

A branch-and-price algorithm to solve the integrated berth allocation and yard assignment problem in bulk ports

Tomáš Robenek * Nitish Umang * Michel Bierlaire * Stefan Røpke †

October 31, 2012

Report TRANSP-OR 120617
Transport and Mobility Laboratory
École Polytechnique Fédérale de Lausanne
transp-or.epfl.ch

*Transport and Mobility Laboratory (TRANSP-OR), School of Architecture, Civil and Environmental Engineering (ENAC), École Polytechnique Fédérale de Lausanne (EPFL), CH-1015 Lausanne, Switzerland, {tomas.robeneck, nitish.umang, michel.bierlaire}@epfl.ch

†Department of Transport, Technical University of Denmark, Bygningstorvet 116 Vest, DK-2800 Kgs. Lyngby, Denmark, {sr}@transport.dtu.dk

Abstract

In this research, two crucial optimization problems of berth allocation and yard assignment in the context of bulk ports are studied. We discuss how these problems are interrelated and can be combined and solved as a single large scale optimization problem. More importantly we emphasize the differences in operations between bulk ports and container terminals which emphasizes the need to devise specific solutions for bulk ports. The objective is to minimize the total service time of vessels berthing at the port. We propose an exact solution algorithm based on branch and price framework to solve the integrated problem. In the proposed model, the master problem is formulated as a set-partitioning problem, and subproblems to identify columns with negative reduced costs are solved using mixed integer programming. The proposed algorithm is tested and validated through numerical experiments based on instances inspired from real bulk port data. The results indicate that the algorithm can be successfully used to solve instances containing up to 40 vessels within reasonable computational time.

1 Introduction

Maritime transportation is a major channel of international trade. In the last decade, the shipping tonnage for dry bulk and liquid bulk cargo has risen by 52% and 48% respectively. The total volume of dry bulk cargoes loaded in 2008 stood at 5.4 billion tons, accounting for 66.3 per cent of total world goods loaded UNCTAD (2009). The proper planning and management of port operations in view of the ever growing demand represents a big challenge. A bulk port terminal is a zone of the port where sea-freight docks on a berth and is stored in a buffer area called yard for loading, unloading or transshipment of cargo. In general, the bulk terminal managers are faced with the challenge of maximizing efficiency both along the quay side and the yard. From the past research, it is well established that operations research methods and techniques can be successfully used to optimize port operations and enhance terminal efficiency. However while significant contributions have been made in the field of large scale optimization for container terminals, relatively little attention has been directed to bulk port operations.

Bulk terminal operations planning can be divided into two decision levels depending on the time frame of decisions: Tactical Level and Operational Level. Tactical level decisions involve medium to short term decisions regarding resource allocation such as port equipment and labor, berth and yard management, storage policies etc. In practice, these decisions could be based on "rules of thumb" in which the experience of the port managers plays an important role, or alternatively more scientific approaches based on operations research methods could be in use. The operational level involves making daily and real time decisions such as crane scheduling, yard equipment deployment and last minute changes in response to disruptions in the existing schedule. This paper focuses on the tactical level decision planning for the integrated berth and yard management in the context of bulk ports. We focus in particular on two crucial optimization problems in the context of bulk port terminals: The Berth Allocation Problem (BAP) and the Yard Assignment Problem.

The tactical berth allocation problem refers to the problem of assigning a set of vessels to a given berthing layout within a given time horizon. There could be several objectives such as the minimization of the service times of vessels, the minimization of the port stay time, the minimization of the number of rejected vessels, the minimization of the deviation between actual and planned berthing schedules etc. There are several spatial and temporal constraints involved in the BAP, which lead to a multitude of BAP formulations. The temporal attributes include the vessel arrival process, the start of service, the handling times of vessels, while the spatial attributes relate to the berth layout, the draft restrictions and others. In a container terminal, all cargo is packed into containers, and thus there is no need for any specialized equipment to handle any particular type of cargo. In contrast in bulk ports, depending on the vessel requirements and cargo properties, a wide variety of equipments are used for discharging or loading operations. Thus, the cargo type on the vessel needs to be explicitly taken into consideration while modeling the berth allocation problem in bulk ports. The tactical yard assignment problem refers to decisions that concern the storage location and the routing of materials. This affects the travel distance between the assigned berth to the vessel and storage location of the cargo type of the vessel on the yard, and furthermore determines the storage efficiency of the yard. Thus, the problems of berth allocation and yard management are interrelated. The start times and end times of operations of vessels determine the workload distribution and the deployment of yard equipment such

as loading shovels and wheel loaders in the yardside. Moreover, berthing locations of vessels determine the storage locations of specific cargo types to specific yard locations, which minimize the total travel distance between the assigned berthing positions to the vessels and the yard locations storing the cargo type for the vessel. Similarly, the yard assignment of specific cargo types has an impact on the best berthing assignment for vessels berthing at the port. In this study, we present an integrated model for the dynamic, integrated berth allocation problem and yard assignment in context of bulk ports. Few scholars have investigated this problem in context of container terminals, and there is no published literature for bulk ports. We present an exact solution algorithm based on branch-and-price to solve the combined large scale problem. Numerical experiments based on real-life-inspired port data indicate that the proposed algorithm can be successfully used to solve even large sized instances.

2 Literature Review

From the past OR literature on container terminal operations, it is well established that integrated planning of operations can allow port terminals to reduce congestion, lower delay costs and enhance efficiency. Significant contributions have been made in the field of large scale optimization and integrated planning of operations in container terminals. Bulk ports on the other hand have received almost no attention in the operations research literature. The integrated berth allocation and quay crane assignment or scheduling problem has been studied in the past by Park and Kim (2003), Meisel and Bierwirth (2006), Imai, Chen, Nishimura and Papadimitriou (2008), Meisel and Bierwirth (2008), and more recently by Giallombardo et al. (2010) and Vacca (2011) for container terminals. Comprehensive literature surveys on container terminal operations can be found in Steenken et al. (2004), Stahlbock and Voss (2008), Bierwirth and Meisel (2010).

The dynamic, hybrid berth allocation problem in context of bulk ports is studied by Umang et al. (2012). The berth allocation problem in container terminals has been widely studied in the past. Imai et al. (1997), Imai et al. (2001), Imai, Nishimura and Papadimitriou (2008), Imai et al. (2003), Monaco and Sammarra (2007), Buhrkal et al. (2011), Zhou and Kang (2008), Han et al. (2010), Cordeau et al. (2005), Mauri et al. (2008) propose methods to solve the discrete berth allocation problem. The continuous berth allocation problem is studied by Li et al. (1998), Guan et al. (2002), Park and Kim (2003), Guan and Cheung (2004), Park and Kim (2002), Kim and Moon (2003), Lim (1998), Tong et al. (1999), Imai et al. (2005) and Chang et al. (2008). The berth allocation problem with hybrid layout is addressed by Moorthy and Teo (2006), Dai et al. (2008), Nishimura et al. (2001) and Cheong et al. (2010), and position-dependent handling times are considered by Cordeau et al. (2005) and Imai et al. (2007) for indented berths.

Yard management in container terminals involves several tactical and operational level decision problems. Scheduling and deployment of yard cranes is addressed by Cheung et al. (2002), Zhang et al. (2002), Ng and Mak (2005), Ng (2005) and Jung and Kim (2006). Storage and space allocation, stacking and re-marshalling strategies have been studied by Kim and Kim (1999), Kim et al. (2003), Lee et al. (2006) and few others. Nishimura et al. (2009) investigate the storage plan for transshipment hubs, and propose an optimization model to minimize the sum of the waiting time of feeders and the handling times for transshipment containers flow. Transfer operations that consist of routing and

scheduling of internal trucks, straddle carriers and AGV's have been studied by Liu et al. (2004), Vis et al. (2005), and Cheng et al. (2005) among others. Works on integrated problems related to yard management in container terminals include Bish et al. (2001) and Kozan and Preston (2006) who propose the integration of yard allocation and container transfers, whereas Chen et al. (2007) and Lau and Zhao (2007) study the integrated scheduling of handling equipment in a container terminal. In the following, we discuss in more detail some articles relevant to our study.

Moorthy and Teo (2006) discuss the concepts of berth template and yard template in the context of transshipment hubs in container shipping. They study the delicate trade-off between the level of service as indicated by the vessel waiting times and the operational cost for moving containers between the yard and quay in a container terminal. A robust berth allocation plan is developed using the sequence pair approach, with the objective to minimize the total expected delays and connectivity cost that is related to the distance between the berthing positions of vessels belonging to the same transshipment group.

Cordeau et al. (2007) study the Service Allocation Problem (SAP), a tactical problem arising in the yard management of Gioia Tauro Terminal. The SAP is a yard management problem that deals with dedicating specific areas of the yard and the quay to the services or route plans of shipping companies which are planned in order to match the demand for freight transportation. The objective of the SAP is the minimization of container rehandling operations in the yard and it is formulated as a Generalized Quadratic Assignment Problem (GQAP, see e.g. Cordeau et al. (2006), and Hahn et al. (2008)). An evolutionary heuristic is developed to solve larger instances obtained from the real port data.

More recently, Zhen et al. (2011) propose a mixed integer model to simultaneously solve the tactical berth template and yard template planning in transshipment hubs. The objective is to minimize the sum of service cost derived from the violation of the vessels expected turnaround time intervals and the operation cost related to the route length of transshipment container flows in the yard. A heuristic algorithm is developed to solve large scale instances within reasonable time and numerical experiments are conducted on instances from real world data to validate the efficiency of the proposed algorithm.

To the best of our knowledge, operations research problems have received almost no attention thus far in the context of bulk port terminals. In the context of container terminals, the major focus in the field of large scale optimization has been on studying the integrated berth allocation and quay crane scheduling or assignment problem, while very few scholars have attempted to solve the combined problem of berth allocation and yard assignment as a single large scale optimization problem. This is also the first paper to present an exact solution algorithm (based on branch-and-price) to solve the integrated problem in context of seaside port operations planning. Moreover we discuss and emphasize the specific features in bulk port operations that necessitate the need to devise specific solutions for bulk terminals.

3 Problem Statement

In this section we elaborate on the background for the integrated berth allocation and yard assignment problem in the context of bulk ports. A schematic representation of a bulk port terminal is shown in Figure 1. We consider a set of vessels N , to be berthed on a continuous quay of length L over a time horizon H . We consider dynamic vessel arrivals and a hybrid berth layout in which the quay boundary is discretized into a set M of sections of variable lengths. In a feasible berthing assignment, a given vessel may occupy more than one section, however a given section cannot be occupied by more than one vessel or part of a vessel at any given time. The dynamic, hybrid berth allocation problem in bulk ports is studied by Umang et al. (2012), in which two alternative exact solution methods and a heuristic approach are proposed to solve the problem. In the present work, we extend the berth allocation problem to account for the assignment of different yard locations to specific cargo types and vessels berthing at the port. Thus unlike the berth allocation model in which the problem was solved for a given yard layout and the unit handling times for given sections along the quay and cargo types were provided as input parameters to the model, in the integrated framework the assignment of cargo locations to specific cargo types and vessels are also decision variables.

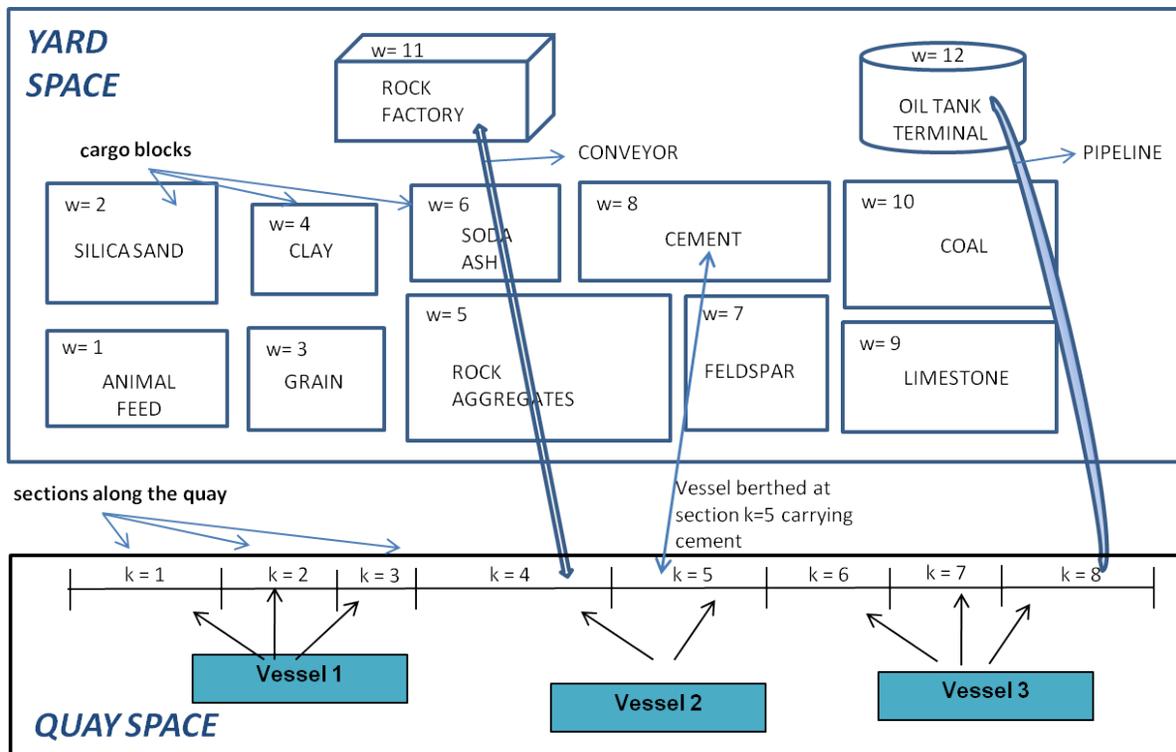


Figure 1: Schematic representation of a bulk port terminal

A major difference between bulk port and container terminal operations is the need to explicitly account for the cargo type on the vessel in bulk ports. Depending on the vessel requirements and cargo types, a wide variety of specialized equipment such as conveyors and pipelines are used for discharging or loading operations. For example, liquid bulk is generally discharged using pipelines which

are installed at only certain sections along the quay. Similarly, a vessel may require the conveyor facility to load cargo from a nearby factory outlet to the vessel. In contrast in a container terminal, all cargo is packed into containers, and thus there is no need for any specialized equipment to handle any particular type of cargo. Furthermore in bulk ports, depending on the cargo properties, there may be additional restrictions on the storage of specific cargo types in the yard which forbids two or more cargo types to be stored in adjacent yard locations to avoid intermixing.

In our model the main assumption in the computation of handling times is that all sections occupied by the berthed vessel are being operated on simultaneously. The amount of cargo handled at each section is assumed proportional to the section length. The handling time of the vessel is the time taken to load or discharge the section whose operation finishes last. The unit processing or handling time of a given vessel has a fixed component dependent on the number of quay cranes operating on the vessel, and a variable component which is dependent on the distance between the section occupied by the vessel along the quay and the storage location of the cargo type of the vessel on the yard. In the integrated model, we assume that each vessel has a single cargo type but that can be discharged (loaded) and transferred to (from) multiple yard locations. Thus, the distance used to calculate this variable component of handling time is the weighted average of the distance between the vessel and the assigned yard locations, where the weights are equal to the cargo quantities that are transferred to (from) each yard location from (to) the vessel. Another assumption in the model is that in the given planning horizon, a given yard location is either assigned to a single cargo type, or alternatively the yard location is not assigned to any cargo type.

Based on the preceding discussion, the unit handling time h_{ik}^w for vessel i with cargo type w occupying section k along the quay includes the time taken to transfer the unit quantity of cargo between the cargo location on the yard and section k , and the time taken to load (or unload) the cargo from the quay side to the vessel. These can be denoted by β_{ik}^w and α_{ik}^w respectively. Thus we have, $h_{ik}^w = \alpha_{ik}^w + \beta_{ik}^w$, where $\alpha_{ik}^w = T/n_{ik}^w$ and $\beta_{ik}^w = V_w r_k^i$. Here T is the crane handling rate for loading or discharging operations, and n_{ik}^w is the number of cranes operating in section k on vessel i for cargo type w . β_{ik}^w is the time taken to transfer a unit quantity of cargo between the cargo location w on the yard and the section k for vessel i , which is assumed to be a linear function of the weighted average distance r_k^i between the section k and all cargo locations assigned to the vessel i . The parameter V^w depends on the rate of transfer of cargo type w . Thus for example, if a vessel is using the conveyor facility to load rock aggregates from the rock factory directly into the vessel, the parameter V_w is equal to the cargo transfer rate for the conveyor facility, and if there are no additional cranes operating on the vessel, the parameter α_{ik}^w which is provided as an input parameter to the model is equal to zero. In practice, the fixed specialized equipment facilities such as conveyors and pipelines are dedicated to handling certain cargo types. For example, liquid bulk is transferred using pipelines, and rock aggregates are transferred using conveyor facility. Thus the specialized facilities are themselves modeled as cargo types in the proposed model. The objective of the integrated optimization model that we solve is to minimize the sum of the service times of all vessels, which includes the handling or processing times and the berthing delays for all vessels berthing at the port.

4 Model Formulation

In this section, we present a mixed integer programming formulation for the integrated berth allocation and yard assignment problem in bulk ports.

4.1 Notation

Input parameters In the formulation of the integrated model, the following input data is assumed available:-

$N =$	set of vessels
$M =$	set of sections
$P =$	set of cargo locations
$W =$	set of cargo types
$H =$	set of time steps
$W_i =$	cargo type to be loaded or discharged from vessel i
$\bar{P}(p) =$	set of cargo locations neighbouring cargo location p
$\bar{W}(w) =$	set of cargo types that cannot be stored adjacent to cargo type w
$A_i =$	expected arrival time of vessel i
$D_i =$	draft of vessel i
$L_i =$	length of vessel i
$Q_i =$	quantity of cargo for vessel i
$d_k =$	draft of section k
$\ell_k =$	length of section k
$b_k =$	starting coordinate of section k
$\alpha_{ik}^w =$	deterministic component of handling time for cargo type w of vessel i berthed at section k
$V_w =$	constant dependent on the rate of transfer of cargo type w
$r_k^p =$	distance between cargo location p and section k
$R_w =$	maximum amount of cargo type w that can be handled in a single time step
$L =$	total length of quay
$B =$	large positive constant
$F =$	maximum number of cargo locations that can be assigned to a single vessel
$\rho_{i\ell k} =$	fraction of cargo handled at section k when section ℓ is the first section occupied by vessel i
$\delta_{i\ell k} =$	$\begin{cases} 1 & \text{if vessel } i \text{ starting at section } \ell \text{ touches section } k \\ 0 & \text{otherwise.} \end{cases}$

Decision Variables The following decision variables are used in the model:-

m_i	integer ≥ 0 , represents the starting time of handling of vessel $i \in N$;
c_i	integer ≥ 0 , represents the total handling time of vessel $i \in N$;
h_{ik}^w	handling time for unit quantity of cargo type w for vessel i berthed at section k ;
β_{ik}^w	variable component of handling time of vessel i with cargo type w berthed at section k along the quay;
q_{ip}	amount of cargo handled by vessel i at cargo location p ;
η_i	number of cargo locations assigned to vessel i ;
r_k^i	weighted average distance between section k occupied by vessel i and all cargo locations assigned to the vessel;
s_k^i	binary, equals 1 if section $k \in M$ is the starting section of vessel $i \in N$, 0 otherwise;
x_{ik}	binary, equals 1 if vessel $i \in N$ occupies section $k \in M$, 0 otherwise;
y_{ij}	binary, equals 1 if vessel $i \in N$ is berthed to the left of vessel $j \in M$ without any overlapping in space, 0 otherwise;
z_{ij}	binary, equals 1 if handling of vessel $i \in N$ finishes before the start of handling of vessel $j \in N$, 0 otherwise;
μ_w^p	binary, equals 1 if cargo type w is stored at cargo location p , 0 otherwise;
ϕ_{ip}	binary, equals 1 if cargo location p is assigned to vessel i , 0 otherwise;
θ_{it}	binary, equals 1 if vessel i is being handled at time t , 0 otherwise;
ω_t^{ip}	binary, equals 1 if vessel i is being handled at location p at time t , 0 otherwise;

4.2 Mathematical Model

The integrated model for the berth allocation and yard assignment problem in bulk ports can be formulated as follows:

$$\min \sum_{i \in N} (m_i - A_i + c_i) \quad (1)$$

$$\text{s.t. } m_i - A_i \geq 0 \quad \forall i \in N \quad (2)$$

$$\sum_{k \in M} (s_k^j b_k) + B(1 - y_{ij}) \geq \sum_{k \in M} (s_k^i b_k) + L_i \quad \forall i, j \in N, i \neq j \quad (3)$$

$$m_j + B(1 - z_{ij}) \geq m_i + c_i \quad \forall i \in N, \forall j \in N, i \neq j \quad (4)$$

$$y_{ij} + y_{ji} + z_{ij} + z_{ji} \geq 1 \quad \forall i \in N, \forall j \in N, i \neq j \quad (5)$$

$$\sum_{k \in M} s_k^i = 1 \quad \forall i \in N \quad (6)$$

$$\sum_{k \in M} (s_k^i b_k) + L_i \leq L \quad \forall i \in N \quad (7)$$

$$\begin{aligned} \sum_{\ell \in M} (\delta_{i\ell k} s_{\ell}^i) &= x_{ik} && \forall i \in N, \forall k \in M && (8) \\ (d_k - D_i)x_{ik} &\geq 0 && \forall i \in N, \forall k \in M && (9) \\ c_i \geq h_{ik}^w \rho_{ik} Q_i - B(1 - s_{\ell}^i) &&& \forall i \in N, \forall \ell \in M, \forall k \in M, \forall w \in W_i && (10) \\ h_{ik}^w &= \alpha_{ik}^w + \beta_{ik}^w && \forall w \in W_i, \forall k \in M && (11) \\ \beta_{ik}^w &= V_w r_k^i && \forall i \in N, \forall w \in W_i, \forall k \in M && (12) \\ r_k^i &= \sum_{p \in P} (r_k^p q_{ip}) / Q_i && \forall i \in N, \forall k \in M && (13) \\ Q_i &= \sum_{p \in P} q_{ip} && \forall i \in N && (14) \\ q_{ip} &\leq \phi_{ip} Q_i && \forall i \in N, \forall p \in P && (15) \\ \phi_{ip} &\leq q_{ip} && \forall i \in N, \forall p \in P && (16) \\ q_{ip} &\leq \sum_{w \in W_i} \sum_{t \in H} (R_w \omega_t^{ip} + B(1 - \mu_w^p)) && \forall i \in N, \forall p \in P && (17) \\ \sum_{p \in P} \phi_{ip} &\leq F && \forall i \in N && (18) \\ \mu_w^p + \mu_{\bar{w}}^{\bar{p}} &\leq 1 && \forall w \in W, \forall \bar{w} \in \bar{W}(w), \forall p \in P, \forall \bar{p} \in \bar{P}(p) && (19) \\ \sum_{i \in N} \omega_t^{ip} &\leq 1 && \forall p \in P, \forall t \in H && (20) \\ \sum_{w \in W} \mu_w^p &\leq 1 && \forall p \in P && (21) \\ \phi_{ip} &\leq \mu_w^p && \forall i \in N, \forall w \in W_i, \forall p \in P && (22) \\ \omega_t^{ip} &\geq \phi_{ip} + \theta_{it} - 1 && \forall i \in N, \forall p \in P, \forall t \in H && (23) \\ \omega_t^{ip} &\leq \phi_{ip} && \forall i \in N, \forall p \in P, \forall t \in H && (24) \\ \omega_t^{ip} &\leq \theta_{it} && \forall i \in N, \forall p \in P, \forall t \in H && (25) \\ \sum_{t \in H} \theta_{it} &= c_i && \forall i \in N && (26) \\ t + B(1 - \theta_{it}) &\geq m_i + 1 && \forall i \in N, \forall t \in H && (27) \\ t \leq m_i + c_i + B(1 - \theta_{it}) &&& \forall i \in N, \forall t \in H && (28) \\ s_k^i, x_{ik} &\in \{0, 1\} && \forall i \in N, \forall k \in M && (29) \\ y_{ij}, z_{ij} &\in \{0, 1\} && \forall i, j \in N && (30) \\ \mu_w^p &\in \{0, 1\} && \forall p \in P, \forall w \in W && (31) \\ \omega_t^{ip} &\in \{0, 1\} && \forall i \in N, \forall p \in P, \forall t \in H && (32) \\ \phi_{ip} &\in \{0, 1\} && \forall i \in N, \forall p \in P && (33) \\ \theta_{it} &\in \{0, 1\} && \forall i \in N, \forall t \in H && (34) \end{aligned}$$

The objective function (1) minimizes the total service time of all vessels, which is the sum of total delays and total handling time of vessels berthing at the port. Constraints (2) are the dynamic arrival constraints that ensure that vessels can be serviced only after their arrival. Constraints (3)-(5) are the non-overlapping restrictions for any two vessels berthing at the port. These ensure that while two vessels may be overlapping in space or in time, they cannot be simultaneously overlapping in both space and time. Note that the constraints (3)-(4) have been linearized by using a large positive constant B . Constraints (6)-(8) ensure that each vessel occupies only as many number of sections as determined by its length and the starting section occupied by the vessel. Constraints (9) ensure that the draft of the vessel does not exceed the draft of any occupied section.

Constraints (10) are used to determine the total handling time for any given vessel which is equal to the time taken to process the section occupied by the vessel whose operation finishes last. The variable component of the handling time for a given vessel i and given occupied section k is determined by the constraint (12), and is a function of the weighted average distance between section k and all cargo locations assigned to vessel i . The average distance is weighted over the cargo quantities transferred between each cargo location p assigned to vessel i and section k occupied by the vessel, and is determined by constraint (13). Constraints (11) determine the unit handling time of vessel i at a given section k , which is the sum of the fixed component dependent on the number of cranes operating on that section and the variable component of the handling time as discussed earlier. Constraints (14)-(16) state that the total cargo quantity to be loaded (discharged) is equal to the sum of the cargo quantities transferred from (to) all the cargo locations assigned to the vessel. Constraints (17) are capacity constraints to ensure that the amount of cargo transferred in a unit time does not exceed the maximum amount of cargo that can be handled as given by parameter R_w for cargo type w . Note that in case of specialized equipment facilities, the parameter R_w may refer to the conveyor speed or the flow rate through the pipeline; in other cases it may refer to the maximum rate of transfer of material between the quay side and the yard side that can be achieved by assigning auxiliary equipment such as loading shovels, wheel loaders etc.

Constraints (18) impose an upper bound on the maximum number of cargo locations that can be assigned to any single vessel. Constraints (19) ensure that two cargo types that cannot be stored together, for example coal and clay, are not assigned to adjacent yard locations. Constraints (20) state that a given cargo location at a given time can be used by at most one vessel to avoid congestion. Constraints (21) state that the yard locations have dedicated cargo types, and given yard location can be assigned to at most one cargo type. Constraints (22) ensure that a vessel is assigned to a yard location only if that yard location stores the cargo type on the vessel.

Constraints (23) - (25) control the values of the binary decision variable ω_t^{ip} which should take value equal to 1, if and only if both the binary variables ϕ_{ip} and θ_{it} are equal to 1. This implies that vessel i is handled at cargo location p at time t if and only if, cargo location p is assigned to vessel i and vessel i is being handled at time t . Similarly, constraints (26) - (28) control the values of the binary decision variable θ_{it} which is equal to 1 at all time intervals between the starting time of the handling of the vessel and the finishing time of the handling operations, and 0 otherwise.

5 Solution Approach

The mixed integer programming formulation of the integrated model is extremely complex and unwieldy, and could not be used to solve even small sized instances, as was validated by numerical experiments. Thus in the following section, we decompose the mixed integer model and formulate it as a set partitioning problem.

5.1 Set Partitioning Model

Let Ω be the set of feasible berthing assignments of all vessels berthing at the port in the given planning horizon. Note that a feasible assignment represents the assignment of a single vessel for a given set of section(s) for a specific time period, assigned to a set of specific cargo location(s) in the yard. Let c_a be the cost of assignment $a \in \Omega$. The following input parameters are used in the set partitioning model:

Input Parameters

$$A_a^i = \begin{cases} 1 & \text{if vessel } i \text{ is assigned in assignment } a, \\ 0 & \text{otherwise.} \end{cases}$$

$$B_a^{kt} = \begin{cases} 1 & \text{if section } k \text{ is occupied at time } t \text{ in assignment } a, \\ 0 & \text{otherwise.} \end{cases}$$

$$C_a^{pw} = \begin{cases} 1 & \text{if cargo type } w \text{ is stored at location } p \text{ in assignment } a, \\ 0 & \text{otherwise.} \end{cases}$$

$$D_a^{pt} = \begin{cases} 1 & \text{if assignment } p \text{ is handled at cargo location } p \text{ at time } t, \\ 0 & \text{otherwise.} \end{cases}$$

There is one additional parameter ϑ_w , which indicates the number of vessels carrying cargo type $w \in W$. The set partitioning model for the integrated berth allocation and yard assignment problem can then be formulated as following:

$$\min \sum_{a \in \Omega} c_a \lambda_a \quad (35)$$

$$\text{s.t. } \sum_{a \in \Omega} A_a^i \lambda_a = 1, \quad \forall i \in N, \quad (36)$$

$$\sum_{a \in \Omega} B_a^{kt} \lambda_a \leq 1, \quad \forall k \in K, \forall t \in H, \quad (37)$$

$$\sum_{a \in \Omega} C_a^{pw} \lambda_a - \vartheta_w \mu_w^p \leq 0, \quad \forall p \in P, \forall w \in W, \quad (38)$$

$$\sum_{w \in W} \mu_w^p \leq 1, \quad \forall p \in P, \quad (39)$$

$$\mu_w^p + \mu_w^{\bar{p}} \leq 1, \quad \forall p \in P, \forall \bar{p} \in \bar{P}, \forall w \in W, \forall \bar{w} \in \bar{W}, \quad (40)$$

$$\sum_{a \in \Omega} D_a^{pt} \lambda_a \leq 1, \quad \forall p \in P, \forall t \in H, \quad (41)$$

$$\lambda_a \in \{0, 1\}, \quad \forall a \in \Omega, \quad (42)$$

$$\mu_w^p \in \{0, 1\}, \quad \forall p \in P, \forall w \in W. \quad (43)$$

In the above formulation, λ_a indicates if assignment $a \in \Omega$ is part of the optimal solution (i.e. $\lambda_a = 1$), and the decision variable μ_w^p is retained from the original formulation and indicates if location $p \in P$ stores cargo type $w \in W$.

The objective function (35) minimizes the total service time of vessels berthing at the port. Constraints (36) ensure that there is exactly one feasible berthing assignment for each vessel in the optimal solution. Constraints (37) state that a given section at a given time can be occupied by at most one vessel. Constraints (38) and (39) state that a given cargo location on the yard can store at most one cargo type. Constraints (40) are identical to constraints (19) in the original formulation, ensuring that cargo types that cannot be stored together are not stored at adjacent locations in the yard. Constraints (41) ensure that at most one vessel can be handled at a cargo location at a given time to avoid congestion. Constraints (42) and (43) state that both the decision variables λ_p and μ_w^p are binary integer variables and can only take (0,1) values.

The above formulation was tested for small instances inspired from real port data. However, the number of feasible assignments given by the size of (Ω) grows exponentially with the problem size. This in turn leads to an exponential growth in the computational time. Hence in order to avoid the "explosion" of the solution time, we propose to solve the linear programming relaxation of the above problem using column generation, as described in the next section.

5.2 Column Generation

The main idea behind the column generation method is to keep the set partitioning model as the so called "restricted master problem" (RMP). We call it 'restricted', because instead of using the set of all feasible assignments Ω , only a subset of the assignments denoted by Ω_1 are considered in any given iteration. Moreover, the binary decision variables λ_a and μ_w^p are relaxed in order to speed up the convergence of the process.

In the first iteration of column generation, the RMP is solved using the set Ω_1 consisting of vessel assignments in the initial feasible solution provided by the greedy heuristic (Section 5.2.1). Thereafter, in each successive iteration of the column generation process, the following dual variables are passed to the subproblem for identifying feasible assignments with negative reduced cost:

- π_i – dual variables corresponding to constraints (36)
- τ_{kt} – dual variables corresponding to constraints (37)
- ξ_{pw} – dual variables corresponding to constraints (38)

Algorithm 1: Branch and Price

Data: data file, Ω , finished - boolean, duals - float

Result: $\Omega_1 \subset \Omega$, solution

```
1 begin
2    $\Omega_1 \leftarrow \text{greedy}(\Omega)$ 
3   duals  $\leftarrow \emptyset$ 
4   solution  $\leftarrow \emptyset$ 
5   repeat
6     duals  $\leftarrow \text{solveMaster}(\Omega_1)$ 
7     finished  $\leftarrow \text{true}$ 
8     for  $i \in N$  do
9       temp  $\leftarrow \text{solveSubProblem}(i, \text{duals})$ 
10      if  $\text{reducedCost}(\text{temp}) < 0$  then
11         $\Omega_1 \cup \text{temp}$ 
12        finished  $\leftarrow \text{false}$ 
13  until finished
14  solution  $\leftarrow \text{solveMaster}(\Omega_1)$ 
15  if solution  $\notin \mathbb{Z}$  then
16    ub  $\leftarrow \text{solveMaster}(\Omega_1, \text{integral})$ 
17    if solution = ub then
18      break
19    solution  $\leftarrow \text{branch\&bound}(\text{solution})$ 
20  print solution
```

- γ_{pt} – dual variables corresponding to constraints (41)

The rest of the dual variables are not considered, since from the definition of dual variables (also known as shadow prices), as described in Hillier and Lieberman (2001):

"The shadow price for resource i (denoted by y_i^*) measures the marginal value of this resource, i.e., the rate at which Z (objective function) could be increased by (slightly) increasing the amount of this resource being made available."

Since there is no direct effect on the objective function when the μ variable value(s) are adjusted in constraints 39 and 40, it implies that the dual variable values for these constraints would be always zero. Thus the dual variable values for these constraints are not reported to the sub-problem (Section 5.2.2). Based on the dual variables from RMP, the subproblem generates new columns to enter the active pool of columns Ω_1 by calculating the most negative reduced cost column for each vessel separately in each iteration of the column generation process. When there are no columns with negative reduced cost for any subproblem to enter Ω_1 , the column generation terminates.

The column generation in pseudocode can be seen in Lines (1) - (13) in Algorithm 1. For mathematical justification of column generation, please refer to Barnhart et al. (1998), Desaulniers et al. (2005) and Feillet (2010). In general, this method decomposes the model into a master problem and sub-problem, and hence reduces the solution space and enhances the convergence speed.

5.2.1 Initial Solution

Algorithm 2: Greedy Heuristic

Data: N - set of vessels, C - set of columns

Result: initial solution

```

1 begin
2   initial solution  $\leftarrow \emptyset$ 
3   for  $i \in N$  do
4     for  $j \in C$  do
5       if initial solution =  $\emptyset$  then
6         initial solution  $\leftarrow j$ 
7         break
8       else if  $i \in j$  and  $j$  is compatible with initial solution then
9         initial solution  $\cup j$ 
10        break
11  return initial solution

```

In order to execute the column generation, an initial feasible solution is needed. Note that if the problem is infeasible, no dual variables are produced causing the column generation approach to fail.

A greedy heuristic algorithm (Algorithm 2) is designed to extract an initial feasible solution to the master problem. The heuristic searches through the space of all feasible columns, where a column represents the assignment of a single vessel containing the following information:

- set of sections occupied by the vessel
- set of cargo locations in the yard assigned to the vessel (from this information related to the cargo types stored at specific yard locations can be derived)
- time period for which the given set of section(s) and cargo location(s) in the yard are occupied by the vessel (from this information related to the berthing delay and the handling time of the vessel can be derived)

5.2.2 Sub-Problem

In each iteration of column generation, we solve $|N|$ subproblems, one for each vessel $i \in N$. In each subproblem, the objective is to identify the feasible assignment for that particular vessel with the most negative reduced cost to be added to the current pool of active columns Ω_1 in the restricted master problem. Note that the index $i \in N$ is removed from all decision variables and input parameters in the sub-problem, since it is solved separately for each vessel $i \in N$. The objective function in a single sub-problem can be written as:

$$\min (m - a + c) - (\pi + \sum_{k \in K} \sum_{t \in H} \tau_{kt} \cdot \sigma_{kt} + \sum_{p \in P} \sum_{t \in H} \gamma_{pt} \cdot \psi_{pt} + \sum_{p \in P} \sum_{w \in W} \xi_{pw} \cdot \mu_{pw}) \quad (44)$$

where:

- $\pi, \tau_{kt}, \gamma_{pt}, \xi_{pw}$ are the dual variables obtained from the restricted master problem
- binary decision variables connecting the duals:
 - $\sigma_{kt} \in (0, 1)$ – 1 if section k is occupied at time t , 0 otherwise
 - $\psi_{pt} \in (0, 1)$ – 1 if cargo location p is used at time t , 0 otherwise
 - $\mu_{pw} \in (0, 1)$ – 1 if cargo type w is stored at location p , 0 otherwise
- m is the starting time of handling the vessel
- a is the arrival time of the vessel
- c is the handling time of the vessel

Input Parameters The input parameters used in the subproblem are:

- w – cargo type on the vessel
- δ_{lk} – fraction of cargo handled at section k , if the starting section of the vessel is l
- B – large positive constant (set to 1,000,000)
- b_k – starting coordinate of section k
- L – length of the vessel
- Q – quantity of cargo on the vessel
- Y – quay length
- ρ_{lk} – binary, equals 1 if section k is occupied by the vessel when section l is the starting section, 0 otherwise
- F – maximum number of cargo locations that can be assigned to the vessel
- r_k^p – distance between section k and cargo location p
- α_k^w – number of cranes operating in section k for cargo type w
- V_w – transfer rate of cargo type w

Decision Variables The decision variables used in the subproblem are:

- $h_k^w \geq 0$ – handling time of the vessel at section k
- $q_p \geq 0$ – quantity of cargo handled by the vessel at cargo location p
- $r_k \geq 0$ – weighted average distance for section k occupied by the vessel
- $s_l \in (0, 1)$ – binary, equals 1 if section l is the starting section of the vessel, 0 otherwise
- $x_k \in (0, 1)$ – binary, equals 1 if section k is occupied by the vessel, 0 otherwise
- $\phi_p \in (0, 1)$ – binary, equals 1 if cargo location p is assigned to the vessel, 0 otherwise
- $\theta_t \in (0, 1)$ – binary, equals 1 if the vessel is served at time t , 0 otherwise

Sub-problem formulation The subproblem can be formulated as a mixed integer linear program as follows:

$$m - a \geq 0, \quad (45)$$

$$c \geq h_k^w \rho_{lk} - B(1 - s_l), \quad \forall l, k \in M \quad (46)$$

$$\sum_{l \in M} s_l = 1, \quad (47)$$

$$\sum_{k \in M} (s_k b_k) + L \leq Y, \quad (48)$$

$$\sum_{l \in M} \rho_{lk} s_l = x_k, \quad \forall k \in M, \quad (49)$$

$$\sum_{l \in M} \phi_l \leq F, \quad (50)$$

$$\phi_p \leq \mu_w^p, \quad \forall p \in P, \quad (51)$$

$$\sum_{p \in P} q_p = Q, \quad (52)$$

$$q_p \leq \phi_p Q, \quad \forall p \in P, \quad (53)$$

$$\phi_p \leq q_p, \quad \forall p \in P, \quad (54)$$

$$r_k = \left(\sum_{p \in P} r_k^p q_p \right) / Q, \quad \forall k \in M, \quad (55)$$

$$h_k^w = \alpha_k^w + V_w r_k, \quad \forall k \in M, \quad (56)$$

$$\sum_{t \in T} \theta_t = c, \quad (57)$$

$$t + B(1 - \theta_t) \geq m + 1, \quad \forall t \in H, \quad (58)$$

$$t \leq m + c + B(1 - \theta_t), \quad \forall t \in H, \quad (59)$$

$$\sigma_{kt} \geq x_k + \theta_t - 1, \quad \forall k \in M, \forall t \in H, \quad (60)$$

$$\sigma_{kt} \leq x_k, \quad \forall k \in M, \forall t \in H, \quad (61)$$

$$\sigma_{kt} \leq \theta_t, \quad \forall k \in M, \forall t \in H, \quad (62)$$

$$\psi_{pt} \geq \phi_p + \theta_t - 1, \quad \forall p \in P, \forall t \in H, \quad (63)$$

$$\psi_{pt} \leq \phi_p, \quad \forall p \in P, \forall t \in H, \quad (64)$$

$$\psi_{pt} \leq \theta_t, \quad \forall p \in P, \forall t \in H. \quad (65)$$

Constraints (45) ensure that vessels can be served after their arrival. Constraints (46) are used to determine the total handling time of the vessel. Constraints (47) state that each vessel has exactly one starting section. Constraints (48) state that the vessel should be berthed such that it does not extend beyond the length of the quay. Constraints (49) determine if a particular section is occupied by the vessel. Constraints (50) impose an upper bound on the number of cargo locations that can be assigned to a single vessel. Constraints (51) ensure that a vessel is assigned to a yard location only if that yard location stores the cargo type on the vessel. Constraints (52) - (54) state that the total cargo quantity to be loaded (discharged) is equal to the sum of the cargo quantities transferred from (to) all the cargo locations assigned to the vessel. Constraints (55) calculate the weighted average distance over cargo quantities and constraints (56) calculate the handling time for given vessel and berthed section. Constraints (57) - (59) control the values of the binary decision variable θ_t ensuring they take value equal to 1 at all times when the vessel is berthed along the quay. Constraints (58) - (60) control the values of the binary decision variable σ_{kt} to ensure that they take value equal to 1 if and only if both x_k and θ_t are equal to 1. Similarly, constraints (61) - (63) control the values of the binary decision variable ψ_{pt} to ensure that they take value equal to 1 if and only if both ϕ_p and θ_t are equal to 1.

Since the sub-problem is solved separately for each vessel, there are no non-overlapping constraints in the above formulation and the complexity of the problem is significantly reduced. Thus in the proposed algorithm, the sub-problems are solved as mixed integer programs directly using CPLEX solver.

5.3 Branch and Bound

Since we solve the linear relaxation of the restricted master problem, the final solution obtained after the convergence of the column generation process is typically not an integer solution. In the "worst" case, the solution of the relaxed version of the restricted master problem provides a lower bound to the solution of the original problem and an integer solution can be obtained by applying the branch and bound algorithm (for more details on branch and bound, please refer to Hillier and Lieberman (2001)) to the obtained solution. However, this does not guarantee that we obtain the optimal integer solution to the original problem, since there might be a column that would price out favorably but is not present in the final pool of active columns Ω_1 . Therefore to find the optimal solution, column generation has to be executed again at every node of the branch and bound tree (Barnhart et al. (1998)).

The branching is held on the 2 decision variables: λ and μ . Since λ variables are not restricting the solution space dramatically, we branch first on the μ variables. Since μ variables restrict the cargo type stored in a given cargo location, several columns are discarded, hence the convergence of branch and bound is faster than with the restriction of λ variables. When the list of μ variables has been exhausted, i.e. there are no fractional μ variables, then we start branching on the λ variables.

Note that while branching, both the RMP and the subproblem(s) have to be modified by adding extra constraints. In RMP, when μ variables are set to 0 or 1 in a particular iteration, constraints enforcing the μ variables to take the same values are also added in each subproblem in that iteration. When λ_a is set to zero, it is ensured that the subproblem for the vessel i in assignment a does not generate that assignment, otherwise (when set to 1) there is no change.

The combination of column generation and branch and bound is called branch-and-price. The complete algorithm for branch-and-price can be seen in Algorithm 1.

6 Improvement Methods

The branch and price method has been designed to tackle computationally heavy problems and to achieve a tight bound (lower bound in case of minimization). The convergence of the method is very much dependent on how well the decomposition of the original problem is carried out. The recommended method is to decompose the model in a way such that the sub-problem has a block angular structure (see Alvelos (2005) and Desaulniers et al. (2005)), so that each subproblem can be solved independently of each other. In this study, each sub-problem is trying to find the berthing and yard assignment of a given vessel that has the most negative reduced cost for a given set of dual variables reported from the solution output of the restricted master problem in the last iteration. It is straightforward to see that the berthing schedule of one vessel (output of the specific sub-problem) solved separately, does not have any effect on the remaining sub-problems. Since the effect of dependency rises only in the master problem, block angular structure is achieved for our problem.

Since we have access to the Ω (from the generation of the initial solution), the first strategy considered was to solve the master problem on Ω_1 , retrieve dual variables and calculate reduced costs on Ω . However this method proved to be computationally heavy, hence not feasible.

Another acceleration strategy to speed up the rate of convergence of the column generation process is to retrieve more than one negative reduced cost columns per sub-problem. In this study, up to 40

columns are added per sub-problem. This is achieved via modification of the original formulation of the sub-problem(s). The objective function is removed, and instead following constraint is inserted into the model:

$$(m - a + c) - (\pi + \sum_{k \in K} \sum_{t \in H} \tau_{kt} \cdot \sigma_{kt} + \sum_{p \in P} \sum_{t \in H} \gamma_{pt} \cdot \psi_{pt} + \sum_{p \in P} \sum_{w \in W} \xi_{pw} \cdot \mu_{pw}) \leq 0 \quad (66)$$

The above constraint together with constraints (45) - (65) yield as solution any feasible assignment with reduced cost less than or equal to zero.

There are many other techniques to improve the convergence of the column generation process such as stabilization of the dual variables (Pigatti et al. (2005)) and dynamic constraint aggregation (Elhallaoui et al. (2005) and Elhallaoui et al. (2008)).

These methods are not universal and depend on the structure of the specific problem under study. We studied these two techniques for our problem, however the results didn't look very promising and hence have not been included in this paper. In the dual stabilization method, the basic assumption that the dual variables are oscillating is not fulfilled and hence it leads to the failure of the method, although it needs lesser number of columns to finish the column generation process.

The dynamic constraint aggregation technique that we tested is based on the aggregation of constraints of the same class and same coefficient values in the restricted master problem. This technique didn't speed up the convergence of the column generation process significantly, though the bounds were found to be tighter and it was found to converge faster than standard column generation for larger problem size. This can be attributed to the fact that most of the computation time needed for the convergence of the column generation process is the time taken to solve the sub-problems, while the computation time of the master problem is already very fast for smaller problem size.

7 Results and Discussion

7.1 Generation of Instances

The proposed branch-and-price algorithm was run on an x64-bit Intel Core i7, (1.60 GHz) using a 64-bit version of CPLEX 12.3. The algorithm was implemented in Java Programming language.

The test instances are based on a sample of real port data obtained from the SAQR port, Ras-Al-Khaimah, UAE, the biggest bulk port in the middle east. In terms of the level of congestion as determined by the inter-arrival times of the incoming vessels, the instances consist of 3 different types of scenarios - congested, mildly congested and uncongested. In all the instances, the total quay length is 1600 meters, and the vessel lengths lie in the range 50-260 meters as in SAQR.

The cargo quantities on the vessels vary between 1 and 15 cargo units units. In the tested instances, we consider five cargo types - liquid bulk that needs the pipeline facility for discharging, rock aggregates that need the conveyor facility for loading and three other general type dry cargoes (clay, coal and cement). The incompatible pair of cargo types that cannot be stored at adjacent locations in the yard are liquid bulk - clay, and rock - cement. The berthing layout along with the location of the fixed facilities such as conveyors and pipelines used in the instances is shown in the Table 5.

Section	Length	Facility
1	150	C
2	50	C
3	200	P
4	150	C,P
5	125	C,P
6	250	-
7	250	-
8	75	P
9	150	P
10	200	-

Table 5: Berthing layout and fixed facility locations for $|M|=10$ (C and P stand for conveyor and pipeline respectively)

In all the tested instances, the planning horizon is 5 days or 120 hours. As illustrated on Figure 2, we have 10 cargo locations in the yard: location 1 has the pipeline facility and is hence designated for liquid cargo, likewise location 10 has the conveyor facility and is hence designated to the rock aggregates; locations 2 to 9 are designated to other cargo types, which is a part of the output of the optimization problem being solved.

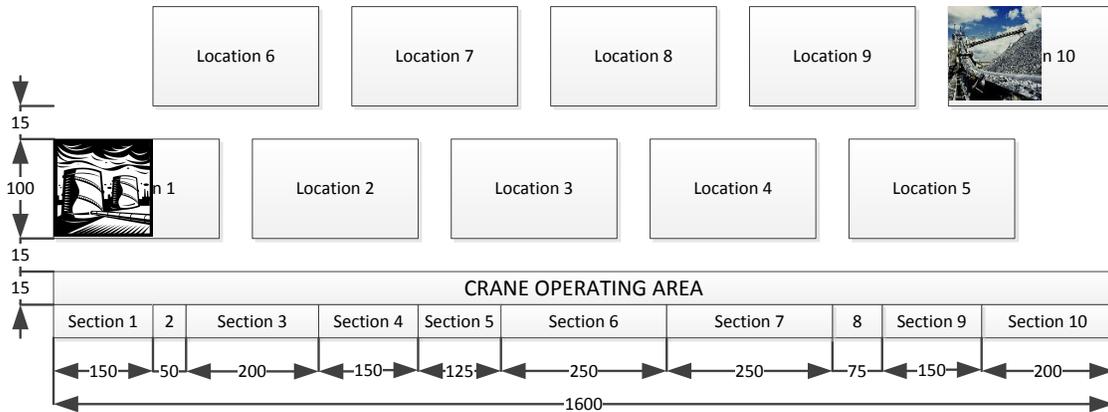


Figure 2: Schematic representation of the port used in instances

7.2 Computational Results and Analysis

In Tables 6, 8 and 9, the results obtained from the column generation method for instances containing $|N|=10, 25$ and 40 vessels respectively are shown. As discussed earlier, the solution value obtained from the column generation method provides a lower bound to the optimal solution of the original problem.

instance #	congestion	time	# iter.	Ω_1	lb	ub	gap1	gap2	gap
1	yes	4m 16s	15	1 434	125	127	1.5%	0%	1.5%
2	mild	3m 16s	14	996	131	134	1.5%	0.7%	2.2%
3	no	1m 47s	11	737	113	117	–	–	3.4% ²
4	yes	5m 13s	21	1 294	130	131	0%	0.8%	0.8%
5	mild	3m 18s	18	1 027	131	132	0.7%	0%	0.7%
6	no	1m 39s	17	692	109	111	1.8%	0%	1.8%
7	yes	3m 59s	17	1 311	126	126	0%	0%	0% ¹
8	mild	2m 37s	13	1 025	122	122	0%	0%	0% ¹
9	no	1m 29s	9	502	110	117	1.8%	4.5%	6.3%

¹ Zero optimality gap obtained from the column generation method without running the B&B algorithm.

² Optimality gap without running the B&B algorithm, because of its increased time complexity.

Table 6: Results obtained from the column generation for the instances containing $|N| = 10$ vessels.

The upper bound to the optimal solution is obtained by running the RMP with integer decision variables at the end of the column generation method.

In the tables shown, the optimality gap between the lower bound and the optimal solution is represented by *gap1*, whereas *gap2* represents the optimality gap between the optimal solution and the upper bound. The total gap between the lower bound and the upper bound, represented by *gap*, is then simply the sum of *gap1* and *gap2*.

As shown in Table 6, for $|N| = 10$ vessels, for instances 7 and 8, the optimal solution is obtained directly from the column generation method, and hence there is no need for running the branch and bound algorithm for these instances. For the 3rd instance, the computational time of the branch and bound algorithm explodes and the optimal solution remains unknown.

Note that because of the exponential increase in the time complexity of the branch and bound algorithm with increase in the instance size, the branch and bound algorithm is executed for only instances containing $|N| = 10$ vessels. The results are as shown in Table 7, strongly indicating that branching on the μ variables is sufficient in almost all cases. Note that for instances containing $|N| = 25, 40$ vessels, the total optimality gap is still small, and thus we do not execute the branch and bound algorithm and save on the computational time. In these cases, the upper bound is accepted as the final sub-optimal solution to our problem.

Overall from the results of the column generation shown in Tables 6, 8 and 9, we can observe the following: barring a few exceptions, the computational time increases with the level of congestion, as well as with the final size of the active pool of columns given by Ω_1 and the number of iterations of the column generation process. On the other hand, the total optimality gap represented by *gap* in the tables, is smaller for the instances with higher level of congestion. This can be attributed to the fact that the total number of feasible solutions are lesser for congested scenarios.

instance #	time	value	# of nodes	# of μ	# of λ
1	10m 02s	127	23	22	0
2	12m 21s	133	43	32	10
3	> 4h 00m 00s	–	–	–	–
4	01m 57s	130	5	4	0
5	03m 15s	132	7	6	0
6	03m 12s	111	33	32	0
7	–	–	–	–	–
8	–	–	–	–	–
9	02m 22s	112	29	28	0

Table 7: Results obtained from the branch and bound for the instances containing $|\mathcal{N}| = 10$ vessels.

instance #	congestion	time	# iter.	Ω_1	lb	ub	gap
1	yes	5h 03m 13s	60	6 566	512	529	3.2%
2	mild	2h 18m 38s	38	6 454	459	477	3.8%
3	no	48m 08s	25	5 300	413	449	8.0%
4	yes	1h 34m 32s	26	6 868	520	532	2.3%
5	mild	1h 02m 18s	18	6 051	469	481	2.5%
6	no	1h 13m 39s	32	5 030	436	470	7.2%
7	yes	2h 18m 05s	33	5 757	506	513	1.4%
8	mild	2h 12m 00s	47	6 184	454	467	2.8%
9	no	40m 00s	28	4 877	388	412	5.8%

Table 8: Results obtained from the column generation for the instances containing $|\mathcal{N}| = 25$ vessels.

instance #	congestion	time	# iter.	Ω_1	lb	ub	gap
1	yes	3h 12m 15s	35	9 662	677	698	3.0%
2	mild	1h 52m 46s	31	9 017	586	608	3.6%
3	no	1h 26m 20s	39	8 174	457	483	5.4%
4	yes	7h 17m 20s	79	10 743	664	678	2.0%
5	mild	4h 02m 14s	55	9 771	581	593	2.0%
6	no	3h 00m 27s	54	8 620	470	486	3.3%
7	yes	4h 02m 58s	44	8 727	721	758	4.9%
8	mild	3h 03m 38s	35	8 412	641	676	5.2%
9	no	1h 20m 27s	33	7 791	508	532	4.5%

Table 9: Results obtained from the column generation for the instances containing $|\mathcal{N}| = 40$ vessels.

8 Conclusions and Future Work

In this research, we study the integrated problem of berth allocation and yard assignment in context of bulk ports. In the past, few scholars have attempted to study the berth allocation problem in integration with the yard assignment problem in the context of container terminals, while in the context of bulk ports the problem has not been studied at all. The specific issues in bulk port operations that necessitate the need to devise specific solutions for bulk ports are emphasized in this study. An exact solution algorithm based on branch-and-price is proposed to solve the large scale integrated problem of berth allocation and yard assignment. To the best of our knowledge, this is the first study that proposes an exact solution method to solve the large scale problem in the context of port operations planning, since all the previous studies in the context of container terminals use metaheuristic methods to solve the problem.

The mathematical formulation of the integrated berth allocation and yard assignment problem is complex and extensive, however there is still scope for improvement. An important assumption in our model is that each vessel carries only a single type of cargo. In order to model and solve the problem for multiple cargo types, the location of each cargo type on the vessel needs to be explicitly modeled which would require the modification of some of the existing constraints in the current formulation as well as addition of some new constraints. Another possible extension of the current work is to account for the uncertainty in arrival times of the vessels and delays in handling operations due to factors such as breakdown of handling equipment including conveyors, pipelines and/or mobile harbor cranes etc. As for the solution method, the major source of the time complexity of the column generation framework has been identified as the solution of the sub-problems. Thus it may be worth investigating the reduction in the solution time of the sub-problems by using heuristic methods such as dynamic programming, instead of using exact methods and optimization solvers. There is also scope to obtain a sub-optimal integer solutions in a more time efficient manner by using heuristic methods (GRASP for instance) instead of using the branch and bound algorithm proposed in this study. Finally more sophisticated techniques such as dual stabilization and dynamic constraint aggregation to speed up the convergence of the column generation process need to be studied and implemented in a way such that they exploit the structure of the specific problem under study.

References

- Alvelos, F. (2005). *Branch-and-Price and Multicommodity Flows*, PhD thesis, Universidade do Minho.
- Barnhart, C., Johnson, E. L., Nemhauser, G. L., Savelsbergh, M. W. P. and Vance, P. H. (1998). Branch-and-price: Column generation for solving huge integer programs, *Operations Research* **46**(3): 316–329.
- Bierwirth, C. and Meisel, F. (2010). A survey of berth allocation and quay crane scheduling problems in container terminals, *European Journal of Operational Research* **202**(3): 615–627.
- Bish, E. K., Leong, T. Y., Li, C. L., Ng, J. W. C. and Simchi-Levi, D. (2001). Analysis of a new vehicle scheduling and location problem, *Naval Research Logistics* **48**: 363–385.
- Buhrkal, K., Zuglian, S., Ropke, S., Larsen, J. and Lusby, R. (2011). Models for the discrete berth allocation problem: A computational comparison, *Transportation Research Part E* **47**(4): 461 – 473.
- Chang, D., Yan, W., Chen, C.-H. and Jiang, Z. (2008). A berth allocation strategy using heuristics algorithm and simulation optimisation, *Int. J. Comput. Appl. Technol.* **32**: 272–281.
- Chen, Z. L., Lei, L. and Zhong, H. (2007). Container vessel scheduling with bi-directional flows, *Operations Research Letters* **35**: 186 – 194.
- Cheng, Y. L., Sen, H. C., Natarajan, K., Teo, C. P. and Tan, K. C. (2005). Dispatching automated guided vehicles in a container terminal, in J. Geunes and P. M. Pardalos (eds), *Supply Chain Optimization*, Applied Optimization, Springer.
- Cheong, C., Tan, K., Liu, D. and Lin, C. (2010). Multi-objective and prioritized berth allocation in container ports, *Annals of Operations Research* **180**: 63–103.
- Cheung, R. K., Li, C. L. and Lin, W. (2002). Interblock crane deployment in container terminals, *Transportation Science* **36**: 79–93.
- Cordeau, J. F., Gaudioso, M., Laporte, G. and Moccia, L. (2007). The service allocation problem at the Gioia Tauro maritime terminal, *European Journal of Operational Research* **176**(2): 1167–1184.
- Cordeau, J. F., Laporte, G., Legato, P. and Moccia, L. (2005). Models and tabu search heuristics for the berth-allocation problem, *Transportation Science* **39**(4): 526–538.
- Dai, J., Lin, W., Moorthy, R. and Teo, C. P. (2008). Berth allocation planning optimization in container terminals, in C. S. Tang, C. P. Teo and K. K. Wei (eds), *Supply Chain Analysis*, Vol. 119 of *International Series in Operations Research & Management Science*, Springer, pp. 69–104.

- Desaulniers, G., Desrosiers, J. and Solomon, M. (eds) (2005). *Column Generation*, GERAD 25th Anniversary Series, Springer.
- Elhallaoui, I., Desaulniers, G., Metrane, A. and Soumis, F. (2008). Bi-dynamic constraint aggregation and subproblem reduction, *Comput. Oper. Res.* **35**(5): 1713–1724.
URL: <http://dx.doi.org/10.1016/j.cor.2006.10.007>
- Elhallaoui, I., Villeneuve, D., Soumis, F. and Desaulniers, G. (2005). Dynamic aggregation of set-partitioning constraints in column generation, *Oper. Res.* **53**(4): 632–645.
URL: <http://dx.doi.org/10.1287/opre.1050.0222>
- Feillet, D. (2010). A tutorial on column generation and branch-and-price for vehicle routing problems, *4or* **8**(4): 407–424.
URL: <http://www.springerlink.com/index/10.1007/s10288-010-0130-z>
- Giallombardo, G., Moccia, L., Salani, M. and Vacca, I. (2010). Modeling and solving the tactical berth allocation problem, *Transportation Research Part B* **44**(2): 232–245.
- Guan, Y. and Cheung, R. K. (2004). The berth allocation problem: models and solution methods, *OR Spectrum* **26**: 75–92.
- Guan, Y., Xiao, W.-Q., Cheung, R. K. and Li, C.-L. (2002). A multiprocessor task scheduling model for berth allocation: heuristic and worst-case analysis, *Operations Research Letters* **30**(5): 343–350.
- Han, X.-l., Lu, Z.-q. and Xi, L.-f. (2010). A proactive approach for simultaneous berth and quay crane scheduling problem with stochastic arrival and handling time, *European Journal of Operational Research* **207**(3): 1327 – 1340.
- Hillier, F. S. and Lieberman, G. J. (eds) (2001). *Introduction to Operations Research*, McGraw-Hill Science/Engineering/Math.
- Imai, A., Chen, H., Nishimura, E. and Papadimitriou, S. (2008). The simultaneous berth and quay crane allocation problem, *Transportation Research Part E* **44**(5): 900–920.
- Imai, A., Nagaiwa, K. and Chan, W. T. (1997). Efficient planning of berth allocation for container terminals in Asia, *Journal of Advanced Transportation* **31**(1): 75–94.
- Imai, A., Nishimura, E., Hattori, M. and Papadimitriou, S. (2007). Berth allocation at indented berths for mega-containerships, *European Journal of Operational Research* **179**(2): 579–593.
- Imai, A., Nishimura, E. and Papadimitriou, S. (2001). The dynamic berth allocation problem for a container port, *Transportation Research Part B* **35**(4): 401–417.
- Imai, A., Nishimura, E. and Papadimitriou, S. (2003). Berth allocation with service priority, *Transportation Research Part B* **37**(5): 437–457.

- Imai, A., Nishimura, E. and Papadimitriou, S. (2008). Berthing ships at a multi-user container terminal with a limited quay capacity, *Transportation Research Part E* **44**: 136–151.
- Imai, A., Sun, X., Nishimura, E. and Papadimitriou, S. (2005). Berth allocation in a container port: using a continuous location space approach, *Transportation Research Part B* **39**(3): 199–221.
- Jung, S. H. and Kim, K. H. (2006). Load scheduling for multiple quay cranes in port container terminals, *Journal of Intelligent Manufacturing* **17**: 479–492.
- Kim, K. H. and Kim, K. Y. (1999). Routing straddle carriers for the loading operation of containers using a beam search algorithm, *Computers and Industrial Engineering* **36**(1): 109–136.
- Kim, K. H., Lee, K. M. and Hwang, H. (2003). Sequencing delivery and receiving operations for yard cranes in port container terminals, *International Journal of Production Economics* **84**: 283–292.
- Kim, K. H. and Moon, K. C. (2003). Berth scheduling by simulated annealing, *Transportation Research Part B* **37**(6): 541–560.
- Kozan, E. and Preston, P. (2006). Mathematical modelling of container transfers and storage locations at seaport terminals, *OR Spectrum* **28**: 519–537.
- Lau, H. and Zhao, Y. (2007). Integrated scheduling of handling equipment at automated container terminals, *International Journal of Production Economics* p. doi:10.1016/j.ijpe.2007.05.015.
- Lee, L. H., Chew, E. P., Tan, K. C. and Han, Y. (2006). An optimization model for storage yard management in transshipment hubs, *OR Spectrum* **28**: 539–561.
- Li, C. L., Cai, X. and Lee, C. Y. (1998). Scheduling with multiple-job-on-one-processor pattern, *IIE Transactions* **30**: 433–445.
- Lim, A. (1998). The berth planning problem, *Operations Research Letters* **22**(2-3): 105–110.
- Liu, C. I., Jula, H., Vukadinovic, K. and Ioannou, P. (2004). Automated guided vehicle system for two container yard layouts, *Transportation Research Part C* **12**: 349–368.
- Mauri, G., Oliveira, A. and Lorena, L. (2008). A hybrid column generation approach for the berth allocation problem, in J. van Hemert and C. Cotta (eds), *Evolutionary Computation in Combinatorial Optimization*, Vol. 4972 of *Lecture Notes in Computer Science*, Springer Berlin / Heidelberg, pp. 110–122.
- Meisel, F. and Bierwirth, C. (2006). Integration of berth allocation and crane assignment to improve the resource utilization at a seaport container terminal, *Operations Research Proceedings 2005*, Springer, pp. 105–110.
- Meisel, F. and Bierwirth, C. (2008). Heuristics for the integration of crane productivity in the berth allocation problem, *Transportation Research Part E* **45**(1): 196–209.

- Monaco, M. F. and Sammarra, M. (2007). The berth allocation problem: a strong formulation solved by a lagrangean approach, *Transportation Science* **41**(2): 265–280.
- Moorthy, R. and Teo, C. P. (2006). Berth management in container terminal: the template design problem, *OR Spectrum* **28**(4): 495–518.
- Ng, J. W. C. (2005). Crane scheduling in container yards with inter-crane interference, *European Journal of Operational Research* **164**: 64–78.
- Ng, J. W. C. and Mak, K. L. (2005). Yard crane scheduling in port container terminals, *Applied Mathematical Modelling* **29**: 263–276.
- Nishimura, E., Imai, A. and Papadimitriou, S. (2001). Berth allocation planning in the public berth system by genetic algorithms, *European Journal of Operational Research* **131**(2): 282 – 292.
- Nishimura, E., Imai, A. and Papadimitriou, S. (2009). Container storage and transshipment marine terminals, *European Journal of Operational Research* **45**: 771–786.
- Park, K. T. and Kim, K. H. (2002). Berth scheduling for container terminals by using a sub-gradient optimization technique, *Journal of the Operational Research Society* **53**: 1054–1062.
- Park, Y. M. and Kim, K. H. (2003). A scheduling method for berth and quay cranes, *OR Spectrum* **25**(1): 1–23.
- Pigatti, A., Poggi and Uchoa, E. (2005). Stabilized Branch-and-cut-and-price for the Generalized Assignment Problem, *GRACO'05*, Vol. 19, pp. 389–395.
- Stahlbock, R. and Voss, S. (2008). Operations research at container terminals: a literature update, *OR Spectrum* **30**(1): 1–52.
- Steenken, D., Voss, S. and Stahlbock, R. (2004). Container terminal operation and operations research - a classification and literature review, *OR Spectrum* **26**(1): 3–49.
- Tong, C., Lau, H. and Lim, A. (1999). Ant colony optimization for the ship berthing problem, in P. Thiagarajan and R. Yap (eds), *ASIAN '99 Proceedings of the 5th Asian Computing Science Conference on Advances in Computer Science*, Vol. 1742 of *Lecture Notes in Computer Science*, Springer, pp. 789–789.
- Umang, N., M.Bierlaire and Vacca, I. (2012). Exact and heuristic methods to solve the berth allocation problem in bulk ports, *Technical report*, TRANSP-OR, Ecole Polytechnique Federale De Lausanne.
- UNCTAD (2009). *Review of Maritime Transport*, United Nations Conference on Trade and Development, <http://www.unctad.org>.
- Vacca, I. (2011). *Container terminal management: integrated models and large-scale optimization algorithms*, PhD thesis, Ecole Polytechnique Fédérale de Lausanne.

- Vis, I. F. A., de Koster, R. and Savelsbergh, M. W. P. (2005). Minimum vehicle fleet size under time-window constraints at a container terminal, *Transportation Science* **39**: 249–260.
- Zhang, C., Wan, Y. W., Liu, J. and Linn, R. (2002). Dynamic crane deployment in container storage yards, *Transportation Research Part B* **36**: 537–555.
- Zhen, L., Chew, E. P. and Lee, L. H. (2011). An integrated model for berth template and yard template planning in transshipment hubs, *Transportation Science* **45**: 483–504.
- Zhou, P.-f. and Kang, H.-g. (2008). Study on berth and quay-crane allocation under stochastic environments in container terminal, *Systems Engineering - Theory & Practice* **28**(1): 161 – 169.