

## Research Article

# Integrated Berth Allocation and Time-Variant Quay Crane Scheduling with Tidal Impact in Approach Channel

Xiaogang Jiao <sup>1,2</sup>, Feifeng Zheng,<sup>1</sup> Ming Liu <sup>3</sup> and Yinfeng Xu<sup>1</sup>

<sup>1</sup>Glorious Sun School of Business and Management, Donghua University, Shanghai 200051, China

<sup>2</sup>School of Information Engineering, Ningxia University, Yinchuan 750021, China

<sup>3</sup>School of Economics & Management, Tongji University, Shanghai 200092, China

Correspondence should be addressed to Ming Liu; [mingliu@tongji.edu.cn](mailto:mingliu@tongji.edu.cn)

Received 21 July 2018; Accepted 27 September 2018; Published 15 October 2018

Academic Editor: Giulio E. Cantarella

Copyright © 2018 Xiaogang Jiao et al. 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.

This paper addresses the integrated problem of dynamic continuous berth allocation and time-variant quay crane scheduling in container terminals and introduces the factor of tidal impacts into the problem. We mainly consider the impact of tides on the transport capacity of an approach channel that connects quay and anchorage. An integer linear programming model is developed, and then three heuristic algorithms, Genetic Algorithm, Hybrid Particle Swarm Optimization, and Hybrid Simulated Annealing, are proposed to solve the model. Numerical experiments are conducted to verify the efficient performances of the proposed heuristics. Moreover, experimental results also demonstrate the nonignorable impact of tides on the vessel turnaround time and the utilization of berth and quay cranes.

## 1. Introduction

In the last decades, container shipping plays an important and irreplaceable role in the international trade due to its low shipping cost and high reliability. Huge maritime transport demand brings container ports more business opportunities as well as more challenges [1]. To stand out in the fierce competition, container port managers have to improve the efficiency of services in order to lower service costs and attract more vessels calling. One critical aspect is to maximize the use of key resources, such as berths and quay cranes (QCs). With the quick increase of containership size, the processing schedule of vessels in many container seaports is much impacted by tides. Because their required deep drafts are usually more than the channel's water depth, mega-containerships have to pass through the approach channel by the aid of tides.

Figure 1 illustrates the tidal effect. As shown in the figure, the tide fluctuates and it alters the water depth of the channel over time (taking a tidal cycle as an example). When a vessel arrives at the port, the vessel draft  $h_1$  becomes known. To ensure the safety during passing through the channel, the

under keel clearance  $h_2$ , which is the distance from the vessel bottom to the fairway bottom, is required to be at least 12% of the vessel draft [2]. Let  $h_3$  be the tide datum depth. Then a minimum tide height  $h^*$  required for the vessel to pass through the channel is formulated as  $h^* = h_1 + h_2 - h_3$ . Based on this, the available tide time window  $[t_1, t_2]$  can be calculated according to  $h^*$  and the tidal rhythm together with tidal amplitude at the seaport. The vessel can only pass through channel during time interval  $[t_1, t_2]$  in the tidal cycle. Specifically, if the vessel calls at the port before  $t_1$ , it waits to the time point. If otherwise it arrives after time  $t_2$ , the vessel has to wait until the next available tide time window. Similarly, if the vessel has completed service and plans to leave the port via the channel, the same tidal requirement has to be met. So the container terminal manager has to well consider the interaction of berth, QCs, vessels' drafts, and the fluctuation of tide so as to provide appropriate berthing time for each vessel to prevent its unnecessary waiting at the anchorage ground. An optimal processing schedule of vessels not only maximizes the use of the critical resources, but also greatly reduces vessel bunker fuel costs and greenhouse gas emissions.

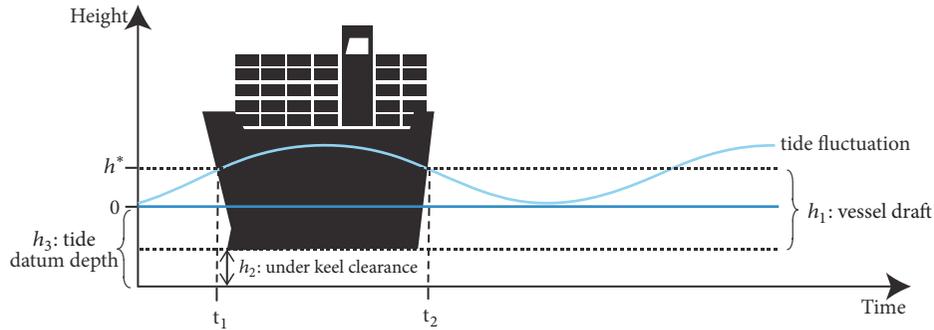


FIGURE 1: Illustration of a vessel passing through the approach channel.

Many recent works [3–7] have addressed the impact of tides on the seaside operations in a container port. These papers consider the tidal impact in berth allocation problem (BAP), in which the processing times of vessels are generally assumed to be constants. That is, QC assignment is not involved. In practice, however, the turnaround time of a vessel is a variable and depends on not only the number of containers to be processed but also the number of QCs assigned to the vessel. Motivated by this, we investigate the integrated berth and QC scheduling problem with tidal impact in this work such that the turnaround time of any vessel relies on its schedule.

The main contributions of this work are threefold. (i) We introduce the impact of tides into the integrated problem of dynamic continuous berth allocation and time-variant QC scheduling. (ii) An integer linear programming model is established and three heuristic algorithms are exploited to solve the model. (iii) Numerical experiments are conducted to reveal the nonneglected tidal impacts on the turnaround times of vessels and the utilization of berth and QCs.

The rest of this paper is organized as follows. Section 2 reviews relevant literature. The description of the problem is presented in Section 3. Section 4 formulates the integer linear programming model of the considered problem, and three heuristic algorithms are employed in Section 5. In Section 6 we conduct extensive numerical experiments to further verify the impact of tides in scheduling and make comparison between the proposed heuristics. Finally, conclusions and some suggestions of further research are provided in Section 7.

## 2. Literature Review

Seaside operations scheduling in container terminals have been of large interest in the fields of management science, operations research, and computer sciences in the last decades. For a comprehensive understanding of container terminal operations, readers are referred to literature [8–12]. According to Bierwirth and Meisel [10], there are three basic scheduling problems in seaside operations, i.e., the berth allocation problem (BAP), the QC assignment problem (QCAP), and the QC scheduling problem (QCSP). Moreover, the integration of these problems, such as BAP-QCAP and BAP-QCSP, has also caught a lot of interest in literature.

*2.1. Integrated Seaside Operations Scheduling.* Park and Kim [13] put forward a pioneering integration method for scheduling berth and QCs. They establish a mixed integer linear programming (MILP) model and propose a two-phase solution procedure to solve the model. At the first stage, the assignment of berths and QCs to vessels is determined by Lagrangian relaxation and a subgradient optimization method. Then a dynamic programming technique is adopted to schedule the assigned QCs for the vessels. Liang et al. [14] study an integration of discrete berth allocation problem and QC dynamic assignment problem. They formulate a multi-objective mathematical model to minimize both turnaround time of vessels and the number of QC movements simultaneously. To solve the model, a multiobjective hybrid genetic algorithm approach with a priority-based encoding method is applied. Zhang et al. [15] address a BAP-QCAP problem considering the coverage ranges and limited adjustments of QCs during processing. They construct a mixed integer programming model and apply a subgradient optimization algorithm to solve the model. Giallombardo et al. [16] propose a novel concept of QC assignment profiles to integrate berth allocation and QC assignment. A mixed integer quadratic programming formulation and a mixed integer linear program are constructed, which aim to maximize the total value of chosen QC-profiles and yard costs. Hu et al. [17] construct a nonlinear multiobjective mixed integer programming model for integrating BAP and QCAP, which considers vessel fuel consumption and emissions. A second-order mixed integer cone programming model is adopted in solving the model.

Meisel and Bierwirth [18] establish a framework for the integration of berth allocation, QC assignment and scheduling problem. The framework proceeds in three phases. Phase 1 calculates the crane productivity rates according to vessel stowage plans. Berthing times and positions as well as crane capacity assignments are determined in Phase 2. Finally, Phase 3 provides QC schedules. Turkogullari et al. [19] study the integration of berth allocation, QC assignment, and time-invariant QC scheduling problem (BACASP) such that the QCs assigned to each vessel stay at where they are until the vessel has been completed. First, they integrate the berth allocation and QC assignment problem (BACAP) in a binary integer linear programming model. In order to incorporate QC scheduling problem, they extend the

BACAP and introduce a mixed integer linear programming model for the BACASP. Two algorithms are proposed to generate the optimal solution for the BACASP based on the optimal solution of BACAP. Considering the scenario where the number of QCs assigned to any vessel is allowed to vary during the processing of the vessel, Turkogullari et al. [20] build a mixed integer linear program to combine berth allocation, time-variant QC assignment and scheduling with consideration of QC setup cost owing to the movement between two working locations along the berth. They split it into a relaxed master problem and a subproblem and propose an exact solution algorithm by means of directed, layered, single source, and single sink network. The simulation results demonstrate that the algorithm may obtain optimal solutions for large-scale realistic problem instances.

**2.2. Seaside Operations considering Tidal Impacts.** Although seaside operations in container terminals have caught much interest in literature, there are only a few studies considering the influence of tide in the BAP problem. Most of these works consider the tide impact on the water depth along the berths. Barros et al. [3] investigate a BAP model with tidal time windows and stock level conditions in a tidal bulk port of Brazil. They present an integer linear programming model with homogeneous berths, which aims to minimize the overall demurrage in the planning horizon. A simulated-annealing-based algorithm (SA) is proposed to obtain good solutions. Xu et al. [4] study a discrete BAP model in both static and dynamic cases for container terminals in which the assignment of berths is limited by water depth due to tides. They assume that the time horizon is divided into only one high water period and one low water period. Qin et al. [7] address the BAP at a tidal river port in a more precise manner than [3, 4] by introducing time-varying draft constraints on vessel berthing based on each vessel's draft and tide conditions of different berths. They build six integer programming models which are various combinations of three objectives (i.e., total weighted turnaround time, total weighted departure delay, and makespan) and two vessel arrival cases (i.e., static and dynamic). The models are then transformed into the innovative constraint programming counterparts. Lalla-Ruiz et al. [6] modify the model proposed in [4] to improve the feasibility of solutions based on the generalized Set Partitioning problem. Numerical experiments indicate that the computational efficiency is also improved significantly.

Some other authors investigate the BAP problem with the impact of tide in the approach channel. Sheikholeslami et al. [5] take into consideration not only the tidal constraints in the approach channel but also the variations of water depth in different berths in the continuous dynamic berth allocation. A genetic algorithm combined with pattern search algorithm is proposed to solve the problem. Du et al. [2] develop two mathematical models to figure out the BAP in a tidal port, in which it takes into account the restriction of tides when vessels enter or leave the port through the approach channel. The first model aims to minimize the total departure delay of vessels, and the second model takes vessel arrival

times as decision variables and takes the minimum weighted summation of departure delay and fuel consumption as the objective. The second-order cone programming technique is adopted to simplify the nonlinear complexity of the second model. Numerical simulations show that the second model can effectively mitigate the tidal impacts on container terminal operations and vessel emissions. To the best of our knowledge, there are no results on the BACASP problem where the influence of tides is involved. In this work we focus on the BACASP problem with time-variant QC scheduling and tidal impacts.

### 3. Problem Statement

We consider the integrated berth allocation, QC assignment, and time-variant QC scheduling problem with tide constraint and denote it as BACASP-TC as in [19, 20], where the notation TC denotes the factor of tide constraint. By reasonable discretization of the continuous berth and time horizon, we can treat the considered problem as a discrete optimization model. We discretize the planning time horizon into time steps  $1, 2, 3, \dots$  with one hour per time step [15, 18–20]. Moreover, the continuous quay berth is partitioned into discretized segments with 50 meters per berth segment [15, 19–21]. Thus, the length  $l_i$  of vessel  $i$  can be measured by the number of required berth segments. For example,  $l_i = 2$  means it is of length equal to 100 meters. Considering the average productivity of one QC, we assume based on field investigation that any QC processes 30 containers in each hour. Together with the above time discretization method, we define a notation *QC-time-step* to measure the workload of a vessel such that one QC-time-step means a workload of 30 containers. For example, if vessel  $i$  is of workload  $c_i = 10$ , then it means there are  $30 \times c_i = 300$  containers to be handled for the vessel. Notice that if there are only 275 but not 300 containers to be processed for the vessel, we still set  $c_i = 10$  since nine QC-time-steps can only satisfy 270 containers.

We partition vessels into three types by their sizes, i.e., feeders, mediums, and jumbos, where feeders are of the smallest size while jumbos are of the largest size. Considering tidal impact, we assume that only jumbos are *tide dependent*; that is, these vessels can only pass through the approach channel in the duration of high water level. This constraint is enforced when jumbos are either approaching the quay berth for processing or leaving the quay after processing. The other two kinds of vessels, however, are *tide independent* and they can pass through the channel at any time step. It is assumed as in [2] that any vessel takes one time step to pass through the approach channel. The vessels arrive at the anchorage at various time points and can only pass through the channel on or after arrival. Considering tidal rule, we assume that there are high and low water level time windows alternatively, each of which consists of six consecutive time steps. It is assumed without loss of generality that the first six time steps are of high water level.

For QC scheduling in processing any vessel, we consider the time-variant version in this work such that the number of QCs assigned to the same vessel can be different in two

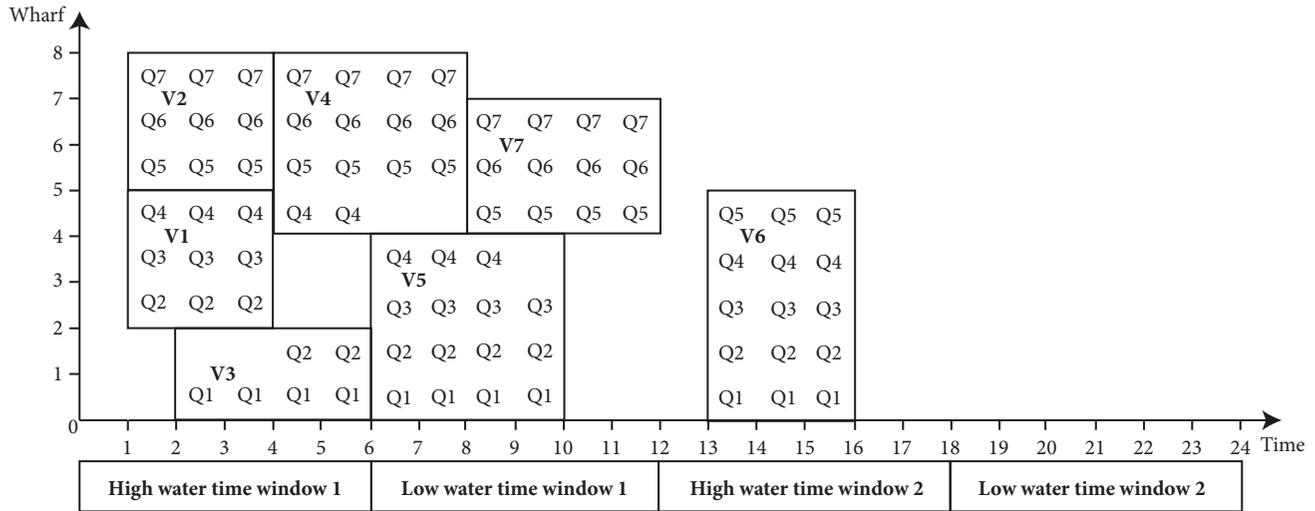


FIGURE 2: An optimal solution of a BACASP-TC instance.

consecutive time steps. Compared with the time-invariant scenario where the set of QCs assigned to each vessel is predetermined and keeps unchanged during processing [16, 21–23], the time-variant QC scheduling fashion may improve the utilization of both QCs and berth segments. As all the QCs move on the same rail along the quay, their relative positions keep unchanged all the time. Moreover, any vessel stays still during its processing. Together with the fact that each QC is of width equal to about one berth segment length, it is reasonable to assume that at most one QC serving vessel  $i$  is allowed to leave for processing a neighbor vessel before completing vessel  $i$  at any time step.

To illustrate the integrated berth and QC scheduling in the BACASP-TC problem, we give an example with 8 berth segments, 7 QCs, and 7 vessels, where vessels  $V_1, V_2, V_3$ , and  $V_7$  are of small sizes and tide independent, while  $V_4, V_5$ , and  $V_6$  are tide dependent. As shown in Figure 2, the vertical axis denotes the berth segments along the wharf, and the horizontal axis represents the time steps, which are divided into high water level time windows and low water level time windows. Each rectangle with solid borderlines stands for the processing duration and position of a vessel, denoted by  $V_i$ . The width of the rectangle denotes the processing time of the vessel, and the height represents its length. Each symbol  $Q_j$  within a rectangle denotes the QC  $j$  assigned to the vessel at one time step. For example, the processing of tide-independent vessel  $V_3$  occupies berth segments 1 and 2. The vessel passes through the channel in the second time step, starts processing at the beginning of time step 3, and ends processing at the end of time step 6. It then leaves the wharf in time step 7. For QC scheduling, only QC1 processes  $V_3$  in time steps 3 and 4, while both QC1 and QC2 process the vessel in time steps 5 and 6. QC4 serves vessel  $V_4$  in time steps 5 and 6, but it moves to serve vessel  $V_5$  at time step 7, at which  $V_4$  is still on processing.

Notice that it is not allowed for any two vessels that share the same processing berth segments to be of overlapped processing time steps, while this is not necessary for their

setup times, i.e., time steps for entering or leaving the wharf. In Figure 2, for example, both  $V_3$  and  $V_5$  share berth segments 1 and 2.  $V_5$  passes through the channel at time step 6 and starts processing at time step 7, while its preceding vessel  $V_3$  leaves the wharf at time step 7, which is later than the entering time of  $V_5$ . When  $V_5$  arrives at the port at time step 6, it immediately enters into the channel taking advantage of the high water level and moors at berth segments 1 to 4 one time step later. After finishing processing at the end of time step 10, it vacates the berth segments at once and waits for two time steps to leave the quay at the beginning of the next high water time window. A similar waiting situation occurs for  $V_6$  between time step 7 and time step 13 before it enters the channel.

There are some fundamental assumptions in the considered BACASP-TC problem as follows.

- (1) The time horizon is divided into uniform time steps with one hour per time step, and the wharf is partitioned into equal-sized berth segments with 50 meters per segment.
- (2) There are three types of vessels based on their respective lengths, where feeders and mediums are tide independent, while jumbos are tide dependent.
- (3) By the tidal rule, in every 12 consecutive time steps or equally 12 hours, the first six time steps are in high water level, and the following six consecutive time steps are in low water level.
- (4) It consumes exactly one time step for any vessel to pass through the approach channel.
- (5) For any vessel being processed, the difference in the number of its assigned QCs in any two consecutive time steps is at most one.
- (6) The setup time of QC, including its move along the berth line, is ignored.

## 4. Mathematical Formulation

In this section we establish an integer linear programming model for the BACASP-TC problem. We first present basic notations and then propose the mathematical model.

### 4.1. Notations

#### Sets

- (i)  $I^{TIDE}$ : set of tide-dependent vessels, and  $I^{TIDE} = \{1, 2, \dots, |I^{TIDE}|\}$ .
- (ii)  $I^{FREE}$ : set of tide-independent vessels, and  $I^{FREE} = \{1, 2, \dots, |I^{FREE}|\}$ .
- (iii)  $I$ : set of vessels, where  $I = I^{TIDE} \cup I^{FREE}$  and  $|I| = |I^{TIDE}| + |I^{FREE}|$ .
- (iv)  $Q$ : set of QCs, and  $Q = \{1, 2, 3, \dots, |Q|\}$ .
- (v)  $B$ : set of berth segments, and  $B = \{1, 2, 3, \dots, |B|\}$ .
- (vi)  $N$ : set of high/low water time windows in the time horizon, and  $N = \{1, 2, 3, \dots, |N|\}$ .
- (vii)  $T$ : set of time steps, and  $T = \{1, 2, 3, \dots, |T|\}$  with  $|T| = 2|N|V$ , where  $V$  is the duration of a high (or low) water time window.

#### Indices

- (i)  $i, i'$ : indices of vessels.
- (ii)  $q, q'$ : indices of QCs.
- (iii)  $t, t'$ : indices of time steps.
- (iv)  $b, b'$ : indices of berth segments.
- (v)  $n$ : index of high/low water time windows.

#### Input Parameters

- (i)  $a_i$ : arrival time of vessel  $i$ .
- (ii)  $l_i$ : length of vessel  $i$ , and it is measured in the number of berth segments.
- (iii)  $c_i$ : workload of vessel  $i$ , and it is measured in QC-time-step.
- (iv)  $q_i^{\min}$ : the minimum number of QCs required to be assigned to vessel  $i$ .
- (v)  $q_i^{\max}$ : the maximum number of QCs allowed to be assigned to vessel  $i$ .
- (vi)  $V$ : duration of a high/low water time window.
- (vii)  $C$ : the time for any vessel to pass through the approach channel, and  $C = 1$  time step.
- (viii)  $M$ : a sufficiently large constant that is commonly used in describing linear inequality constraints.

#### Decision Variables

- (i)  $\lambda_i$ : the time at which vessel  $i$  passes through the channel for processing.
- (ii)  $\varepsilon_i$ : the start time of processing vessel  $i$ . The vessel starts processing at the beginning of time step  $\varepsilon_i$ .

- (iii)  $\rho_i$ : the processing time of vessel  $i$ .
- (iv)  $\sigma_i$ : the end time of processing vessel  $i$ . The vessel completes processing at the end of time step  $\sigma_i$ .
- (v)  $\psi_i$ : the time at which vessel  $i$  passes through the channel and leaves the port.
- (vi)  $\zeta_i$ : the departure time of vessel  $i$ , and  $\zeta_i = \psi_i + C$ .
- (vii)  $\beta_i$ : the berth segment occupied by the stem of vessel  $i$ .
- (viii)  $\eta_{qit}$ : (Binary) equal to 1 if QC  $q$  is assigned to vessel  $i$  in time step  $t$ , 0 otherwise.

#### Auxiliary Decision Variables

- (i)  $\pi_{it}$ : (binary) equal to 1 if vessel  $i$  starts processing at time step  $t$ ; i.e.,  $\varepsilon_i = t$ , 0 otherwise.
- (ii)  $\phi_{it}$ : (binary) equal to 1 if vessel  $i$  finishes processing at time step  $t$ ; i.e.,  $\sigma_i = t$ , 0 otherwise.
- (iii)  $\theta_{it}$ : (binary) equal to 1 if vessel  $i$  is moored and processed in time step  $t$ , 0 otherwise.
- (iv)  $\omega_{bit}$ : (binary) equal to 1 if the stem of vessel  $i$  is located at berth segment  $b$  in time step  $t$ , 0 otherwise.
- (v)  $\delta_{i'it}$ : (binary) equal to 1 if vessel  $i$  is moored below vessel  $i'$ ; i.e., vessel  $i$  is moored at berth segments with indices smaller than those of vessel  $i'$ , 0 otherwise. See Figure 2 for illustration.
- (vi)  $\mu_{in}$ : (binary) equal to 1 if vessel  $i$  enters the approach channel to berth in the  $n$ th high water time window, 0 otherwise.
- (vii)  $\nu_{in}$ : (binary) equal to 1 if vessel  $i$  passes through the approach channel and leaves the port in the  $n$ th high water time window, 0 otherwise.

### 4.2. Formulation of the BACASP-TC

**Objective Function.** The objective function is to minimize the total turnaround time of vessels, which is one of the most studied objectives in literature [9–12].

$$\text{Minimize } \sum_{i \in I} (\zeta_i - a_i) \quad (1)$$

#### Tidal Restriction Constraints

$$\lambda_i \geq a_i, \quad \forall i \in I \quad (2)$$

$$\varepsilon_i = \lambda_i + C, \quad \forall i \in I \quad (3)$$

$$\psi_i \geq \sigma_i, \quad \forall i \in I \quad (4)$$

$$\zeta_i = \psi_i + C, \quad \forall i \in I \quad (5)$$

$$\sum_{n \in N} \mu_{in} = 1, \quad \forall i \in I^{TIDE} \quad (6)$$

$$2(n-1)V + 1 - M(1 - \mu_{in}) \leq \lambda_i \leq (2n-1)V + M(1 - \mu_{in}), \quad \forall i \in I^{TIDE}, \forall n \in N \quad (7)$$

$$\sum_{n \in N} \nu_{in} = 1, \quad \forall i \in I^{TIDE} \quad (8)$$

$$\begin{aligned} 2(n-1)V + 1 - M(1 - \nu_{in}) &\leq \psi_i \\ &\leq (2n-1)V + M(1 - \nu_{in}), \quad \forall i \in I^{TIDE}, \forall n \in N \end{aligned} \quad (9)$$

Constraint (2) ensures that any vessel cannot enter the approach channel before its arrival time. Constraint (3) indicates that any vessel must pass through the approach channel before starting processing. Constraint (4) guarantees that a vessel must pass through the approach channel for leaving the port after processing. Constraint (5) defines the leave time of any vessel. Constraints (6) to (9) ensure that a tide-dependent vessel passes through the approach channel within a high water time window for either berthing or leaving the port.

#### Berth Allocation Constraints

$$\sum_{t \in T} \pi_{it} = 1, \quad \forall i \in I \quad (10)$$

$$\sum_{t \in T} \varphi_{it} = 1, \quad \forall i \in I \quad (11)$$

$$\varepsilon_i = \sum_{t \in T} t \pi_{it}, \quad \forall i \in I \quad (12)$$

$$\rho_i = \sum_{t \in T} \theta_{it}, \quad \forall i \in I \quad (13)$$

$$\sigma_i = \sum_{t \in T} t \varphi_{it}, \quad \forall i \in I \quad (14)$$

$$\sigma_i = \varepsilon_i + \rho_i - 1, \quad \forall i \in I \quad (15)$$

$$\sum_{t'=1}^{t-1} \theta_{it'} \leq M(1 - \pi_{it}), \quad (16)$$

$$\forall i \in I, \forall t \in [2, |T|]$$

$$\sum_{t'=t+1}^{|T|} \theta_{it'} \leq M(1 - \varphi_{it}), \quad (17)$$

$$\forall i \in I, \forall t \in [1, |T| - 1]$$

$$\theta_{it} \geq 1 - M(1 - \pi_{it}), \quad \forall i \in I, \forall t \in T \quad (18)$$

$$\begin{aligned} \theta_{it'} + \theta_{i,t-1} - \theta_{it} &\leq 1, \\ \forall i \in I, \forall t \in [2, |T|], \forall t' \in [t+1, |T|] \end{aligned} \quad (19)$$

$$\sum_{b \in B} \omega_{bit} = \theta_{it}, \quad \forall i \in I, \forall t \in T \quad (20)$$

$$\beta_i \geq \sum_{b \in B} (b \omega_{bit}) - M(1 - \theta_{it}), \quad (21)$$

$$\forall i \in I, \forall t \in T$$

$$\beta_i \leq \sum_{b \in B} (b \omega_{bit}) + M(1 - \theta_{it}), \quad (22)$$

$$\forall i \in I, \forall t \in T$$

$$\begin{aligned} \omega_{b,i,t+1} - \omega_{bit} &\geq -M(2 - \theta_{i,t+1} - \theta_{it}), \\ \forall i \in I, \forall b \in B, \forall t \in [1, |T| - 1] \end{aligned} \quad (23)$$

$$\begin{aligned} \omega_{b,i,t+1} - \omega_{bit} &\leq M(2 - \theta_{i,t+1} - \theta_{it}), \\ \forall i \in I, \forall b \in B, \forall t \in [1, |T| - 1] \end{aligned} \quad (24)$$

$$\beta_i + l_i - 1 \leq |B| + M(1 - \theta_{it}), \quad \forall i \in I, \forall t \in T \quad (25)$$

$$\sum_{i \in I} (\theta_{it} l_i) \leq |B|, \quad \forall t \in T \quad (26)$$

$$\sum_{b'=b-l_i+1}^{b+l_i-1} \omega_{b',i',t} \leq M(1 - \omega_{bit}), \quad (27)$$

$$\forall i, i' \in I, i \neq i', \forall t \in T, \forall b \in [l_{i'}, |B| - l_i + 1]$$

$$\begin{aligned} \beta_i + l_i &\leq \beta_{i'} + M(1 - \delta_{ii'}), \\ \forall i, i' \in I, i \neq i' \end{aligned} \quad (28)$$

$$\begin{aligned} \delta_{ii'} + \delta_{i'i} &\leq 1 + M(2 - \theta_{it} - \theta_{i't}), \\ \forall i, i' \in I, \forall t \in T, i \neq i' \end{aligned} \quad (29)$$

$$\begin{aligned} \delta_{ii'} + \delta_{i'i} &\geq 1 - M(2 - \theta_{it} - \theta_{i't}), \\ \forall i, i' \in I, \forall t \in T, i \neq i' \end{aligned} \quad (30)$$

Constraints (10) and (11) state that each vessel starts or ends processing at a certain time step. Constraints (12) and (13) determine the processing start time of vessel  $i$  and its processing time, respectively. Constraints (14) and (15) determine the processing end time  $\sigma_i$  of vessel  $i$ . Constraints (16) and (17) guarantee that vessel  $i$  cannot be processed either before its berthing time or after its completion time. Constraint (18) ensures that vessel  $i$  starts processing once it has been berthed. Constraint (19) guarantees that the processing of any vessel cannot be interrupted. Constraint (20) states that when a vessel is on processing, its stem must be located at some berth segment. Constraints (21) and (22) determine the berth location of vessel  $i$ 's stem. Constraints (23) and (24) denote that vessel  $i$  stays still during processing. Constraint (25) shows that all the vessels must berth within the boundaries of the quay. Constraint (26) means that the total length of vessels being on processing is no more than the quay length at any time. Constraint (27) avoids the conflicting situation where two vessels occupy the same berth segments at any time step. Constraint (28) defines the variable  $\delta_{ii'}$ , which is equal to 1 if vessel  $i$  is moored at berth segments with indices smaller than those of vessel  $i'$ , 0 otherwise. Constraints (29) and (30) guarantee that only one of  $\delta_{ii'}$  and  $\delta_{i'i}$  equals 1.

#### QC Scheduling Constraints

$$\sum_{i \in I} \sum_{q \in Q} \eta_{qit} \leq |Q|, \quad \forall t \in T \quad (31)$$

$$\sum_{t \in T} \sum_{q \in Q} \eta_{qit} \geq c_i, \quad \forall i \in I \quad (32)$$

$$\sum_{i \in I} \eta_{qit} \leq 1, \quad \forall t \in T, \forall q \in Q \quad (33)$$

$$\sum_{q \in Q} \eta_{qit} \leq q_i^{\max} + M(1 - \theta_{it}), \quad (34)$$

$$\forall i \in I, \forall t \in T$$

$$\sum_{q \in Q} \eta_{qit} \geq q_i^{\min} - M(1 - \theta_{it}), \quad (35)$$

$$\forall i \in I, \forall t \in T$$

$$\sum_{q \in Q} \eta_{qit} \leq M\theta_{it}, \quad \forall i \in I, \forall t \in T \quad (36)$$

$$\eta_{q'it} + \eta_{q-1,it} - \eta_{qit} \leq 1, \quad (37)$$

$$\forall i \in I, \forall t \in T, \forall q \in [2, |Q|], \forall q' \in [q+1, |Q|]$$

$$q < q' + M(3 - \eta_{qit} - \eta_{q'i't} - \delta_{ii'}), \quad (38)$$

$$\forall i, i' \in I, \forall t \in T, \forall q, q' \in Q, i \neq i', q \neq q'$$

$$\sum_{q \in Q} \eta_{qit} - \sum_{q \in Q} \eta_{qit'} \leq 1 + M(2 - \theta_{it} - \theta_{it'}), \quad (39)$$

$$\forall i \in I, \forall t, t' \in T, t \neq t'$$

$$\sum_{q \in Q} \eta_{qit} - \sum_{q \in Q} \eta_{qit'} \geq -1 + M(2 - \theta_{it} - \theta_{it'}), \quad (40)$$

$$\forall i \in I, \forall t, t' \in T, t \neq t'$$

Constraint (31) guarantees that the total number of QCs assigned to vessels at any time cannot exceed the number of available QCs. Constraint (32) ensures that each vessel must complete its processing within the time horizon. Constraint (33) makes sure that each QC processes at most one vessel in any time step. Constraints (34) and (35) guarantee when vessel  $i$  is under processing, the number of QCs assigned to it must be within the range  $[q_i^{\min}, q_i^{\max}]$ . Constraint (36) states that any QC cannot be assigned to a vessel not being processed. Constraint (37) describes the constraint of QC scheduling such that if both QCs  $q+1$  and  $q-1$  are assigned to vessel  $i$ , then QC  $q$  must also be assigned to the vessel. Constraint (38) states the relative location consistence between QCs and the vessels being processed at any time. Constraints (39) and (40) ensure that for any two consecutive time steps, the difference on the number of QCs assigned to the same vessel is no more than one.

#### Variable Constraints

$$\lambda_i, \varepsilon_i, \rho_i, \sigma_i, \psi_i, \zeta_i, \beta_i \in \mathbb{Z}_0^+, \quad \forall i \in I \quad (41)$$

$$\eta_{qit}, \pi_{it}, \varphi_{it}, \theta_{it}, \omega_{bit}, \delta_{ii'}, \mu_{in}, \gamma_{in} \in \{0, 1\}, \quad (42)$$

$$\forall i, i' \in I, i \neq i', \forall t \in T, \forall b \in B, \forall q \in Q, \forall n \in N$$

## 5. Solution Algorithms

As the BAP is NP-hard [24], the NP-hardness of the BACASP-TC problem is straightforward. For small instances, it can be solved by CPLEX or LINGO solver, in which the branch and cut algorithm is mainly employed. In practice, however, the problem cannot be solved by the exact algorithm within a reasonable time due to its large input scale. Hence, we employ heuristic algorithms to solve the BACASP-TC problem with large instances. Heuristic algorithms are to seek near-optimal solutions for optimization problems, especially in the cases with large inputs. We propose three heuristic algorithms, Genetic Algorithm (GA), Hybrid Particle Swarm Optimization (HPSO), and Hybrid Simulated Annealing (HSA) below.

**5.1. Solution Representation.** In implementing the three heuristics, the first step is to code any solution. We use an integer code scheme such that the code consists of  $|I|$  subsections, corresponding to the  $|I|$  vessels, respectively. Each subsection comprises two codes. The first code denotes the index of a vessel, and the position of the index in the code sequence represents the order of vessel berthing or processing. The second code indicates the processing time of the vessel. First, we randomly generate a vessel processing sequence. We also calculate the minimum possible processing time  $\rho_i^{\min}$  and the maximum possible processing time  $\rho_i^{\max}$  for vessel  $i$ , where  $\rho_i^{\min} = c_i/q_i^{\max}$  and  $\rho_i^{\max} = c_i/q_i^{\min}$ . A random value within interval  $[\rho_i^{\min}, \rho_i^{\max}]$  is then generated as the preset processing time. A code representation for an instance with six vessels is depicted in Figure 3. Notice that QC scheduling is not included in the code, which will be elaborated in Section 5.2.

**5.2. Objective Value Calculation.** The objective function value is used to evaluate the quality of a solution produced by any heuristic algorithm. Below we present a specific method in three steps to calculate the objective value of any given solution.

*Step 1.* Determine the berthing location and the time with tidal constraints in the approach channel for each vessel by Algorithm 1 to be described later.

*Step 2.* Assign QCs to vessels in every time step by Algorithm 2, as shown later on.

*Step 3.* If the assignment of QCs for vessels in Step 2 is feasible (i.e.,  $f = True$ ), then calculate the objective value  $obj$  of the solution where  $obj = \sum_{i \in I} (\zeta_i - a_i)$ ; otherwise, set  $obj$  as a sufficiently large constant  $M$  such as 10000, implying that the solution is unreasonable and is to be eliminated during calculation iterations of the heuristics. As the considered BACASP-TC problem is of a minimization objective and the heuristics take the reciprocal of the objective value  $obj$  of a solution as its fitness value, so a solution with a larger objective value is more likely to be eliminated in the evolution process of any heuristic algorithm.

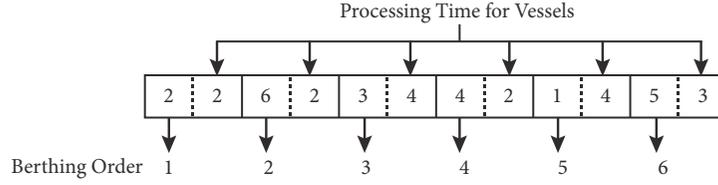


FIGURE 3: A solution representation for an instance with six vessels.

**Input:** A solution representation;  
**Output:** The berthing position  $\beta_i$  and the processing start time  $\varepsilon_i$  for each vessel  $i$ ;

- (1) **Initialize**  $V_{berthed} = \emptyset$ ; // Set of berthed vessels;
- (2) **for**  $j = 1 : |I|$  **do**
- (3) Let  $i$  be the index of the vessel whose berthing order in the solution representation is  $j$ ;
- (4) **Initialize**  $berthed = false$ ,  $y_{rec} = 0$ ,  $t_i^{enter} = 0$ ; //  $t_i^{enter}$  is the earliest time for vessel  $i$  to enter the approach channel;
- (5) **if** vessel  $i \in I^{TIDE}$  **then**
- (6) Let  $t_i^{enter}$  be the earliest time step in high water time windows, which is greater than or equal to  $a_i$ ;
- (7) **else**
- (8)  $t_i^{enter} = a_i$ ;
- (9) **end**
- (10)  $x_{rec} = t_i^{enter} + C$ ;
- (11)  $rect_i = (x_{rec}, y_{rec}, \rho_i, l_i)$ ; //  $rect_i$  is the rectangle in the time-wharf plan, which represents the schedule of vessel  $i$ ;
- (12) **Search** the possible locations for  $rect_i$  and **sort** them in the ascending order using the time-coordinate as the primary key and the wharf-coordinate as the secondary key;
- (13) **while**  $berthed == false$  **do**
- (14) **if** ( $i \in I^{FREE}$ , and  $rect_i$  does not overlap with any rectangle in  $V_{berthed}$  and is within the boundary of the time-wharf plan) **or** ( $i \in I^{TIDE}$ , the time-coordinate of the bottom-left corner of  $rect_i$  is in high water time window, and  $rect_i$  does not overlap with any rectangle in  $V_{berthed}$  and is within the boundary of the time-wharf plan) **then**
- (15)  $\varepsilon_i = x_{rec}$ ,  $\beta_i = y_{rec}$ ;
- (16)  $V_{berthed} = \{v_i\} \cup V_{berthed}$ ;
- (17)  $berthed = true$ ;
- (18) **break**;
- (19) **else**
- (20) **Move**  $rect_i$  to the next possible location and **update**  $x_{rec}$  and  $y_{rec}$ ;
- (21) **if** there is no more candidate for possible location of  $rect_i$  **then**
- (22)  $\beta_i = 0$ ;
- (23) let  $\varepsilon_i = \bar{t} + C$  where  $\bar{t}$  is the first time step in the next high water time window;
- (24)  $V_{berthed} = \{v_i\} \cup V_{berthed}$ ;
- (25)  $berthed = true$ ;
- (26) **end**
- (27) **end**
- (28) **end**
- (29) **end**

ALGORITHM 1: The greedy heuristic for BAP.

After the coding process in Section 5.1, a simple and preliminary scheduling scheme for the BACASP-TC problem is presented, where it includes the vessel berthing sequence and the processing times of vessels. The two algorithms, Algorithms 1 and 2, determine the berthing locations, the processing start times of vessels, and QC scheduling.

In Algorithm 1, a greedy heuristic method is constructed to assign berthing locations and start times of processing

vessels with the tidal constraint, based on the bottom-left bin-packing heuristic in [25]. First, according to a vessel's preset processing time in the solution together with its length, both the width and height of the rectangle can be determined, as shown in Figure 4. Secondly, the rectangle is to be placed in the time-wharf plan, where point  $(t_i^{enter} + C, 0)$  is set as the original possible location. In Figure 4(a), the original possible location for processing vessel 6 is point 1. The rest

**Input:** The berthing position  $\beta_i$  and the processing start time  $\varepsilon_i$ ;  
**Output:** The time-variant QC scheduling for vessels and the assignment feasibility  $f$ ;

```

(1) Initialize  $f = true$ ;
(2) for  $t = 1 : |T|$  do
(3)   Initialize  $distr\{t, 1\} = \emptyset, distr\{t, 2\} = 0, distr\{t, 3\} = 0$ ;
(4)   for  $i = 1 : |I|$  do
(5)     if vessel  $i$  is being processed in time step  $t$  then
(6)        $distr\{t, 1\} = \{i\} \cup distr\{t, 1\}$ ;
(7)        $distr\{t, 2\} = distr\{t, 2\} + 1$ ;
(8)     end
(9)   end
(10) end
(11) for  $t = 1 : |T|$  do
(12)    $sum = 0, avg = \emptyset, percent = 0$ ;
(13)   for  $j = 1 : distr\{t, 2\}$  do
(14)      $i = distr\{t, 1\}(j)$ ;
(15)     if vessel  $i$  starts processing at time step  $t$  then
(16)        $remain_i^{con} = c_i$ ;
(17)        $remain_i^{tim} = \rho_i$ ; //  $\rho_i$  is the preset processing time for vessel  $i$  in the solution representation;
(18)     end
(19)      $avg_j = \frac{remain_i^{con}}{remain_i^{tim}}$ ;
(20)      $sum = sum + avg_j$ ;
(21)   end
(22)   for  $j = 1 : distr\{t, 2\}$  do
(23)      $i = distr\{t, 1\}(j)$ ;
(24)      $percent_j = \frac{avg_j}{sum}$ ;
(25)      $distr\{t, 3\}(j) = round(|Q| * percent_j)$ ;
(26)     if  $distr\{t, 3\}(j) < q_i^{min}$  then
(27)        $distr\{t, 3\}(j) = q_i^{min}$ ;
(28)     end
(29)     if  $distr\{t, 3\}(j) > q_i^{max}$  then
(30)        $distr\{t, 3\}(j) = q_i^{max}$ ;
(31)     end
(32)     Adjust  $distr\{t, 3\}(j)$  to satisfy Constraints (39) and (40);
(33)      $remain_i^{con} = remain_i^{con} - distr\{t, 3\}(j), remain_i^{tim} = remain_i^{tim} - 1$ ;
(34)     if  $remain_i^{tim} == 0$  and  $remain_i^{con} > 0$  then
(35)        $f = false$ ;
(36)       break;
(37)     end
(38)   end
(39) end

```

ALGORITHM 2: The greedy heuristic for time-variant QC scheduling.

four possible locations are determined from bottom to top and from left to right. Next, the feasibility of placing vessel 6 is verified from point 1 to 5, according to the conditions in Line 14 of Algorithm 1. If vessel 6 is tide independent, it is placed in point 4, i.e., rectangle (1) in Figure 4(b). If otherwise vessel 6 is tide dependent, then all the five points are verified infeasible and the location of the vessel is to be determined by Lines 22 to 25 of Algorithm 1. In this case, the rectangle (2) in Figure 4(b) is the final location of vessel 6. Note that vessel 6 starts processing at time step 14 but not 13, because it takes one time step to pass through the approach channel in the high water time window.

After determining the vessel berthing locations and processing start times, we employ Algorithm 2 to obtain the time-variant QC scheduling for vessels and check the assignment feasibility as well. According to the schedule of vessels from Algorithm 1, the specific berthed vessels and the number of vessels being processed in each time step  $t$  are determined and stored in sets  $distr\{t, 1\}$  and  $distr\{t, 2\}$ , respectively. The algorithm greedily assigns QCs to the vessels in each time step based on the proportions of their average workloads. The numbers of QCs assigned to the vessels in set  $distr\{t, 1\}$  at each time step are placed in set  $distr\{t, 3\}$ . If there is any vessel uncompleted in its preset processing time steps, it implies that the assignment of vessel location and the QC scheduling

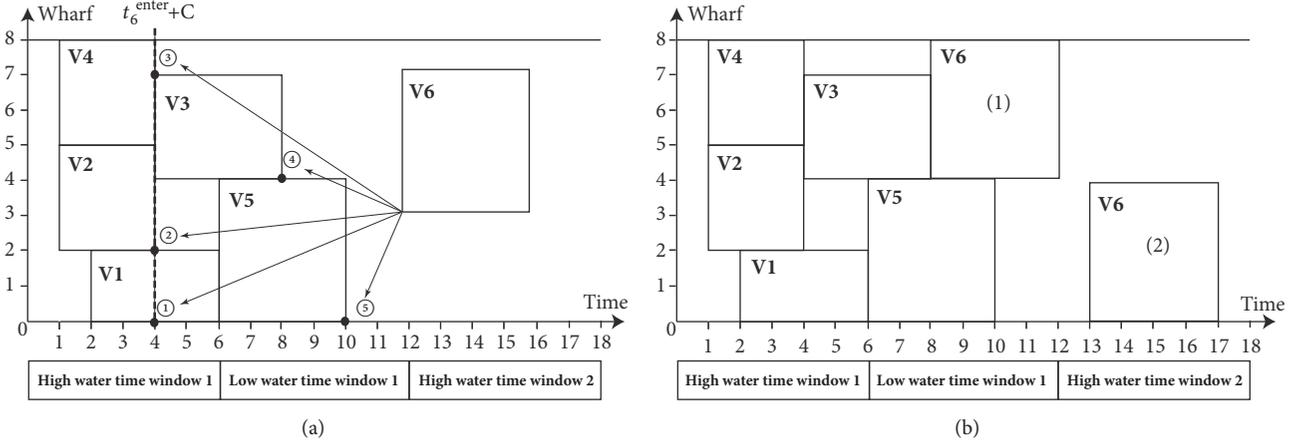


FIGURE 4: Illustration of possible locations for processing vessel  $V_6$ .

are infeasible. In this case the corresponding solution will be discarded in the iterations of the three heuristics. To better understand Algorithm 2, the process of solving the example in Figure 2 is given in Table 1, where the workloads of vessels 1 to 7 are equal to 9, 9, 5, 14, 14, 15, and 11 QC-time-steps, respectively. In time step 1 or the first row of Table 1, there is no vessel on processing and all the columns are of initial values 0 or 0. In time step 2 or the second row, the numbers in the sixth column  $distr\{t, 1\}$  mean that vessels 1 and 2 start their processing. Columns  $remain_i^{con}$  and  $remain_i^{tim}$  show their original workloads {9,9} and preset processing time steps in the solution representation {3,3}. So, column  $avg$  gives their average workloads {3,3} in the three time steps, and the resulting proportions {0.5,0.5} between the two vessels are given in column  $percnt$ . In column  $distr\{t, 3\}$ , the original allocated QC numbers (4,4), as given in the brackets, for the two vessels are out of their allowed ranges. Thus they are replaced with their nearest available numbers 3, 3 of allocated QCs. That is, if the original allocated QC number for any vessel in  $distr\{t, 1\}$  is smaller (or larger) than its lower (or upper) bound of allocated QC number, then the number is replaced by the lower (or upper) bound. The remaining rows of the table can be similarly explained as in the second row.

The above two greedy algorithms decode a solution of BACASP-TC and calculate the objective function value. Next we propose three heuristic algorithms, Genetic Algorithm (GA), Hybrid Particle Swarm Optimization (HPSO), and Hybrid Simulated Annealing (HSA) to produce optimal or near-optimal solutions for large instances. We only present the frameworks of the three heuristics due to page limitation. For details of the heuristics, readers may refer to [26–28].

**5.3. Genetic Algorithm.** Genetic Algorithm (GA) is a method for tackling both constrained and unconstrained optimization problems by mimicking biological evolution in a natural selection process. Since its first introduction by Holland in 1975 [29], GA has demonstrated its broad applicability in many areas. In this paper, we apply GA to solve BAP and QCSP simultaneously. It generates an initial population of solutions or chromosomes and then repeatedly generates

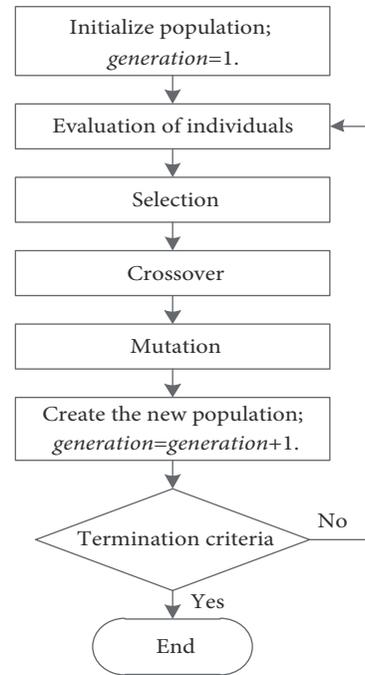


FIGURE 5: Flowchart of GA.

offspring generations by a series of bioinspired operations including selection, crossover, and mutation, which are adopted from [30]. In each evolution, Algorithms 1 and 2 are adopted to calculate the objective values as well as fitness values of individual solutions. Solutions with high fitness values are selected to form the population in the next generation. The flow chart of GA is depicted in Figure 5.

**5.4. Hybrid Particle Swarm Optimization Algorithm.** Particle Swarm Optimization (PSO) is inspired by social behavior of bird flocking or fish schooling. The algorithm is carried out by means of a swarm of candidate solutions or particles. According to some rules, these particles move around in the search space. The movements of the particles are guided by their own

TABLE 1: An example for Algorithm 2.

$t$	$remain_i^{con}$	$remain_i^{tim}$	$avg$	$percnt$	$distr\{t, 1\}$	$distr\{t, 2\}$	$distr\{t, 3\}$
1	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$	0	$\emptyset$
2	9, 9	3, 3	3, 3	0.5, 0.5	1, 2	2	3, 3 (4, 4)
3	6, 6, 5	2, 2, 4	3, 3, 1.25	0.41, 0.41, 0.17	1, 2, 3	3	3, 3, 1
4	3, 3, 4	1, 1, 3	3, 3, 1.33	0.41, 0.41, 0.18	1, 2, 3	3	3, 3, 1
5	3, 14	2, 4	1.5, 3.5	0.3, 0.7	3, 4	2	2, 4 (2, 5)
6	1, 10	1, 3	1, 3.33	0.23, 0.77	3, 4	2	2, 4 (2, 5)
7	6, 14	2, 4	3, 3.5	0.46, 0.54	4, 5	2	3, 4
8	3, 10	1, 3	3, 3.33	0.47, 0.53	4, 5	2	3, 4
9	6, 11	2, 4	3, 2.75	0.52, 0.48	5, 7	2	4, 3
10	2, 8	1, 3	2, 2.67	0.43, 0.57	5, 7	2	3, 3 (3, 4)
11	5	2	2.5	1	7	1	3 (7)
12	2	1	2	1	7	1	3 (7)
13	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$	0	$\emptyset$
14	15	3	5	1	6	1	5 (7)
15	10	2	5	1	6	1	5 (7)
16	5	1	5	1	6	1	5 (7)

best known positions in the search space as well as the entire swarm's best known position. As illustrated in Figure 6, the proposed Hybrid PSO (HPSO) initializes a set of particles with their original locations  $X_i$  and objective function values  $f(X_i)$  in the solving space. The original locations  $X_i$  of particles are taken as their initial individual best locations  $Pbest_i$ . Among the original locations of particles, the one that has the smallest objective function value is selected as the population best location  $Gbest$ . In order to seek a better location or solution, each particle location needs crossover and mutation with its own previous best location  $Pbest_i$  and the population previous best location  $Gbest$ , respectively. The crossover and mutation evolution operators are from the GA in the above subsection. After moving to the new location, each particle updates its  $Pbest_i$ , provided that the objective function value of its new location is less than the objective function value of its  $Pbest_i$ . Likewise, update the  $Gbest$  if the objective function value of its new location is less than the objective function value of  $Gbest$ . By repeating the above process, a satisfying solution is eventually obtained.

**5.5. Hybrid Simulated Annealing Algorithm.** The inspiration of simulated annealing (SA) algorithm comes from metallurgical annealing process, and it is first applied to the field of combinatorial optimization by Kirkpatrick et al. [31]. SA initializes with a solution  $X_1$  at a high temperature  $T = T_0$ , and the temperature  $T$  gradually drops to a set value  $T_{end}$  at a certain rate  $q$ . At each temperature  $T$ , the algorithm generates a new solution  $X_2$  and measures its quality  $Y_2$ . If the new solution  $X_2$  is better than the current  $X_1$ ,  $X_1$  is replaced by  $X_2$ . Otherwise, whether the current solution  $X_1$  is replaced or not depends on a probability that is progressively decreased with the reducing of temperature  $T$ . The above operations are repeated for specified times that is called chain length  $L$  at every temperature  $T$ . This algorithm can effectively avoid being trapped in local minima and is able to

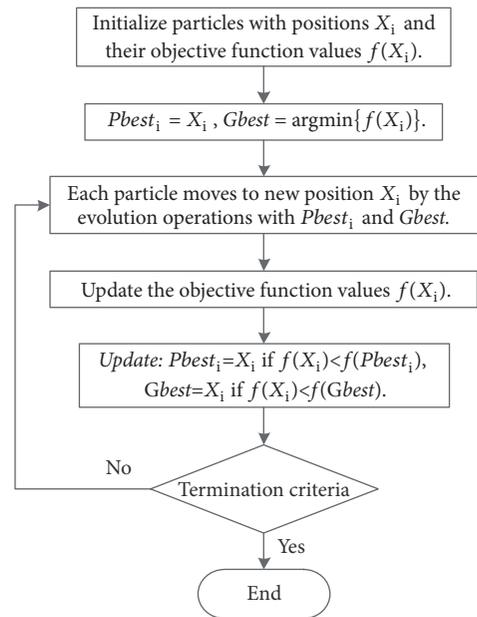


FIGURE 6: Flowchart of HPSO.

explore globally for optimal solutions. In this work the Hybrid SA (HSA) combines the search mechanism of simulated annealing algorithm with genetic operators such as crossover and mutation when a new solution is generated. The flow chart of HSA is given in Figure 7.

## 6. Computational Experiments

In this section, a series of computational experiments are conducted to evaluate the performances of the proposed heuristics and demonstrate the impact of tides in the integrated scheduling of berth and QCs. First, we compare

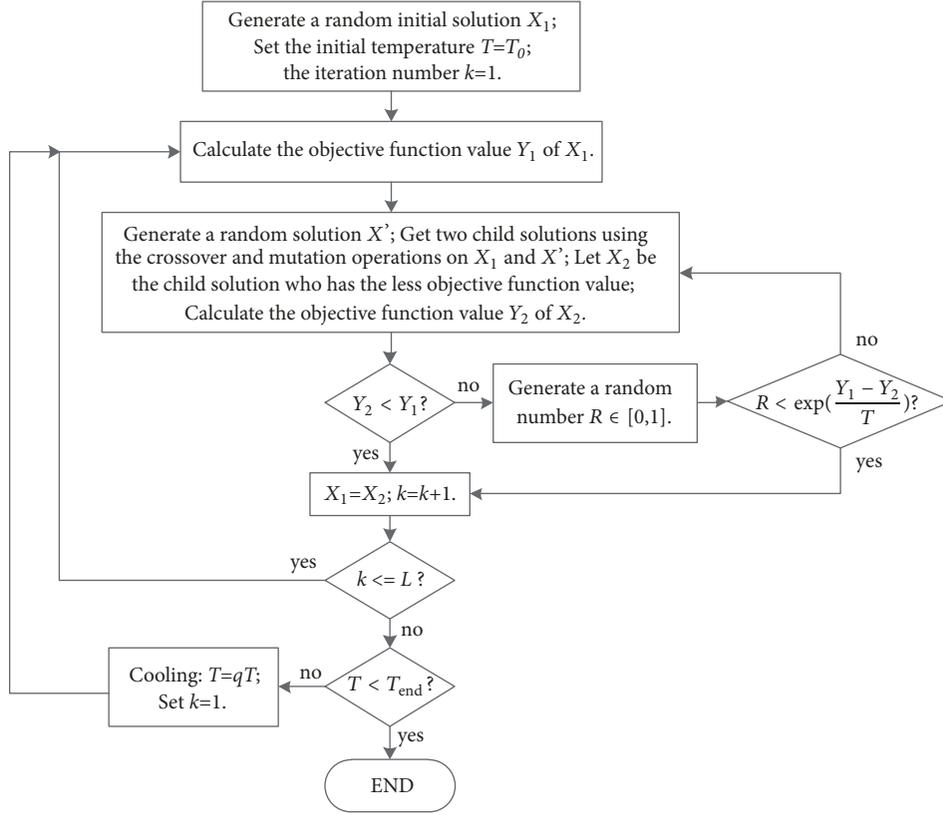


FIGURE 7: Flowchart of HSA.

the performances of the three heuristics with the exact solutions obtained via CPLEX for small instances. We then make comparison between the heuristic algorithms for large instances. Moreover, we analyze the tidal influence on the processing schedule of vessels and the utilization of berth and QCs.

We first make selections on the parameter settings for the heuristics to enhance their experimental performances. To carry out the preliminary test, five instances are randomly generated from a small size scenario  $(|I|, |B|, |Q|, |T|) = (6, 12, 8, 42)$ , and five instances from a large size scenario  $(|I|, |B|, |Q|, |T|) = (30, 30, 15, 120)$ . For all the heuristic algorithms, the various combinations of their parameter candidate values are tested on the 10 instances. The parameter values in the best parameter combinations are listed in Table 2 and adopted in all subsequent computational experiments. The three proposed heuristic algorithms will be terminated if one of the following termination criteria is met: (1) the number of generations or the terminal temperature is reached; (2) the quality of the solution is not improved in 50 consecutive generations; and (3) the running time exceeds 2000 seconds.

The computational experiments are conducted using a PC with Intel Core i7-4700 3.4 GHz processor and 16 GB RAM. The algorithms are coded in Matlab R2014b, and the BACASP-TC model is solved with CPLEX 12.6 called by Matlab. In accordance with [32], the heuristics run each

instance 10 times, and the best results among the 10 runs are reported.

**6.1. Instance Generation.** Following the fundamental assumptions of BACASP-TC, the duration of any high/low water time window is equal to 6 time steps. For the three types of vessels, feeder and medium are tide independent, while jumbo is tide dependent. In accordance with [18], the detailed information of vessels in any instance is listed in Table 3, where the workload of a vessel is in units of QC-time-step. For example, the workload of a feeder varies between 10 and 20 QC-time-steps. The interarrival times of vessels are exponentially distributed with a mean of 2 time steps, as in [33].

## 6.2. Algorithm Performance Evaluation

**6.2.1. Small Instances.** For small instances, the wharf length, the number of available QCs, and the planning horizon  $(|B|, |Q|, |T|)$  are set as  $(12, 8, 42)$ . For two cases with  $|I| = 5$  and  $|I| = 6$  vessels, we randomly generate ten instances for each. Table 4 shows the computational results of CPLEX and the three heuristic algorithms. In the table, column 1 lists the indices of the instances, while columns 2 and 3 are, respectively, the objective function value of exact solution and the computational time by CPLEX. Columns 4, 7, and 10 are, respectively, the objective function values of the optimal

TABLE 2: Parameter selection for heuristic algorithms.

Heuristic algorithm	Parameters	Parameter candidate values	Best parameter value
GA	Crossover probability	0.7, 0.8, 0.9	0.9
	Mutation probability	0.05, 0.10, 0.15	0.1
	Population size	100, 200	200
	Number of generations	150, 200	150
HPSO	Population size	100, 200, 300	200
	Number of generations	100, 150, 200	150
HSA	Initial temperature	800, 1000, 1200	1000
	Cooling rate	0.85, 0.90, 0.95	0.9
	Chain length	100, 200	200
	Terminal temperature	0.005, 0.001	0.001

TABLE 3: Specifications of different types of vessels.

Vessel class	Percentage	Vessel length	Range of used QCs	Range of workload
Feeder	1/3	3-4	1-2	10-20
Medium	1/3	5-6	2-4	20-40
Jumbo	1/3	7-8	4-6	40-70

solutions by GA, HPSO, and HSA, and the corresponding running times are given in columns 5, 8, and 11, respectively. The relative errors of objective value between CPLEX and the three heuristics are defined as follows:  $G_{GA} = (O_{GA} - Obj)/Obj \times 100\%$ ,  $G_{HPSO} = (O_{HPSO} - Obj)/Obj \times 100\%$ , and  $G_{HSA} = (O_{HSA} - Obj)/Obj \times 100\%$ . First, we observe that all the heuristics obtain optimal solutions as CPLEX does for most of the small instances. For instances 6-2 and 6-7, CPLEX outputs feasible but not exact solutions within the preset 36000 seconds, while the heuristics produce better solutions in much less computational time. Secondly, the running time of CPLEX increases rapidly even for a small growth in the number of vessels, while the heuristics solve all the instances within 600 seconds. Thirdly, although the three heuristics produce the same optimal solution as CPLEX does, the convergence speed of GA outperforms HPSO and HSA.

**6.2.2. Large Instances.** For large instances with more than 6 vessels, CPLEX can hardly produce exact solutions in 10 hours. Thus, we compare the performances between the three heuristic algorithms for large instances. We test three scenarios with  $(|I|, |B|, |Q|, |T|) = (20, 20, 10, 90)$ ,  $(30, 30, 15, 120)$  and  $(40, 40, 20, 150)$ , and 10 instances are randomly generated for each scenario. Table 5 summarizes the computational results for the 30 instances, where columns 2, 5, and 8 display, respectively, the best objective function value obtained by GA, HPSO, and HSA in 10 runs. The corresponding running times are reported in columns 3, 6, and 9, respectively.  $Gap_{GA} = (Obj_{GA} - Obj_{HSA})/Obj_{HSA} \times 100\%$  and  $Gap_{HPSO} = (Obj_{HPSO} - Obj_{HSA})/Obj_{HSA} \times 100\%$  are the relative errors of the objective values. It is observed that HSA outperforms GA and HPSO with respect to the solution quality. Moreover, for most instances the relative errors between the best objective values of solutions produced by the three heuristics are within 5%. Hence, we conclude that all the three algorithms can solve large size problems effectively.

**6.2.3. Analysis of Tidal Impacts.** We carry out numerical experiments to further reveal how the factor of tides impacts the turnaround times of vessels and the utilization of berth and QCs. Four cases 0.1:0.3:0.6(P1), 0.3:0.3:0.4(P2), 0.5:0.2:0.3(P3), and 0.7:0.2:0.1(P4) are investigated according to the different portions of jumbos, mediums, and feeders. For example, in case P1, 10%, 30%, and 60% percentages of the vessels are jumbos, mediums, and feeders, respectively. For each scenario, again 10 instances with  $(|I|, |B|, |Q|, |T|) = (30, 30, 15, 120)$  are randomly generated and solved by HSA algorithm, which performs best among the three heuristics in the above experiments. Moreover, we consider three scenarios with different durations  $V = 0, 6$  and  $12$  of high/low water time window. For example, in the first scenario with  $V = 0$ , there is no tidal impacts involved and all the vessels can pass through the channel at any time, while in the third scenario with  $V = 12$ , each high or low water time window is of length equal to 12 time steps.

The objective values of solutions and the gaps between them are listed in Table 7, in which  $Gap_{V6} = (Obj_{V6} - Obj_{V0})/Obj_{V0} \times 100\%$  and  $Gap_{V12} = (Obj_{V12} - Obj_{V0})/Obj_{V0} \times 100\%$ . As shown in Figure 8, the average total turnaround times of vessels increase as the proportion of the tide-dependent vessels (i.e., jumbos) rises in any duration of high/low water time window. This result verifies that the large amount of loading and unloading of jumbos inevitably extends their turnaround times. Moreover, for each case with a fixed proportion of different vessels, it can be seen that the tides result in an obvious increase in vessel turnaround times, compared with the ideal situation without tides, and the tidal effects become more significant with the increase of the duration  $V$  of high/low water time window.

In order to explain the above results, we summarize the average waiting times and handling times of tide-dependent vessels (i.e., jumbos) in Table 6 based on the instance solutions in Table 7. It shall be noted that column

TABLE 4: Experimental comparison of CPLEX and the heuristic algorithms for small instances.

Instance	CPLEX		GA			HPSO			HSA		
	Obj	Time	$O_{GA}$	Time	$G_{GA}$	$O_{HPSO}$	Time	$G_{HPSO}$	$O_{HSA}$	Time	$G_{HSA}$
5-1	89	1190	89	39	0.0%	89	513	0.0%	89	246	0.0%
5-2	87	5901	87	40	0.0%	87	534	0.0%	87	254	0.0%
5-3	91	10417	91	37	0.0%	91	489	0.0%	91	249	0.0%
5-4	73	1580	73	34	0.0%	73	466	0.0%	73	230	0.0%
5-5	84	1538	84	38	0.0%	84	502	0.0%	84	245	0.0%
5-6	87	1288	87	41	0.0%	87	527	0.0%	87	251	0.0%
5-7	92	1258	92	40	0.0%	92	504	0.0%	92	255	0.0%
5-8	86	1287	86	40	0.0%	86	505	0.0%	86	247	0.0%
5-9	78	834	78	43	0.0%	78	559	0.0%	78	261	0.0%
5-10	83	391	83	36	0.0%	83	479	0.0%	83	232	0.0%
6-1	91	8637	91	45	0.0%	91	512	0.0%	91	263	0.0%
6-2	116*	36283	106	48	-	106	580	-	106	277	-
6-3	96	14822	96	46	0.0%	96	530	0.0%	96	254	0.0%
6-4	84	1442	84	45	0.0%	84	525	0.0%	84	263	0.0%
6-5	89	3497	89	47	0.0%	89	552	0.0%	89	271	0.0%
6-6	83	11358	83	45	0.0%	83	513	0.0%	83	257	0.0%
6-7	99*	36324	97	52	-	97	597	-	97	290	-
6-8	89	4474	89	48	0.0%	89	548	0.0%	89	278	0.0%
6-9	108	10970	108	52	0.0%	108	601	0.0%	108	292	0.0%
6-10	95	33375	95	47	0.0%	95	534	0.0%	95	265	0.0%

Note: The running time is in seconds.

\*The running time of CPLEX exceeds 36000 seconds in the instance.

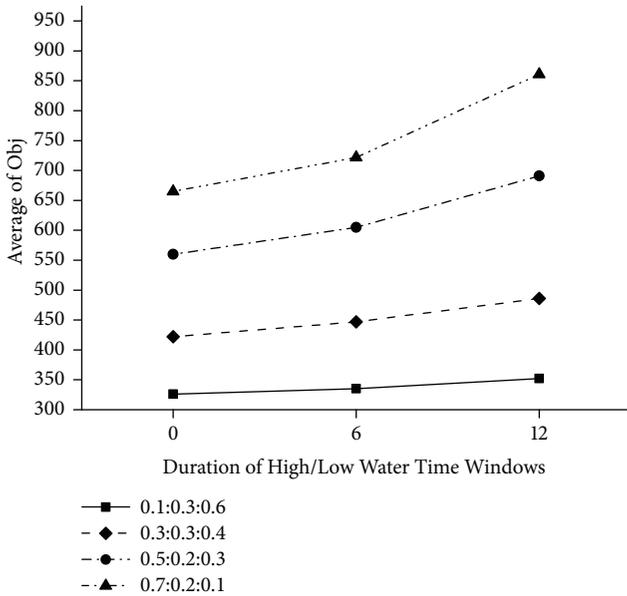


FIGURE 8: Tidal impact on vessel turnaround time.

$n_{jumbo}$  represents the total number of tide-dependent vessels in ten instances for each combination of vessel ratio and  $V$ . Columns  $Avg_{hdling}$ ,  $Avg_{wtng}_{pre}$ , and  $Avg_{wtng}_{post}$  mean the average handling time, the average waiting time

before berthing and after processing of  $n_{jumbo}$  tide-dependent vessels, respectively. Columns  $value$  and  $Incrs_{td}$  denote the corresponding value and increase caused by tides. To take an example of Row 0.3:0.3:0.4, the number of tide-dependent vessels in each instance is  $30 * 30\% = 9$ , and  $n_{jumbo} = 10 * 9 = 90$ . First, these results show that the intensive arriving of more and more jumbos and the increase of duration of high/low water time window prolong the waiting times to enter the approach channel for berthing. Secondly, it is found that the tide effect on the processing times is very small, and the average processing times are all close to the mean processing time of jumbos, which is the mean workload divided by the mean number of used QCs, i.e.,  $55/5=11$ . Because the average processing time is nearly twice of  $V = 6$  or close to  $V = 12$  and jumbos must start processing in high water time window, jumbos have larger chances to meet low water time windows in tidal condition of  $V = 12$  than  $V = 6$  when they finish their stevedoring to pass through the approach channel. It results in the increasing waiting times after processing as the value of  $V$  becomes larger. Thirdly, tides have a greater effect on the waiting time of a jumbo before berthing than that after processing. This provides valuable suggestions for container terminal managers that reasonable times for vessels calling at the port shall be given to reduce unnecessary waiting and improve service quality.

Next, according to the solutions of the instances in Table 7, the corresponding utilization ratios of berth and QCs are further reported in Tables 8 and 9 where BU and QCU

TABLE 5: Numerical comparison between the heuristics for large instances.

Instance	GA			HPSO			HSA	
	$Obj_{GA}$	$Time(s)$	$Gap_{GA}$	$Obj_{HPSO}$	$Time(s)$	$Gap_{HPSO}$	$Obj_{HSA}$	$Time(s)$
20-1	373	1240	6.9%	351	578	0.6%	349	1338
20-2	401	729	7.5%	394	1456	5.6%	373	2000
20-3	432	1026	4.9%	436	1094	5.8%	412	1467
20-4	420	1821	4.5%	418	906	4.0%	402	1985
20-5	414	1823	7.8%	402	1845	4.7%	384	1779
20-6	467	1939	2.0%	479	670	4.6%	458	1971
20-7	494	1719	5.8%	495	504	6.0%	467	1678
20-8	350	1807	0.9%	358	1796	3.2%	347	1680
20-9	319	674	2.2%	330	975	5.8%	312	1329
20-10	377	1915	4.1%	378	972	4.4%	362	1447
30-1	449	1793	2.0%	448	1180	1.8%	440	1822
30-2	445	1977	5.7%	435	1522	3.3%	421	1903
30-3	463	1439	5.0%	457	1130	3.6%	441	1531
30-4	431	1530	4.1%	430	896	3.9%	414	1700
30-5	445	1913	2.3%	440	1538	1.1%	435	1518
30-6	418	1231	0.5%	435	1661	4.6%	416	1482
30-7	527	1858	6.7%	497	1200	0.6%	494	1920
30-8	599	1716	8.5%	571	1642	3.4%	552	1845
30-9	405	1409	0.5%	413	1415	2.5%	403	1864
30-10	431	1189	8.0%	403	1503	1.0%	399	1880
40-1	570	1514	4.2%	553	1796	1.1%	547	1713
40-2	596	1411	2.6%	585	1588	0.7%	581	1636
40-3	553	1871	0.4%	575	1660	4.4%	551	1759
40-4	628	1903	2.1%	644	1633	4.7%	615	1861
40-5	481	1979	1.1%	486	1689	2.1%	476	1827
40-6	503	1177	2.0%	515	1154	4.5%	493	1557
40-7	474	1545	0.2%	476	1569	0.6%	473	1860
40-8	634	1686	0.3%	635	1854	0.5%	632	1821
40-9	489	1709	1.5%	485	1743	0.6%	482	1855
40-10	464	1892	0.7%	472	1736	2.4%	461	1978

TABLE 6: Tidal impact on jumbos waiting times.

Vessel ratio	V	$n_{jumbo}$	$Avgwtng_{pre}$		$Avghdng$		$Avgwtng_{post}$	
			value	$Incrs_{td}$	value	$Incrs_{td}$	value	$Incrs_{td}$
0.1:0.3:0.6	0	30	4.87	—	10.40	—	0	—
	6	30	5.53	12%	10.47	1%	0.20	100%
	12	30	9.67	50%	11.10	6%	2.00	100%
0.3:0.3:0.4	0	90	6.71	—	10.36	—	0	—
	6	90	8.87	24%	10.66	3%	0.37	100%
	12	90	10.70	37%	11.11	7%	2.31	100%
0.5:0.2:0.3	0	150	12.05	—	10.90	—	0	—
	6	150	14.58	17%	11.09	2%	0.23	100%
	12	150	18.14	34%	11.62	6%	1.93	100%
0.7:0.2:0.1	0	210	13.22	—	10.58	—	0	—
	6	210	15.28	13%	10.92	3%	0.35	100%
	12	210	19.90	34%	11.27	6%	1.77	100%



TABLE 8: Tidal impact on berth utilization ratio.

Instance	P1			P2			P3			P4		
	$BU_{V_0}$	$BU_{V_6}$	$BU_{V_{12}}$									
1	51.28%	51.16%	50.99%	58.41%	56.75%	53.72%	71.12%	71.20%	63.16%	67.43%	64.95%	60.17%
2	46.42%	44.94%	42.11%	66.48%	67.15%	61.95%	71.12%	72.46%	67.26%	71.53%	65.83%	69.43%
3	61.51%	63.98%	54.65%	59.65%	54.18%	58.64%	65.85%	61.91%	65.30%	67.22%	61.34%	65.43%
4	50.09%	50.50%	47.54%	59.28%	56.36%	50.04%	64.58%	58.54%	54.01%	72.81%	71.72%	65.71%
5	65.63%	60.60%	64.44%	69.77%	71.93%	65.24%	67.04%	63.51%	67.47%	69.42%	64.49%	60.31%
6	51.22%	52.34%	52.84%	54.64%	50.95%	54.53%	66.30%	62.81%	63.75%	69.14%	77.15%	69.37%
7	42.08%	40.39%	40.00%	66.46%	61.85%	62.36%	65.82%	61.72%	58.21%	63.27%	60.28%	56.86%
8	37.96%	38.24%	38.21%	71.51%	62.85%	64.77%	69.11%	66.07%	56.19%	67.86%	67.07%	63.56%
9	62.79%	62.62%	55.12%	66.20%	64.30%	58.98%	72.76%	64.70%	69.09%	73.57%	68.70%	62.44%
10	50.72%	52.21%	47.49%	67.81%	62.21%	55.72%	68.24%	64.98%	66.35%	65.73%	70.00%	65.88%
<i>Average</i>	51.97%	51.70%	49.34%	64.02%	60.85%	58.60%	68.19%	64.79%	63.08%	68.80%	67.15%	63.92%

TABLE 9: Tidal impact on QC utilization ratio.

Instance	P1			P2			P3			P4		
	$QCU_{V_0}$	$QCU_{V_6}$	$QCU_{V_{12}}$									
1	66.12%	64.62%	65.14%	77.56%	76.63%	67.37%	93.82%	94.62%	82.53%	88.78%	84.36%	75.91%
2	63.33%	61.12%	53.40%	87.03%	84.39%	77.64%	91.55%	94.29%	83.79%	92.59%	86.73%	86.79%
3	78.82%	78.17%	68.26%	77.66%	72.61%	74.40%	86.89%	81.88%	83.58%	89.01%	81.90%	80.95%
4	62.52%	63.65%	58.17%	75.26%	73.10%	65.47%	82.15%	76.95%	68.80%	92.84%	91.96%	85.32%
5	80.23%	77.05%	77.89%	90.00%	89.39%	82.03%	86.52%	81.96%	83.23%	92.99%	84.61%	76.53%
6	65.23%	65.68%	66.22%	72.34%	65.89%	66.53%	87.04%	83.30%	82.29%	91.02%	94.98%	87.42%
7	50.54%	50.47%	50.25%	84.30%	79.92%	82.20%	85.96%	82.29%	77.67%	86.08%	83.61%	74.85%
8	50.75%	50.82%	51.68%	91.05%	81.45%	82.30%	88.51%	87.65%	70.36%	89.84%	87.17%	79.22%
9	80.87%	80.44%	69.39%	87.51%	83.05%	73.05%	89.27%	82.18%	82.39%	94.81%	87.41%	79.72%
10	67.30%	67.39%	61.81%	87.46%	78.96%	69.05%	89.49%	80.28%	80.35%	86.84%	88.47%	82.75%
<i>Average</i>	66.57%	65.94%	62.22%	83.02%	78.54%	74.00%	88.12%	84.54%	79.50%	90.48%	87.12%	80.95%

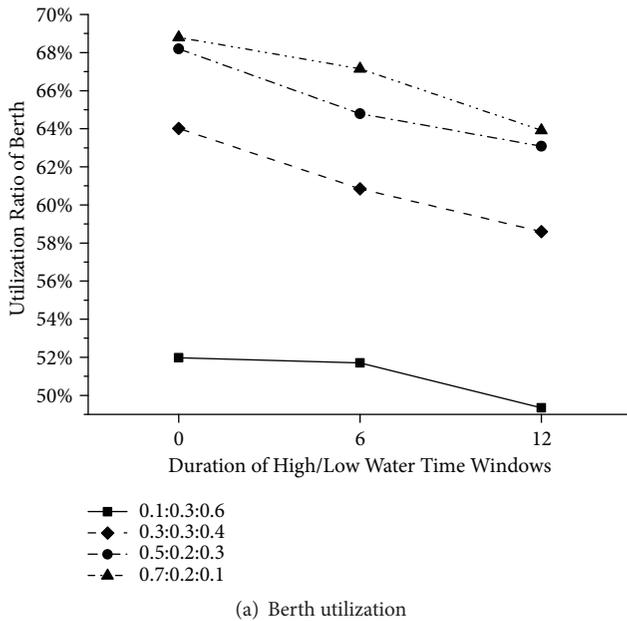
represent the utilization ratio of berth and QCs, respectively. For intuitive understanding, the average utilization ratios of berth and QCs are plotted in Figures 9(a) and 9(b), respectively. It is evident that the berth utilization ratio and QC utilization ratio in any scenario become worse as the duration  $V$  of high/low water time window increases. The reason behind this is that the increasing waiting times of jumbos resulted from the increasing duration of high/low water time window lead to more idleness of berth and QCs. Furthermore, it is observed that both utilization ratios of berth and QCs are enhanced with the increasing portion of the tide-dependent vessels for any given  $V$ , and the increasing extent reduces gradually. This is because the average processing time of jumbos is longer than mediums or feeders. So the utilization ratios of berth and QCs increase in the proportion of jumbos for any given duration of high/low water time window.

From the above experimental results, we may conclude that tides have a nonnegligible impact on the vessel turnaround times and the utilization of critical resources in container terminals. Therefore, the integrated scheduling of berth and QCs considering tidal impacts seems important and meaningful.

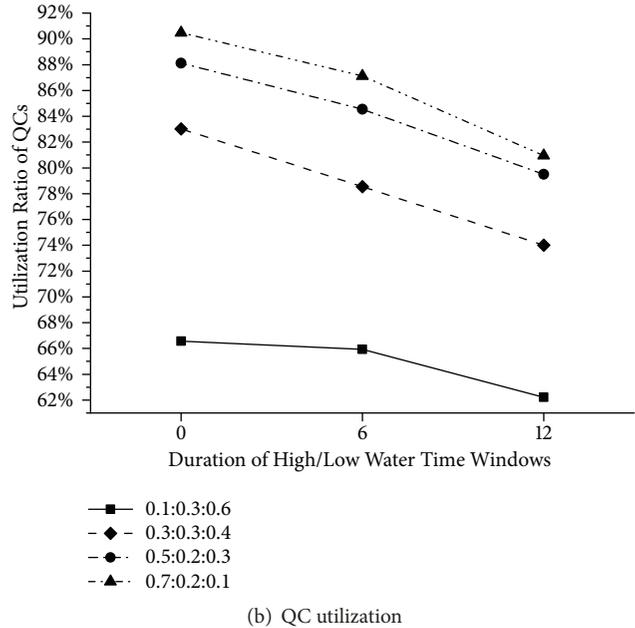
## 7. Conclusion

This work introduces tidal impacts into the integrated berth and QC scheduling problem. We focus on the time-variant version of QC scheduling such that the set of QCs assigned to the same vessel can be different in two consecutive time steps. An integer linear programming model for the considered problem is proposed. Three heuristics, GA, HPSO, and HSA, based on two greedy algorithms are proposed to solve the problem. Computational experiments are conducted to demonstrate the good performances of the proposed algorithms, which produce optimal solutions for small size instances and generate high quality solutions for large size instances. Numerical results also reveal that tides have a significant impact on the total turnaround time of vessels as well as the resource utilization.

The following two extensions may be of interest for the studied problem in this work. The first one is to investigate multiobjective models which minimize simultaneously more than one objective function such as the total turnaround time, the departure delay cost of vessels, and the penalties of non-desired berthing location assignments. Another extension is to study the scenario with uncertain arrival times or processing



(a) Berth utilization



(b) QC utilization

FIGURE 9: Tidal impact on berth and QC utilization.

times of vessels due to bad weather and traffic jam in the port. Robust schedules are usually the target for the model with uncertainty, and it may be an interesting challenge.

## Data Availability

Our computational experiments in our manuscript are based on simulation and all experiments parameters are listed in the manuscript. The data used to support the findings of this study are available from the corresponding author upon request.

## Conflicts of Interest

The authors declare that they have no conflicts of interest.

## Acknowledgments

This work was partially supported by the National Science Foundation of China (71832001, 71771048, 71531011, 71571061, 71571134, and 71428002). The work was also supported by the Fundamental Research Funds for the Central Universities, Shanghai Pujiang Program and Scientific Research Innovation Project of First Class University in Western China (ZKZD2017005).

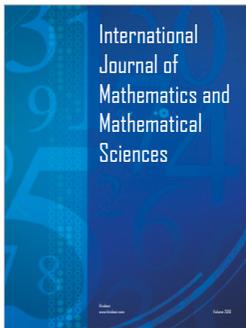
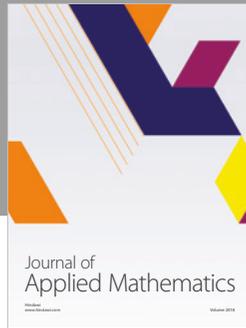
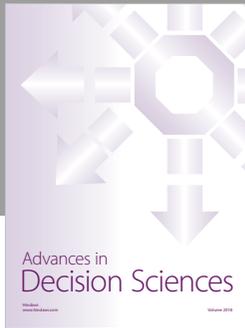
## References

- [1] A. Imai, Y. Yamakawa, and K. Huang, "The strategic berth template problem," *Transportation Research Part E: Logistics and Transportation Review*, vol. 72, pp. 77–100, 2014.
- [2] Y. Du, Q. Chen, J. S. L. Lam, Y. Xu, and J. X. Cao, "Modeling the impacts of tides and the virtual arrival policy in berth

allocation," *Transportation Science*, vol. 49, no. 4, pp. 939–956, 2015.

- [3] V. H. Barros, T. S. Costa, A. C. M. Oliveira, and L. A. N. Lorena, "Model and heuristic for berth allocation in tidal bulk ports with stock level constraints," *Computers & Industrial Engineering*, vol. 60, no. 4, pp. 606–613, 2011.
- [4] D. Xu, C.-L. Li, and J. Y. Leung, "Berth allocation with time-dependent physical limitations on vessels," *European Journal of Operational Research*, vol. 216, no. 1, pp. 47–56, 2012.
- [5] A. Sheikholeslami, G. Ilati, and M. Kobari, "The continuous dynamic berth allocation problem at a marine container terminal with tidal constraints in the access channel," *International Journal of Civil Engineering*, vol. 12, pp. 344–353, 2014.
- [6] E. Lalla-Ruiz, C. Expósito-Izquierdo, B. Melián-Batista, and J. M. Moreno-Vega, "A set-partitioning-based model for the berth allocation problem under time-dependent limitations," *European Journal of Operational Research*, vol. 250, no. 3, pp. 1001–1012, 2016.
- [7] T. Qin, Y. Du, and M. Sha, "Evaluating the solution performance of IP and CP for berth allocation with time-varying water depth," *Transportation Research Part E: Logistics and Transportation Review*, vol. 87, pp. 167–185, 2016.
- [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] D. Steenken, S. Voß, and R. Stahlbock, "Container terminal operation and operations research - a classification and literature review," *OR Spectrum*, vol. 26, no. 1, pp. 3–49, 2004.
- [10] 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.
- [11] C. Bierwirth and F. Meisel, "A follow-up survey of berth allocation and quay crane scheduling problems in container terminals," *European Journal of Operational Research*, vol. 244, no. 3, pp. 675–689, 2015.

- [12] H. J. Carlo, I. F. A. Vis, and K. J. Roodbergen, "Seaside operations in container terminals: literature overview, trends, and research directions," *Flexible Services and Manufacturing Journal*, vol. 27, no. 2-3, pp. 224–262, 2015.
- [13] 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.
- [14] C. Liang, J. Guo, and Y. Yang, "Multi-objective hybrid genetic algorithm for quay crane dynamic assignment in berth allocation planning," *Journal of Intelligent Manufacturing*, vol. 22, no. 3, pp. 471–479, 2011.
- [15] 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.
- [16] G. Giallombardo, L. Moccia, M. Salani, and I. Vacca, "Modeling and solving the tactical berth allocation problem," *Transportation Research Part B: Methodological*, vol. 44, no. 2, pp. 232–245, 2010.
- [17] Q. M. Hu, Z. H. Hu, and Y. Q. Du, "Berth and quay-crane allocation problem considering fuel consumption and emissions from vessels," *Computers & Industrial Engineering*, vol. 70, pp. 1–10, 2014.
- [18] F. Meisel and C. Bierwirth, "A framework for integrated berth allocation and crane operations planning in seaport container terminals," *Transportation Science*, vol. 47, no. 2, pp. 131–147, 2013.
- [19] Y. B. Turkogullari, Z. C. Taskin, N. Aras, and Altinel. I. K., "Optimal berth allocation and time-invariant quay crane assignment in container terminals," *European Journal of Operational Research*, vol. 235, no. 1, pp. 88–101, 2014.
- [20] Y. B. Turkogullari, Z. C. Taşkın, N. Aras, and Altinel. K., "Optimal berth allocation, time-variant quay crane assignment and scheduling with crane setups in container terminals," *European Journal of Operational Research*, vol. 254, no. 3, pp. 985–1001, 2016.
- [21] L. Zhen, E. P. Chew, and L. H. Lee, "An integrated model for berth template and yard template planning in transshipment hubs," *Transportation Science*, vol. 45, no. 4, pp. 483–504, 2011.
- [22] I. Vacca, M. Salani, and M. Bierlaire, "An exact algorithm for the integrated planning of berth allocation and quay crane assignment," *Transportation Science*, vol. 47, no. 2, pp. 148–161, 2013.
- [23] M. Liu, C.-Y. Lee, Z. Zhang, and C. Chu, "Bi-objective optimization for the container terminal integrated planning," *Transportation Research Part B: Methodological*, vol. 93, pp. 720–749, 2016.
- [24] A. Lim, "The berth planning problem," *Operations Research Letters*, vol. 22, no. 2-3, pp. 105–110, 1998.
- [25] B. Chazelle, "bottom-left bin packing heuristic: an efficient implementation," *IEEE Transactions on Computers*, vol. 32, no. 8, pp. 697–707, 1983.
- [26] K. F. Man, K. S. Tang, and S. Kwong, "Genetic algorithms: concepts and applications," *IEEE Transactions on Industrial Electronics*, vol. 43, no. 5, pp. 519–534, 1996.
- [27] Y. Zhang, S. Wang, and G. Ji, "A Comprehensive Survey on Particle Swarm Optimization Algorithm and Its Applications," *Mathematical Problems in Engineering*, vol. 2015, Article ID 931256, 38 pages, 2015.
- [28] B. Suman and P. Kumar, "A survey of simulated annealing as a tool for single and multiobjective optimization," *Journal of the Operational Research Society*, vol. 57, no. 10, pp. 1143–1160, 2006.
- [29] J. H. Holland, *Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence*, University of Michigan Press, Oxford, UK, 1975.
- [30] D.-H. Lee and H. Q. Wang, "Integrated discrete berth allocation and quay crane scheduling in port container terminals," *Engineering Optimization*, vol. 42, no. 8, pp. 747–761, 2010.
- [31] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi, "Optimization by simulated annealing," *Science*, vol. 220, no. 4598, pp. 671–680, 1983.
- [32] S.-W. Lin and C.-J. Ting, "Solving the dynamic berth allocation problem by simulated annealing," *Engineering Optimization*, vol. 46, no. 3, pp. 308–327, 2014.
- [33] M. Rodriguez-Molins, M. A. Salido, and F. Barber, "A GRASP-based metaheuristic for the berth allocation problem and the quay crane assignment problem by managing vessel cargo holds," *Applied Intelligence*, vol. 40, no. 2, pp. 273–290, 2014.



**Hindawi**

Submit your manuscripts at  
[www.hindawi.com](http://www.hindawi.com)

