Port Everglades, Puerto Limón, <strong>Manzanillo-Panama</strong>, Cristobal, Puerto Limón, Port Everglades, Miami</td>
</tr>
<tr>
<td>NYK RoRo - US Gulf Express PCTC Loop 1</td>
<td>Kobe, Ningbo, Shanghai, Busan, Tokyo, <strong>Manzanillo-Panama</strong>, Savannah, Jacksonville, Charleston, Norfolk, <strong>Manzanillo-Panama</strong>, Balboa, Los Angeles, Oakland, Tokyo, Kobe</td>
</tr>
<tr>
<td>Caribbean Feeder/Alianca/Seaboard/Hamburg Sud/Zim - Carico Service</td>
<td>Kingston, Rio Haina, Caucedo, San Juan, Kingston, Cartagena, Cartagena</td>
</tr>
<tr>
<td>航线</td>
<td>港口目的地</td>
</tr>
<tr>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>Barranquilla, Cartagena, <strong>Manzanillo-Panama</strong>, Kingston</td>
<td>Cork, Tilbury, Rotterdam, Bremerhaven, Mariel, Big Creek, <strong>Manzanillo-Panama</strong>, Moin Bay, Cork</td>
</tr>
<tr>
<td>Maersk Line/Safmarine - CRX</td>
<td>Seaboard - North Central America vía Houston</td>
</tr>
<tr>
<td>Houston, New Orleans, Puerto Cortes, Santo Tomas de Castilla, Cartagena, Turbo, <strong>Manzanillo-Panama</strong>, Puerto Limón, Puerto Cortes, Santo Tomas de Castilla, Houston</td>
<td>Cork, Tilbury, Rotterdam, Bremerhaven, Mariel, Big Creek, <strong>Manzanillo-Panama</strong>, Moin Bay, Cork</td>
</tr>
<tr>
<td>Evergreen/MOL/SeaLand/X Press/Zim - CAN</td>
<td>Colon Container Terminal S.A., <strong>Manzanillo-Panama</strong>, Caucedo, Rio Haina, San Juan, Caucedo, Colon Container Terminal S.A.</td>
</tr>
<tr>
<td>MSC-MOL-Amazone Feeder/CX1</td>
<td><strong>Manzanillo-Panama</strong>, Cristobal, Port of Spain, Macapa, Vila do Conde, <strong>Manzanillo-Panama</strong></td>
</tr>
</tbody>
</table>

Source: (Comision Centroamericana de Transporte Marítimo [Central American Maritime Transport commission], 2017)
Appendix B. Python code for BAP/BAQCASP MILP solution, database manipulation, graph and schedule .csv file creation

```python
1. from itertools import *
2. from collections import defaultdict
3. from pulp import *
4. from datetime import datetime, timedelta
5. import math
6. import pandas as pd
7. import matplotlib.pyplot as plt
8. import matplotlib.patches as patches
9. import numpy as np
10. from Tkinter import *
11.
12. # Steady inputs
13. # Number of Quays
14. Q=3
15. quays=range(1,Q+1)
16.
17. # Quays length
18. lq={1:1240,2:400,3:400}
19. lmax=max(lq[x] for x in lq)
20.
21. # Cranes info
22. # Crane reach
24. total_cranes=range(1,len(cr.keys())+1)
25.
26. # Distribution of cranes per group
27. distr={1:[1,2,3,4,5,6],2:[7,8,9,10,11],3:[12,13,14,15],4:[16,17,18,19]}
28.
29. # Processing time per crane of each group
30. proc={1:28,2:30,3:31,4:32}
31.
32. # Cranes distribution to their respective quays
33. distr_quays={1:[1,2],2:[3],3:[4]}
34. ######
35. # Input of data absorbed from database in csv of planning + particulars + arrival and departure of vessels
```
file = open("Merged_df2.csv")
calls = pd.read_csv(file)
file.close()
calls["ATA_Day"] = pd.to_datetime(calls["ATA_Day"], format='%m/%d/%Y')
calls["ATD_Day"] = pd.to_datetime(calls["ATD_Day"], format='%m/%d/%Y')
calls["ATA_Time"] = pd.to_datetime(calls["ATA_Time"], format='%H:%M:%S')
calls["ATD_Time"] = pd.to_datetime(calls["ATD_Time"], format='%H:%M:%S')
calls["ATD_Time_dec"] = calls["ATD_Time"].dt.hour + calls["ATD_Time"].dt.minute / 60
calls["ATA_Time_dec"] = calls["ATA_Time"].dt.hour + calls["ATA_Time"].dt.minute / 60
date_str = calls["ATA_Day"].dt.strftime("%Y-%m-%d")
list_date = []
for i in range(date_str.count()):
    list_date.append(date_str.iloc[i])

Assigned departure time. 2 hours prior transit, Time to sail and arrive next destination with average of 20 kts.

si = defaultdict(dict)
li = defaultdict(dict)
raw_li = defaultdict(dict)
b = defaultdict(dict)
ai = defaultdict(dict)
c = defaultdict(dict)
reef = defaultdict(dict)
reef = defaultdict(dict)
dc = defaultdict(dict)
qcmin = defaultdict(dict)
qcmax = defaultdict(dict)
diq = defaultdict(dict)

Desired position for vessel depending in condition.

Cost of unit of time of delay prior berthing. Bunkering cost 103.80 per hour H.F.O 380 cst for Deep Sea vessels and
75. # 30.30 per hour D.O for feeder vessels 150m. Anchorage not payable, out of breakwater
76. ciw=defaultdict(dict)
77. #Cost of delay of one hour from desired departure time si. Bunker + Canal cancellation fee 100% of reserve.
78. cid=defaultdict(dict)
79. #Cost of not being located in desired position
80. cpi=defaultdict(dict)
81. #Name of vessel
82. name=defaultdict(dict)
83. #Cost of berth assignment
84. caiq=defaultdict(dict)
85. 86. ###Function to start system
87. def input_BAP(schedule_name,date_YYYYmmdd):
88.     schedule=calls.loc[calls["ATA_Day"]=='%s'%date_YYYYmmdd,:]
89.     schedule.loc[:,"Day_number"]=[i for i in range(1,schedule["Name"].count()+1)]
90.     # #Departure time
91.     for i in range(schedule["Name"].count()):
92.         if schedule["ATA_Day"].iloc[i]==schedule["ATD_Day"].iloc[i]:
93.             si[schedule["Day_number"].iloc[i]]=int(math.ceil(schedule["ATD_Time_dec"].iloc[i]))
94.         else:
95.             t=schedule["ATD_Day"].iloc[i]-schedule["ATA_Day"].iloc[i]
96.             a=np.timedelta64(t,"D")
97.             timeadded=(a/np.timedelta64(1,"D")).astype(int)*24
98.             next_day=math.ceil(timeadded + schedule["ATD_Time_dec"].iloc[i])
99.             si[schedule["Day_number"].iloc[i]]=int(next_day)
100. # Vessel values
101.     for i in range(schedule["Name"].count()):
102.         ##Beam
103.         b[schedule["Day_number"].iloc[i]]=schedule["Beam"].iloc[i]
104.         ##Arrival
105.         ai[schedule["Day_number"].iloc[i]]=int(math.ceil(schedule["ATA_Time_dec"].iloc[i]))
106.         ##Containers
107.         c[schedule["Day_number"].iloc[i]]=int(schedule["Containers"].iloc[i])
108.         ##Reefer containers
80
reef[schedule["Day_number"].iloc[i]]=int(schedule["Reefers"].iloc[i])
#Dangerous cargo
dc[schedule["Day_number"].iloc[i]]=int(schedule["Dangerous_cargo"].iloc[i])
#Minimum cranes allowed
qcmin[schedule["Day_number"].iloc[i]]=int(schedule["Min_cranes"].iloc[i])
#Maximum cranes allowed
qcmax[schedule["Day_number"].iloc[i]]=int(schedule["Max_cranes"].iloc[i])
##Minimum cranes allowed
qcmin[schedule["Day_number"].iloc[i]]=int(schedule["Min_cranes"].iloc[i])
##Maximum cranes allowed
qcmax[schedule["Day_number"].iloc[i]]=int(schedule["Max_cranes"].iloc[i])
###Length
if schedule["Length"].iloc[i] > 130:
    li[schedule["Day_number"].iloc[i]]=schedule["Length"].iloc[i]+(int(
math.ceil(schedule["Length"].iloc[i]*0.10)))
else:
    li[schedule["Day_number"].iloc[i]]=schedule["Length"].iloc[i]+10
#Raw length not including safety margin
raw_li[schedule["Day_number"].iloc[i]]=schedule["Length"].iloc[i]

#Adjustable quays description and desired position of vessels
if schedule["Berth_pref"].iloc[i] == 0:
    if schedule["Length"].iloc[i] >= 250:
        diq[schedule["Day_number"].iloc[i][1]=lq[1]-schedule["Length"].iloc[i]+(int(
math.ceil(schedule["Length"].iloc[i]*0.10))))
        diq[schedule["Day_number"].iloc[i][2]=1
        diq[schedule["Day_number"].iloc[i][3]=1
    elif schedule["Reefers"].iloc[i] >=50:
        diq[schedule["Day_number"].iloc[i][1]=lq[1]/2
        diq[schedule["Day_number"].iloc[i][2]=1
        diq[schedule["Day_number"].iloc[i][3]=1
    elif schedule["Dangerous_cargo"].iloc[i] >=50:
        diq[schedule["Day_number"].iloc[i][1]=lq[1]-schedule["Length"].iloc[i]+(int(
math.ceil(schedule["Length"].iloc[i]*0.10))))
        diq[schedule["Day_number"].iloc[i][2]=1
        diq[schedule["Day_number"].iloc[i][3]=1
    elif float(schedule["Transhipment"].iloc[i]/schedule["Containers"].iloc[i]) >=0.65:
        diq[schedule["Day_number"].iloc[i][1]=lq[1]/2
        diq[schedule["Day_number"].iloc[i][2]=1
        diq[schedule["Day_number"].iloc[i][3]=1
    else:
        diq[schedule["Day_number"].iloc[i][1]=1
        diq[schedule["Day_number"].iloc[i][2]=1
diq[schedule["Day_number"][i]][3]=1

elif schedule["Berth_pref"][i] == 1:
    diq[schedule["Day_number"][i]][1]=schedule["Berth_Position"][i]
    diq[schedule["Day_number"][i]][2]=1
    diq[schedule["Day_number"][i]][3]=1

elif schedule["Berth_pref"][i] == 2:
    diq[schedule["Day_number"][i]][1]=1
    diq[schedule["Day_number"][i]][2]=schedule["Berth_Position"][i]
    diq[schedule["Day_number"][i]][3]=1

elif schedule["Berth_pref"][i] == 3:
    diq[schedule["Day_number"][i]][1]=1
    diq[schedule["Day_number"][i]][2]=1
    diq[schedule["Day_number"][i]][3]=schedule["Berth_Position"][i]

#Cost of berth delay
if schedule["Berth_pref"][i] == 0:
    if schedule["Length"][i] >= 150:
        ciw[schedule["Day_number"][i]]=104
    else:
        ciw[schedule["Day_number"][i]]=31
    else:
        if schedule["Length"][i] >= 150:
            ciw[schedule["Day_number"][i]]=104+500
        else:
            ciw[schedule["Day_number"][i]]=31+500

#Cost of delay of processing
if schedule["Berth_pref"][i] == 0:
    if (schedule["Length"][i] < 150) & (schedule["Transit"][i] == "Yes"):
        cid[schedule["Day_number"][i]]=31*2500
    elif (schedule["Beam"][i] < 24) & (schedule["Transit"][i] == "Yes") & (schedule["Length"][i] >= 150):
        cid[schedule["Day_number"][i]]=104 +5500
    elif (24<=schedule["Beam"][i] < 27) & (schedule["Transit"][i] == "Yes") & (schedule["Length"][i] >= 150):
        cid[schedule["Day_number"][i]]=104 +10500
    elif (27<=schedule["Beam"][i] < 31) & (schedule["Transit"][i] == "Yes") & (schedule["Length"][i] >= 150):
        cid[schedule["Day_number"][i]]=104 +18500
elif (schedule["Beam"][i] >= 31) & (schedule["Transit"][i] == "Yes"):
    cid[schedule["Day_number"][i]] = 104
elif (schedule["Length"][i] >= 150):
    cid[schedule["Day_number"][i]] = 104
else:
    cid[schedule["Day_number"][i]] = 31
else:
    if schedule["Transit"][i] == "Dummy":
        cid[schedule["Day_number"][i]] = 500000
    else:
        # Cost of moving away from position
        if schedule["Berth_pref"][i] == 0:
            cpi[schedule["Day_number"][i]] = (c[i+1] * 0.001)
        else:
            cpi[schedule["Day_number"][i]] = (c[i+1] * 1)

        name[schedule["Day_number"][i]] = schedule["Name"][i]

        # Cost of berthing in a berth. Rolling horizon enhancer
        if schedule["Berth_pref"][i] == 0:
            caiq[schedule["Day_number"][i]][1] = 1
            caiq[schedule["Day_number"][i]][2] = 50000
            caiq[schedule["Day_number"][i]][3] = 50000
        elif schedule["Berth_pref"][i] == 2:
            caiq[schedule["Day_number"][i]][1] = 50000
            caiq[schedule["Day_number"][i]][2] = 50000
            caiq[schedule["Day_number"][i]][3] = 1
        elif schedule["Berth_pref"][i] == 3:
            caiq[schedule["Day_number"][i]][1] = 50000
            caiq[schedule["Day_number"][i]][2] = 50000
            caiq[schedule["Day_number"][i]][3] = 1

        date = raw_input("Date of planning (Introduce as YYYY-MM-dd, including hyphens): ")
while (len(date) != 10) or ("-" not in date[4]) or ("-" not in date[7]) or ("2" not in date[0]):
```
    date=raw_input("Try again, remember to add hyphens and introduce date as YYYY-MM-dd: ")
else:
    while date not in list_date:
        date=raw_input("Date not in database. Try again: ")
else:
    input_BAP("schedule1",date)
    date_to_dt= datetime.strptime(date, "%Y-%m-%d")
    modified_date = date_to_dt + timedelta(days=1)
    modified_date2= date_to_dt + timedelta(days=2)
    modified_date3= date_to_dt + timedelta(days=3)
    date_for_rh=datetim
    date_for_rh2=datetime.strftime(modified_date2, "%m/%d/%Y")
    date_for_rh3=datetime.strftime(modified_date3, "%m/%d/%Y")

    #Time horizon
    H=max(15,max(si.values()))
    T=range(H+1)
    V=len(si.keys())
    vessels=range(1,V+1)

    #Groups of possible cranes attending vessel i
    #Dictionary of list of cranes available per vessel based on cranes reach and vessel beam
    comb_cranes=defaultdict(list)

    for i in vessels:
        for qc in total_cranes:
            if b[i]<=cr[qc]:
                comb_cranes[i].append(qc)

    # Ranges needed to have min and max attached to each vessel.
    ranges=defaultdict(dict)
    #Dictionary of list of tuples with combination vessel/possible crane combinations
    gcomb=defaultdict(dict)
    #List of all combinations given min max needed for every vessel irrespective of vessel
    qnamelist=[]
    for i in vessels:
        ranges[i]=range(qcmin[i],qcmax[i]+1)
```
for d in distr.keys():
    if set(distr[d]).issubset(set(comb_crane[i])):
        for s in ranges[i]:
            a=zip(*{distr[d][p:] for p in range(0,s)})
            qnamelist.append(a)
            gcomb[i].append(a)

#Conversion of qnamelist (List of list of tuples) to List of tuples
flat_list=[]
for sublist in qnamelist:
    for item in sublist:
        flat_list.append(item)

#Eliminate duplicates from list of tuples
dupl_list=list(set(flat_list))

#Dictionary with vessels as key and all combinations attached to them
g_combflat=defaultdict(list)
for i in vessels:
    for sublist in gcomb[i]:
        for item in sublist:
            g_combflat[i].append(item)

#Name of each individual group of combinations. 47 total
g_name=defaultdict(dict)
for i in range(len(dupl_list)):
    g_name[i+1]=dupl_list[i]

#Dictionary of combinations by vessel:[name of combination] as QC[i]=[name of valid combinations of quaycranes]
QC_1=defaultdict(list)
QC_reverse=defaultdict(list)
for i in g_combflat:
    for g in g_name:
        for x in g_combflat[i]:
            if x==g_name[g]:
                QC_1[i].append(g)
                QC_reverse[g].append(i)
# Number of cranes in group
n = defaultdict(dict)

# First crane in group
f = defaultdict(dict)

# Last crane in group
z = defaultdict(dict)

groups = []

g_name = 
for g in g_name:
    n[g] = len(g_name[g])
    f[g] = g_name[g][0]
    z[g] = g_name[g][-1]
    groups.append(g)

# Processing time of vessel i when processed by QC combining (1,47)
u = defaultdict(dict)
proc Comb = defaultdict(list)

for g, t in product(g_name, distr):
a = [x for x in g_name[g]]
for y in a:
    if y in distr[t]:
        proc_comb[g].append(proc[t])

proc_g = defaultdict(dict)

for g in proc_comb:
    proc_g[g] = sum(proc_comb[g])

for g, i in product(proc_g, vessels):
    if g in QC_1[i]:
        u[i][g] = math.ceil((c[i] * 1.00) / proc_g[g])

# Processing time of vessels based on containers/crane profiles
bi = defaultdict(dict)

for i in vessels:
    bi[i] = int(round(sum([u[i][x] for x in u[i]]) / float(len([u[i][x] for x in u[i]]))))

# Cranes per quay
listQq=defaultdict(list)
Qq=defaultdict(dict)
for q,d in product(quays,distr):
    if d in distr_quays[q]:
        listQq[q].append(len(distr[d]))
for q in quays:
    Qq[q]=sum(item for item in listQq[q])

#Vessels feasibility considering crane reach

cranes_quay=defaultdict(list)
for q in quays:
    for x in distr_quays[q]:
        cranes_quay[q].append(distr[x])

cranes_quay2=defaultdict(list)
for q in quays:
    cranes_quay2[q].append(list(itertools.chain.from_iterable(cranes_quay[q])))

vessel_cranes=defaultdict(list)
for i in vessels:
    for g in total_cranes:
        if b[i] <= cr[g]:
            vessel_cranes[i].append(g)

vessel_quays=defaultdict(list)
for i in vessels:
    for q in quays:
        for x in range(len(vessel_cranes[i])):
            if vessel_cranes[i][x] in cranes_quay2[q]:
                vessel_quays[i].append(q)

vessel_quays2=defaultdict(list)
for i in vessels:
    vessel_quays2[i].append(list(set(vessel_quays[i])))

#Specific combinations of cranes and vessels combinations in their specific cranes. QC[i][q]
cranes_quay3 = defaultdict(list)
for q, g in product(quays, g_name):
    for x in range(len(cranes_quay2[q][0])):
        if cranes_quay2[q][0][x] in g_name[g]:
            cranes_quay3[q].append(g)

cranes_quay4 = defaultdict(list)
for q in quays:
    cranes_quay4[q].append(list(set(cranes_quay3[q])))

QC = defaultdict(lambda: defaultdict(list))
for i in vessels:
    for q in cranes_quay4:
        for x in range(len(cranes_quay4[q][0])):
            if cranes_quay4[q][0][x] in QC_1[i]:
                QC[i][q].append(cranes_quay4[q][0][x])

#Lp problem
obj1 = LpProblem("BAP", LpMinimize)

#Decision variable
t = LpVariable.dicts("BerthTime", vessels, lowBound=0, cat=LpInteger)
p = LpVariable.dicts("BerthPos", vessels, lowBound=1)
h = LpVariable.dicts("Delay", vessels, lowBound=0)
e = LpVariable.dicts("Deviation", vessels, lowBound=0)

#Binary variables
m = LpVariable.dicts("m", (vessels, quays), cat=LpBinary)
sigma = LpVariable.dicts("sigma", (vessels, vessels), cat=LpBinary)
delta = LpVariable.dicts("delta", (vessels, vessels), cat=LpBinary)

#New variable
r = LpVariable.dicts("Vessel_handled", (vessels, groups, T), cat=LpBinary)

#Objective function
obj1 += lpSum([ciw[i]*(t[i] - ai[i]) + (cid[i]*h[i])
               for i in vessels])
+ lpSum([caiq[i][q]*m[i][q]
               for (i, q) in ves_quay ]+[cpi[i]*e[i]
               for i in vessels]), "Minimum cost"
# Constraints

# Overlapping in time and space avoidance

# Time

for i in vessels:
    for j in vessels:
        if i!=j:
            obj1 += t[j] - (t[i]+bi[i]) - (sigma[i][j]-1)*H >= 0, "Time_comb(%d,%d)" % (i, j)

# Space

for i in vessels:
    for j in vessels:
        if i!=j:
            obj1 += p[j] - (p[i]+li[i]) - (delta[i][j]-1)*lmax >= 0, "Space_comb(%d,%d)" % (i, j)

# Overlapping of time space

for i in vessels:
    for j in vessels:
        for q in quays:
            if q in vessel_quays2[i][0] and i!=j and q in vessel_quays2[j][0] :
                obj1 += sigma[i][j]+sigma[j][i]+delta[i][j]+delta[j][i]==m[i][q]+m[j][q]-1, "Vessel(%d,%d)" % (i, j)+"_"+"Quay(%d)" % q

# Valid quay

# Space Max

for i in vessels:
    obj1 += p[i]+li[i]<=lpSum([m[i][q]*lq[q] for q in quays if q in vessel_quays2[i][0]]), "Valid_quay(%d)" % i

# Service Time Max Optional if willinf to constraint time to planning horizon-

# for i in vessels:
#    obj1 += t[i]+bi[i]<=lpSum([m[i][q]*H for q in quays])

# Every vessel assigned to a quay

for i in vessels:
    obj1 += lpSum([m[i][q]*lq[q] for q in quays if q in vessel_quays2[i][0]])==1 , "Vessel_per_quay(%d,%d)" % (i, q)

# Avoid mooring vessel prior arrival

for i in vessels:
    obj1 += t[i]>=ai[i], "BerthTimeLowBound(%d)" % i
#Define delay of each vessel that cannot be negative
for i in vessels:
    obj1 += h[i] >= t[i] + bi[i] - si[i], "VesselDelay(%d)"%i

#Deviation from vessel from desired position at the assigned quay
for i in vessels:
    obj1 += e[i] >= p[i] - lpSum([diq[i][q]*m[i][q] for q in quays if q in vessel_quays2[i][0]]), "Deviationvesselfromidealquay(%d,%d)"%(i,q)

for i in vessels:
    obj1 += e[i] >= lpSum([diq[i][q]*m[i][q] for q in quays if q in vessel_quays2[i][0]]) - p[i], "2ndDeviationvesselfromidealquay(%d,%d)"%(i,q)

# LP file
# obj1.writeLP("BAPW.lp")
# Sol.
# obj1.solve(GUROBI())
# Status
# Values for BAQCASPS
results=[]
for v in obj1.variables():
    plp_exp=v.name,v.varValue
    results.append(plp_exp)

# Dataframe for BAQCASPS
df=pd.DataFrame.from_records(results,index=0)

quay_vessel=defaultdict(list)
for i in vessels:
    for q in vessel_quays2[i][0]:
        quay_vessel[q].append(i)
if df[1]["m_%s_%s"%(i,q)] == 1:
    quay_vessel[q].append(i)

######## GRAPH
horizonlist=defaultdict(list)
def graph(objective,list_vessels,quay):
    graph=[]
    for v in objective.variables():
        v.name,"=" v.varValue
        plp_exp=v.name,v.varValue
        graph.append(plp_exp)
    if quay == 0:
        for i in list_vessels:
            bi_exp=Processing_of_vessel_%d"%i, bi[i]
            graph.append(bi_exp)
            li_exp=Length_of_vessel_%d"%i, li[i]
            graph.append(li_exp)
            name_exp=Name_of_vessel_%d"%i, name[i]
            graph.append(name_exp)
    else:
        for i in list_vessels:
            li_exp=Length_of_vessel_%d"%i, li[i]
            graph.append(li_exp)
            name_exp=Name_of_vessel_%d"%i, name[i]
            graph.append(name_exp)
            beam_exp=Beam_%d"%i, b[i]
            graph.append(beam_exp)
            raw_liexp=Raw_length_%d"%i,raw_li[i]
            graph.append(raw_liexp)
            arrival_exp=Arrival_%d"%i,ai[i]
            graph.append(arrival_exp)
            departure_exp=Departure_%d"%i,si[i]
            graph.append(departure_exp)
    for g in QC[i][quay]:
        bi_exp=Processing_of_vessel_%d_%d"%(i,g), u[i][g]
graph.append(bi_exp)

# Dataframe for graphic
df=pd.DataFrame.from_records(graph,index=0)

# Table of values for graph
graph_plot=[]
if quay == 0:
    for i in list_vessels:
        plot="Vessel_%d"%i,next(q for q in vessel_quays2[i][0] if df[1]["m_%d_%d"%(i,q)]!=0),df[1]["BerthPos_%d"%i],df[1]["BerthTime_%d"%i],df[1]["Processing_of_vessel_%d"%i],df[1]["Length_of_vessel_%d"%i],df[1]["Name_of_vessel_%d"%i]
        graph_plot.append(plot)
else:
    for i in list_vessels:
        for g in QC[i][quay]:
            for tm in (range(ai[i],H+1)):
                if df[1]["Vessel_handled_%d_%d_%d"%(i,g,tm)] == 1:
                    plot="Vessel_%d"%i,df[1]["BerthPos_%d"%i],df[1]["BerthTime_%d"%i],df[1]["Length_of_vessel_%d"%i],df[1]["Name_of_vessel_%d"%i],df[1]["Vessel_Name"],df[1]["Arrival_%d"%i],df[1]["Departure_%d"%i]
                    graph_plot.append(plot)

df2=pd.DataFrame.from_records(graph_plot)
if quay == 0:
df2.columns=["Vessel","Assigned_Berth","Berth_Position","Berth_Time","Processing_Time","Length","Vessel_Name"]
else:
df2.columns=["Vessel","Berth_Position","Berth_Time","Length","Processing_Time","Crane_Group","Cranes_in_use","Vessel_Name","Arrival","Departure"]
df3=df2.set_index("Vessel")

# if quay != 0:
df3.to_csv("Schedule_quay_%d.csv"%quay)

# Table for planning horizon
horizon=[]

if quay!=0:
    for i in list_vessels:
        for g in QC[i][quay]:
            for tm in (range(ai[i],H+1)):
                if df[1]["Vessel_handled_%d_%d_%d"%(i,g,tm)] == 1:
                    hplot="Vessel_%d",df[1]["BerthPos_%d_%d"]%i,g,tm],g,g_name[g],df[1][
                        "Raw_length_%d_%d",len(g_name[g]),len(g_name[g]),quay
                    horizon.append(hplot)

    df4=pd.DataFrame.from_records(horizon)
    if quay!=0:
        df4.columns=
            ["Vessel","Berth_Position","Berth_Time","Length","Processing_Time","Crane_Group","Cranes_in_use","Vessel_Name","Beam","Min_crane","Max_crane","Berth_pref"]

if quay!=0:
    for i in range(int(df4["Vessel_Name"].count())):
        if df[1]["Berth_Time"].iloc[i]+df[1]["Processing_Time"].iloc[i] >24:
            atd=df[1]["Berth_Time"].iloc[i]+df[1]["Processing_Time"].iloc[i]-24
            containers_remaining=df[1]["Min_crane"].iloc[i]*int(proc_comb["Crane_Group"][0])*atd
            list_accepted=df[1]["Vessel_Name"].iloc[i],date_for_rh,"00:00:00",date_for_rh,"%d:00:00"%atd,df[1][
                "Length"].iloc[i],df[1]["Beam"].iloc[i],df4["Min_crane"],df[1][
                "Max_crane"].iloc[i],containers_remaining,"Dummy",df4["Berth_pref"].iloc[i],0,0,df[1][
                "Berth_Position"].iloc[i]
            horizonlist[quay].append(list_accepted)

# Graph plot
def graphplot(figure):
    if quay == 0:
```python
figure = plt.figure()

# Create dic of ax
ax = defaultdict(dict)
for q in quays:
    ax[q] = figure.add_subplot("22%d" % q)

# Edit quays graphs
for q in quays:
    ax[q].set_xlim([0, H])
    ax[q].set_ylim([0, lq[q]])
    ax[q].set_title("Quay %s", fontsize=10)
    ax[q].set_xlabel("Time", fontsize=6)
    ax[q].xaxis.set_label_coords(0.50, -0.025)
    ax[q].set_ylabel("Length", fontsize=6)
figure.suptitle("Berth Allocation Problem Manzanillo International Terminal")

def plot(Assigned_Berth, Berth_Time, Berth_Position, Processing_Time, Length):
    return ax[Assigned_Berth].add_patch(patches.Rectangle((Berth_Time, Berth_Position), Processing_Time, Length, fill=False))

# Vessel plot in quays
for i in list_vessels:
    plot(df3["Assigned_Berth"], df3["Vessel_%s"] % i, df3["Berth_Time"], df3["Vessel_%s"] % i, df3["Berth_Position"], df3["Vessel_%s"] % i, df3["Processing_Time"], df3["Vessel_%s"] % i, df3["Length"])

# Create dic of vessels
vx = defaultdict(dict)
for i in list_vessels:
    vx[i] = patches.Rectangle((df3["Berth_Time"]), df3["Vessel_%s"] % i, df3["Berth_Position"]), df3["Vessel_%s"] % i, df3["Processing_Time"]), df3["Vessel_%s"] % i, df3["Length"])

# Number label in vessel
for key, value in ax.items():
    for i in list_vessels:
        if df3["Assigned_Berth"] == key:
            ax[key].add_artist(vx[i])
            rx, ry = vx[i].get_xy()
```
cx = rx + vx[i].get_width()/8.0
cy = ry + vx[i].get_height()/1.5

ax[key].annotate(i, (cx, cy), color='black', weight='bold', fontsize=8, ha='center', va='center')

figure.savefig('quay0.png', dpi=90, bbox_inches='tight')
plt.show()

else:
    figure = plt.figure(figsize=(5,3.5))
ax = figure.add_subplot(111)

    #Edit quay graph
    ax.set_xlim([0,H])
    ax.set_ylim([0,lq[quay]])
    ax.set_title("Quay %s"%str(quay),fontsize=10)
    ax.set_xlabel("Time", fontsize=6)
    ax.xaxis.set_label_coords(0.50,-0.025)
    ax.set_ylabel("Length", fontsize=6)
    figure.suptitle("MIT Quay_%d"%quay)

    def plot(Berth_Time,Berth_Position,Processing_Time,Length):
        return ax.add_patch(patches.Rectangle((Berth_Time,Berth_Position),Processing_Time,Length,fill=False))

    #Vessel plot in quays
    for i in list_vessels:
        plot(df3["Berth_Time"]["Vessel_%s"%i],df3["Berth_Position"]["Vessel_%s"%i],df3["Processing_Time"]["Vessel_%s"%i],df3["Length"]["Vessel_%s"%i])

    #Create dic of vessels
    vx=defaultdict(dict)
    for i in list_vessels:
        vx[i]=patches.Rectangle((df3["Berth_Time"]["Vessel_%s"%i],df3["Berth_Position"]["Vessel_%s"%i]),
                                 df3["Processing_Time"]["Vessel_%s"%i],df3["Length"]["Vessel_%s"%i],fill=False)

    #Number label in vessel
    for i in list_vessels:
```python
ax.add_artist(vx[i])
rx, ry = vx[i].get_xy()

cx = rx + vx[i].get_width()/2

cy = ry + vx[i].get_height()/1.3
cy2 = ry + vx[i].get_height()/2.8
cy3 = ry + vx[i].get_height()/5

ax.annotate(i, (cx, cy), color='black', weight='bold', fontsize=7, ha='center', va='center')

ax.annotate(df3['Cranes_in_use']["Vessel_%d"]%i, (cx, cy2), color='black', weight='bold', fontsize=8, ha='center', va='center')

ax.annotate(df3['Vessel_Name']["Vessel_%d"]%i, (cx, cy3), color='black', weight='bold', fontsize=7, ha='center', va='center')

figure.savefig('quay%d.png'%quay, dpi=90, bbox_inches='tight')

plt.show()

if quay == 0:
    graphplot("fig1")
else:
    graphplot("fig%s"% str(quay+1))

### Function for optimizing BAP into BAQCASPS

def BAQCASP(objective, quay):
    ships=quay_vessel[quay]
    #Lpproblem
    objective=LpProblem("BAP",LpMinimize)
    #Function for independently introduce BAQCASPS for each quay
    def BAQCASP(objective, quay):
        ships=quay_vessel[quay]
        #Lpproblem
        objective=LpProblem("BAP",LpMinimize)
        #Objective function
        objective +=lpSum([ciw[i]*(t[i]-ai[i])+(cid[i]*h[i])+(cpi[i]*e[i]) for i in ships]), "Minimum cost"
```

95
#Constraints

#Handling of vessel i with group g in time t
for i in ships:
    objective += lpSum([r[i][g][tm] for g, tm in product(QC[i][quay], range(ai[i], H+1))]) == 1, "r_%d%i"

#Processing time equal to
for i in ships:
    objective += t[i] == lpSum([r[i][g][tm] * tm for g, tm in product(QC[i][quay], range(ai[i], H+1))]), "t_%d=r_%d_%d_%d"%(i, i, g, tm)

#Avoid overlapping time on vessels
for i in ships:
    for j in ships:
        if i != j:
            objective += t[j] - t[i] - (lpSum([r[i][g][tm] * u[i][g] for g, tm in product(QC[i][quay], range(ai[i], H+1))]) - (sigma[i][j] - 1)*H) >= 0, "Time_comb_%d_%d"%(i, j)

#Avoid overlapping space between vessels
for i in ships:
    for j in ships:
        if i != j:
            objective += p[j] - (p[i] + li[i]) - (delta[i][j] - 1)*lq[quay] >= 0, "Space_comb_%d_%d"%(i, j)

#Avoid overlapping time_space
for i in ships:
    for j in ships:
        if i != j:
            objective += sigma[i][j] + sigma[j][i] + delta[i][j] + delta[j][i] >= 1, "Sig_Delta_%d_%d"%(i, j)

for i in ships:
    objective += p[i] + li[i] <= lq[quay] + 1, "Valid_space_vessel_%d%i"

###Solution 1

# Cranes utilized in any time period couldnt be more tha maximum available in any period
r_dict=defaultdict(list)
g_ro=defaultdict(list)
for tm, i in product(T, ships):
    if objective[tm][i] > 0:
        r_dict[tm].append(i)
        g_ro[i].append(tm)
[r_dict[ro].append((i,g,tm)) for g in QC[i][quay] for ro in range(max(ai[i],(int(math.ceil(tm-u[i][g]+1)))),tm+1)]

g_ro[ro,ro].append(i) for g in QC[i][quay] for ro in range(max(ai[i],(int(math.ceil(tm-u[i][g]+1)))),tm+1)]

r_dict2=defaultdict(list)
g_ro2=defaultdict(list)

for i in r_dict:
r_dict2[i].append(list(set(r_dict[i])))

for i in g_ro:
g_ro2[i].append(list(set(g_ro[i])))

#Cranes always less or equal than maximum allowed
riqro2=defaultdict(list)

for tm in T:
    for i in ships:
        for x in range(len(QC[i][3])):
            a=([(i,QC[i][3][x],ro[y]) for ro in range(max(ai[i],int(tm-u[i][QC[i][3][x]][x][x])),tm+1)] if tm>=ai[i]
            else [None]) for y in range(len(ro))

        for x in sublist in riqro2[tm]:
            pr[tm].append(item)

        for tm in T:
            objective += lpSum([r[g_ro2[ro][0]][g][ro] for x in range(len(g_ro2[g,ro][0]))]) <= Qq[quay], "ro_%d"%tm

#One group per vessel per period
for g,ro in product(g_name,r_dict2):
    if g_ro2[g,ro]!=[]:
        objective += lpSum([r[g_ro2[g,ro][0]][g][ro] for x in range(len(g_ro2[g,ro][0]))]) <= 1, "r_n_%d"%tm

#Numbers of cranes constraint
for i in ships:
    for j in ships:
if i!=j:
    objective += (lpSum([f[g]*r[j][g][tm] for g,tm in product(QC[j][quay],range(ai[j],H+1))])-(
    lpSum([z[g]*r[i][g][tm] for g,tm in product(QC[i][quay],range(ai[i],H+1))])>=1-Qq[quay]-(
    delta[j][i]+sigma[i][j]+sigma[j][i]),"Crane_number_assigned_vessel_%d_%d"%(i,j))

#Define delay of vessel and
for i in ships:
    objective += h[i]>=t[i]-si[i] + (lpSum([(r[i][g][tm]*u[i][g]) for g,tm in product(QC[i][quay],range(ai[i],H+1))])-1,"Delay_%d"%i

#Define deviation from desired position
for i in ships:
    objective += e[i]>=p[i]-diq[i][q], "Deviationvesselfromidealquay_%d"%i

for i in ships:
    objective += e[i]>=diq[i][q]-p[i], "2ndDeviationvesselfromidealquay_%d"%i

# Sol.
objective.solve(GUROBI())

# #LP file
# objective.writeLP("BAQCASP\_S\_quay%d.lp"%quay)

#Status
status=LpStatus[objective.status]
status_result[quay]=status
costs[quay]=value(objective.objective)
graph(objective,ships,quay)

#Graph for BAP
graph(obj1,vessels,0)

min_bap=value(obj1.objective)

#Graph + solutions for individual quays with BAQCASP
listquays=[]
for q in quays:
    for i in vessels:
        if df[1]["m_%s_%s"]%(i,q) == 1:
            listquays.append(q)
quays_accepted=list(set(listquays))
for q in quays_accepted:
    BAQCASP("obj%s"%str(q+1),q)
fil2=open("Merged_df.csv")
calls2=pd.read_csv(fil2)
fil2.close()
calls["ATA_Day"].dt.strftime("%m/%d/%Y")
calls["ATD_Day"].dt.strftime("%m/%d/%Y")
calls["ATA_Time"].dt.strftime("%H:%M:%S")
calls["ATD_Time"].dt.strftime("%H:%M:%S")
dflist=[]
for x in horizonlist.keys():
    for y in range(len(horizonlist[x])):
        dflist.append(horizonlist[x][y])
df5=pd.DataFrame.from_records(dflist, columns=["Name","ATA_Day","ATA_Time","ATD_Day","ATD_Time","Length","Beam", "Min_cranes","Max_cranes","Containers","Transit","Berth_pref","Reefers","Dangerous_cargo","Berth_Position"])
df6=pd.concat([calls2,df5],ignore_index=True,axis=0)
df6.to_csv("Merged_df2.csv")
#Dialogue box with results
root=Toplevel()
root.wm_title("Results for Manzanillo International Terminal scheduling")
#Width and Height of root
w=700
h = 300

# get screen width and height
ws = root.winfo_screenwidth()  # width of the screen
hs = root.winfo_screenheight()  # height of the screen

# calculate x and y coordinates for the Tk root window
x = (ws/4) - (w/2)
y = (hs/5) - (h/2)

# set the dimensions of the screen
# and where it is placed
root.geometry('%dx%d+%d+%d' % (w, h, x, y))

text1 = Text(root, height=30, width=47)
photo = PhotoImage(file="Manzanillo_port.gif")
text1.image_create(END, image=photo)
text1.pack(side=LEFT)

text2 = Text(root, height=30, width=40)
text2.tag_configure("bold", font=("Arial", 14, "bold"))
text2.tag_configure("text", foreground="#091221", font=('Arial', 12))
text2.insert(END, '
MIT Results for BAQCASP

for q in quays_accepted:
    text2.insert(END, ' BERTH %d: %s. Cost %d

%(q, status_result[q], costs[q]), "text")
text2.insert(END, 'Minimum cost Multistep MILP: %d

% sum(costs.values()), "text")
text2.pack(side=LEFT)
root.mainloop()