

## Research Article

# Berth Allocation Problem with Quay Crane Assignment for Container Terminals Based on Rolling-Horizon Strategy

Ling Xiao<sup>1</sup> and Zhi-Hua Hu<sup>2</sup>

<sup>1</sup> Logistics Research Center, Shanghai Maritime University, Shanghai 200135, China

<sup>2</sup> School of Economics and Management, Tongji University, Shanghai 200092, China

Correspondence should be addressed to Zhi-Hua Hu; [zhhu@shmtu.edu.cn](mailto:zhhu@shmtu.edu.cn)

Received 13 January 2014; Accepted 27 March 2014; Published 7 May 2014

Academic Editor: Hu Shao

Copyright © 2014 L. Xiao and Z.-H. Hu. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

In order to solve the large-scale integral dynamic scheduling of continuous berths and quay cranes problem, a method based on rolling-horizon strategy is proposed. A multiobjective optimization model that is established minimizes the total penalty costs considering vessels' deviations to their preferred berthing positions, delayed times for berthing comparing to their estimated arrival times, and delayed times for departure comparing to their estimated departure times. Then, the scheduling process was divided into a set of continual scheduling interval according to the dynamic arrival sequences. Meanwhile, rolling-horizon strategies for setting rolling and frozen windows and the parameter updating strategy are designed. The input parameters of the model in the next rolling window are updated according to the optimal results of each time window which have been obtained. The model is solved by choosing appropriate rolling and freezing window lengths that represents the numbers of adjacent vessels in the sequence of calling vessels. The holistic optimal solution is obtained by gradually rolling and combining the results of each window. Finally, a case study indicated that the rolling schedule can solve large-scale scheduling problems, and the efficiency of the proposed approach relates to the size of rolling window, freeze ship quantity, and rolling frequency.

## 1. Introduction

Along with the rapid development of large-scale container terminals and large container ships, the container handling efficiency plays an increasingly important role in competition environment of container terminals, which increases the difficulty and complexity of large-scale integral dynamic scheduling of berths and quay cranes. Berths and quay cranes are critical resources in container terminals. So, the berth allocation problem (BAP) and quay crane assignment problem (QCAP) are important issues which improve the whole efficiency of a container terminal. A reasonable arrangement of berth and quay cranes reduces the stay time of vessels at berths as well as the operational cost and improves the turnover rate of vessels at then container terminal and profit and service ability of the container terminal. For most large-scale container terminals in China, many methods mentioned in the literature cannot be applied in practice because of unacceptable computational performance, even for solving

a 24-hour berth schedule. While rolling-horizon strategy is widely used in real operational plan of container terminals, especially suitable for dynamic continuous berth allocation problem (DCBAP) which conforms well with real situations. Rolling-horizon scheduling method is a decomposition strategy for large-scale problem, which means to solve original problem by decomposing it into small subproblems. It is an effective method to solve large-scale complex problems [1–3]. Berthing plan is a kind of multiprocessor task scheduling [4, 5], so the rolling strategy can be used for reference.

Compared with the existing literature, including that of [6], which formulated the continuous CBAP, the main contributions of this paper are elucidated as follows. First, this paper formulated two constraints including the adjustment of the amount of quay cranes allocated to a vessel and the limitation of total number. A mixed-integer multiobjective optimization model was established which minimizes the total penalty cost that consists of deviations of the solution to the preferred berthing position and estimated berthing and

departure times. Then, the scheduling process was divided into a set of continual scheduling intervals according to the vessels' arrival sequences. Third, this paper developed the rolling-horizon strategies for setting rolling and frozen windows and updating methods for the present window based on past windows. Through large-scale numerical case study, it is indicated that the model and rolling-horizon strategy proposed in our paper can solve large-scale scheduling problems and that the relationship between the efficiency and the amount of vessels is almost linear; it can be applied to decision support system for DCBAP large-scale container terminals.

## 2. Related Studies

BAP and QCAP are two important issues in operations optimization of container terminals. According to whether mooring points are continuous, BAPs can be divided into discrete BAP (DBAP) and continuous BAP (CBAP) [7]. In DBAP, the available berths are regarded as a limited set of mooring points. Vessels cannot berth across more than one berth [8], while CBAP takes berths as a continuous mooring space. Vessels can moor and be handled anywhere in this space, not subjected to the limitations of berthing positions [9]. According to the vessel's arrival time, BAPs can be classified into static BAP and dynamic BAP. Static BAP means all the vessels have already arrived in port before berth allocation [10]. While in Dynamic BAP, some vessels are yet in terminal at the beginning of the berth allocation, but others will arrive at a certain time during the allocation [11]. The dynamic arrival sequence of the vessels provides the foundation for the rolling scheduling in this work.

According to the literature, the CBAP can be divided into two categories. The first is to consider BAP and QCAP as two independent stages. In berth allocation, the duration of a vessel is estimated by its capacity and distance between the actual berthing position and the preference berthing position. Then, the berthing position, berthing time, and departure time can be obtained. However, It may cause problems when BAP and QCAP are studied independently. For instance, if the container terminal is busy, the limited quay cranes cannot meet the demand of operations and it may result in increasing waiting times of vessels at the terminal, while if the container terminal is idle, it may result in wasting resources [5, 10, 12, 13]. Considering that the operation time of a vessel depends on the amount of cranes assigned to it, the second method integrated scheduling raised a lot of attention. In integrated scheduling, the amount of quay cranes is taken into consideration in BAP. Imai et al. [13] established a model for DBAP that optimized the berth allocation and quay cranes assignment with the consideration of path optimization of quay cranes. Legato et al. [14] designed a model in stochastic dynamic environment with the minimization of vessel's operation time and quay cranes. Though many models and algorithms are developed for BAPs, the quay-crane constraints (the amount of quay cranes allocated to a vessel determines the vessel's handling time and that the total number of quay cranes is limited) are simplified for computing performance problem. For instance,

Zhen et al. [15] conducted a scenario analysis that aimed at the uncertainty of berth allocation by heuristic method. An experiment which has a scale of 40 vessels was performed, while the effect of the quantity of cranes on the operating time was not considered. Sammarra et al. [16] solved a QCAP with Tabu search algorithm and also ignored this effect. To our knowledge, few literature studied large-scale scheduling problems (e.g., over 80 vessels) that considered the two quay-crane constraints.

The large-scale problems of dynamic CBAP with quay assignment are difficult to be solved in acceptable computation times. Previous researches used heuristic algorithms, for example, Tabu search and simulated annealing algorithm, to solve the problems [14]. Exact algorithms are generally used to solve small-scale problem [16]. Rolling-horizon scheduling has been widely used in manufacture, and the main principle is to solve a series of small optimization problems instead of solving large-scale scheduling problem [17, 18]. Raa et al. [19] solved the integral scheduling of dynamic DBAP and QCAP by rolling-horizon scheduling, and the length of rolling window was optimized by a hybrid heuristic algorithm. Chang et al. [20] considered a CBAP from the perspective of energy consumption and proposed the rolling-horizon optimization strategy but ignored the impact of the assignment of quay cranes on the results.

In this work, DCBAP considered dynamic berthing, continuous berth allocation, and quay cranes assignment at the same time. DCBAP conforms well with the actual situation of dynamic optimization in container terminals. Few literature have considered DCBAP. Kim and Park [21] designed a two-phase solution procedure for DCBAP. The first phase determines the berthing position and time of each vessel as well as the number of cranes assigned to each vessel at each time segment. In the second phase, the subgradient optimization technique is applied to obtain a near-optimal solution of the first phase. In this phase, a detailed schedule for each quay crane is constructed. But in the experiment, the method can only obtain approximate solutions for small- or medium-sized problems with 9 quay cranes, 40 vessels, and 1200 meters berthing line. Zhang et al. [22] considered the coverage ranges of quay cranes and allowed for limited adjustments of quay cranes during loading and discharging based on Park and Kim [6]. Lim [23] transformed BAP to a restricted form of the two-dimensional packing problem. A graph-based representation is used to capture the problem, and an effective heuristic is proposed to solve the problem.

## 3. DCBAP Problem

The scheduling of continuous berth allocation problem with quay assignment needs to determine the berthing positions, berthing times, and the handling cranes for all vessels at each time. Generally, before the vessel moor alongside, the container terminal will allocate a preferred berth that the distance to the yard is the shortest. To improve the operating efficiency and reduce the operation cost, the real berthing time should be close to the expected berthing time. If the berthing time delays, the vessel should speed down or wait

in the anchorage, which will increase fuel consumption and affect the sail plan. The handling operations of a vessel should be completed in time so it can depart in time; delays will cause penalties and will reduce the service quality of the container terminal.

The berth in most of modern large container terminals is continuous. Park and Kim [6] solved DCBAP by dividing berth and time into several intervals and established a two-dimensional coordinate system with time as  $X$ -axis and berth as  $Y$ -axis; the  $X$ -axis denotes the berthing times and the  $Y$ -axis denotes the berthing positions. Both of the positions and times are continuous, and each vessel can only occupy one rectangle in the  $X$ - $Y$  space. The horizontal length of rectangle depends on vessel length, and the width of rectangle depends on the container handling time that varies inversely to the number of cranes assigned to this vessel. Each rectangle represents a berth plan for a vessel. Based on the model proposed by Park and Kim [6], the DCBAP model built in this work considers the following reasonable situations. First, each vessel has a preferred berth that is determined in advance. Both of the position and time are continuous, and each vessel can only occupy one rectangle. The width of the rectangle is only related to vessel length and is irrelevant to vessel type. Second, the horizontal length of rectangle depends on the container handling time that varies inversely to the cranes assigned to this vessel. Third, the quay cranes available for each vessel are constrained by a minimum quantity and maximum quantity. The two quantities usually are determined by the contract between the shipping company and the terminal. Fourth, the ship handling tasks must be finished without interruption (when no crane is allocated to the vessel).

## 4. Formulation

### 4.1. Notations

(1) Sets are as follows:

- (a)  $sl = \{1, 2, \dots, sls\}$ : the set of vessels;
- (b)  $sm = \{1, 2, \dots, sms\}$ : the set of berthing positions;
- (c)  $sn = \{1, 2, \dots, sns\}$ : the set of time segments.

(2) Parameters are as follows:

- (a)  $e_k$ : the expected arrival time of vessel  $k$ ;
- (b)  $a_k$ : the total operation time of cranes, which means the total amount of time segments required to handle all containers for vessel  $k$ ;
- (c)  $b_k$ : the length of vessel  $k$ ;
- (d)  $d_k$ : the due time for the departure of vessel  $k$ ;
- (e)  $s_k$ : the preferred berthing position of vessel  $k$ ;
- (f)  $c_k^1$ : the penalty cost per unit distance of vessel  $k$  between the berthing position and preferred berthing position;
- (g)  $c_k^2$ : the penalty cost of vessel  $k$  per unit time of arrival after  $e_k$ ;

- (h)  $c_k^3$ : the penalty cost of vessel  $k$  per unit time of delay beyond the due time;
- (i)  $l_k$ : the minimum quantity of cranes that should be assigned to vessel  $k$ ;
- (j)  $u_k$ : the maximum quantity of cranes that can be assigned to vessel  $k$ ;
- (k)  $c$ : the total number of available cranes ( $C > u_k$ );
- (l)  $A_{i,j}$ : if the cell  $(i, j)$  in the time-space grid is occupied, it equals 1; otherwise, it is equal to 0;
- (m)  $D_j$ : the number of cranes that is assigned at time  $j$  (the available cranes number equals  $c - D_j$ );
- (n)  $M$ : a large positive real number.

(3) Decision variables are as follows:

- (a)  $x_{i,j,k}$ : if the grid square  $(i, j)$  is covered by the rectangle for vessel  $k \in sv$ , it is equal to 1; otherwise, it is equal to 0;
- (b)  $z_{i,j,k}$ : 1, if the preference point of vessel  $k \in sv$  is located at  $(i, j)$  in the lower-left corner point of the rectangle corresponding to the vessel; otherwise, it is equal to 0;
- (c)  $v_{kj}$ : 1, if vessel  $k$  is operated at time segment  $j$ ; otherwise, it is equal to 0;
- (d)  $u_{ki}$ : 1, if vessel  $k$  is berthed at berthing position  $i$ ; otherwise, it is equal to 0;
- (e)  $Y_{kj}$ : the number of cranes allocated to vessel  $k \in sv$  at time  $j$ ;
- (f)  $C_k$ : the completion time of container handling for vessel  $k$ ;
- (g)  $BL_k$ : the left most berthing position of a vessel  $k$ ;
- (h)  $BR_k$ : the right most berthing position of a vessel  $k$ ;
- (i)  $TL_k$ : the delayed amount of berthing time of vessel  $k$ ;
- (j)  $DL_k$ : the delayed amount of departure time of vessel  $k$ .

The following variables can be expressed by the above variables and parameters:

- (1)  $B_k = \sum_{i \in sm, j \in sn} (Z_{k,i,j} \cdot i)$ : the berthing position of vessel  $k \in sv$ ;
- (2)  $T_k = \sum_{i \in sm, j \in sn} (Z_{k,i,j} \cdot j)$ : the berthing time of vessel  $k \in sv$ ;
- (3)  $POS_{i,j} = \sum_{k \in sn} (Z_{k,i,j} \cdot k)$ : the vessel that occupies the cell  $(i, j)$  in the time-space grid.

**4.2. The Model.** The goals of the DCBAP include two aspects: first, the optimization of berthing position, which means that the berthing position should be as close as possible to the preferred berthing position so as to reduce the time to move the container from vessel to container yard and increase the operation efficiency; second, the optimization of berthing time, which means the operations should be completed as

early as possible in due time. Aimed at the optimization of berth and time, a multiobjective model proposed as (M1) minimizes the total penalty cost including deviation of berthing, berthing delay, and departure delay. The objective functions are defined in (1)–(4), and the constraints are defined in (5)–(29). Consider the following:

$$(M1) \text{ Minimize } f = (f^1, f^2, f^3), \quad (1)$$

$$f^1 = \sum_k (BL_k + BR_k) c_k^1, \quad (2)$$

$$f^2 = \sum_k (DL_k \cdot C_k^2), \quad (3)$$

$$f^3 = \sum_k (TL_k \cdot C_k^3), \quad (4)$$

$$\text{s.t. } BL_k \geq B_k - S_k, \quad \forall k, \quad (5)$$

$$BR_k \geq S_k - B_k, \quad \forall k, \quad (6)$$

$$TL_k \geq T_k - e_k, \quad \forall k, \quad (7)$$

$$T_k \geq e_k, \quad \forall k, \quad (8)$$

$$DL_k \geq C_k - d_k, \quad \forall k, \quad (9)$$

$$C_k \geq V_{k,j} (j + 1), \quad \forall k, \quad (10)$$

$$\sum_k X_{k,i,j} \leq 1, \quad \forall i, j, \quad (11)$$

$$\sum_k Y_{k,j} \leq c, \quad \forall j, \quad (12)$$

$$\sum_j Y_{k,j} \geq a_k, \quad \forall k, \quad (13)$$

$$V_{k,j} \leq Y_{k,j}, \quad \forall j, k, \quad (14)$$

$$Y_{k,j} \leq M \cdot V_{k,j}, \quad \forall j, k, \quad (15)$$

$$Y_{k,j} + M(1 - V_{k,j}) \geq l_k, \quad \forall j, k, \quad (16)$$

$$Y_{k,j} \leq u_k, \quad \forall j, k, \quad (17)$$

$$Y_{k,j} \leq \sum_i X_{k,i,j}, \quad \forall j, k, \quad (18)$$

$$M \cdot V_{k,j} \geq \sum_i X_{k,i,j}, \quad \forall j, k, \quad (19)$$

$$U_{k,i} \leq \sum_j X_{k,i,j}, \quad \forall i, k, \quad (20)$$

$$M \cdot U_{k,i} \geq \sum_j X_{k,i,j}, \quad \forall i, k, \quad (21)$$

$$\begin{aligned} ib - ia + 1 &\leq \sum_{ia \leq i \leq ib} U_{k,i} \\ &+ M(2 - U_{k,ia} - U_{k,ib}), \\ &\forall k, 1 < ia < ib < sms, \end{aligned} \quad (22)$$

$$\begin{aligned} jb - ja + 1 &\leq \sum_{ja \leq j \leq jb} V_{k,j} \\ &+ M(2 - V_{k,ja} - V_{k,jb}), \\ &\forall k, 1 < ja < jb < sms, \end{aligned} \quad (23)$$

$$V_{k,j} \leq \sum_{i, ja \leq j} Z_{k,i,ja}, \quad \forall j, k, \quad (24)$$

$$\sum_{i,j} Z_{k,i,j} = 1, \quad \forall k, \quad (25)$$

$$\begin{aligned} &\sum_{i,j, (i < ia, \text{ or } i \geq ia + b_k)} X_{k,i,j} \\ &\leq M \left( 1 - \sum_j Z_{k,ia,j} \right), \end{aligned} \quad (26)$$

$$\forall k, 2 \leq ia \leq (sms - b_k),$$

$$\sum_{i,j,i > b_k} X_{k,i,j} \leq M \left( 1 - \sum_j Z_{k,1,j} \right), \quad \forall k, \quad (27)$$

$$\begin{aligned} &\sum_{i,j,i \geq sms - b_k} X_{k,i,j} \\ &\leq M \left( 1 - \sum_j Z_{k,sms - b_k + 1, j} \right), \quad \forall k, \end{aligned} \quad (28)$$

$$b_k - \sum_i X_{k,i,j} \leq M(1 - V_{k,j}), \quad \forall k, j. \quad (29)$$

The objective equation (1) specifies the total penalties including the cost for deviation of the actual berthing position to the preferred berthing position, the cost for vessels' berthing delay, and the cost for vessels' departure delay, as, respectively, defined in (2)–(4). Equations (5)–(6) indicate that the berthing position deviates to the preferred berthing position. Equation (7) constrains the delayed amount of berthing time of vessel. Equation (8) constrains that the arrival time must be earlier than the berthing time. Equation (9) calculates the delayed amount of departure time. Equation (10) constrains that the departure time of vessel must be later than the completion time. Equation (11) indicates that each time-space grid can only be occupied by one vessel. Equation (12) represents the fact that the total number of cranes allocated to vessels at each time segment is restricted by  $c$ . Equation (13) constrains that the total operation time of the vessel should exceed its total operation time of cranes. Equations (14)–(15) restrict the sustainability of vessel handling operation; that is, if the loading and unloading operations of a vessel start, more than one crane should be



allocated to it during the operation time until all loading and unloading operations are completed. Equations (16)–(17) indicate that the quantity of cranes available for each vessel is restricted by the maximum and minimum quantities. Equations (18)–(19) represent the relations between  $V_{k,j}$  and  $X_{k,i,j}$ . Equations (20)–(21) represent the relations between  $U_{k,i}$  and  $X_{k,i,j}$ . Equations (22)–(24) constrains that the vessel occupies continuous time and space after berthing. Equation (25) ensures that each vessel has only one preference point. Equations (26)–(29) ensure that the time-space grid can only be occupied by a vessel that is berthed and handled in the schedule.

The multiobjective model (M1) is transformed into a single objective model. The objective of the new model is the weighted sum of three objectives, namely,  $f = \sum_* (W^* \cdot f^*)$ , where a weight vector  $W^*$  ( $* \in \{1, 2, 3\}$ ) represents the significance of the three subobjectives and the preferences of decision-makers. Therefore, a new model (M2) is devised as follows:

$$(M2) \text{ Minimize } f = W^* \cdot f^* \\ \text{s.t.} \quad \text{Constraints (2)–(10) and (13)–(29).} \quad (30)$$

## 5. Rolling-Horizon Strategy

In the process of the operation of the rolling scheduling, all vessels that need to be allocated are classified as three sets including vessels berthed set, set of vessels waiting for berthing, and set of vessels waiting for scheduling. Vessels berthed represent those whose handling operations have already completed or are ongoing. The vessels waiting for berthing are those who are in scheduling but have not yet started the handling operations. The vessels waiting for scheduling are those who have arrived at the port but are not yet in the schedule. During the operations of the container terminal, the vessels arrive to the port according to the expected berthing time, and the vessels stay in the set of vessels waiting for scheduling in order of arrival times. In each optimization of rolling-horizon schedule, the vessels that have finished the handling operation can be removed from the rolling window and the vessels waiting for scheduling will be added into the rolling window. Then, the schedule of the present rolling window can be obtained through optimal algorithm.

In the process of rolling scheduling, the input parameters of vessels waiting for berthing are updated by the scheduling information of vessels berthed. Based on the DCBAP model (M2), a new model (M3) was proposed as (31)–(33). Equation (32) expands (11), which indicates that a time-space grid can be occupied by present scheduling vessels and the vessels that have already berthed but the operation is still ongoing right now. The parameter  $A$  is set as the time-space grids that have been occupied in past rolling-horizon schedule (frozen ship). That is,  $A$  represents the time-space grids occupied by those vessels that have already been berthed. If the time-space grid is occupied, set the element of  $A$  to 1; otherwise, set it to 0. Equation (33) expands (12), and the parameter  $D$  is

introduced to calculate the quantity of quay cranes occupied by vessels berthed in the past rolling-horizon schedules in each time so as to update the available quantity of quay cranes in each time. Consider

$$(M3) \text{ Minimize } f = W^1 \cdot f^1 + W^2 \cdot f^2 + W^3 \cdot f^3, \quad (31)$$

$$\text{s.t.} \quad \sum_k (X_{k,i,j} + A_{i,j}) \leq 1, \quad \forall i, j, \quad (32)$$

$$\sum_k Y_{k,j} \leq c - D_j, \quad \forall j,$$

$$\text{Constraints (2)–(10) and (13)–(29).} \quad (33)$$

The rolling-horizon strategy decomposes the scheduling horizon into a series of scheduling windows, and optimal results for each window are obtained as a small-scale optimization of (M3). Then, the scheduling window is shifted to the next and the input parameters of (M3) are updated. The transition interval length of a shift and the size of the rolling window are basic factors of the rolling-horizon strategy. The transition interval is the interval between present rolling window and next rolling window. In general, decreasing the transition interval length can make the optimal result close to the result of global optimization; the number of iterations of rolling-horizon optimization increases the computation time. The rolling window for the  $t$ th time of rolling contains a set of top  $k$  of vessels waiting for scheduling, defined as  $k(t)$ .  $k$  is the size of the rolling window. When  $k$  is normal, the number of the vessels in each rolling window equals to  $k$  except the last rolling window which may be less than  $k$ . A smaller rolling window indicates less vessels scheduled for each time and less computation time, while the number of iterations of rolling optimization improves the result compared to the result of global optimization. Figure 1 shows the rolling strategy from  $t$ th time to  $t + 1$ th time.

In the optimal results for the  $t$ th window of the rolling-horizon optimization, the first  $\lambda$  vessels are frozen from the rolling window  $k(t)$  according to the sequence of scheduled berthing times.  $\lambda$  is the size of the frozen window in the rolling-horizon strategy, that is, the transition interval. The scheduled vessels in next time ( $t + 1$ ) for rolling-horizon optimization are obtained by merging the  $\lambda$  scheduled vessels into  $V(t - 1)$  which is the vessels scheduled at  $(t - 1)$ th time of rolling-horizon optimization. The other vessels in  $k(t)$  which are not frozen still stay in  $\bar{V}(t)$  and wait for the next time of rolling-horizon optimization. Then, a new time of rolling-horizon optimization,  $k(t + 1)$ , is obtained by choosing the first  $k$  vessels from  $\bar{V}(t)$  according to their berthing times. If the vessels left in  $\bar{V}(t)$  are less than  $k$ , we choose all vessels left. When all vessels are in  $V(t)$ , all the vessels are believed to have been scheduled, and the entire rolling-horizon optimization is finished. The holistic optimal solution is obtained by the rolling-horizon optimization process and combining the results returned at each rolling window.

Based on the above analysis, the rolling-horizon strategy for DCBAP involves the following four steps.

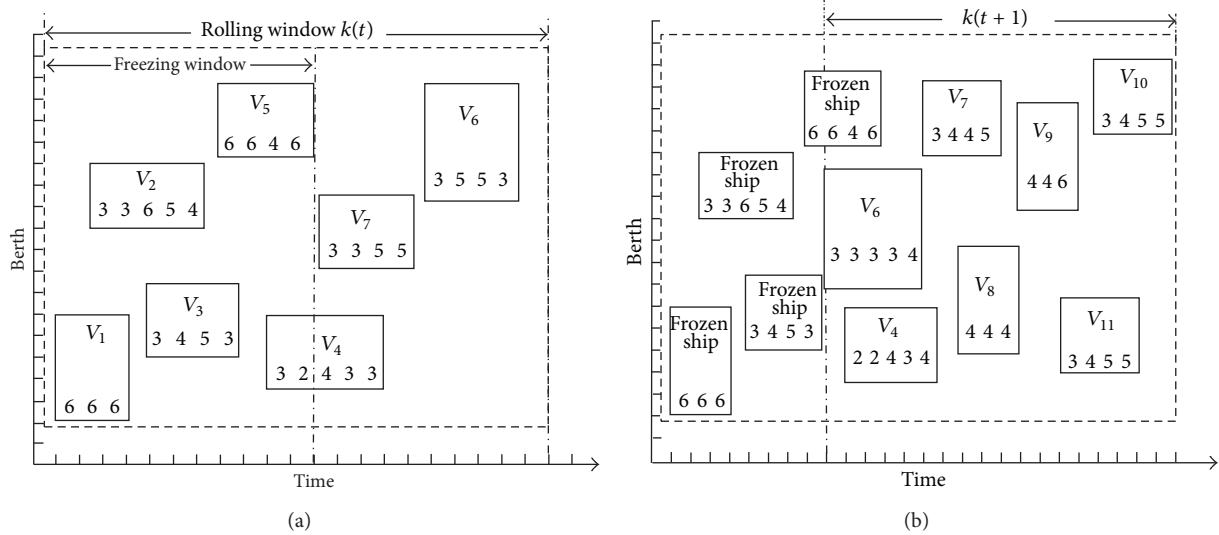


FIGURE 1: An illustration of rolling strategy.

*Step 1.* Sort all vessels that are to be scheduled according to the arrival times, initialize the total vessels set as  $sl$ , and set  $t = 1$ .

*Step 2.* Initialize the set of scheduled vessels by setting  $V(t) = \Phi$  and initialize the set of vessels waiting for scheduling by setting  $\bar{V}(t) = sl$ , obtain the rolling window  $k(t)$  by choosing the first  $k$  vessels in set  $\bar{V}(t)$ , and update  $\bar{V}(t)$  by setting  $\bar{V}(t) = \bar{V}(t) \setminus k(t)$ . Then, the parameter  $A$  is updated according to whether the time-space grid is occupied by those vessels that have already berthed; update the parameter  $D$  by calculating the quantity of quay cranes occupied in each time segment.

*Step 3.* After  $sl \leftarrow k(t)$ , solve (M2) and obtain the berthing schedule of  $k(t)$ ; freeze the first  $\lambda$  vessels from the rolling window  $k(t)$  according to the sequence of scheduled berthing times of vessels in optimal results; set the vessels scheduled as  $V(t+1)$  by merging the  $\lambda$  scheduled vessels into  $V(t)$ ; set the vessels waiting for scheduling as  $\bar{V}(t+1)$  by merging the vessels left in rolling window ( $k - \lambda$ ) into  $\bar{V}(t)$ .

*Step 4.* If  $|V| = sls$  (all vessels have been scheduled), stop and combine the results of each window as the optimal solution; or else, set  $t = t + 1$  and go to Step 2.

## 6. Experiments

The known data and the initialized schedule are set as follows.

- (1) The length of the quay line is 1200 meters (m), which is divided into 24 grids by the unit of 50 m, while the planning horizon is divided into 48 grids (i.e., 48 time segments). During this period, 20 vessels arrive at the terminal for vessel handling operations successively.
- (2) For vessel  $k$ ,  $e_k$  is generated from the uniform distribution  $U[1, 37]$ . Similarly,  $r_k$  is generated from

$U[1, 18]$ ;  $a_k$  is generated from  $U[10, 30]$ .  $d_k$  is generated based on  $e_k$  and  $a_k$ . The length of the vessel ( $b_k$ ) is generated from  $U[4, 8]$  based on the setting of berth axis. The maximum and minimum numbers of cranes are set as Table 1.

- (3) The penalty cost per unit is set to the same value;  $c_*^1 = 100$ ,  $c_*^2 = 200$ , and  $c_*^3 = 300$  for all vessels.
- (4) The maximum available crane number is set to  $c = 9$ . Set  $M = 9999$ .
- (5) The size of the rolling window is set to  $k = 5$ . And the length of the transition interval is set to  $\lambda = 2$ .

Based on Table 1, CPLEX is used as the optimization engine for solving the mixed-integer linear programming model (M2), and the rolling-horizon strategy was implemented with C#. A berthing schedule for 20 vessels in 48 hours is obtained, as shown in Figure 2. The berthing time, departure time, berthing position, and quay cranes assigned are indicated in Figure 2. Each rectangle represents a set if time-space grids are occupied by berthing vessels, and the number represents the quay cranes assigned to the vessel at that time segment.

Two important parameters in the rolling strategy are the size of rolling window ( $k$ ) and transition interval ( $\lambda$ ). Table 2 presents a series of results of adjusting  $k$  and  $\lambda$ . When  $\lambda = 4$ , the total penalties decrease first and then increase along with the increasing of  $k$  which changes from 5 to 8. The minimum objective function value is  $\lambda = 15600$  when  $k = 7$ . When  $k$  changes from 5 to 7, the total penalty decreases with the increase of the size of the rolling window ( $k$ ). This is mainly due to the fact that with the increase of the size of the rolling window, the quantity of vessels scheduled at a time increases, so that the result of the rolling-horizon optimization gets close to that of the global optimization. The decrease of total penalty decreases the berthing delay penalty and departure delay penalty, because the penalty for

TABLE 1: Input data of vessels.

Vessels	ETA	Operations	Vessel length	Due time	Ref. position	Min. cranes	Max. cranes
	$e_k$	$a_k$	$b_k$	$d_k$	$r_k$	$l_k$	$u_k$
1	1	15	8	11	1	3	6
2	2	10	4	14	16	2	4
3	4	27	6	16	3	3	6
4	4	11	4	17	10	2	4
5	5	24	7	17	5	3	6
6	9	30	6	16	14	3	5
7	10	21	5	20	7	3	5
8	10	16	4	20	13	3	5
9	13	23	5	23	10	3	5
10	16	18	6	26	15	3	6
11	17	15	6	27	2	3	6
12	20	20	6	30	10	3	5
13	21	21	6	31	15	2	5
14	26	18	4	36	8	3	6
15	27	20	4	37	6	3	6
16	27	17	6	37	15	3	5
17	27	22	5	37	9	3	5
18	28	15	7	38	1	3	6
19	34	18	6	44	6	3	5
20	36	24	6	46	7	3	6

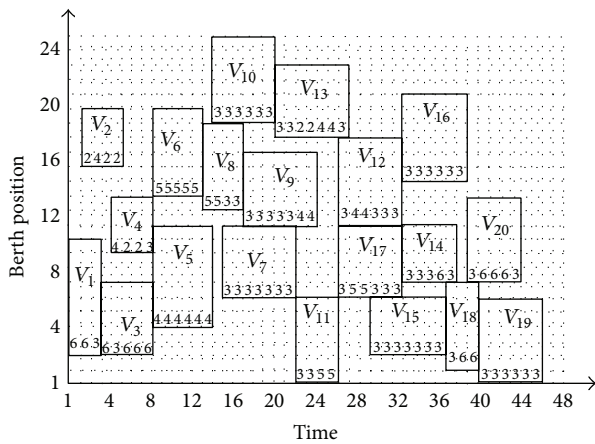


FIGURE 2: The schedule of a DCBAP.

berthing delay or departure delay is much more than the penalty for deviating from the preferred positions. So when the size of the rolling window gets bigger, the optimal result prefers to be deviated from the preferred position or assigns more quay cranes instead of delay to keep the total penalty lower. Meanwhile, with the increase of  $k$ , the utilization rate of the quay cranes also increases, the fluctuation of quay cranes (the variance yields of the quantity of quay cranes assigned to vessels at each time) gets lower, the quantity of quay cranes assigned to the vessel at each time increases the balancing degree, the operation cost of quay cranes decreases, and the total stay time at the terminal increases slightly. A reasonable and efficient rolling-horizon schedule needs the cooperation

between the container terminal operators and the shipping company. Moreover, optimal results can be reached under the balance of the stakeholders.

In the rolling-horizon strategy, decreasing the transition interval length will increase the rolling times but the result can be improved. Based on the comparisons of the objectives and the vessel's total stay times when  $\lambda$  changes from 5, 7, and 8 in Table 2, it is found that decreasing  $\lambda$  can improve the result. For instance, when  $k = 7$  and  $\lambda$  changes from 5 to 4,  $f$ ,  $f^1$ ,  $f^2$ , and  $f^3$  can be improved by 13.8%, 25%, 13.7%, 0%, and 5%, respectively. Obvious effects of optimization can also be seen in other experiments.

When the size of the rolling window increases from 7 to 8, the rolling times increase, and the total penalty cost increases. The size of the rolling window and the quantity of the frozen vessels have opposite effects on the rolling times. The former has a negative correlation with rolling times while the latter's correlation with rolling times is positive. Therefore, when the container terminal operator makes a rolling-horizon schedule, the size of the rolling window and quantity of the frozen vessels should be set reasonably, so as to improve the efficiency of the rolling-horizon scheduling process.

Still as presented in Table 2, the quantity of frozen vessels is varied to study the effect of it on the result. With the increase of the number of the frozen vessels, the penalty increases and the performance of rolling-horizon strategy decreases. With the increase of the frozen vessels, the number of the fixed vessels in each time of rolling scheduling increases. The fixed vessels are not involved in the next times of optimization, so that the rolling-horizon optimization

TABLE 2: The results under different  $\lambda$ .

$k$	$\lambda$	$f$	$f^1$	$f^2$	$f^3$	TSP (h)	FC	LRW
5	3	17300	2400	13400	<b>1500</b>	80	1.597	6
5	4	17600	1800	14000	1800	100	1.436	5
6	4	16800	1700	13600	<b>1500</b>	100	1.388	5
7	4	<u>15600</u>	<u>1500</u>	<u>12600</u>	<u>1500</u>	<u>95</u>	1.344	5
7	5	18100	2000	14600	<b>1500</b>	100	1.320	<b>4</b>
8	4	18100	2000	14600	<b>1500</b>	105	<b>1.313</b>	<b>4</b>
8	5	17600	2100	14000	<b>1500</b>	105	1.350	<b>4</b>

Note: TST: total time staying at terminal; FC: fluctuations of cranes; LRW: length of the rolling window.

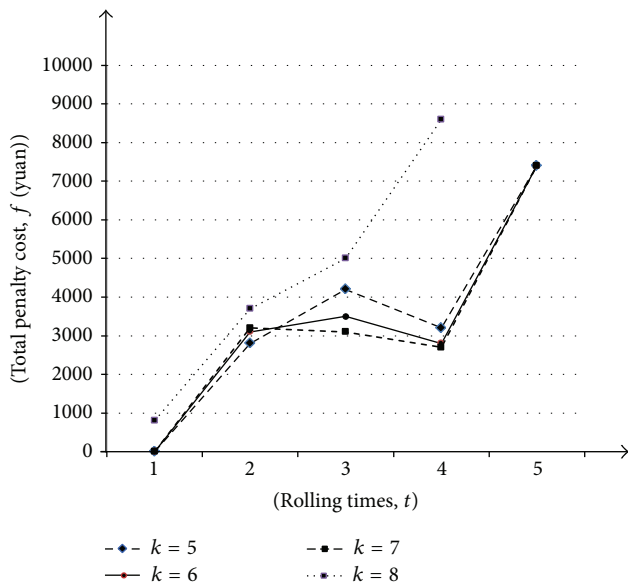


FIGURE 3: The penalty cost of each rolling time under different size of rolling window.

method cannot obtain the best result. Setting the quantity of the frozen ships to a small value may result in rolling too many times and computing for a long time. Therefore, in an actual schedule, a reasonable configuration of quantity of frozen vessels can decrease the effect of the number of fixed vessels and optimize the computation time as well as optimize the performance of the rolling-horizon strategy.

The coupling degree of vessels' arrival times has a big impact on the efficiency of the rolling strategy. As shown in Figure 3, when the size of the rolling window changes, the similar tendency can be seen that the penalty costs increase sharply in the third and fifth times of rolling under different size of rolling windows. Due to the high coupling degree in vessels' arrival in the last time of rolling, vessels arrive intensely and bring too many handling operations, so resources including quay cranes and berths are occupied for a long time and the available resources for the next rolling schedule are insufficient. Therefore, the successor ships have to delay the times for berth and departure. As a result, the efficiency of the rolling schedule decreases and the penalty increases.

The factors that affect the rolling strategy mainly include the size of rolling window, quantity of the frozen vessels, and the rolling times. So appropriate size of rolling window and quantity of frozen vessels need to be chosen to improve the performance of the rolling-horizon strategy. When choosing the size of the rolling window, both operation penalty cost and computation time should be considered. On the one hand, the size should be bigger and the quantity of vessels in the rolling window should be sufficient enough so that the local optimal solution (returned by solving (M3)) will be closer to the result of global optimization. Moreover, it will decrease the operations cost of the container terminal and improve the service quality. On the other hand, if the size is too big and the computation time of each time of the rolling window gets too long, which may lead to delay of the berthing time of vessels and affect the normal operations of container terminals and increase the cost inversely. So the size of the rolling window should balance both cost and computation times to improve the operation efficiency of the container terminal. And based on this, an appropriate quantity of frozen vessels should be chosen to achieve the best performance of the rolling-horizon strategy.

In this paper, the multiobjective function is changed to a single-objective function by using weighting efficient. In order to analyze the tradeoff between berth deviation and time delay, the objectives are divided into two categories, berth-varied objective  $f^1$ , and time-varied objectives  $f^2$  and  $f^3$ . The weights of  $f^2$  and  $f^3$  are set as equal. By adjusting the weight of the objects ( $W^1 = 0.1, 0.2, \dots, 0.9$ ,  $W^2 = 1 - W^1$ ), 10 experimental variables can be obtained based on M2. Then, the objective function value can be calculated based on each experimental variable. The change trend of the two objective functions is shown in Figure 4. With the increase of the weight  $W^1$ , the berthing deviation is reduced from 7200 to 0, and when  $W^1 = 0.9$  each vessel can be assigned to the preferred berth. But the result is that the waiting time for handling is increasing and the number of vessels that delayed departure and the delay time is also increasing, which lead to the fact that the penalty cost including berthing delay and departure delay increases to 152000, the waiting time is added by 76%, and the departure delay is added by 57%. Therefore, controlling weight  $W^1$  reasonably can seek a balance between the deviation from the preferred berth and the waiting time.

As shown in Table 2, the optimal solution is obtained when  $k = 7$ ,  $\lambda = 4$ . So a comparative analysis between



TABLE 3: Comparison between rolling-horizon and normal schedule for different vessel quantity.

Quantity of vessels	Scheduling method	$f$	$f^{c1}$	$f^{c2}$	$f^{c3}$	Computation time (s)
10	Rolling-horizon	4400	900	3200	300	564.4
	Normal schedule	4000	900	2800	300	1133.3
11	Rolling-horizon	5300	900	4000	0	763.2
	Normal schedule	5300	900	4400	0	2054.8
12	Rolling-horizon	6200	1200	5000	0	778.2
	Normal schedule	6000	1000	5000	0	7871.1
13	Rolling-horizon	7000	1000	6000	0	853.3
	Normal schedule	7000	1000	6000	0	21092.1
14	Rolling-horizon	7500	1300	6200	0	954.9
	Normal schedule	—	—	—	—	—
15	Rolling-horizon	8500	1300	7200	0	1125.7
	Normal schedule	—	—	—	—	—

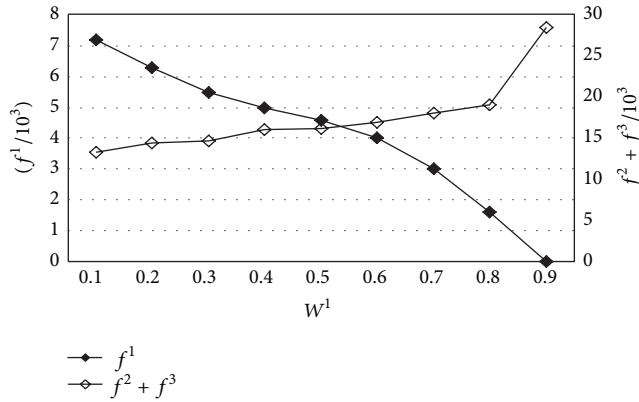


FIGURE 4: Tradeoff between penalty of berthing deviation and time delay.

normal schedule and rolling schedule is conducted as shown in Table 3, where “—” indicates that the result cannot afford the CPLEX solver running out of memory.

As shown in Table 3, when using rolling-horizon strategy, optimal results are obtained for all the six examples. While using normal schedule, the solution cannot be obtained when the quantity of vessels is increasing to 14. It indicates that the rolling-horizon strategy can decrease the complexity of computation compared to normal schedule and solve large-scale problems.

When the quantity of vessels increases from 10 to 15, the optimal result of the rolling-horizon strategy is similar to that of the normal schedule. The rolling-horizon strategy performs well. As shown in Figure 5, from the tendency of computation time of the rolling-horizon strategy and normal schedule, the time increases sharply with the increase of the total quantity of vessels. When the total quantity of vessels is 10, the computation time of normal schedule is twice that of the rolling schedule. When the total quantity is 13, the time of the normal schedule is 25 times that of the rolling schedule. The computation time of normal schedule increases exponentially with the number of vessels, and when the total quantity is 14, the optimal result cannot be obtained. On

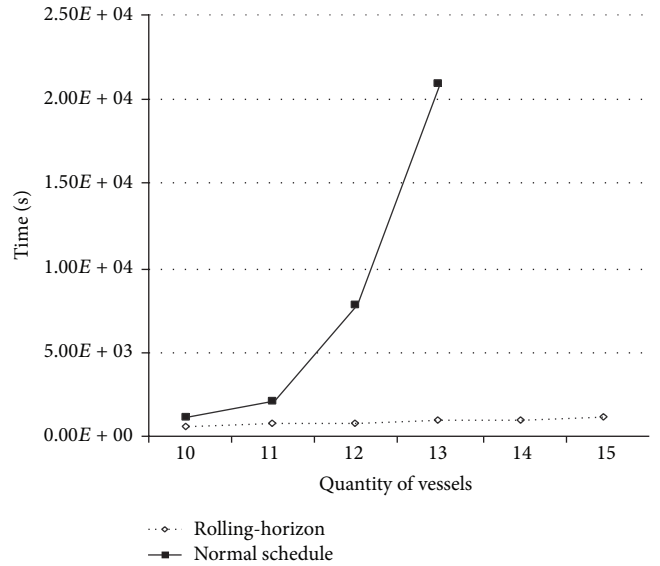


FIGURE 5: The computation time tendency of rolling-horizon and normal schedule.

the contrary, the computation time of the rolling-horizon strategy increases slightly, which is directly in proportion to the total quantity. The normal schedule searches for a large amount of nodes once, which burdens the CPU and memory much. The rolling schedule has better computational efficiency than normal schedule, and it can solve large-scale problems in the limited time.

Though in the above experiment only 20 vessels are considered in the problems, the quantity of the vessels is directly in proportion to the computation time. So the rolling-horizon strategy proposed in this paper can be used to solve large-scale problems and meet practical requirements. Based on the above experiments, the algorithm was performed to solve large-scale problems whose quantities of vessels range from 40 to 200 and the planning horizons range from 48 to 400 hours. The results show that the computation performance is also almost directly in proportion to the quantity.

## 7. Conclusions

To solve the large-scale integral scheduling of DCBAP, a method based on the rolling-horizon strategy is developed and a mixed-integer multiobjective linear programming model is established. The model minimizes the total penalty including deviation of berthing, berthing delay, and departure delay. Moreover, the model is revised for the rolling-horizon strategy. The experimental results indicate that the method can be applied to solve large-scale DCBAP problem with the computation time linear to the quantity of involved vessels. In the rolling-horizon strategy, the parameters include the sizes of the rolling window and transition intervals have big influences on the optimality and the overall performance of the rolling-horizon strategy. The increase of the length of the rolling window can improve the results, while the computation time increases exponentially. The decrease of the length of the transition interval increases the rolling times and improves the results. The experiments prove that the settings of  $k = 7$ ,  $\lambda = 4$ , can obtain better results than other settings. Meanwhile, the high coupling degree of vessels' arrival times will decrease the efficiency of the rolling strategy. The actual scheduling should choose appropriate combination of the size of rolling window and transition interval by a series of experiments on the balance of optimality and computation performance to improve the efficiency of berthing schedule and the service quantity of the container terminal and decrease the operations cost. The rolling-horizon strategy in this paper considers the dynamic characteristics of the vessels' expected berthing time. Compared to the model proposed by Park and Kim [6], the problem scale (vessels involved) is greatly expanded, but the possibility that vessels may arrive and be scheduled in advance is not considered. Besides, though the on-the-shelf solver for mixed-integer linear programming model is convenient for researching the effectiveness of the rolling-horizon strategy, this solver becomes a bottleneck that restricts improvement of the computing performance. To meet the demands of practical applications, the work on the way is to design a heuristic algorithm to solve the model (M2) effectively and merge it into the rolling-horizon strategy for further improving the optimality and computational performance.

## Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

## Acknowledgments

This study is partially supported by the National Nature Science Foundation of China (71101088, 71171129, and 71390521), the Science Foundation of Ministry of Education of China and Shanghai (20113121120002, 14YZ100, 20123121110004, and 13SG48), the Science and Technology Commission of Shanghai (12ZR1412800 and 12510501600), the high-tech research and development program of China (2013A2041106), and

Graduate Innovation Fund Project of Shanghai Maritime University (GK2013022).

## References

- [1] S. Chand, R. Traub, and R. Uzsoy, "Rolling horizon procedures for the single machine deterministic total completion time scheduling problem with release dates," *Annals of Operations Research*, vol. 70, pp. 115–125, 1997.
- [2] J. Fang and Y. Xi, "A rolling horizon job shop rescheduling strategy in the dynamic environment," *The International Journal of Advanced Manufacturing Technology*, vol. 13, no. 3, pp. 227–232, 1997.
- [3] I. M. Ovacik and R. Uzsoy, "Rolling horizon procedures for dynamic parallel machine scheduling with sequence-dependent setup times," *The International Journal of Production Research*, vol. 33, no. 11, pp. 3173–3192, 1995.
- [4] Y. Guan, W.-Q. Xiao, R. K. Cheung, and C.-L. Li, "A multiprocessor task scheduling model for berth allocation: heuristic and worst-case analysis," *Operations Research Letters*, vol. 30, no. 5, pp. 343–350, 2002.
- [5] C. L. Li, X. Q. Cai, and C. Y. Lee, "Scheduling with multiple-job-on-one-processor pattern," *IIE Transactions*, vol. 30, no. 5, pp. 433–445, 1998.
- [6] Y.-M. Park and K. H. Kim, "A scheduling method for berth and quay cranes," *OR Spectrum*, vol. 25, no. 1, pp. 1–23, 2003.
- [7] C. Bierwirth and F. Meisel, "A survey of berth allocation and quay crane scheduling problems in container terminals," *European Journal of Operational Research*, vol. 202, no. 3, pp. 615–627, 2010.
- [8] I. F. A. Vis and R. de Koster, "Transshipment of containers at a container terminal: an overview," *European Journal of Operational Research*, vol. 147, no. 1, pp. 1–16, 2003.
- [9] K. H. Kim and K. C. Moon, "Berth scheduling by simulated annealing," *Transportation Research B: Methodological*, vol. 37, no. 6, pp. 541–560, 2003.
- [10] Y.-M. Park and K. H. Kim, "Berth scheduling for container terminals by using a sub-gradient optimization technique," *Journal of the Operational Research Society*, vol. 53, no. 9, pp. 1054–1062, 2002.
- [11] A. Imai, E. Nishimura, and S. Papadimitriou, "The dynamic berth allocation problem for a container port," *Transportation Research B: Methodological*, vol. 35, no. 4, pp. 401–417, 2001.
- [12] G. G. Brown, S. Lawphongpanich, and K. P. Thurman, "Optimizing ship berthing," *Naval Research Logistics*, vol. 41, no. 1, pp. 1–15, 1994.
- [13] A. Imai, H. C. Chen, E. Nishimura, and S. Papadimitriou, "The simultaneous berth and quay crane allocation problem," *Transportation Research E: Logistics and Transportation Review*, vol. 44, no. 5, pp. 900–920, 2008.
- [14] P. Legato, D. Gullí, and R. Trunfio, "The quay crane deployment problem at a maritime container terminal," in *Proceedings of the 22nd European Conference on Modelling and Simulation (ECMS '08)*, pp. 196–209, Nicosia, Cyprus, June 2008.
- [15] L. Zhen, L. H. Lee, and E. P. Chew, "A decision model for berth allocation under uncertainty," *European Journal of Operational Research*, vol. 212, no. 1, pp. 54–68, 2011.
- [16] M. Sammarra, J.-F. Cordeau, G. Laporte, and M. F. Monaco, "A tabu search heuristic for the quay crane scheduling problem," *Journal of Scheduling*, vol. 10, no. 4–5, pp. 327–336, 2007.

- [17] I. M. Ovacik and R. Uzsoy, "Rolling horizon algorithms for a single-machine dynamic scheduling problem with sequence-dependent setup times," *The International Journal of Production Research*, vol. 32, no. 6, pp. 1243–1263, 1994.
- [18] M. Singer, "Decomposition methods for large job shops," *Computers & Operations Research*, vol. 28, no. 3, pp. 193–207, 2000.
- [19] B. Raa, W. Dullaert, and R. van Schaeren, "An enriched model for the integrated berth allocation and quay crane assignment problem," *Expert Systems with Applications*, vol. 38, no. 11, pp. 14136–14147, 2011.
- [20] D. Chang, J. He, and H. Zhang, "A rule-based joint berth allocation and quay crane assignment," in *Proceedings of the 2nd International Conference on Industrial and Information Systems (IIS '10)*, pp. 464–467, IEEE, Dalian, China, July 2010.
- [21] K. H. Kim and Y.-M. Park, "A crane scheduling method for port container terminals," *European Journal of Operational Research*, vol. 156, no. 3, pp. 752–768, 2004.
- [22] C. Zhang, L. Zheng, Z. Zhang, L. Shi, and A. J. Armstrong, "The allocation of berths and quay cranes by using a sub-gradient optimization technique," *Computers & Industrial Engineering*, vol. 58, no. 1, pp. 40–50, 2010.
- [23] A. Lim, "The berth planning problem," *Operations Research Letters*, vol. 22, no. 2-3, pp. 105–110, 1998.



# Hindawi

Submit your manuscripts at  
<http://www.hindawi.com>

