MIP approaches for the integrated berth allocation and quay crane assignment and scheduling problem *

Agostinho Agra
aagra@ua.pt
Department of Mathematics and CIDMA, University of Aveiro

Maryse Oliveira
maryse.oliveira@ua.pt
Department of Mathematics, University of Aveiro

September 2, 2017

Abstract

In this paper we consider an integrated berth allocation and quay crane assignment and scheduling problem motivated by a real case where a heterogeneous set of cranes is considered. A first mathematical model based on the *relative position formulation* (RPF) for the berth allocation aspects is presented. Then, a new model is introduced to avoid the big-M constraints included in the RPF. This model results from a discretization of the time and space variables. For the new discretized model several enhancements, such as valid inequalities, are introduced. In order to derive good feasible solutions, a rolling horizon heuristic (RHH) is presented. A branch and cut approach that uses the enhanced discretized model and incorporates the upper bounds provided by the RHH solution is proposed. Computational tests are reported to show (i) the quality of the linear relaxation of the enhanced models; (ii) the effectiveness of the exact approach to solve to optimality a set of real instances; and (iii) the scalability of the RHH based on the enhanced mathematical model which is able to provide good feasible solutions for large size instances.

keywords: Logistics; Berth allocation; Quay crane scheduling; mixed integer formulations; branch and cut; rolling horizon heuristic.

1 Introduction

Maritime transportation is a major mode of transportation. Around 80% of global trade by volume and over 70% of global trade by value are carried by sea and are handled by ports worldwide [37]. Port activities involve several interrelated decisions, as berth assignment of vessels, quay crane assignment and scheduling, cargo placement, etc. During peak periods of activity delays may occur leading to large waiting costs. The request for operational efficiency at ports has motivated the increase of research during the last decades on such optimization problems, which is even more

[*] Published in European Journal of Operations Research, 264 (1), 138– 148 2018

visible during the last years with a rapid increase on the number of papers, some of them focused on real applications. For a recent survey see [7].

This paper considers an integrated berth allocation and quay crane assignment and scheduling problem occurring at a port whose main activities are related to short sea shipping operations. For a given time period, a sets of vessels is considered. For each vessel the arrival time and cargo quantity (to load or unload) are known. The objective is to manage the load and unload operations in order to minimize the total service completion time. These operations are performed with a set of cranes that transfer the cargo between the vessels and the storage area at the yard, as depicted in Figure 1. This problem integrates two subproblems: the Berth Allocation Problem (BAP), which aims to assign arriving vessels to berthing positions, and the quay crane assignment and scheduling problem (QCASP) where cranes are assigned to vessels, and their operations are scheduled. The complete problem is known by the acronym BACASP [41]. The storage management of the cargo at the yard is not considered here.

The cranes are mounted on rails. This physical limitation creates operational restrictions on the quay areas where each crane can operate, and enforces non-crossing constraints, that is, the relative position of the cranes cannot be exchanged, see Figure 1. An additional complexity of this real problem is the existence of a heterogeneous set of cranes types, with different processing rates. The physical operational limitation of the cranes combined with the cranes efficiency, make some berthing areas more attractive than others. Typically an area that is served by a more efficient crane tends to be used more often than the other areas. Other physical aspects, such as the proximity to the storage yard, the structure of the berth, may also influence berth allocation. See for example [5]. This fact makes the BACASP that results from the integration of both subproblems (BAP and QCASP) even more relevant than in the case of homogeneous cranes.

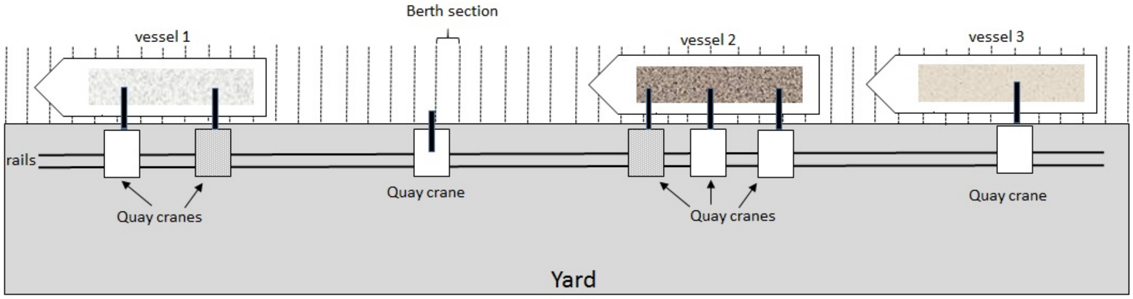

Figure 1: Example of a quay operation on three ships and seven cranes (five of them of one type and the other two of other type), six of them operating.

Following the classical approach for BAP (see [16]), a time and space discretization is assumed. The wharf is divided into sections of the same length. Contrary to the case of the discrete berth allocation, where each vessel is assigned to a single berth position, in the continuous berth allocation several adjacent sections are assigned to each vessel, corresponding to its length. The continuous berth allocation assumption is more flexible and such flexibility is required in practical cases, as the one we consider, where the wharf length is a binding restriction and the set of cranes is heterogeneous.

As BACASP has many variants which depend on the assumptions made, next we list the assumptions for our problem:

1. A dynamic berth allocation where vessels are allowed to arrive any time within the planning horizon but their arriving time is deterministic.

2. No service priorities are considered.

3. Vessels are allowed to moor at any place along the wharf, which is known in the literature as the continuous berth allocation.

4. The berth is divided into berth sections of equal length.

5. The time horizon is divided into time periods of equal length.

6. A time-variant assignment of cranes, that is, a crane can be assigned to a vessel and move to another vessel in the following period while the first vessel is still operating.

7. Non-crossing constraints and safety space between cranes must be obeyed.

Both subproblems BAP and QCASP have received great attention in the past years, see the survey [6] and its follow-up [7] with a very recent overview and classification. As the integrated problem is very complex many approaches consider the two problems separately. For the BAP see, for instance, [16, 18, 19, 27, 30, 42, 46, 47]. For the QCASP see [4, 13, 14, 15, 25, 28, 29, 32, 35, 38, 40]. Some authors consider the travelling times of cranes when moving between quay positions, see [25, 32]. Here we assume that such travelling times are negligible. Crane scheduling problems at port terminals may occur not only at the quay area but at other areas, such as the storage area. Although such storage management issues can be very complex, see for example [23, 22], here we consider only the scheduling of the quay cranes. For a recent classification of the crane scheduling problems see [9].

It is well-known that better solutions can in general be obtained by analysing all the decisions together. The integrated problem has been considered before in several papers such as [3, 8, 10, 11, 17, 20, 21, 24, 26, 31, 34, 36, 39, 40, 41, 43, 44, 45, 48].

Next we briefly review some of the most relevant references for the integrated model. For a more detailed information on the problem characteristics and approaches we suggest the very recent work [21].

Park and Kim [34] present an integer programming model which is used in a two-phases solution procedure. Ak [3] provides both a mathematical analysis and heuristics. Chang et al. [10] focus on heuristic approaches: a rolling-horizon approach and a hybrid parallel genetic algorithm. Model formulations and genetic algorithms are also presented by Imai et al. [20] and Liang et al. [26]. Meisel and Bierwirth [31] give several heuristic procedures including a construction heuristic, local refinement procedures, and two meta-heuristics. Heuristic procedures are developed also by Zhang et al. [44] (a subgradiente based heuristic) and by Yang et al. [48] (an evolutionary algorithm). Blazewicz et al. [8] consider the integrated problem as a scheduling problem where tasks are considered the ships and processors are the quay cranes. Giallombardo et al. [24] consider a MILP model to solve small instances and a Tabu search heuristic for generating feasible solutions. Raa et al. [36] present a formulation that is used in a hybrid heuristic solution procedure. Song et al. [39] follow a bi-level programming approach where the BAP is considered in the upper-level problem and the QCASP is considered in the lower-level. Recently a constraint programming approach was followed by Zampelli et al. [45]. In a different perspective, Han et al. [17] consider the case with uncertainty on the vessel arrival times. They propose a mixed integer programming model, and use a simulation based genetic algorithm search procedure to generate robust solutions. Chen et al. [11] propose a Benders decomposition method over the berth-level model introduced by Liu et al. They also use valid inequalities to improve the mathematical model. [29]. Vacca et al. [43] propose a model based on an exponential number of variables that is solved with a branch-and-price scheme.

More recently, Turkogullari et al. [41] present a mathematical model for QCASP assuming a time-invariant quay crane assignment policy. First they solve the model considering only the number of cranes assigned to each vessel and then they extend the approach to general assignment case. Iris et al. [21] propose new set partition models and provide variable reduction techniques. Their improved models are compared with the one given in [31]. Both time-invariant and time-variant problems are considered.

The development of commercial solvers combined with the use of integer programming theory to create tuned models has been allowing to solve an increasing number of practical problems. The trend is to increase the size of those instances that can be solved to optimality using such exact tools. When the problems are not solved to optimality, these tuned models can often be used within heuristics. Following such line of research, in this paper we present and discuss mixed integer formulations to model the integrated problem with a heterogenous set of cranes. Those formulations are used within a branch and cut algorithm which can be used either to solve the problem instances directly or within heuristic schemes.

It is well-known that the performance of the exact methods such as the branch and bound and the branch and cut depend on the quality of the bounds (lower and upper) to the optimal solution. In a minimization problem, the quality of the lower bounds depend on the quality of the formulation, that is, depend on how distant is the linear relaxation value to the optimum value. The quality of the upper bound depend on the distance between the value of the feasible solution to the optimum value. If the bounds are bad, the number of nodes in the enumeration search tree of these exact methods tend to be large, making those exact approaches unattractive from a practical point of view. Conversely, if the formulation provides tight lower bounds and a good feasible solution is known, one can expect that the search tree nodes are pruned at early stages of the enumeration scheme. Hence, in this paper we address the two issues: (i) how to derive good mixed integer formulations, and (ii) how to derive good upper bounds. Next, we consider each case separately and relate it with the literature.

Lower bounds. For the continuous BAP, the classical formulations are the well-known *Relative Position Formulation* (RPF) and the *Position Assignment Formulation* (PAF) that are based on the space-time diagram [16]. The RPF uses variables that order each pair vessels in time and space, separately, while the PAF uses variables that assign blocks of the space-time diagram to each vessel. A RPF is used in [31] for an integrated berth and crane allocation problem. The main drawback of these formulations are the big-M constraints used to link the variables. Those constraints lead to models with very weak linear relaxation based lower bounds. Although the PAF avoids some of those constraints, it includes many more variables. For a detailed analysis of these formulations see [3, 16]. New formulations have been used recently where the binary variables are associated with an assignment of a vessel to a berth position, a time period and a number of cranes [21, 41, 43]. Such formulations, known as *Set Partition Formulations* (SPF), are known to be tighter and avoid the big-M constraints. However they originate an exponential number of variables and can only be used directly to solve small size instances of simpler problems, such as the discrete berth assignment (where the number of berth positions is small and it is not necessary to enforce a vessel to occupy adjacent berth sections) and the case of time-invariant quay crane assignment (where the schedule of the cranes is ignored, and only the number of cranes assigned to the vessel is relevant). Even for such less complex problems, solving instances to optimality may require the use of column generation based techniques such as the branch and price algorithm, see [43].

Here we propose a new formulation that uses variables that discretize the integer variables used in the RPF (the berth sections and the time). This new model will be named as discretized RPF and to the best of our knowledge it has never been used before. This model has the advantage of avoiding the big-M constraints and to avoid the large number of variables used in the SPF. Additionally, families of valid inequalities are introduced to enhance the discretized formulation. The use of valid inequalities is not common within recent literature in BAP and QCSP. Exceptions are [11, 32, 41] that introduce valid inequalities which are not directly related to the ones we introduce in this paper and cannot be directly used here. As a result of these model improvements, the linear relaxation of the new formulation produces much better lower bounds than the linear relaxation of the RPF, as it is shown in Section 6. Both the discretized formulation and the valid inequalities can be used for the heterogeneous and homogeneous cranes cases. For the homogeneous cranes case, a family of valid inequalities becomes the classical cover inequalities.

Upper bounds. In order to obtain good upper bounds we introduce a rolling horizon heuristic. The heuristic solves a sequence of restricted subproblems. In each iteration a new subproblem is considered by including a set of a given number of arriving vessels and freezing some decisions taken in relation to the previous vessels. These restricted subproblems are solved by a branch and cut algorithm based on the enhanced discretized model. A Rolling horizon strategy was used in [10] for a related problem with different assumptions.

The heuristic can be used either to provide good upper bounds that are used as cutoff values to speed up the branch and cut algorithm, or to provide good feasible solutions for the large size instances when the exact approach fails.

The computational results show that a commercial solver based on the discretized formulation, enhanced with valid inequalities, and feeded up with the solution of the rolling horizon heuristic solves all the instanced based on real data to optimality. Moreover, the three main contributions of this paper: (i) a new discretized model for the set of feasible solutions of the BAP; (ii) the inclusion of valid inequalities, and (iii) the rolling horizon heuristic, are valid both for the heterogeneous and homogeneous cranes case and can be easily adapted to related problems.

In Section 2 we introduce the complete mixed integer model for the BACASP. For the berth allocation aspects, the model follows [3]. Then, in Section 3, we introduce a new model for berth allocation which avoids the big-M constraints of the *classical* RPF. Enhancements, mainly based on derivation of valid inequalities, are discussed in Section 4. In Section 5 the rolling horizon heuristic is introduced. In Section 6 we report the computational results that compare the different models and test the heuristic.

These testes include the study of the homogeneous quay cranes case. Finally, some conclusions and future research directions are given in Section 7.

2 The integrated quay allocation and crane assignment formulation

In this section we introduce a formulation for the BACASP. We break the formulation into the two parts: the Berth Allocation (BA) formulation and the Quay Crane Assignment and Scheduling (QCAS) formulation. This eases the presentation and allow us to discuss the reformulation of BA independently, as this is one of the main contributions of this paper.

BA formulation

In order to model the BA we use the time-space diagram. The wharf is divided into J berth sections and the time horizon is divided into M periods. There are two classical formulations based on the time-space diagram, the Relative Position Formulation (RPF) and the Position Assignment Formulation (PAF) [16]. Here we follow the RPF since the number of variables is smaller.

Consider the following sets.

$V = \{1, \dots, N\}$ is the set of ships, with indices k, ℓ,
$T = \{0, \dots, M\}$ is the time horizon, with indices i, j,
$B = \{0, \dots, J\}$ is the set of berth sections, with indices n, m.

And define the following parameters:

A_k is the arrival time of vessel k, $k \in V$,
H_k is the length of vessel k measured in number of berth sections, $k \in V$,
F safety time between the departure of a vessel and the berthing of a new one, measured in number of time units.

As in [3, 16, 31] we define the following variables:

$$x_{k\ell} = \begin{cases} 1, \text{if ship } \ell \text{ berths after ship } k \text{ had departed} \\ 0, \text{otherwise} \end{cases} \quad , k, \ell \in V,$$

$$y_{k\ell} = \begin{cases} 1, \text{if ship } \ell \text{ berths below the berth position of ship } k \\ 0, \text{otherwise} \end{cases} \quad k, \ell \in V,$$

b_k berth position of vessel k, for $k, \in V$,
t_k berthing time of vessel k, for $k, \in V$,
c_k departing (completion) time of vessel k, for $k, \in V$.

The x and y variables are binary and model the relative position of vessels. The remaining variables are integer.

The set of constraints is given by:

$$\begin{aligned}
&x_{\ell k} + x_{k\ell} + y_{\ell k} + y_{k\ell} \geq 1, && k,\ell \in V, k < \ell, && (1)\\
&x_{\ell k} + x_{k\ell} \leq 1, && k,\ell \in V, k < \ell, && (2)\\
&y_{\ell k} + y_{k\ell} \leq 1, && k,\ell \in V, k < \ell, && (3)\\
&t_\ell \geq c_k + F + (x_{k\ell} - 1)M, && k,\ell \in V, k \neq \ell, && (4)\\
&b_k \geq b_\ell + H_\ell + (y_{k\ell} - 1)J, && k,\ell \in V, k \neq \ell, && (5)\\
&t_k \geq A_k, && k \in V, && (6)\\
&b_k \leq J - H_k, && k \in V, && (7)\\
&b_k, t_k, c_k \in \mathbb{Z}_0^+, && k \in V, && (8)\\
&x_{k\ell} \in \{0,1\}, y_{k\ell} \in \{0,1\}, && k,\ell \in V, k \neq \ell. && (9)
\end{aligned}$$

Constraints (1) - (3) are the usual constraint enforcing each pair of ships not to overlap either in time or space. Constraints (4) relate the time variable of two vessels t_l, c_k, x_{kl} and (5) relate the space variable of two vessels. Constraints (4), (5) also enforce the definition of variables x_{kl}, y_{kl}. Constraints (6), (7) define the range of variables representing the time and space allocation of vessels, constraints (8) define the integer variables and constraints (9) define the binary variables.

QCAS formulation

Quay cranes are mounted along the same berth and operate on a common set of rails. This prevents quay cranes from passing each other at any time. The set of cranes is assumed to be heterogeneous with different processing rates. Several cranes can be assigned to the same ship and each crane can change its position at the beginning of each time period. For instance, a crane can be assigned to a ship and then move to another ship while the previous one is still being loaded/unloaded. All these assumptions make the model more flexible.

Let $G = \{1, \ldots, \mid G \mid\}$ be an ordered set of cranes where $g, g' \in G$ and $g < g'$ means that crane g has an operating range lower (in term of berth position) than crane g'. We assume a crane can be assigned to a vessel if the vessel is berthed within the range interval of the crane.

Let us define the following parameters:

$$\begin{aligned}
Q_k &\text{ cargo volume to be operated in vessel } k\text{, for } k, \in V;\\
P_g &\text{ processing rate of crane } g \text{ for each time period, for } g \in G;\\
[S_g, E_g] &\text{ range interval of crane } g\text{, for } g, \in G;
\end{aligned}$$

and the binary crane assignment variables,

$$z_{gk}^j = \begin{cases} 1, \text{if crane } g \text{ is assigned to ship } k \text{ in time period } j \\ 0, \text{otherwise} \end{cases}, \ g \in G, k \in V, j \in T.$$

The set of constraints is the following:

$$\sum_{k\in V} z_{gk}^{j} \leq 1, \qquad j \in T, g \in G, \tag{10}$$

$$t_k \leq j z_{gk}^{j} + (1 - z_{gk}^{j})M, \qquad j \in T, k \in V, g \in G, \tag{11}$$

$$c_k \geq (j+1) z_{gk}^{j}, \qquad \forall j \in T, k \in V, g \in G, \tag{12}$$

$$\sum_{j\in T}\sum_{g\in G} P_g z_{gk}^{j} \geq Q_k, \qquad k \in V, \tag{13}$$

$$b_k + H_k \leq E_g z_{gk}^{j} + (1 - z_{gk}^{j})J, \qquad j \in T, k \in V, g \in G, \tag{14}$$

$$b_k \geq S_g z_{gk}^{j}, \qquad j \in T, k \in V, g \in G, \tag{15}$$

$$z_{gk}^{j} + z_{g'\ell}^{j} \leq 2 - y_{k\ell}, \qquad j \in T, k, \ell \in V, g, g' \in G, g' < g, \tag{16}$$

$$\sum_{g\in G} z_{gk}^{j} \leq NC_k, \qquad k \in V, j \in T, \tag{17}$$

$$z_{gk}^{j} \in \{0,1\}, \qquad k \in V, g \in G, j \in T. \tag{18}$$

Constraints (10) ensure that in each time period each crane is operating at most one vessel. Constraints (11) guarantee that a crane can be assigned to a vessel only after its arrival, while constraints (12) ensure that if a crane was operating vessel k in time period j, then this vessel can only depart after the end of time period j ($c_k \geq j+1$). Constraints (13) guarantee that the assigned cranes to vessel k suffice to operate all the cargo of that vessel. Constraints (14) and (15) ensure that a crane can be assigned to a vessel if and only if the vessel is berthed within the range of the crane. Constraints (16) prevent cranes from passing each other at any time period. If $g' < g$ then crane g' must be assigned to a lower berth section than crane g, thus if $y_{kl} = 1$, meaning vessel k is assigned to a berth position lower than the berth position of l, cranes g and g' cannot be assigned to vessel k and l, simultaneously. Constraints (17) impose a maximum number of cranes, denoted by NC_k, assigned to vessel k and allow to consider safety distance between cranes. Finally, constraints (18) define variables z_{gk}^{j} as binary.

Objective function

The objetive is to minimize the service completion time:

$$\min \sum_{k\in V} c_k \tag{19}$$

The model (1) - (19) will be denoted by RPF (Relative Position Formulation) and the set of feasible solutions will be denoted by X_{RPF}.

This problem has many alternative optimal solutions that can be obtaining by exchanging cranes of the same type between vessels. From the practical point of view it is desirable to minimize the unnecessary crane movements. In order to minimize the crane movements without deteriorating the quality of the optimal solution one can solve a new crane schedule problem on the z variables where all variables x, y, b, t, c are fixed to their values in the the optimal solution. Since this problem was solved very efficiently for all tested instances by branch and cut using a commercial solver, we provide the model in the Appendix and omit further discussion on it.

3 A discretized reformulation for the BA

The model (1) - (9) for the BA is very weak since it is based on the big-M constraints (4) and (5). As it is well-known such constraints provide poor linear relaxation lower bounds. Here we propose a reformulation that avoids the use of those constraints by introducing new variables which result from a discretization of time, t_k, and space, b_k, variables. Such discretization technique has been successfully employed in other optimization problems to derive tighter formulations (formulations that provide better lower bounds), see for instance an application to lot-sizing problem in [12]. Within berth allocation and quay crane assignment such technique is not common.

We first consider the case of the berth aspects. As the vessels' length is known in advance this case is simpler than the case of the time aspects, where the operating time of a vessel depends on the cranes assignment and schedule.

Discretizing berth variables

Consider the new binary variables π_{kn} and σ_{kn} defined as follows:

$$\pi_{kn} = \begin{cases} 1, \text{if the first (lower) berth position of vessel } k \text{ is } n \\ 0, \text{otherwise} \end{cases}, \quad k \in V, n \in B,$$

$$\sigma_{kn} = \begin{cases} 1, \text{if berth section } n \text{ is assigned to vessel } k \\ 0, \text{otherwise} \end{cases}, \quad k \in V, n \in B.$$

Variables π_{kn} are related with variables b_k through the equations:

$$b_k = \sum_{n \in B} n\pi_{kn}, \qquad k \in V \tag{20}$$

Additionally, the following set of constraints is added:

$$\sum_{n \in B} \sigma_{kn} = H_k, \qquad k \in V, \tag{21}$$

$$\sum_{n \in B} \pi_{kn} = 1, \qquad k \in V, \tag{22}$$

$$\pi_{kn} \geq \sigma_{kn} - \sigma_{k,n-1}, \qquad k \in V, n \in B, n > 1 \tag{23}$$

$$\pi_{k1} \geq \sigma_{k1}, \qquad k \in V, \tag{24}$$

$$\pi_{kn} \leq \sigma_{kn}, \qquad k \in V, n \in B, \tag{25}$$

$$\pi_{kn} \leq 1 - \sigma_{k,n-1} \qquad k \in V, n \in B, n > 1, \tag{26}$$

$$y_{k\ell} + \sum_{m=max\{n-H_\ell+1,0\}}^{J} \pi_{km} + \pi_{\ell n} \leq 2, \qquad k, \ell \in V, k \neq \ell, n \in B, \tag{27}$$

$$\pi_{kn}, \sigma_{kn} \in \{0,1\}, \qquad k \in V, n \in B. \tag{28}$$

Constraints (21) ensure that the number of berth sections assigned to vessel k coincides with its length. Constraints (22) state that there must exits a first (lower) berth position assigned to each vessel. Constraints (23)-(26) relate the two new sets of variables. Constraints (23) ensure that

π_{kn} must be one if $\sigma_{kn} = 1$ and $\sigma_{k,n-1} = 0$, that is, if berth section n is assigned to vessel k but berth section $n-1$ is not. Conversely, constraints (25) and (26) enforce π_{kn} to be zero when either σ_{kn} is zero or $\sigma_{k,n-1}$ is one, respectively. Constraints (27) are binding only if $y_{k\ell} = 1$ since $\sum_{m=max\{m-H_\ell+1,0\}}^{J} \pi_{km} \leq 1$ and $\pi_{\ell n} \leq 1$. These constraints ensure that if the berth position of vessel k is lower than the position of vessel ℓ ($y_{k\ell} = 1$), and if the starting berthing position of vessel ℓ is n ($\pi_{\ell n} = 1$), then starting berthing position of vessel k cannot be located in the interval $[n - H_\ell + 1, J]$.

The new set of variables can also be related with the crane assignment variables through the constraints

$$z_{gk}^j \leq \sum_{n=S_g}^{E_g - H_k} \pi_{kn}, \qquad k \in V, g \in G, j \in T, \tag{29}$$

which ensure that if crane g is assigned to vessel k, ($z_{gk}^j = 1$ for some time period j), then vessel k is berthed within the range of crane g. If (29) are added, then (14) and (15) can be removed. Preliminar results showed that keeping those constraints provide better results so we keep them in addition to (29).

Discretizing time variables

Next we consider the case of time constraints. Consider the following variables:

$$\alpha_k^j = \begin{cases} 1, \text{if vessel } k \text{ starts operating in time period } j \\ 0, \text{otherwise} \end{cases} \qquad , k \in V, j \in T,$$

$$\beta_k^j = \begin{cases} 1, \text{if vessel } k \text{ is operated in time period } j \\ 0, \text{otherwise} \end{cases} \qquad , k \in V, j \in T.$$

We assume $\alpha_k^j = \beta_k^j = 0$ when $k \in V, j \in T, j < A_k$. Variables t_k are related with variables α_k^j in the following way:

$$t_k = \sum_{j \in T} j \alpha_k^j, \qquad k \in V. \tag{30}$$

The set of additional constraints is similar to the case of the space discretization.

$$c_k \geq (j+1)\beta_k^j, \qquad k \in V, j \in T, \tag{31}$$

$$z_{gk}^j \leq \beta_k^j, \qquad k \in V, j \in T, g \in G, \tag{32}$$

$$\sum_{j \in T} \alpha_k^j = 1, \qquad k \in V, \tag{33}$$

$$\alpha_k^j \geq \beta_k^j - \beta_k^{j-1}, \qquad k \in V, j \in T, j > 1, \tag{34}$$

$$\alpha_k^1 \geq \beta_k^1, \qquad k \in V, \tag{35}$$

$$\alpha_k^j \leq \beta_k^j, \qquad k \in V, j \in T, \tag{36}$$

$$\alpha_k^j \leq 1 - \beta_k^{j-1}, \qquad k \in V, j \in T, j > 1, \tag{37}$$

$$x_{kl} + \beta_k^i + \beta_l^j \leq 2, \qquad k, l \in V, k \neq l, j, i \in T, i \geq j-1, \tag{38}$$

$$\alpha_k^j \in \{0,1\}, \beta_k^j \in \{0,1\}, \qquad k \in V, j \in T. \tag{39}$$

Since the meaning of these constraints is similar to the ones given for the discretized berth allocation subproblem we explain in detail only those that are different, that is, the non-overlapping constraints (38) because the operating time of the vessel is an output of the model, therefore it is not possible to replicate the spatial non-overlapping constraints (27). Constraints (38) are binding only if $x_{k\ell} = 1$. In this case, since vessel k is served before vessel l, then if vessel l berths at time period j, vessel k cannot be served in time interval $[j-1, J]$.

We denote the model obtained by replacing constraints (4) and (5) by (20) - (39) by DRPF (Discretized Relative Position Formulation) and the set of feasible solutions by X_{DRPF}.

Observe that variables t_k, b_k could now be dropped from the model. Overall, in order to avoid the big-M constraints (4) and (5) model DRPF has more variables and constraints. The difference is mainly on the modelling of the BAP. For this subproblem the RPF has $\mathcal{O}(\max\{V^2, M, J\})$ variables and $\mathcal{O}(V^2)$ constraints while the DRPF has $\mathcal{O}(\max\{V^2, VM, VJ\})$ variables and $\mathcal{O}(V^2M^2)$ constraints.

It is worth of notice that including the discretized variables in the RPF using equations (20) and (30) will not lead to a tighter model. Only the use of the information provided by those new variables has allowed us to derive new and tighter constraints.

4 Enhancements

In this section we discuss different enhancements on the DRPF to improve the performance of the branch and cut algorithm implemented in a commercial solver using the default options. Some enhancements were not clearly effective from the computational point of view. For instance, constraints (38) are large in number and one can add them dynamically in order to reduce the size of the model. In order to add dynamically the original and weaker constraints (38) are kept to ensure feasibility of the solutions. For the tested instances this approach was not clearly effective. Next we discuss only those that were effective from the computational point of view.

Additional tightening with new variables

A first approach is to discretize the c_k variables by introducing new binary variables $\gamma_k^j, k \in V, j \in T$, that indicate whether the last time period vessel k is operated is j or not. Although this discretization is not required to eliminate big-M constraints, new inequalities can be derived using the information of the new variables.

The new variables are related with variables β_k^j through the following constraints:

$$\gamma_k^j \geq \beta_k^j - \beta_k^{j+1}, \qquad k \in V, j \in T, j < M, \tag{40}$$

$$\gamma_k^1 \geq \beta_k^M, \qquad k \in V, \tag{41}$$

$$\gamma_k^j \leq \beta_k^j, \qquad k \in V, j \in T, \tag{42}$$

$$\gamma_k^j \leq 1 - \beta_k^{j+1}, \qquad k \in V, j \in T, j < M, \tag{43}$$

$$\sum_{j \in T} \gamma_k^j = 1, \qquad k \in V, \tag{44}$$

$$\gamma_k^j \in \{0,1\}, \qquad k \in V, j \in T. \tag{45}$$

Constraints (40) - (43) have a similar meaning as (34) - (37) with the necessary changes, namely (40) forces γ_k^j to one if $\beta_k^j = 1$ and $\beta_k^{j+1} = 0$. Inequalities (41) have the same meaning but for the last time period. Inequalities (42) and (43) enforce γ_k^j to zero if either $\beta_k^j = 0$ or $\beta_k^{j+1} = 1$. Constraints (44) ensure that there must exist a final operating period assigned to each vessel. Constraints (45) state that the new variables are binary.

Using the discretized variables γ_k^j, c_k can be defined by the following equations:

$$c_k = \sum_{j \in T} (j+1)\gamma_k^j, \qquad k \in V. \tag{46}$$

Time inequalities

As stated above, the use of the information provided by the discretized variables will allow us to derive new valid inequalities. Next a new set of inequalities is derived by relating c_k with the berthing time t_k and the discretized variables β_k^j.

$$c_k \geq t_k + \sum_{j \in T} \beta_k^j \qquad k \in V, g \in G \tag{47}$$

This equation says that the departing time is equal to the starting time plus the number of periods vessel k was operated. If t_k are removed from the model, then (47) can be written in the following equivalent format.

$$c_k \geq \sum_{j \in T} j\alpha_k^j + \sum_{j \in T} \beta_k^j, \;\; k \in V, g \in G$$

Valid inequalities from an integer knapsack relaxation

A relaxation of the feasible sets X_{RPF} and X_{DRPF} can be obtained for each $k \in V$ from the knapsack constraint (13): $\sum_{j \in T} \sum_{g \in G} P_g z_{gk}^j \geq Q_k$. Let L denote the set of different crane types (each crane type corresponds to a processing rate) and consider a partition of G accordingly to that set of cranes types: $G = G_1 \cup G_2 \cup \cdots \cup G_L$. The knapsack constraint can be written as

$$\sum_{t=1}^{L} P_t Y_t \geq Q_k$$

where P_t is the rate of a crane of type t, and the integer variable $Y_t = \sum_{g \in G_t} \sum_{j \in T} z_{gk}^j$ is the number of time periods that cranes of type t are assigned to vessel k. In our case there are two types of cranes (corresponding to two distinct processing rates), thus we obtain the following 2-integer knapsack set.

$$X_K = \Big\{ (Y_1, Y_2) \in \mathbb{Z}^2 \mid P_1 Y_1 + P_2 Y_2 \geq Q_k, Y_1, Y_2 \geq 0 \Big\}.$$

Valid inequalities for X_K can then be converted into valid inequalities for X_{RPF}. The strongest valid inequalities are the facet-defining inequalities which, for such sets, can be derived in polynomial time, see [1]. The coefficients of these inequalities are obtained, in general, using the euclidian algorithm. The number of facet-defining inequalities is polynomial.

Example 4.1. *Suppose* $Q_k = 2800$ *tones, for some* $k \in V$, *and* $P_1 = 250$ *tones/hour and* $P_2 = 320$ *tones/hour. So* $X_K = \{(Y_1, Y_2) \in \mathbb{Z}^2 \mid 250Y_1 + 320Y_2 \geq 2800, Y_1, Y_2 \geq 0\}$. *The segment line* $250Y_1 + 320Y_2 = 2800$ *is drawn by a dashed line in Figure 2. As it is shown in Figure 2, conv(*X_K*) has three non-trivial facet-defining inequalities:* $Y_1 + 2Y_2 \geq 12$, $7Y_1 + 9Y_2 \geq 79$, *and* $Y_1 + Y_2 \geq 9$. *These facets are easily derived from the extreme points in conv(*X_K*). The extreme points are computed as follows. Two extreme points are trivially obtained setting one of the coordinates to zero:* $(0, \lceil\frac{2800}{320}\rceil) = (0, 9)$ *and* $(\lceil\frac{2800}{250}\rceil, 0) = (12, 0)$. *Starting from each one of these points a sequence of extreme points is obtained using vectors* $(p_i, -q_i)$ *and* $(-u_j, v_j)$. *These vectors are obtained from a variant of the Euclidean algorithm applied to the integers 250 and 320 and satisfying the following approximation* $p_1/q_1 < p_2/q_2 < \cdots < p_k/q_k = 320/250 = u_r/v_r < \cdots < u_1/v_1$. *In the case of the example the* $(p, -q)$ *vectors obtained are* $(1, -1), (5, -4), (14, -11), (23, -18), (32, -25)$ *and the* $(-u, v)$ *vectors are* $(-2, 1), (-3, 2), (-4, 3), (-9, 7), (-32, 25)$. *Starting from* $(0, 9)$, *vectors* $(p, -q)$ *are tested in sequence to check whether another extreme point is obtained. In this case only vector* $(1, -1)$ *leads to the extreme points* $(1, 8)$. *Starting from the point* $(12, 0)$ *and using vectors* $(-u, v)$ *we obtain the other extreme point* $(10, 1)$ *with vector* $(-2, 1)$. *Then, starting from* $(10, 1)$, *we obtain the extreme point* $(1, 8)$ *again using the vector* $(-9, 7)$. *The vectors* $(1, -1), (-2, 1), (-9, 7)$ *originate the coefficients of the three non-trivial facet-defining inequalities. For other instances, the number of facets can be greater or lower than the case we illustrate here. That number depends on the coefficients of the knapsack inequalities. The possible facet-defining coefficients are the* p, q *and the* u, v *values.*

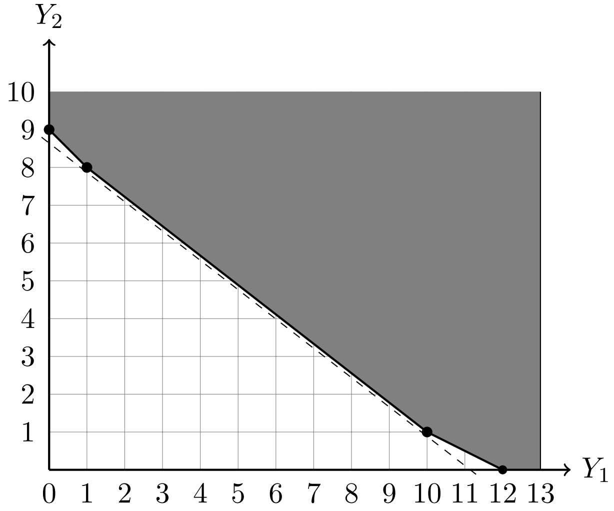

Figure 2: Facet-defining inequalities for conv($\{(Y_1, Y_2) \in \mathbb{Z}^2_+ \mid 25Y_1 + 32Y_2 \geq 280\}$).

Knapsack inequalities can be added *a priori* to the models or added dynamically, that is, an inequality is added when the fractional solution violates such inequality. As the number of inequalities is small we opted by include them *a priori.*

New inequalities can derived by aggregating the cargo of a set of vessels. Let $S \subseteq V$. Then the following 2-integer knapsack constraint is obtained $P_1Y_1(S) + P_2Y_2(S) \geq Q(S)$ where $Q(S) = \sum_{k \in S} Q_k$ and $Y_t(S) = \sum_{g \in G_t} \sum_{j \in T} \sum_{k \in S} z^j_{gk}, t \in \{1, 2\}$, and new inequalities can be derived.

The facet-defining inequalities for the 2-integer knapsack sets can also be extended to the case where more than two types of cranes are considered by lifting, see [2].

Time-windows

If time-windows are not given in advance, constructing tight time-windows for the service of each vessel allow us to handle with large size problems. In particular the number of z variables can be kept under control.

Here we consider time-windows for the start of service time $A_k \leq t_k \leq T_k$ and for the end of service time $A_k + PK_k \leq c_k \leq C_k$. Parameter PK_k is the minimum time required to serve vessel k and can be easily computed. Given an upper bound $\overline{z}$ on the optimal value, the value T_k (resp. C_k) can be computed by providing an lower bound $\underline{z}_k^T$ for the case $t_k > T_k$ (resp. $\underline{z}_k^C$ for the case $c_k > C_k$) such that $\underline{z}_k^T \geq \overline{z}$ (resp. $\underline{z}_k^C \geq \overline{z}$). A good feasible solution can be easily obtained using the heuristic given in Section 5. The lower bound can be obtained by solving a relaxation of the restricted set with $t_k \geq T_k + 1$ (resp. $c_k \geq C_k + 1$). A weaker but easily computable bound can be derived for each vessel k as follows. Let $\underline{c}_\ell = A_\ell + PK_\ell$ denote the lower completion time for vessel ℓ. Then take $\underline{z}_k^T = \sum_{\ell \in V} \underline{c}_\ell + (T_k - A_k)$ and $\underline{z}_k^C = \sum_{\ell \in V} \underline{c}_\ell + (C_k - \underline{c}_k)$.

Branching priorities

It is well known that branching decisions can have a great impact on the performance of the branch and cut algorithm. Base on preliminar tests we chose to give branching priority to variables x_{kl} and y_{kl}.

5 Rolling Horizon Heuristic

The enhanced formulation will allow to solve small/medium practical problems but will certainly fail to solve large size instances. We also have observed from preliminar computational tests that for some instances the solver was not able to find any feasible solution quickly. In order to handle with these two issues we propose a Rolling Horizon Heuristic (RHH) where each subproblem is solved by a branch and cut based on the enhanced formulation.

The main idea of the RHH is to split the planning horizon into smaller sub-horizons, and then repeatedly solve tractable subproblems for each sub-horizon. Such heuristic scheme fits well the dynamic problem where the decisions concerning the last vessels arriving have little connection with those decisions for the first arriving vessels.

At each iteration k of the RHH a new subproblem is chosen. This subproblem is defined by the inclusion of a new set of vessels. The number of vessels included in each iteration is fixed and denoted by nv. Therefore, the time horizon of these subproblems is not constant, but it depends on the last arrival time of the new vessels being considered. When moving from iteration k to iteration $k + 1$ the next nv arriving vessels are considered. The berth locations of the vessels considered in the previous k iterations are freezed. The time variables, and the cranes schedule and assignment are freezed in part of the previous time horizon and kept free during the last ut time units of that horizon. Hence, in iteration $k + 1$ a restricted problem is solved where the berth decision are considered only for the new nv vessels and part of the scheduled decisions are fixed. See Figure 3. The details of the RHH are given in Algorithm 1.

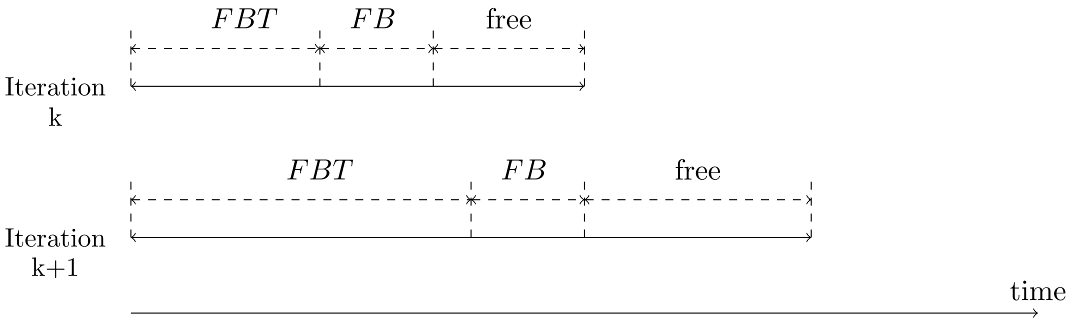

Figure 3: The rolling horizon heuristic, where *FBT* means freeze berth and time variables, *FB* means freeze berth variables, and *free* means all the variables are free.

Algorithm 1 Rolling horizon heuristic

```
$nv \leftarrow$ number of new vessels to consider in each iteration
$nt \leftarrow$ number of time periods to consider for each subproblem after the arrival time of the last vessel
$ut \leftarrow$ number of time periods to consider for each subproblem before the arrival time of the first new vessel
$LT = A_{nv} + nt \leftarrow$ limited time horizon of the first subproblem
$U \leftarrow$ number of iterations to cover the planning horizon
solve the first subproblem defined for $nv$ vessels and with time horizon $LT$
for all $k = 2 \dots U$ do
    Fix $x, y, b, \pi, \sigma$ variables for vessels $1, \dots, (k-1) * nv$
    Fix $t, z, \alpha, \beta$ for vessels $1, \dots, (k-1) * nv$ and time periods $1, \dots, LT - ut$
    $LT \leftarrow A_{k*nv} + nt$
    Solve the restricted mixed integer problem for $k * nv$ vessels and $LT$ time periods
end for
```

In each iteration of the RHH the subproblem is solved using the branch and cut based on the DRPF with the enhancements described in Section 4.

Considering $nv = N$ the complete model is solved in one iteration. For the case $nv = 1$ the RHH resembles a greedy algorithm where at each iteration the berth position of a vessel is fixed but the time and quay crane assignment decisions are not fixed.

6 Computational results

In this section we report the computational results obtained on 21 instances. The formulations were written in Mosel and implemented in Xpress-IVE Version 1.24.06, with 64 bits. All the tests were run on a computer with a CPU Intel(R) Core i7, with 16GB RAM and using the Xpress Optimizer Version 27.01.02 with the default options.

6.1 Test Cases

We consider two sets of instances based on real data extracted from a practical problem occurring at a multi-use terminal of a port devoted to short sea shipping operations. The terminal is mainly used for bulk cargo operations of products such as steel, cement, cereals. The physical aspects as the quays length, the number of cranes, their relative position and efficiency are obtained from real data. All the instances consider 7 cranes and 34 berth sections. Six cranes have the same capacity and another crane has a larger capacity. As the processing rates cannot be obtained directly from the capacities, we used the historical data to estimate the processing rates. For the lower capacity cranes the rate estimated is 263.6 ton/hr and for the larger capacity crane the estimated rate is 319 ton/hr. The time period was set to one hour and each berth section is 25 meters. The vessels' size and operated cargo amounts are also based on real data. The vessel sizes vary between 6 to 9 berth sections (which includes the safety distances) and the cargo varies between 3,000 tons to 8,800 tons. Other details on the practical problem can be found in [33].

The first test set includes the 9 instances and was built to test and tune the mathematical models. These instances are made harder than the real instances by shortening the interval between arrival times of the vessels. Table 1 gives the details of these instances. Also, preliminar tests have shown that the problem becomes much harder when several ships arrive at the same time (which is not common in the real situation) creating a congestion. Instances I3, I5, I7, I9 were created assuming that the first four vessels arrive at the beginning of time horizon. A limit of 4 cranes ($NC_k = 4$) was assumed.

Table 1: Small test instances based on real data.

Instance	$\mid V \mid$	M
I1	7	50
I2	8	55
I3	8	55
I4	10	60
I5	10	60
I6	12	65
I7	12	65
I8	15	65
I9	15	65

The second test set, has 12 instances and it is based on large size instances with a time horizon of one week (168 hours). Instances are labeled as nt, where n indicates the number of vessels and t is the version. For each value of n three instances are generated. So t can assume values a, b, c. For this set of instances, the arrival times are randomly generated in the interval $[0, 150]$; the ship lengths are randomly generated in the set $\{6, 7, 8, 9\}$.

6.2 Computational experiments and results for the hard instances

In this section we report some of the main results obtained with the conducted computational experimentation on the first set of instances. These tests allowed us to compare the models and derive solutions approaches that will be used in the second set (larger size) of instances.

In Table 2 we provide the optimum value for each instance (column Opt); the linear relaxation value (LR) and the linear gap (Gap) obtained using models RPF (columns RPF), DRPF (Columns DRPF), and the enhanced DRPF (columns DRPF+), which includes the additional variables γ_k^j and the corresponding constrains, inequalities (47), the knapsack inequalities derived for each vessel and for the set of all vessels. We can observe that the RPF model based on big-M constraints is very weak providing poor lower bounds. The two discretized models are much tighter than the RPF model and the linear gap of the enhanced model is, in average, 2.3% lower than the linear gap of DRPF.

Table 2: Linear relaxation bounds using the different models.

Instance	Opt	RPF		DRPF		DRPF+	
		LR	Gap	LR	Gap	LR	Gap
I1	125	3.8	97.0	112.6	9.9	116.7	6.6
I2	221	7.1	96.8	202.95	8.2	207.4	6.2
I3	128	7.1	94.4	103.89	18.8	108.2	15.5
I4	327	10.6	96.8	301	8.0	307.3	6.0
I5	202	10.0	95.1	169.4	16.1	174.8	13.5
I6	439	10.7	97.6	410.2	6.8	417.3	5.2
I7	281	12.2	95.6	244	13.2	251	10.7
I8	559	14.9	97.3	523.6	6.3	532	4.8
I9	504	17.9	96.5	458.1	9.1	466.4	7.5
Average			96.3		10.7		8.4

In Table 3 we provide the results (best feasible solution found, denoted by BFS, and the running time in seconds, denoted by Time) obtained with the RHH for $nv = 1$, $nv = 3$ and $nv = 5$, which are denoted by RHH1, RHH3 and RHH5, respectively. When the number of vessels nv is increased, the running times also increase since each subproblem tends to be more difficult to solve. The running times for RHH1 are always below one minute, while the running times for the RHH3 depend on the problem difficulty. Considering the best feasible solution found, RHH5 was able to find the optimal solution in 8 instances and provided a better solution than RHH1 in 5 instances. However, RHH1 determines all the optimal solutions, except for instance I2, when we restrict the analysis to those instances that have not been made hard by creating a congestion artificially. As conclusion, RHH1 is fast and provides very good lower bounds in general, while RHH5 is more time consuming than RHH1 but may lead to better solutions when there is congestion at port.

Next, in Table 4 we compare the models RPF and the enhanced model DRPF with the branch priorities and with the inclusion of the upper bound provided by the heuristic RHH5 (denoted by DRPF++). Columns BFS, BLB, Gap, Time, represent the Best Feasible Solution, Best Lower Bound, linear Gap, and running time, respectively. The running time was limited to one hour. BFS, BLB and Gap are the values obtained at the end of the running time. The running times for DRPF++ do not include the running time of heuristic RHH5. For DRPF++ we used a time window of 24 hours for each vessel (allowing the start of service to be delayed at most by one day, and the end-of-service to occur at most one day after the earliest possible end-of-service time $\underline{c}_k$). For all the 9 tested instances the interval considered suffices to prove optimality. An * means no feasible solution was found within one hour.

We can see that using RPF none of the instances were solved, although in two of them the optimal solution was obtained. The lower bounds at the end of one hour are very low.

Table 3: Rolling Horizon Heuristic with $nv = 1$, $nv = 3$ and $nv = 5$.

Instance	RHH1		RHH3		RHH5	
	BFS	Time	BFS	Time	BFS	Time
I1	125	1	125	18	125	12
I2	222	3	221	19	221	61
I3	130	14	131	87	128	1126
I4	327	14	327	43	327	78
I5	203	23	202	153	203	1179
I6	440	29	439	50	439	230
I7	284	36	283	149	281	904
I8	559	51	560	38	559	223
I9	506	54	506	173	504	433

Table 4: Bounds from tested models RPF and $DRPF++$.

Instance	RPF				DRPF++			
	BFS	BLB	Gap	Time	BFS	BLB	Gap	Time
I1	125	115	8.0	3600	125	125	0	7
I2	221	140	36.7	3600	221	221	0	1
I3	152	54	64.5	3600	128	128	0	22
I4	353	147	58.4	3600	327	327	0	11
I5	355	91	74.4	3600	202	202	0	285
I6	497	156	68.6	3600	439	439	0	26
I7	434	88	79.7	3600	281	281	0	434
I8	*	148	*	3600	559	559	0	713
I9	*	145	*	3600	504	504	0	351

Conversely, the exact approach that uses all the contributions in the paper, that is, the branch and cut based on the enhanced DRPF feeded up with the RHH solution, provides the optimal solution to all the tested instances within acceptable running times.

6.3 Computational experiments for the larger size test instances

Currently, from a practical point of view, extending this approach to solve larger time horizons than those tested previously is not meaningful since no accurate information on the arrival vessels would be available. However, in order to test scalability of our approach and to account for possible future scenarios where such information will be available, we test our approaches on the set of large size instances. Since RHH5 outperformed RHH1 only for the artificially hard instances and since RHH1 runs much faster than RHH5, we used in these experiments RHH1. We first ran RHH1 and then ran DRPF++ with a time limit of one hour, using as cutoff value the value of the feasible solution obtained with RHH1. The results are reported in Table 5. For the RHH1 we provide the value of the best feasible solution (BFS) found, the linear gap (Gap) in relation to the best known lower bound and the running time (Time) in seconds. For the exact approach DRPF++ we provide only the best lower bound found (BLB) and the running time (Time) since the best feasible solution was never improved in relation to that obtained with RHH1 (so the gap is the same as the one for RHH1). Hence, the final gap is the same as the one reported for RHH1 and therefore is omitted. When the running time is lower than 1 hour the optimal solution was found (in all those cases the RHH1 solution was proved to be optimal). We can observe that the RHH1 could provide very good solutions for all the tested instances with running times ranging between half of a minute to 10 and half minutes. All but one instances up to 20 vessels were solved to optimality. For the largest instances the solution obtained has a proved gap lower than 2%.

Table 5: Heuristic RHH1 for large scale instances.

	RHH1			DRPF++	
Instance	BFS	Gap	Time	BLB	Time
15a	1295	0.0	28	1295	6
15b	1331	0.0	32	1331	7
15c	1149	0.0	26	1149	5
20a	1680	0.0	58	1680	11
20b	1654	0.1	98	1653	3600
20c	1492	0.0	63	1492	33
30a	2412	0.2	231	2408	3600
30b	2325	0.5	250	2313	3600
30c	2349	0.3	193	2343	3600
40a	2942	0.4	630	2931	3600
40b	3312	1.8	605	3253	3600
40c	3199	1.3	508	3156	3600

6.4 Computational experiments for the homogeneous cranes case

As most of the reported applications in the literature consider a set of homogeneous cranes, here we assume all the cranes are similar in order to tests the formulations and the valid inequalities for this particular case. The tests are similar to those conducted for the heterogeneous case but assuming the cranes have the same processing rate as the lower one. We denote $P_g = P, \forall g \in G$. In the homogeneous case, the knapsack inequalities are reduced to the following family of cover inequalities:

$$Y(G) \geq \left\lceil \frac{Q(S)}{P} \right\rceil, \forall S \subseteq V,$$

where $Y(G) = \sum_{g \in G} \sum_{j \in T} \sum_{k \in S} z^j_{gk}$, and $Q(S) = \sum_{k \in S} Q_k$.

For the homogeneous cranes case we test the RPF model, the heuristic RHH1 and the improved formulation DRPF++ with the upper bound obtained with the RHH1 heuristic. The results are presented in Table 6. Column *Opt* gives the optimal value, columns *LR* give the linear relaxation value of the corresponding formulation, *IGap* gives the initial gap (computed with the values in columns *Opt* and *LR*), columns *BLB* and *BFS* give the best lower bound and the best upper bounds at the end of the running time, respectively, and columns *Time* give the running time in seconds. A time limit of one hour was considered.

Table 6: Upper and lower bounds using different approaches for the homogeneous cranes case.

Instance	Opt			RPF			RHH1		DRPF++		
		LR	IGap	BLB	BFS	Time	BFS	Time	LR	IGap	Time
I1	125	3.8	97	125	125	1571	125	6	115.2	7.8	10
I2	223	7.1	96.8	152	224	3600	223	4	206.8	7.3	8
I3	129	7.1	94.5	107	135	3600	130	17	107.6	16.6	172
I4	328	10.6	96.8	210	347	3600	328	7	306.9	6.4	29
I5	204	10.0	95.1	119	259	3600	205	24	173.6	14.9	301
I6	441	10.7	97.6	216	459	3600	442	20	416.3	5.6	144
I7	283	12.2	95.7	129.8	414	3600	285	27	251	11.3	702
I8	562	14.9	97.3	212.6	576	3600	562	25	531.1	5.5	234
I9	506	17.9	96.5	202.2	542	3600	514	27	465.4	8	1555
Average			96.4							9.3	

We can observe that the results obtained for the homogeneous cranes case are, in general, similar to those obtained to the general heterogeneous one. The *RPF* model based on big-M constraints provides poor lower bounds again. In this case, with the corresponding formulation, it was possible to solve the easiest instance to optimality. However, for the remaining ones, the lower bounds after one hour are very poor (and below the value of the linear relaxation of the enhanced formulation) and the upper bounds are away from the optimal value. The $RHH1$ runs fast, always below 30 seconds, and obtained the optimal solution in four instances, and very good solutions in the remaining cases. Again, with the approach $DRPF++$ (that uses the discretized model tightened with valid inequalities and feeded up with good upper bounds from the heuristic) all the instances could be solved to optimality in less than half and hour. These results show that all the techniques introduced in the paper (stronger formulations, valid inequalities and hybrid heuristics) also have a good performance in the particular case of homogeneous cranes.

7 Conclusion

A flexible model is presented for a real problem integrating berth assignment and quay crane assignment. A heterogenous set of cranes is considered. A new reformulation based on a time and space discretization is introduced and several enhancements are discussed. The reformulation allows to avoid the classical big-M constraints for the Berth Allocation Problem. In order to handle with large size instances, and in order to improve the efficiency of the exact method by feeding it with a good initial upper bound, a rolling horizon heuristic is proposed. The exact approach that is based on a branch and cut algorithm that combines all the contributions: the discretized formulation, the inclusion of valid inequalities and the use of the upper bound given by the heuristic, was very effective in solving all the practical instances within one hour. Computational tests also show that the proposed heuristic can be used to provide good quality solutions to large size instances, cautioning for possible increase of the instances complexity in the future.

From the practical point view, the proposed approach provides the port management with a tool to support the berth assignment planing of the incoming ships and to suggest a schedule for the quay cranes. Although this approach is able to handle current and future larger instances, it has a few shortcomings as it doesn't consider some practical relevant aspects. One such aspect is the storage space at the yard. With the increase of the port activity, the limited storage space is becoming more relevant as it restricts other port activities. Hence, the coordination of the quay load/unload operations with the cargo storage management is needed. The other relevant aspect not considered here is the uncertainty associated with some events such as vessels arrival times, processing times, etc. For future research we aim to integrate uncertainty and storage management with the berth assignment and quay crane scheduling.

Acknowledgements

This research was carried out with financial support from project UID/MAT/04106/2013 funded by the Portuguese Foundation for Science and Technology.

We thank the two anonymous referees for their helpful suggestions.

Appendix: minimizing the number of crane changes

The goal in this problem is to minimize the number of crane movements while keeping the same overall completion time of a given solution. We consider the following variables:

$$w_{gk}^{j} = \begin{cases} 1, \text{if crane } g \text{ starts operating vessel } k \text{ in time period } j \\ 0, \text{otherwise} \end{cases} \qquad , g \in G, k \in V, j \in T.$$

To minimize the number of movements is equivalente to minimize the number of variables w_{gk}^{j} equal to 1. The objective function becomes as follows.

$$min \sum_{g \in G} \sum_{k \in V} \sum_{j \in T} w_{gk}^{j} \tag{48}$$

Given a solution to the RPF, variables $x_{k\ell}$, $y_{k\ell}$, b_k, t_k and c_k are fixed to their value in the crane movements minimization problem. Hence only constraints for the QCASP are considered.

In addition to constraints (10) - (18), the following constraints are considered:

$$z^{j}_{gk} - z^{j-1}_{gk} \leq w^{j}_{gk}, \qquad g \in G, k \in V, j \in T, j > 1, \tag{49}$$

$$w^{j}_{gk} \leq z^{j}_{gk}, \qquad g \in G, k \in V, j \in T, \tag{50}$$

$$w^{j}_{gk} \leq 1 - z^{j-1}_{gk}, \qquad g \in G, k \in V, j \in T, j > 1, \tag{51}$$

$$w^{j}_{gk} \in \{0, 1\}, \qquad g \in G, k \in V, j \in T. \tag{52}$$

Constraints (49) state that if crane g does not operate vessel k at time period $j-1$ and has not started to operate in time period j, then variable z^{j}_{gk} must be zero. Constraints (50) relate variables z^{j}_{gk} and w^{j}_{gk} and guarantee that if crane g starts to operate vessel k in time period j, then z^{j}_{gk} must be 1. Constraints (51) state that if crane g starts to operate in time period j, then it cannot have operated in time period $j-1$ and therefore $z^{j-1}_{gk} = 0$. Finally, constraints (52) state that variables w^{j}_{gk} are binary.

References

[1] Agra, A., Constantino, M.. Description of 2-integer continuous knapsack polyhedra. *Discrete Optimization* 3 (2), 95–110, (2006).

[2] Agra, A., Constantino, M.. *Lifting two-integer knapsack inequalities.* Mathematical Programming, 119, 115–154, (2007).

[3] Ak, A.. *Berth and quay crane scheduling: Problems, models and solution methods.* PhD Thesis, Georgia Institute of Technology, (2008).

[4] Al-Dhaheri, N., Diabat, A.. *The Quay Crane Scheduling Problem.* Journal of Manufacturing Systems, 36, 87–94, (2015).

[5] Beens, M.A., Ursavas, E., *Scheduling cranes at an indented berth.* European Journal of Operational Research, 253, 298–313, (2016).

[6] Bierwirth C., Meisel F.. *A survey of berth allocation and quay crane scheduling problems in container terminals.* European Journal of Operational Research, 202, 615–627, (2010).

[7] Bierwirth, C., Meisel, F.. *A follow-up survey of berth allocation and quay crane scheduling problems in container terminals.* European Journal of Operational Research, 244, 675–689, (2015).

[8] Blazewicz, J., E Cheng, T.C., Machowiak, M., Oguz, C.. *Berth and quay crane allocation: a moldable task scheduling model.* J. Oper. Res. Soc., 1189–1197, (2010).

[9] Boysen, N., Briskorn, D., Meisel, F.. *A generalized classification scheme for crane scheduling with interference.* European Journal of Operational Research, 258, 343–357, (2017).

[10] Chang, D., Jiang, Z., Yan, W., He, J.. *Integrating berth allocation and quay crane assignments.* Transportation Research Part E, 46, 975–990, (2010).

[11] Chen, J. H., Lee, D.-H., Cao, J. X.. *A combinatorial benders cuts algorithm for the quayside operation problem at container terminals.* Transportation Research Part E: Logist. Transp. Rev. 48, 266–275, (2012).

[12] Constantino, M., Gouveia, L.. *Reformulation by discretization: application to Economic Lot Sizing.* Operations Research Letters, 35, 645–650, (2007)

[13] Daganzo, C. F.. *The crane scheduling problem.* Transportation Research Part B, 23, 159–175, (1998).

[14] Diabat, A., Theodorou, E.. *An integrated quay crane assignment and scheduling problem.* Computers and Industrial Engineering, 73, 115–123, (2014).

[15] Guan, Y., Yan, K.H., Zhou Z.. *The crane scheduling problem: models and solution approaches.* Annals of Operations Research, 203, 119–139, (2010).

[16] Guan, Y., Cheung R. K.. *The berth allocation problem: models and solution methods* OR Spectrum, 26, 75–92, (2004).

[17] Han, X., Lu, Z., Xi, L.. *A proactive approach for simultaneous berth and quay crane scheduling problem with stochastic arrival and handling time.* European Journal of Operational Research, 207, 1327–1340, (2010).

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

[19] Imai A., Sun X., Nishimura E., Papadimitriou S.. *Berth allocation in a container port: using a continuous location approach.* Transportation Research Part B, 39, 199–221, (2005).

[20] Imai, A., Chen, H.C., Nishimura E., Papadimitriou S.. *The simultaneous berth and quay crane allocation problem.* Transportation Research Part E, 44, 900–920, (2008).

[21] Iris C., Pacino D., Ropke S., Larsen A.. *Integrated Berth Allocation and Quay Crane Assignment Problem: Set partitioning models and computational results.* Transportation Research Part E, 81, 75–97, (2015).

[22] Gharehgozli, A. H., Yu, Y., Zhang, X., de Koster, R.. *Polynomial Time Algorithms to Minimize Total Travel Time in a Two-Depot Automated Storage/Retrieval System.* Transportation Science, doi: 10.1287/trsc.2014.0562.

[23] Gharehgozli, A. H., Laporte, G., Yu, Y., de Koster, R.. *Scheduling Twin Yard Cranes in a Container Block.* Transportation Science, 49, 686–705, (2015).

[24] Giallombardo, G., Moccia, L., Salani, M., Vacca, I.. *Modeling and solving the tactical berth allocation problem.* Transp. Res. Part B: Methodol. 44, 232–245, (2010).

[25] Kim, K. H., Park, Y. M.. *A crane scheduling method for port container terminals.* European Journal of Operational Research, 156, 752–768, (2004).

[26] Liang, C., Huang, Y., Yang, Y.. *A quay crane dynamic scheduling problem by hybrid evolutionary algorithm for berth allocation planning.* Computers & Industrial Engineering, 56, 1021–1028, (2009).

[27] Lim, A.. *The berth planning problem.* Operation Research Letters, 22, 105–110, (1998).

[28] Lim, A., Rodrigues B., Xiao, F., Zhu, Y.. *Crane scheduling with spatial constraints.* Naval Research Logistics, 51, 386–406, (2004).

[29] Liu, J., Wan, Y.-W., Wang, L.. *Quay Crane Scheduling at Container Terminals To Minimize the Maximum Relative Tardiness of Vessel Departures.* Naval Research Logistics, 53, 60–74, (2006).

[30] Mauri, G., Ribeiro, G., Lorena, L., Laporte, G.. *An adaptive large neighborhood search for the discrete and continuous Berth allocation problem* Computers & Operations Research, 70, 140–154, (2016).

[31] Meisel, F., Bierwirth, C.. *Heuristics for the integration of crane productivity in the berth allocation problem.* Transportation Research Part E, 45, 196–209, (2009).

[32] Moccia, L., Cordeau, J., Gaudioso, M., Laporte, G.. *A branch-and-cut algorithm for the quay crane scheduling problem in a container terminal.* Naval Research Logistics, 53, 45–59, (2006).

[33] Oliveira, M. S.. *Porto de Aveiro: Modelos de afetação de recursos aos navios.* University of Aveiro, http://hdl.handle.net/10773/16797, (2014).

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

[35] Peterkofsky, R. I., Daganzo, C. F.. *A branch and bound solution method for the crane scheduling problem.* Transportation Research Part B, 24, 159–172, (1990).

[36] Raa, B., Dullaert, W., Van Schaeren, R.. *An enriched model for the integrated berth allocation and quay crane assignment problem.* Expert Systems with Applications, 38, 14136–14147, (2011).

[37] UNCTAD. Review of maritime transport, 2014. United Nations, New York and Geneva, (2014).

[38] Sampaio, A., Urrutia, S., Oppen, J.. *A Decomposition Approach to Solve The Quay Crane Scheduling Problem.* arXiv.org, https://arxiv.org/abs/1604.00527, (2016).

[39] Song, L., Cherrett, T., Guan, W.. *Study on berth planning problem in a container seaport: Using an integrated programming approach.* Computers & Industrial Engineering, 62, 119–128, (2012).

[40] Theodorou, E., Diabat, A.. *A joint quay crane assignment and scheduling problem: formulation, solution algorithm and computational results.* Optimization Letters, 9, 799-817, (2015).

[41] Turkogullari, Y. B., Taskin, Z.C., Aras, N., Altinel, I. K.. *Optimal berth allocation and time-invariant quay crane assignment in container terminals.* European Journal of Operational Research, 235, 88–101, (2014).

[42] Ursavas, E., Zhu, S.X.. *Optimal policies for the berth allocation problem under stochastic nature.* European Journal of Operational Research, 255, 380–387, (2016).

[43] Vacca, I., Salani, M., Bierlaire, M.. *An exact algorithm for the integrated planning of berth allocation and quay crane assignment.* Transportation Science, 47, 148–161, (2013).

[44] Zhang, C., Zheng, L., Zhang, Z., Shi, L., Armstrong, A. J.. *The allocation of berths and quay cranes by using a sub-gradient optimization technique.* Computers and Industrial Engineering, 58, 40–50, (2010).

[45] Zampelli, S., Vergados, Y., Van Schaeren, R., Dulleart, W., Birger, R.. *The berth allocation and quay crane assignment problem using a CP approach.* Lecture Notes in Computer Science, 8124, 880–896, (2013).

[46] Zhen, L.. *Tactical berth allocation under uncertainty.* European Journal of Operational Research, 247 , 928–944, (2015).

[47] Zhen, L., Lee, L., Chew, E.. *A decision model for berth allocation under uncertainty.* European Journal of Operational Research, 212, 54–68, (2011).

[48] Yang, C., Wang, X., Li, Z.. *An optimization approach for coupling problem of berth allocation and quay crane assignment in container terminal.* Computers & Industrial Engineering, 63, 243–253, (2012).