

Article

A Proactive-Reactive-Based Approach for Continuous Berth Allocation and Quay Crane Assignment Problems with Hybrid Uncertainty

Zhu Wang ¹ , Junfeng Cheng ¹ and Hongtao Hu ^{1,2,*}

¹ Logistics Engineering College, Shanghai Maritime University, Shanghai 201306, China; wangzhu_anita@163.com (Z.W.); 202230210128@stu.shmtu.edu.cn (J.C.)

² The Container Supply Chain Technology Engineering Research Center of the Ministry of Education, Shanghai Maritime University, Shanghai 201306, China

* Correspondence: hu.hongtao@foxmail.com

Abstract: Port operations have been suffering from hybrid uncertainty, leading to various disruptions in efficiency and tenacity. However, these essential uncertain factors are often considered separately in literature during berth and quay crane assignments, leading to defective, even infeasible schedules. This paper addressed the integrated berth allocation and quay crane assignment problem (BACAP) with stochastic vessel delays under different conditions. A novel approach that combines both proactive and reactive strategies is proposed. First, a mixed-integer programming model is formulated for BACAP with quay crane maintenance activities under the ideal state of no delay. Then, for minor delays, buffer time is added to absorb the uncertainty of the arrival time of vessels. Thus, a robust optimization model for minimizing the total service time of vessels and maximizing the buffer time is developed. Considering that the schedule is infeasible when a vessel is seriously delayed, a reactive model is built to minimize adjustment costs. According to the characteristics of the problem, this article combined local search with the genetic algorithm and proposed an improved genetic algorithm (IGA). Numerical experiments validate the efficiency of the proposed algorithm with CPLEX and Squeaky Wheel Optimization (SWO) in different delay conditions and problem scales. An in-depth analysis presents some management insights on the coefficient setting, uncertainty, and buffer time.



Citation: Wang, Z.; Cheng, J.; Hu, H. A Proactive-Reactive-Based Approach for Continuous Berth Allocation and Quay Crane Assignment Problems with Hybrid Uncertainty. *J. Mar. Sci. Eng.* **2024**, *12*, 182. <https://doi.org/10.3390/jmse12010182>

Academic Editor: Mihalis Golias

Received: 18 December 2023

Revised: 14 January 2024

Accepted: 16 January 2024

Published: 18 January 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

Keywords: berth allocation and quay crane assignment; vessel delay; proactive-reactive; local search; genetic algorithm

1. Introduction

With the development of global trade, container shipment has been rising continuously. According to statistics by the Chinese Ministry of Transport, container throughput has increased 180% in the past decade. With the continuous increment in container port workload, the industry is increasingly concerned about improving port operation efficiency and reliability. Port is an open system, and its operations are disrupted by various uncertain factors at different stages, including natural, human, and internal and external factors. Reducing losses caused by uncertain events has always been one of the critical issues in port operations [1].

Berths and quay cranes connect vessels to terminals for cargo transit, making their operations vulnerable to comprehensive uncertain events, such as vessel delay (external and natural), and quay crane maintenance (internal and human). Under the combined effect of uncertain factors, how to improve berth and quay crane allocation in terms of robustness and agility has always been widely a concern by industry and academia.

The berth and quay crane allocation problem refers to allocating berthing position and time for every vessel within a given planning horizon and assigning reasonable numbers of quay cranes (QCs) to vessels during the berthing period [2]. Numerous studies have

been conducted on the deterministic BACAP. These studies can be found in Bierwirth and Meisel [3], Carlo et al. [4], and Iris et al. [5]. In particular, Iris et al. [5] proposed set partitioning formulations and column reduction techniques for BACAP, including time-variant and time-invariant QC assignments. The study conducted by Salhi et al. [6] presents an integrated optimization model for BACAP and quay crane scheduling with the utilization of a genetic algorithm for large-scale instances. It is worth noting that a considerable number of papers have been published on the subject of quayside problems with a focus on certainty.

However, frequent mechanical operations render QC components prone to fatigue and even breakdowns [7]. One of the most effective ways of preventing accidents is to carry out preventive maintenance on QCs. During the maintenance period, QC cannot handle any vessel, and the berths segment where the QC is occupied cannot be assigned to any vessel. As such, the impact of maintenance activities on BACAP should not be ignored [8]. To be noted, both berth and quay crane allocation decisions are made before the vessel arrives. If the vessel arrives earlier than expected, it will dock at the anchorage and wait for berthing. But if the vessel is delayed, initial plans may become infeasible, disrupting the following two operational stages.

Previously, plenty of research was devoted to improving the robustness of the berthing plan, and the challenge was to control conservativeness. Zhen [9] compared stochastic and robust optimization formulation of tactical-level berth allocation scheduling problems. He found that the robust method can derive an approximate optimal solution to the stochastic model in a fast way. Shang et al. [10] established a robust optimization model with price constraints. Liu et al. [11] developed a two-stage robust optimization approach consisting of a baseline schedule and recovery operation. Xiang and Liu [12] introduced a controlled violation of constraints and proposed an almost robust optimization approach. Some papers formulated a two-stage stochastic programming model based on a set of scenarios, and they developed many heuristic algorithms to improve operational efficiency [13,14]. Zhen et al. [15] proposed a solution using the column generation technique. Park et al. [16] developed an algorithm for time buffer insertion, which accommodates the adaptive search procedure for the time buffer into the particle swarm optimization algorithm. Agra and Rodrigues [17] formulated a distributionally robust two-stage model for BAP under uncertain handling time, and an exact decomposition algorithm was developed to solve the problem. Chargui et al. [18] established a robust model to handle the uncertainty of renewable energy availability. The above literature adopts a proactive strategy to absorb the interference of uncertain factors by improving the robustness to obtain the system's stability.

On the contrary, some research was intended to enhance system agility. They used reactive strategies to respond to a disturbance in real time [19,20]. Xiang et al. [21] and Lv et al. [22] addressed the berth allocation and quay crane assignment under the uncertainty of equipment breakdowns and vessel arrival time to minimize deviations from the original solution. Since real-time reactive inevitably increases the difficulty of management, some scholars integrated proactive and reactive strategies, hoping to reduce the efficiency loss caused by excessive robustness and the difficulty of real-time disruption recovery management [23]. Iris and Lam [24] addressed the uncertainty of vessel arrivals and fluctuations in the container handling rates of quay cranes by introducing recovery plans for each scenario, along with an adaptive large neighborhood-based heuristic framework based on the study by Iris et al. [25]. Tan and He [26] formulated the BACAP under the uncertain vessels' arrival times as a stochastic programming model and proposed a two-stage meta-heuristic framework based on GA for solving this problem. Dai et al. [27] constructed a bi-layer model for BAP to reduce the impact of uncertain arrival time. They used adaptive search and heuristic adjustment strategies to improve the cross-entropy algorithm. The QC maintenance is not within the scope of the research mentioned above.

Li et al. [28] established a stochastic integer linear programming model for BACAP with uncertain quay crane maintenance activities. Li et al. [29] studied a bi-objective

optimization model of BACAP with preventive QC maintenance activities. They proposed an ε -constraint-based two-phase iterative heuristic design. To the best of our knowledge, research on BACAP with maintenance activities in uncertain environments is still in its early stages.

Under the combined effect of the above-mentioned multiple uncertain factors, intending to promote the reliability of container terminals, this article proposed an integrated proactive and reactive framework. The main contributions are as follows:

- (1) Novel proactive-reactive-based BACAP mathematical models are established. Different indicators are adopted for various uncertain factors to balance robustness and agility.
- (2) An improved genetic algorithm (IGA) is proposed, which can find a solution that is 22.86% superior to that of squawky wheel optimization within 225 s.
- (3) A sensitivity analysis is conducted, which implies that port operators should define the coefficient and buffer time according to the uncertainty of vessel arrival time; the more extensive the buffer time, the less significant the effect of risk resistance capability.

The remainder of this paper is organized as follows. Section 2 describes the problem thoroughly, including the maintenance activities of QCs and vessel delay. In Section 3, the deterministic, baseline planning, and reactive models are subsequently established. An improved GA is proposed in Section 4. Computational experiments are reported and analyzed in Section 5. Finally, Section 6 concludes the study.

2. Problem Description

This study focuses on a container terminal with a continuous coastline. The berth is divided into segments of 10 m, and the planning horizon is discretized into one h. All berths and QCs are available at the beginning of the planning horizon. The quantity of QCs assigned to a vessel is time-variant, and the specific QC allocated to the vessel can vary with time. Thus, the quay crane assignment includes allocating the quantity and the specific quay cranes.

2.1. Maintenance and Virtual Vessels

For reliability, preventive maintenance (PM) of quay cranes is performed within a specific period. However, the starting time of PM is a decision that needs to be made in combination with the berthing plan. Vessels are not allowed to enter the berthing area where the quay crane under maintenance is located. Virtual vessels are introduced to accurately describe the relationship between PM and the allocation of berths and QCs. For instance, when QC k is under maintenance, it is assumed that a virtual vessel (Vessel i') is served by QC k . The length of the virtual vessel is consistent with the width of the work coverage of QC k , and the berthing period of Vessel i' (denoted by σ_k) is the same as the actual PM period of QC k . Thus, the PM decision-making problem of QCs is integrated into the BACAP. Furthermore, the berth allocation with discontinuous shorelines can be transformed into a continuous berth allocation problem.

2.2. Uncertainty and Buffer Time

This paper focuses on the vessels' arrival time at the terminal exceeding the expected time due to uncertain factors. To increase the robustness of the berthing and QC allocation plan, the buffer time is added to the berthing time. As shown in Figure 1, a_i is the expected arrival time of a vessel i , e_i is the end of handling time of vessel i . The buffer time of vessel i and j are denoted by θ_i and θ_j , respectively. Thus, the original plan can achieve the desired result when the delay is within this buffer time. It is worth noting that the greater the buffer time, the greater the ability of the corresponding allocation plan to absorb the uncertainty, but the lower the service efficiency of the terminal. Therefore, weighing the service efficiency and robustness index and setting an appropriate buffer time is necessary.

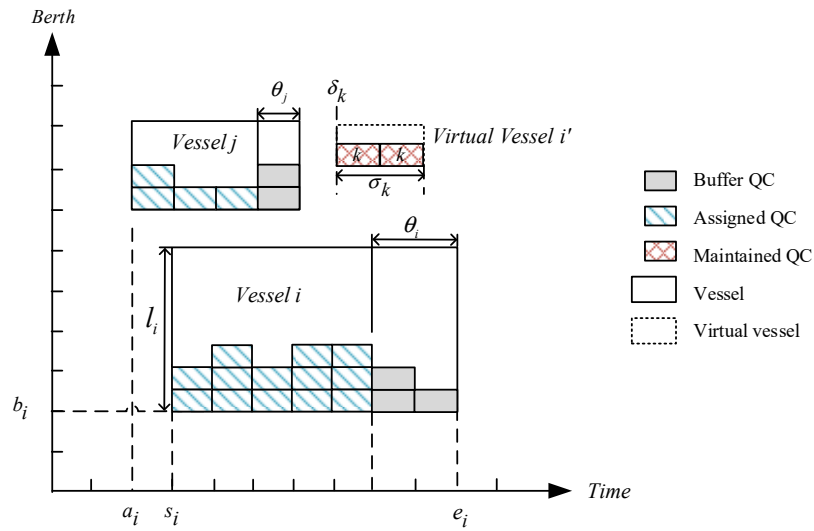


Figure 1. An illustration of QC maintenance, virtual vessels, and buffer time.

3. Mathematical Formulation

Three mathematical models—deterministic, baseline planning, and reactive models—are constructed to obtain a robust plan and effectively assess the impact of uncertainties on berth and QC allocation and the PM plan. The relationship among the models is shown in Figure 2.

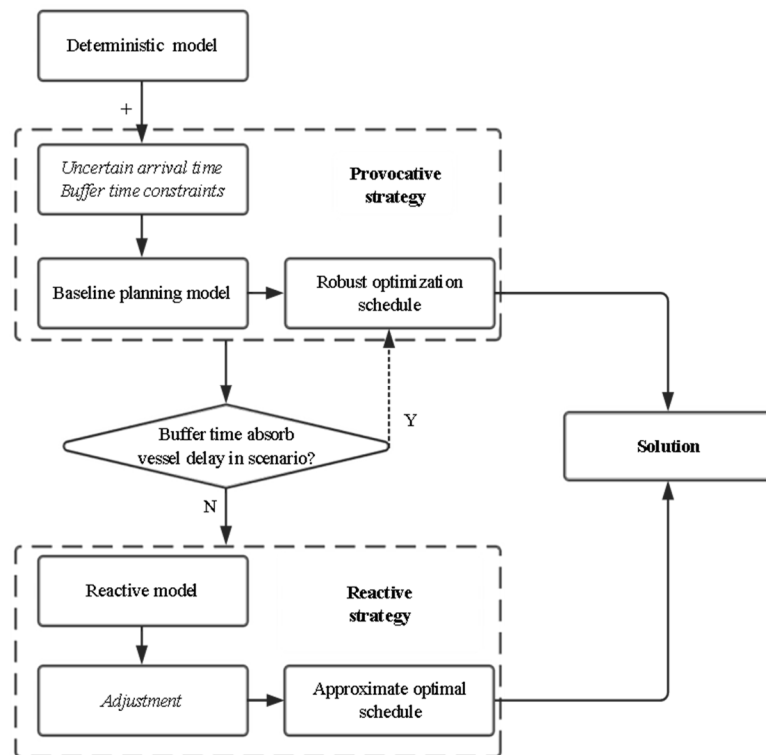


Figure 2. The relationship among the three models.

Figure 3a shows a baseline planning model schedule in a time/berth diagram. Each vessel is represented as a black rectangle with thick lines for length. Small squares indicate QCs inside each vessel rectangle, with diagonal line-filled blue squares being assigned QCs; the number inside the square is the specific number of QCs. Dark grey squares indicate the buffer time. The virtual vessel is represented as a dotted black rectangle. The netted red squares are maintained QCs. The schedule for the reactive model is depicted in

Figure 3b using a time/berth diagram. In contrast to Figure 3a, the berth with buffer time is illustrated by a red rectangle with thick lines. In this diagram, the delay of vessels 1 and 5 is noticeable; however, the buffer time absorbs the uncertain vessel delay, ensuring that the following schedule is not interrupted. Conversely, vessel 3 is significantly delayed and must wait until the berth and quay crane become available.

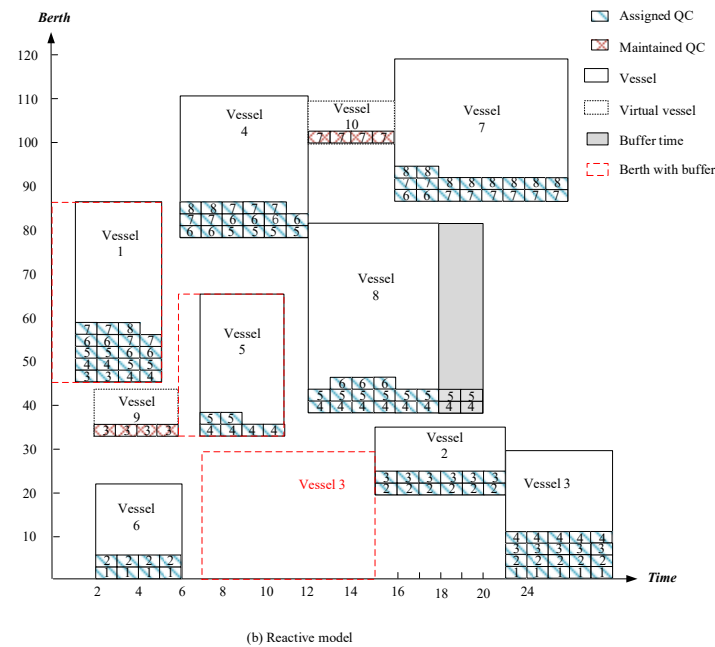
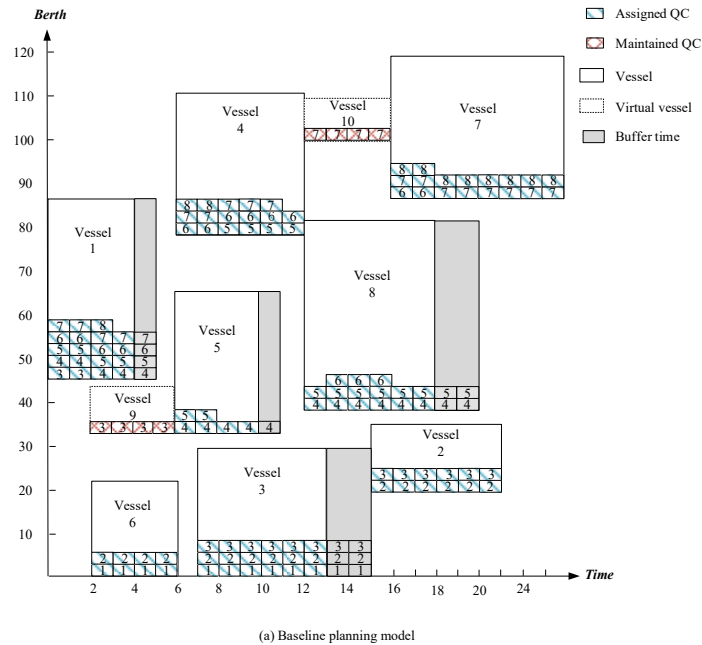


Figure 3. Schedules obtained by baseline line model and reactive model.

3.1. Assumptions

For clarity, the mathematical models are formulated based on the following assumptions.

- (a) All vessels met the physical or technical requirements of the berth, such as water depth, vessel draft, etc.
- (b) The QC assignment was time-invariant, and the number of QCs serving each vessel had upper and lower limits.

- (c) Each vessel had a desired berthing position.
- (d) No difference was present in the productivity between QCs.
- (e) QC assignment is time-invariant, but specific QC reallocation is allowed.
- (f) The PM period is predetermined, and the required PM time is known too.

3.2. Model Formulation

3.2.1. Deterministic Model

Objective:

$$\min f_1 = \sum_{i \in I} [(e_i - a_i) + \beta_i \cdot \Delta b_i] \tag{1}$$

s.t.

$$b_i + l_i \leq b_j + L(1 - x_{ij}), \forall i, j \in I, i \neq j \tag{2}$$

$$e_i \leq s_j + H(1 - y_{ij}), \forall i, j \in I, i \neq j \tag{3}$$

$$x_{ij} + x_{ji} + y_{ij} + y_{ji} \geq 1, \forall i, j \in I, i \neq j \tag{4}$$

$$b_i + l_i \leq L, \forall i \in I \tag{5}$$

$$\Delta b_i \geq b_i - b_i^0, \forall i \in I_1 \tag{6}$$

$$\Delta b_i \geq b_i^0 - b_i, \forall i \in I_1 \tag{7}$$

$$a_i \leq s_i, \forall i \in I \tag{8}$$

$$s_i \leq tw_{it} + H(1 - w_{it}), \forall i \in I, \forall t \in T \tag{9}$$

$$(t + 1)w_{it} \leq e_j, \forall i, j \in I, \forall t \in T \tag{10}$$

$$\sum_{t \in T} w_{it} = e_i - s_i, \forall i \in I \tag{11}$$

$$\sum_{i \in I} p_{itk} \leq 1, \forall t \in T, \forall k \in K \tag{12}$$

$$w_{it} = \sum_{q \in K} \theta_{itq}, \forall i \in I, \forall t \in T \tag{13}$$

$$\sum_{q \in K} q \cdot \theta_{itq} = \sum_{k \in K} p_{itk}, \forall i \in I, \forall t \in T \tag{14}$$

$$q_i^{min} - M \cdot (1 - w_{it}) \leq \sum_{k \in K} p_{itk} \leq q_i^{max} + M \cdot (1 - w_{it}), \forall i \in I, \forall t \in T \tag{15}$$

$$\sum_{t \in T} \sum_{k \in K} p_{itk} \geq (1 + \Delta b_i \cdot \alpha) F_i, \forall i \in I \tag{16}$$

$$\sum_{i \in I} \sum_{k \in K} p_{itk} \leq N, \forall t \in T \tag{17}$$

$$p_{it(k-1)} - p_{itk} + p_{itg} \leq 1, \forall i \in I, \forall t \in T, \forall k \in [2, N - 1], g \in [k + 1, N] \tag{18}$$

$$k + 1 \leq g + N(3 - p_{itk} - p_{jtg} - x_{ij}), \forall i, j \in I, i \neq j, t \in T, k, g \in K, k \neq g \tag{19}$$

$$l_i = d_k, \forall (i, k) \in Z, \forall i \in I_2, \forall k \in K_1 \tag{20}$$

$$F_i = \sigma_k, \forall (i, k) \in Z, \forall i \in I_2, \forall k \in K_1 \tag{21}$$

$$a_i = \delta_k^{min}, \forall (i, k) \in Z, \forall i \in I_2, \forall k \in K_1 \tag{22}$$

$$s_i = \delta_k, \forall (i, k) \in Z, \forall i \in I_2, \forall k \in K_1 \tag{23}$$

$$s_i \leq \delta_k^{max}, \forall (i, k) \in Z, \forall i \in I_2, \forall k \in K_1 \tag{24}$$

$$p_{itk} \leq \beta_{ik}, \forall i \in I_2, \forall k \in K_1, \forall t \in T \tag{25}$$

$$(k - 1)d_k \leq b_i \leq L - (N - k + 1)d_k, \forall i \in I_2, \forall k \in K_1 \tag{26}$$

$$s_i, e_i, b_i, \delta_k, \Delta b_i \geq 0, \forall i \in I, \forall k \in K \tag{27}$$

$$x_{ij}, y_{ij}, p_{itk}, w_{it}, u_i \in 0, 1, \forall t \in T, \forall k, q \in K, \forall i, j \in I, i \neq j \tag{28}$$

Equation (1) represents the objective function, which is to minimize the total turnaround time of vessels in the port and additional time due to the deviation from the desired berth. Constraints (2)–(4) ensure no overlapping of the handling of vessels in the space–time diagram. Constraint (5) provides that the berthing position of each vessel does not exceed the total length of the quay. Constraints (6) and (7) determine the deviations from the desired berthing position. Constraint (8) ensures that the berthing time of each vessel is not shorter than its arrival time. Constraints (9)–(11) define the starting and ending times for serving vessels without pre-emption. Constraint (12) states that each QC can handle at most one vessel at a time. Constraint (13) ensures a consistent set of the corresponding variables w_{it} and θ_{itq} . Constraint (14) guarantees that θ_{itq} and p_{itk} are consistent. Constraint (15) defines the relationship between p_{itk} and w_{it} . Constraint (16) indicates the number of QCs assigned to the vessel is sufficient considering the deviation of the berthing position from the expected berthing position. Constraint (17) ensures that at most N cranes are utilized in a period. Constraint (18) guarantees the continuity of QCs assigned to a vessel to prevent QCs from being assigned but not undertaking operational tasks. Constraint (19) ensures no crossing between adjacent QCs at any instant. Constraint (20) defines the length of the virtual vessel, which is equal to the work coverage of the QC under maintenance. Constraint (21) indicates that the length of maintenance required for the QC is equal to the amount of operation of the virtual vessel. Constraint (22) suggests that the arrival time of the virtual vessel is the earliest start time of the corresponding QC maintenance activities. Constraint (23) indicates that the actual starting time of the maintenance of QC is the start handling time of the corresponding virtual vessel. Constraint (24) restricts the range of the start handling of the virtual vessel. Constraint (25) ensures that one virtual vessel corresponds to one QC. Constraint (26) implies that the berthing position of the virtual vessel is within the movable range of the maintenance QC. Constraints (27) and (28) define domains for the decision variables.

3.2.2. Baseline Planning Model

In this study, the baseline planning model is conducted by introducing the buffer time. Generally, the longer the buffer time, the better the robustness of the schedule, but the lower the efficiency of service. To balance indicators of service and robustness, the weight coefficients are introduced to balance the weight of each part of the objective function, which can be expressed as Equation (30).

$$f_2 = \max \sum_{i \in I} \theta_i \tag{29}$$

$$\min f = \{\omega f_1, (1 - \omega)(-f_2)\} \tag{30}$$

The constraints that the baseline planning model needs to satisfy are as follows.
s.t. constraints (2)–(27)

$$\theta_i = 0, \forall i \in I_2 \tag{31}$$

$$\sum_{\tau \in T} \sum_{q \in K} \theta_{i\tau q} = 1, \forall i \in I_1 \tag{32}$$

$$\theta_i = \sum_{\tau \in T} \sum_{q \in K} \theta_{i\tau q} \cdot \tau, \forall i \in I_1 \tag{33}$$

$$\sum_{t \in T} \sum_{q \in K} \theta_{itq} \cdot q^\gamma \geq F_i + \sum_{\tau \in T} \sum_{q \in K} \theta_{i\tau q} \cdot \tau \cdot q^\gamma, \forall i \in I \tag{34}$$

$$x_{ij}, y_{ij}, p_{itk}, w_{it}, u_i, \theta_{i\tau q} \in \{0, 1\}, \forall t \in T, k, q \in K, i, j \in I, i \neq j \tag{35}$$

Constraint (31) sets the buffer time of the virtual vessel to zero. Constraints (32) and (33) define the domain of $\theta_{i\tau q}$ and the relationship between θ_i and $\theta_{i\tau q}$, respectively. Constraint (34) guarantees that the total productivity of QCs assigned to a vessel is sufficient. The left side of constraint (34) represents the total productivity of QCs assigned to a vessel. The right side of constraint (34) represents the required QC hours, including the crane capacity demand of vessel i within handling time and buffer time. Constraint (35) defines domains for the decision variables.

3.2.3. Reactive Model

This section explains a reactive model to evaluate the deviation between the new robust schedule and the original deterministic schedule in three dimensions, i.e., the berthing time, berthing position, and number of QCs. The objective function (36) indicates the minimization of the time cost of total turnaround, desired berth deviation cost, and variable costs among the revised berthing time, berthing position, number of QCs, and original plan.

$$\begin{aligned} \text{Minimize } \sum_{i \in I} \{ & c_i^1(e_i - a_i) + c_i^2 \cdot \Delta b_i + c_i^3(s_i^0(\omega) + s_i^-(\omega)) \\ & + c_i^4(b_i^+(\omega) + b_i^-(\omega)) + c_i^5 \cdot (q_i^+(\omega) + q_i^-(\omega)) \} \end{aligned} \quad (36)$$

s.t.

$$b_i = b_i^* + b_i^+(\omega) - b_i^-(\omega), \forall i \in I, i \neq j, \omega \in \Omega \quad (37)$$

$$q_i = q_i^* + q_i^+(\omega) - q_i^-(\omega), \forall i \in I, i \neq j, \omega \in \Omega \quad (38)$$

$$s_i = s_i^* + s_i^+(\omega) - s_i^-(\omega), \forall i \in I, i \neq j, \omega \in \Omega \quad (39)$$

$$s_i^0(\omega) \geq s_i^+(\omega) - \theta_i, \forall i \in I, i \neq j, \omega \in \Omega \quad (40)$$

$$s_i, e_i, b_i, \delta_k, q_i, \Delta b_i, s_i^0(\omega), s_i^+(\omega), s_i^-(\omega), b_i^+(\omega), q_i^+(\omega), q_i^-(\omega) \geq 0, \forall i \in I \quad (41)$$

Constraints (2)–(26) are consistent with deterministic models. Constraints (37)–(39) define the deviation of the solution of the deterministic model and baseline planning model in berthing position, number of QCs, and berthing time, respectively. Constraint (40) indicates the increment of s_i^+ relative to the buffer time θ_i . Constraint (41) defines domains for the decision variables. Thus, a reactive model with objectives function (36), constraints (2)–(26), and (37)–(41) are established.

4. Solution Methods

The BAP has been proven to be an NP-hard problem [2]. With increasing scales of problems, obtaining an optimal solution for BACAP in an acceptable time is difficult. Therefore, an improved genetic algorithm (IGA) is proposed to provide an approximate optimal solution for BACAP in a reasonable time. This paper improved the genetic algorithm by combining local search algorithms for decoding and evaluation. Furthermore, to enhance the quality of the solution, a neighborhood search algorithm based on the swap strategy of SWO is implemented in mutation.

4.1. Chromosome Encoding

According to the characteristics of decision variables, this paper adopts the mixed coding form of an integer and real number. An illustration of the chromosome is shown in Figure 4, the length of the chromosome is $2I_1 + I$.

The number of QCs assigned to the vessel and buffer time are the main decision variables located in the first and second segments of chromosomes, and both use integer coding. Genes $1 \sim I_1$ indicate q_i , which is randomly generated in the range $[q_i^{min}, q_i^{max}]$; genes $I_1 + 1 \sim 2I_1$ represent the buffer time of the actual vessel randomly generated in the range $[0, \theta_0]$. The vessel priority sequence is also encoded but as a real number.

Genes $2I_1 + 1 \sim 2I_1 + I$ reflect the vessel priority sequence, which is generated randomly within $[0, 1]$.

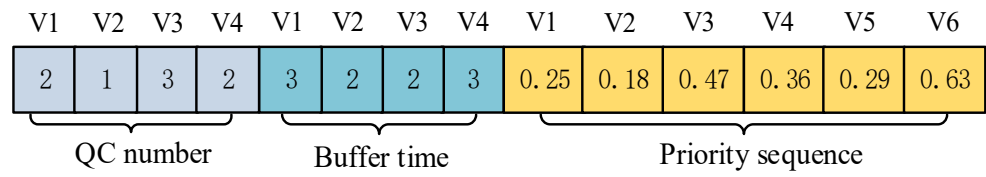


Figure 4. Chromosome description.

4.2. Decoding

Firstly, the vessel priority is decided by the value of genes $2I_1 + 1 \sim 2I_1 + I$ in descending order. For instance, in Figure 4, since $0.63 > 0.47 > 0.36 > 0.29 > 0.25 > 0.18$, the priority sequence of the vessel is 6, 3, 4, 5, 1, and 2. Secondly, to avoid the generation of infeasible chromosome sequences, the vessel’s berthing position and berthing time are obtained by local search, which is based on the established model. The proposed algorithm is called a partition check-based local search algorithm.

4.2.1. QC Assignment and Partition Check

If QC maintenance is not involved, a feasible QC assignment plan needs to meet constraints (17)–(19). In other words, if there is an idle QC and the QCs adjacent to it are working, such a scheme should be avoided. Moreover, the number of QCs assigned to vessels at any time should not exceed the total number of QCs.

When QC maintenance activities are introduced, virtual vessels divide the coastline into independent areas, and an optimal solution meeting the above constraints may become infeasible (as shown in Figure 5). Therefore, after generating the initial solution, this study combines the established model and proposes a partition check process to ensure feasibility.

Step 1: Taking virtual Vessel 1 and virtual Vessel 2 as the nodes, the coastline is divided into three regions.

Step 2. After checking Region 1 from left to right of the coastline, it is considered that the allocation is feasible. Because five QCs are assigned to the actual vessel in Region 1, they are all located on the left side of the first node (i.e., virtual Vessel 1).

Step 3. After checking Region 3 from left to right of the coastline, it is considered that the allocation is feasible. The QC number corresponding to the last node (virtual vessel 2) is 9, and only one QC is assigned to Vessel 4 on the right side of the node, which meets Constraint (17).

Taking Figure 6 as an example, the partition check procedure is as follows:

Step 4. After checking the intermediate Region 2, it was considered that the allocation is feasible. The sum of the QC quantity allocated to this region and the QC number corresponding to the node on the left side of Region 2 does not exceed the QC number corresponding to the node on the right side of that.

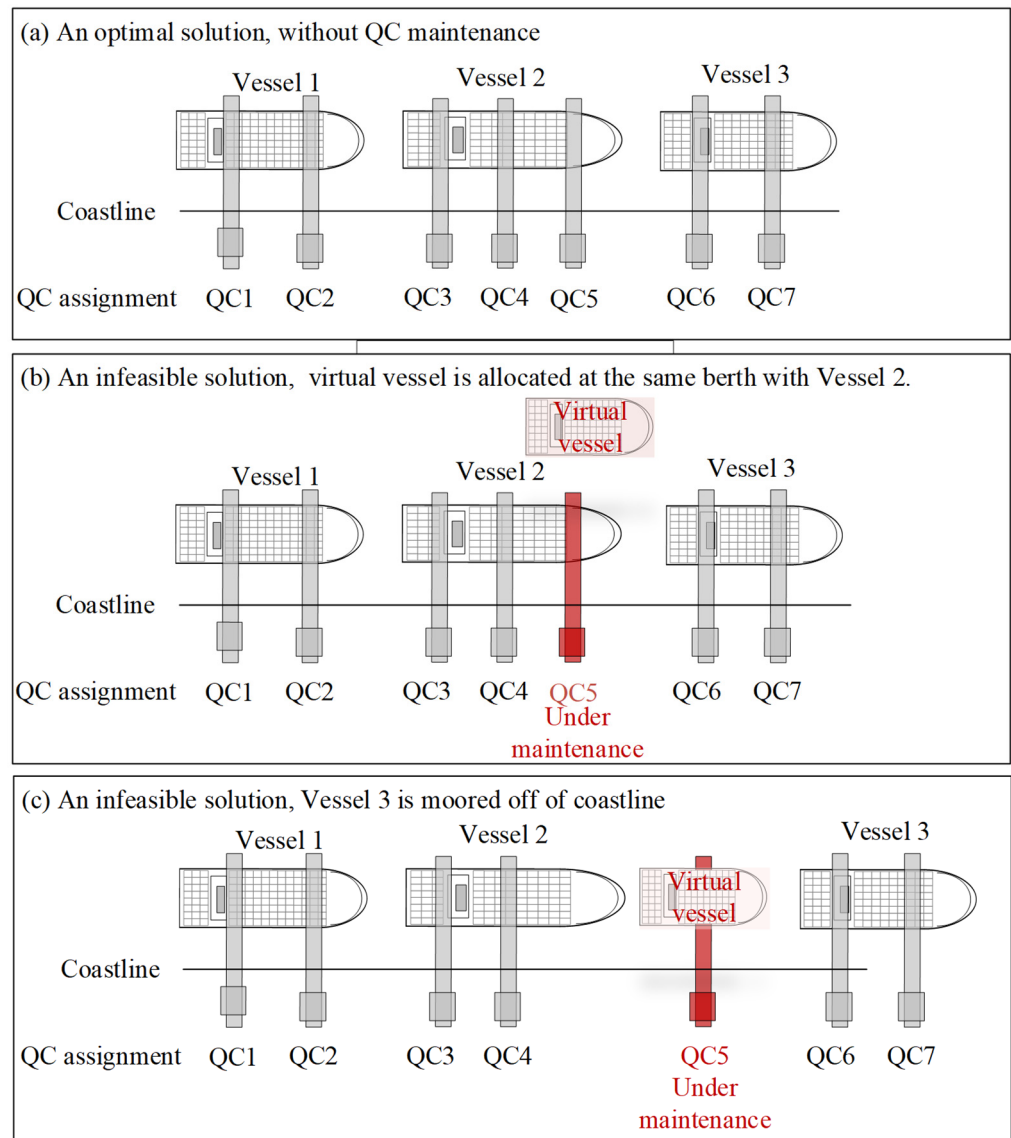


Figure 5. Violation in the presence of a virtual vessel.

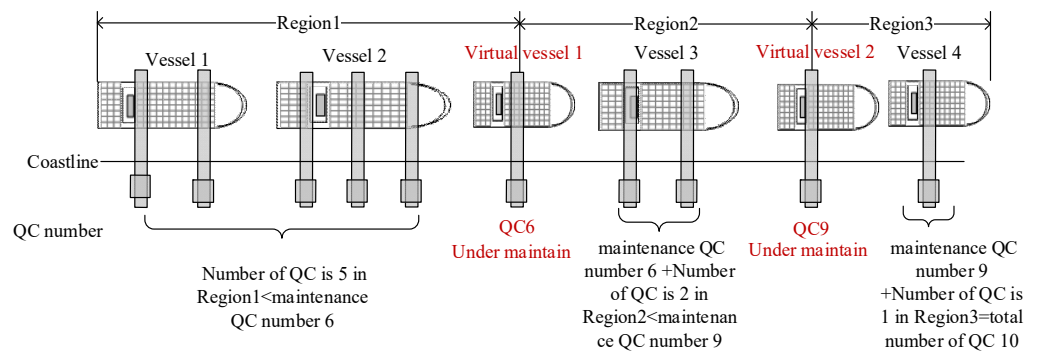


Figure 6. Assignment scheme when virtual vessel exists.

4.2.2. Partition Check-Based Local Search Algorithm

To improve the GA performance, a local search algorithm was designed based on the partition check process. The core idea of the algorithm is to allocate the berthing position, berthing time, and specific QCs of any inserted vessel according to the priority sequence. The flowchart of the proposed algorithm is shown in Figure 7.

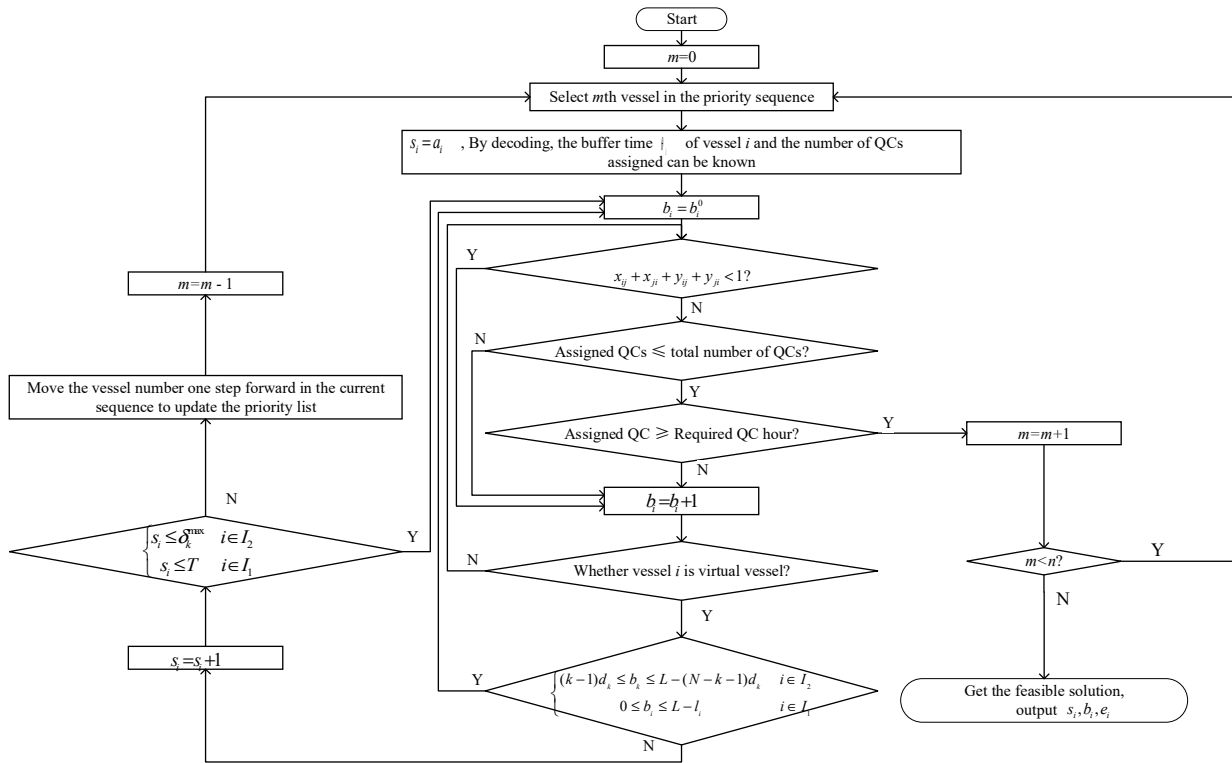


Figure 7. The flow chart of local search.

4.3. Fitness Evaluation

In this paper, the fitness of chromosomes is evaluated by an exponential transformation of the objective function. The calculation method is as follows:

$$Fit(x) = \frac{1}{1 + e^{-\frac{fit(x)}{\eta}}} \tag{42}$$

where η is a problem scale-related parameter and $fit(x)$ is obtained by the objective function of the baseline planning model. The expression of the objective function is

$$fit(x) = \omega \sum_{i \in I} [(e_i - a_i) + \beta_i \cdot \Delta b_i] + (1 - \omega) (-\sum_{i \in I} \theta_i) \tag{43}$$

Thus, all the fitness values obtained using the fitness function are non-negative. Moreover, the higher the fitness value, the better the performance of chromosomes.

4.4. Selection

To retain excellent individuals and avoid significant sampling errors, a truncation selection is used to select chromosome individuals.

4.5. Crossover

The two-point crossover method is employed to operate all individuals in the initial population. Specifically, two crossover points are randomly selected on a chromosome that represents the beginning and end of the swapping segment, and their value ranges are $[1, 2I_1 + I - 1]$ and $[2, 2I_1 + I]$, respectively. After swapping the segments, two new chromosomes are generated.

4.6. Mutation

This study is inspired by the SWO algorithm proposed by Joslin et al. [30]. The Squeaky Wheel Optimization (SWO) algorithm is an important meta-heuristic algorithm

that is often applied to combinatorial optimization problems, especially for those problems that need to be prioritized so that the most urgent problems are solved first. On the basis of Meisel and Bierwirth [31], this study adopts the swap strategy for neighborhood search to improve the convergence speed.

The indicator $shipcost(i)$ is defined to determine the priority sequence of vessels and the objective function value of each vessel is defined as the contribution value of the vessel i . Since vessels with higher priority can more easily obtain sufficient time and space resources, their contribution value is generally lower. In other words, the smaller the contribution value, the smaller the objective, and the better the quality of the solution.

$$shipcost(i) = \omega[(e_i - a_i) + \beta_i \cdot \Delta b_i] - (1 - \omega)\theta_i \tag{44}$$

In this step, the vessel with a higher contribution value is moved forward in the priority list to reduce its contribution value, and the goal of reducing the overall objective value is finally achieved by adjusting the priority sequence list several times. The mutation operator is described as follows (Algorithm 1).

Algorithm 1. Swap strategy-based neighborhood search algorithm.

S: the set of chromosomes needing to mutate;
T: maximum number of iterations for neighborhood search;
J: number of vessels
For $i = 0; i < |S|; i++$
 For $t = 0; t < T; t++$
 decoding $chromosome(i)$;
 obtain the $objectivevalue_j$ and $vesselcost_j$;
 $m = 0$;
 $Best_objectivevalue = objectivevalue_j$;
 $Best_chromosome = chromosome(i)$;
 For $j = 0; j < J; j++$
 If $vessel\ cost\ (j + 1) > vessel\ cost\ (j)$
 obtain the $chromosome'_m$ by swapping the order of two vessels in the priority sequence;
 decoding $chromosome'_m$ and obtaining the $objectivevalue'_m$;
 If $objectivevalue'_m < Best_objectivevalue$
 $Best_objectivevalue = objectivevalue'_m$;
 $Best_chromosome = chromosome'_m$;
 $m = m + 1$;
 End if
 End if
 $chromosome(i) = Best_chromosome$;
 End for
End for

4.7. Restart Mechanism

When the solution does not show any improvement after 50 iterations, the restart mechanism should be started. The next generation inherits the best 15% of the population, 35% of the population regenerates randomly, and 50% of the population mutates with a probability of 0.8, thus forming a new population.

5. Computational Experiments

This section introduces instances provided by Meisel and Bierwirth [31]. The IBM ILOG CPLEX 12.6 and SWO were used as the benchmark to evaluate the performance of the proposed IGA. The mathematical model was solved by Branch and Cut using CPLEX with a computational time limit of 18,000 s. All approaches were implemented by Visual C# using Microsoft Visual Studio 2019 on a PC with Intel Core i7, 3.6 GHz processor, and 8 GB RAM.

5.1. Parameter Settings

Vessel types are divided into Feeder, Medium, and Jumbo. Table 1 shows that 60% of the vessels belong to Feeder, 30% to Medium, and 10% to Jumbo. The planning horizon T is set to 168 h. The length of the quay is 1200 m, discretized into 120 segments. Vessel arrival time follows the distribution $A_i \in [a_i, a_i + 8]$. Maximum buffer time is $\theta_{max} \in \{3, 5, 7\}$ and coefficient $\omega = 0.4$.

Table 1. Parameters for three types of vessels.

Vessel Type	F_i	l_i	q_i^{min}	q_i^{max}	β_i
Feeder	U [10, 20]	U [8, 21]	1	2	0.2
Medium	U [30, 50]	U [21, 30]	2	4	0.2
Jumbo	U [50, 60]	U [30, 40]	4	6	0.2

In addition, to achieve optimal search performance of metaheuristic algorithms, it is crucial to fine-tune specific vital parameters. In IGA, these parameters include maximum iterations (MAXIT), population size (POP), crossover factor (CF), and mutation factor (MF). Their impact on algorithm performance is significant. The ideal parameter combination is determined using Taguchi experiments, where the best individuals serve as the evaluation index. To ensure fairness, both IGA and SWO are run for the same number of iterations on problems of the same size. After conducting experiments, the primary parameter settings for all algorithms are presented in Table 2.

Table 2. Parameters setting of IGA.

Parameter	Problem Scale		
	Small-Scale	Medium-Scale	Large-Scale
MAXIT	200	250	300
POP	150	150	150
CF	0.7	0.7	0.75
MF	0.2	0.25	0.3
η	20	30	35

5.2. Performance Analysis

This section comprehensively compares the IGA, CPLEX, and SWO regarding solution quality and CPU time. The instance is categorized as small, medium, and large, according to the vessel types (I), number of QCs (K), and number of QCs to be maintained (K_1). Each instance is tested ten times, and the average value is recorded. The evaluation criteria are defined as the gap between IGA and benchmark methods. For example, ' $Gap_{IGA} = ((Obj_{IGA} - Obj_{CP}) / Obj_{CP}) \times 100\%$ ' represents the gap between the objective values obtained by the proposed algorithm (Obj_{IGA}) and CPLEX (Obj_{CP}). "CT" represents the CPU time. The computational results of all instances are shown in Tables 3 and 4.

For small-scale instances, the quality of solutions obtained by IGA is better than SWO, and the gap with CPLEX is within 0.6%. In terms of CPU time, SWO and IGA have obvious speed advantages, and CPLEX operation time increases exponentially with the increase in the problem scales. In general, the average CPU time of SWO is less than 0.5 s, and that of IGA is less than 1.3 s, whereas the CPLEX is over 1100 s.

Due to the complexity of the problem, it is hard for CPLEX to obtain optimal solutions for medium- and large-scale problems. In Table 4, IGA is disadvantageous compared to SWO in terms of CPU time, but it is within an acceptable range. In terms of solution, the Gap_{RE} (%) between SWO and IGA increases from 3.67% to 16.54%, suggesting that IGA is superior to the SWO algorithm in most instances, especially in large-scale problems, the Gap_{RE} even exceeds 20%.

Table 3. Comparisons of CPLEX, SWO, and IGA in small-scale instances.

Small Instances	CPLEX		SWO			IGA		
	Obj_cp	CPU (s)	Obj_SWO	CPU (s)	Gap_SWO (%)	Obj_IGA	CPU (s)	Gap_IGA (%)
$(I, K_1, K) = (6, 2, 8)$								
6_1	16.0	21.31	16.0	0.46	0.00	16.0	0.73	0.00
6_2	13.6	69.00	13.6	0.44	0.00	13.6	0.74	0.00
6_3	16.0	75.47	16.0	0.32	0.00	16.0	0.77	0.00
6_4	15.2	370.35	15.2	0.42	0.00	15.2	0.72	0.00
6_5	17.2	99.98	17.2	0.40	0.00	17.2	0.62	0.00
6_6	16.4	182.02	20.2	0.42	0.00	16.4	1.48	0.00
6_7	16.6	58.63	18.1	0.58	23.41	16.6	1.42	0.00
6_8	16.8	30.67	16.8	0.35	15.90	16.8	0.78	0.00
6_9	15.6	181.08	15.6	0.41	0.00	15.6	0.82	0.00
6_10	15.2	71.86	15.2	0.57	0.00	15.2	0.69	0.00
Average	15.8	116.04	16.4	0.44	3.93	15.8	0.88	0.00
$(I, K_1, K) = (8, 2, 8)$								
8_1	20.8	2148.05	24.6	0.64	18.69	20.8	1.69	0.39
8_2	19.8	867.59	19.8	0.44	0.40	19.8	1.00	0.40
8_3	23.0	1260.00	23.1	0.47	0.52	23.1	1.12	0.52
8_4	22.0	1887.00	22.0	0.42	0.00	22.0	1.10	0.00
8_5	16.8	895.53	16.8	0.48	0.00	16.8	1.10	0.00
8_6	20.4	1433.88	20.4	0.44	0.00	20.4	1.08	0.00
8_7	22.0	1458.45	22.0	0.56	0.00	22.0	1.61	0.00
8_8	23.0	707.32	23.0	0.65	0.35	23.0	1.22	0.35
8_9	22.2	239.86	22.8	0.35	2.70	22.2	1.55	0.00
8_10	20.0	149.70	20.0	0.36	0.00	20.0	1.09	0.00
Average	21.0	1104.74	21.5	0.48	2.27	21.0	1.26	0.17
$(I, K_1, K) = (10, 2, 8)$								
10_1	28.3	2673.74	31.1	0.33	10.04	28.8	1.95	1.70
10_2	26.5	8287.09	26.6	0.52	0.30	26.6	1.63	0.30
10_3	27.6	22213.89	27.6	0.31	0.00	27.6	1.83	0.00
10_4	30.0	19121.29	31.1	0.51	3.87	30.0	1.47	0.27
10_5	26.0	1567.89	26.1	0.38	0.31	26.1	1.65	0.31
10_6	25.2	4841.43	25.6	0.40	1.59	25.2	2.24	0.00
10_7	27.6	10015.62	27.6	0.55	0.00	27.6	1.79	0.00
10_8	23.2	3202.77	23.3	0.50	0.34	23.3	1.81	0.34
10_9	26.3	6486.41	26.7	0.53	1.67	26.7	2.73	1.67
10_10	24.8	3393.17	24.8	0.38	0.00	24.8	2.76	0.00
Average	26.6	8180.33	27.1	0.44	1.81	26.7	1.99	0.46
$(I, K_1, K) = (12, 2, 8)$								
12_1	33.5	12633.85	34.1	0.51	2.22	33.7	2.75	0.84
12_2	31.3	9983.61	32.0	0.72	2.11	31.7	2.34	1.15
12_3	32.7	14213.93	33.8	0.45	3.49	32.7	3.23	0.00
12_4	33.7	21123.51	33.9	0.48	0.65	33.7	4.56	0.12
12_5	30.6	11569.45	30.9	0.56	1.11	30.7	2.45	0.26
12_6	29.2	15851.87	30.1	0.63	3.22	29.2	3.39	0.00
12_7	32.4	18036.41	33.3	0.71	2.78	32.4	2.98	0.12
12_8	28.7	17201.89	32.0	0.67	11.43	29.1	3.81	1.46
12_9	32.6	19486.32	33.9	0.73	4.05	32.9	3.89	0.86
12_10	30.2	17393.67	31.0	0.59	2.65	30.2	2.35	0.00
Average	31.5	15749.45	32.4	0.61	3.37	31.6	3.18	0.48

Note: $\text{Gap_SWO} (\%) = (\text{Obj_SWO} - \text{Obj_cp}) / \text{Obj_cp}$, $\text{Gap_IGA} (\%) = (\text{Obj_IGA} - \text{Obj_cp}) / \text{Obj_cp}$.

Table 4. Comparisons of SWO and IGA in medium- and large-scale instances.

Instances	IGA		SWO		GAP
	Obj_IGA	CPU(s)	Obj_SWO	CPU(s)	Gap_RE(%)
		(I, K ₁ , K) = (15, 3, 10)			
15_1	46.12	8.28	48.24	0.73	4.60
15_2	40.64	5.69	40.64	0.64	0.00
15_3	43.68	7.40	43.68	0.48	0.00
15_4	45.88	14.32	47.2	1.04	2.88
15_5	42.68	12.43	45.2	0.88	5.90
15_6	43.72	12.31	44.32	1.03	1.37
15_7	50.24	7.36	57.68	0.98	14.81
15_8	43.64	6.41	43.84	0.97	0.46
15_9	43.32	5.27	43.6	0.90	0.65
15_10	40.88	6.84	43.36	0.76	6.07
Average	44.08	8.63	45.78	0.84	3.67
		(I, K ₁ , K) = (20, 4, 12)			
20_1	69.12	20.18	69.12	1.00	10.27
20_2	56.16	9.80	56.16	1.22	8.84
20_3	76.00	22.49	76.00	1.74	27.01
20_4	55.84	8.65	55.84	1.40	-0.36
20_5	65.36	13.88	65.36	1.61	16.38
20_6	62.40	9.97	62.40	1.41	4.21
20_7	61.52	20.05	61.52	1.62	0.33
20_8	64.40	18.88	64.40	1.42	7.98
20_9	69.28	14.51	69.28	1.44	7.98
20_10	60.56	16.99	60.56	1.57	7.45
Average	58.77	15.54	64.06	1.44	9.01
		(I, K ₁ , K) = (30, 4, 16)			
30_1	94.40	67.15	102.32	2.19	8.39
30_2	109.48	72.81	116.08	2.20	6.03
30_3	95.64	62.62	104.40	2.29	9.16
30_4	95.92	67.02	103.20	2.06	7.59
30_5	98.20	64.43	113.04	2.30	15.11
30_6	101.08	85.82	114.24	1.95	13.02
30_7	93.12	80.38	107.36	2.10	15.29
30_8	94.48	63.13	111.28	1.87	17.78
30_9	105.24	75.61	111.12	2.15	5.59
30_10	101.44	58.08	110.48	2.11	8.91
Average	98.90	69.66	109.35	2.12	10.69
		(I, K ₁ , K) = (40, 5, 12)			
40_1	156.44	216.34	183.56	4.15	17.34
40_2	137.88	206.80	164.44	4.01	19.26
40_3	159.28	224.74	194.92	4.31	22.38
40_4	153.40	200.25	167.80	3.16	9.39
40_5	153.00	182.97	185.64	4.23	21.33
40_6	172.24	187.89	188.68	4.25	9.54
40_7	138.36	219.58	157.48	4.27	13.82
40_8	158.60	207.56	185.96	4.34	17.25
40_9	156.44	173.43	192.20	5.00	22.86
40_10	148.32	196.79	166.44	3.55	12.22
Average	153.40	201.64	178.71	4.13	16.54

Note: Gap_RE (%) = (Obj_SWO – Obj_IGA)/Obj_IGA.

5.3. The Impact of the Coefficient

This section discusses the impact of the weight coefficient ω on the objective function (30). The coefficient ω is set from 0.1 to 1, increasing by 0.1 successively. The computational results are shown in Figure 8.

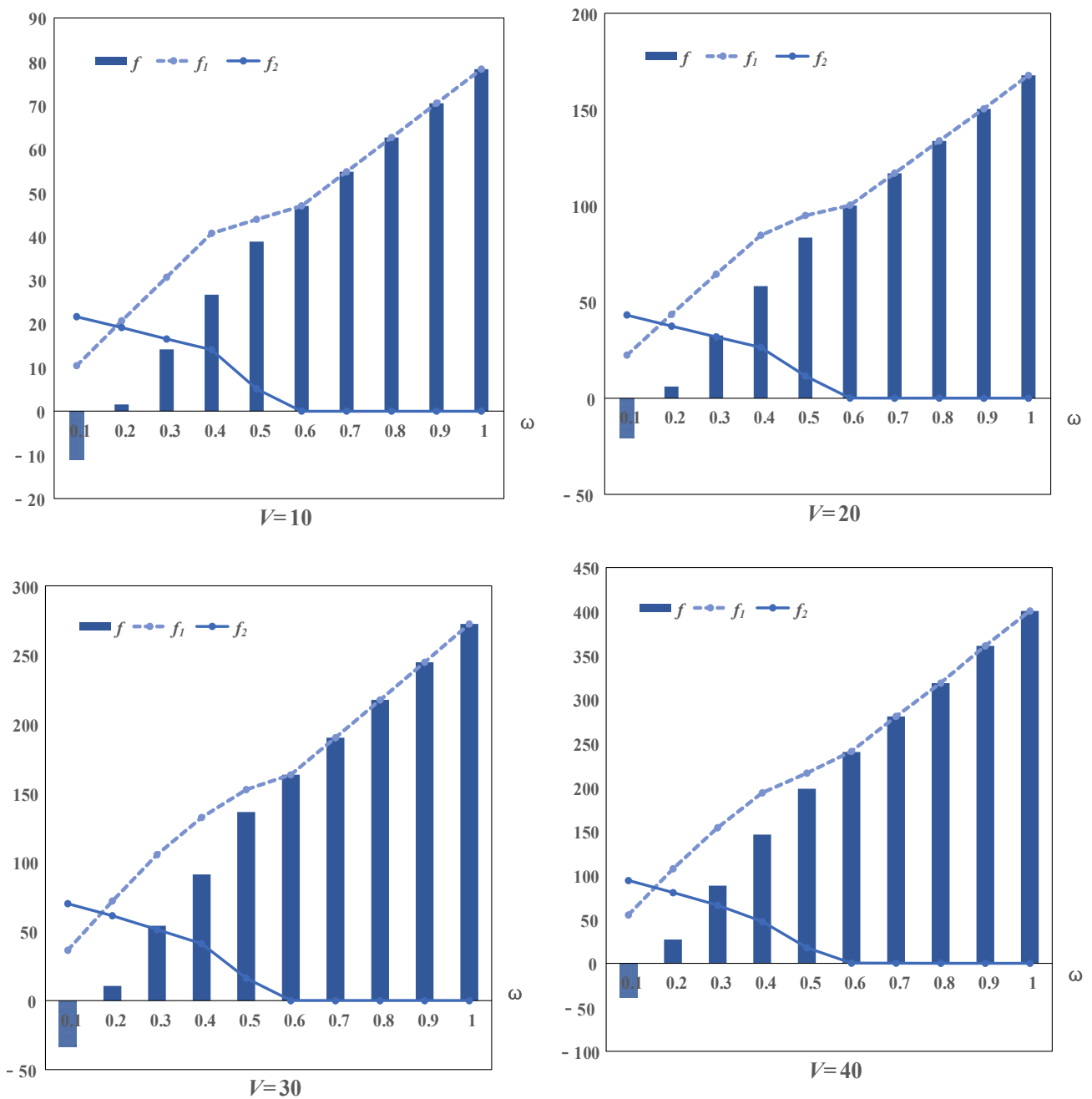


Figure 8. Impact of the coefficient on the objective function.

For different numbers of vessels, the value of the objective function f_1 gradually increases with the coefficient, while the value of the objective function f_2 gradually decreases correspondingly. When the coefficient exceeds 0.6, the f_2 is very close to 0; that is, there is no buffer time.

5.4. Analysis of Uncertainty and Buffer Time

This section evaluates the impact of uncertainty and buffer limit on the original schedule. The vessel arrival time is uniformly distributed as $A_i \in [A_i - a, A_i + a]$, where $a \in \{3, 5, 7\}$; buffer time $\theta \in \{\sigma, 1.5\sigma, 2\sigma\}$, where deviation $\sigma = \frac{a}{\sqrt{3}}$.

R_s represents the robustness of the solution, i.e., the variation between the reactive schedule and the original schedule in berthing time, berthing position, number of QCs, and other decision variables. The less the variation, the better the robustness of the solution.

Since the actual delay of vessels significantly influences the reactive schedule, without loss of generality, for an original schedule, we set up N cases. For each case, the original schedule is substituted into the recovery model to obtain the recovery schedule, and further evaluate the solution robustness of case n (denoted by R_s^n). Based on the objective function (37) of the reactive model, R_s^n is formulated as Equation (45).

$$R_s^n = \sum_i \left(|e_i^R - e_i^*| \right) + \sum_i \left(|s_i^R - s_i^*| \right) + \sum_i \left(|b_i^R - b_i^*| \right) + \sum_i \left(|q_i^R - q_i^*| \right) \quad (45)$$

The robustness of the solution R_s is the expectation of the robustness of N cases, the formulation is as follows.

$$R_s = E \left(R_s^1, \dots, R_s^n, \dots, R_s^N \right) \quad (46)$$

The schedules obtained by the reactive strategy, deterministic model, and baseline model are compared in terms of the robustness of the solution. R_{S_C} and R_{S_R} represents the solution robustness of the deterministic schedule and the baseline schedule, compared with the reactive strategy, respectively. $R_{S_GAP}(\%)$ represents the gap between R_{S_C} and R_{S_R} , i.e., $R_{S_GAP}(\%) = \frac{R_{S_R} - R_{S_C}}{R_{S_C}} \times 100\%$. The experimental results are illustrated in Figure 9.

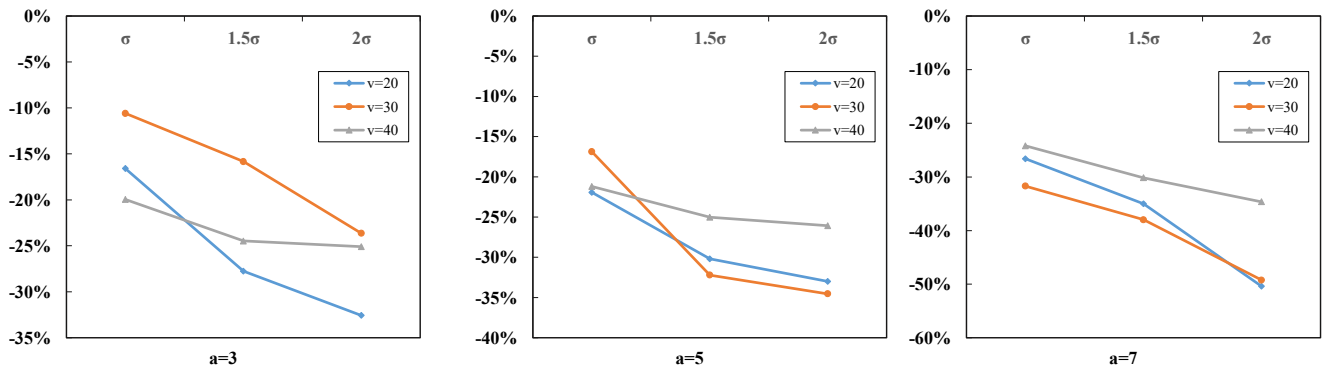


Figure 9. Comparison of $R_{S_GAP}(\%)$ values under different vessel scales, uncertainty, and buffer time.

In Figure 9, for $a = 3$ and $a = 5$, the difference in $R_{S_GAP}(\%)$ between σ and 1.5σ is larger than that between 1.5σ and 2σ ; that is, the growth rate of the difference between the baseline schedule and deterministic schedule decreases. On the contrary, when $a = 7$, the corresponding difference in $R_{S_GAP}(\%)$ increases. Here, the upper bound of buffer time is 2σ . In addition, the buffer time increases, and the $R_{S_GAP}(\%)$ decrease, which means the more extensive the buffer time, the less significant the effect of risk resistance capability.

6. Conclusions

This article proposed an integrated proactive reactive strategy for BACAP with uncertain vessel arrival time and QC preventive maintenance activities. On this basis, a deterministic optimization model, a baseline planning model, and a reactive model were established. In the proactive phase, the buffer time was introduced to absorb uncertain disturbance of vessel arrival time. The reactive phase reduced the deviation between the actual and original plan. To solve this problem efficiently, an improved genetic algorithm was combined with a partition check-based local search. Comprehensive numerical experiments validated the effectiveness of the strategy, the models, and the algorithms. First, by comparing the CPU time and solution, it is found that IGA is much faster than CPLEX when solving small-scale problems, and the solution of IGA for solving medium- and large-scale problems is due to SWO. Second, when the coefficient exceeds 0.6, that is, when the port manager pays more attention to the operation efficiency of the terminal than to the stability of the operation, increasing the buffer time to absorb uncertainty is no longer a good choice. Moreover, when the uncertainty of vessel arrival time is low, an excessive

increase in buffer time will react; when the uncertainty is high, increasing the buffer time can reduce the additional cost brought by the uncertainty to port management.

In the future, more uncertain factors should be involved to further identify the efficiency of the strategy and the proposed algorithm. The accidental failure of QCs and differences in labor efficiency are also interesting topics for future research.

Author Contributions: Conceptualization, H.H.; Methodology, Z.W.; Validation, J.C.; Investigation, J.C.; Writing—original draft, J.C.; Writing—review & editing, Z.W.; Visualization, J.C.; Supervision, Z.W. and H.H.; Funding acquisition, Z.W. and H.H. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by Youth Program of Soft Science Research Project, Science and Technology Innovation Plan of Shanghai Science and Technology Commission [grant number 23692118400]; the National Natural Science Foundation of China [grant number 72271156]; Humanities and Social Sciences Youth Foundation, Ministry of Education of the People’s Republic of China [grant number 20YJC630155]; the Shanghai Shuguang Scholar Project, China [grant number 20SG46].

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: All data, models, or code generated or used during the study are available from the corresponding author by request.

Conflicts of Interest: The authors declare no conflicts of interest.

Nomenclature

Indices and sets

I_1	Set of actual vessels;
I_2	Set of virtual vessels;
I	Set of all vessels, $i, j \in I, I = I_1 \cup I_2, I = \{1, 2, \dots, V\}$;
K_1	Set of QCs requiring maintenance activities;
K	Set of QCs, $k, g \in K, K_1 \in K, K = \{1, 2, \dots, N\}$, k, g are specific QC numbers, and assigned number of QCs is q_i ;
T	Set of time steps in the planning time horizon, $t \in T, T = \{1, 2, \dots, H\}$, H : the length of the time horizon;
Z	Set of correspondence between a virtual vessel i and maintained QC k , $(i, k) \in Z$
Ω	Set of scenario, $\omega \in \Omega$.

Parameters

L	Length of the quay (unit: 10 m);
l_i	Length of the vessel $i \in I$ (unit: 10 m);
β_i	Time penalty coefficient for vessel i deviating from preferred berth;
b_i^0	Tesired berthing position of the vessel i ;
a_i	Expected arrival time of a vessel i , is a random parameter with a probability density function;
F_i	Crane capacity demand of vessel i given as several QC hours;
α	Coefficient of increase in the QC capacity demand with deviation from desired berthing position;
γ	Interference exponent for the QCs. Only q^γ effective QC hours are obtained when assigning q QCs to a vessel for one hour;
q_i^{min}	Minimum number of QCs agreed to serve the vessel i simultaneously;
q_i^{max}	Maximum number of QCs allowed to serve vessel i simultaneously;
β_{ik}	Correspondence relation between the vessel i and QC k ;
σ_k	Duration of quay crane k 's preventive maintenance activity;
d_k	Working coverage width of quay crane k , including the actual working range of QC and the safe working distance between QCs;
δ_k^{min}	Earliest start time for performing quay crane k 's maintenance activity, $\delta_k^{min} \in T$;
δ_k^{max}	Latest start time for performing quay crane k 's maintenance activity, $\delta_k^{max} \in T$;
ω	Weight in the target function;

b_i^*	Berthing position obtained in the original robust model;
s_i^*	Berthing time obtained in the original robust model;
q_i^*	Number of QCs obtained from the original robust model;
c_i^1	Coefficient of the turnaround time of vessels in port cost;
c_i^2	Cost coefficient of deviation from the desired berth;
c_i^3	Cost coefficient, which is caused by the delay (or advance) of the vessel's berthing time relative to the original plan in the recovery process;
c_i^4	Cost coefficient, which is caused by the right (or left) movement of the vessel's berthing position relative to the original plan in the recovery process;
c_i^5	Cost coefficient, which is caused by the increase (or decrease) of the number of QCs relative to the original plan in the recovery process.

Decision variables

s_i	Time of starting the handling of a vessel i (berthing time);
e_i	Time of ending the handling of a vessel i (including buffer time);
b_i	Berthing position of a vessel i ;
Δb_i	The deviation between the preferred and the actual chosen berthing position of the vessel i , $\Delta b_i = b_i - b_i^0 $;
δ_k	The actual time at which quay crane k starts maintenance, $\delta_k \in [\delta_k^{min}, \delta_k^{max}]$;
x_{ij}	Binary, set to 1 if the vessel i is berthed below vessel j , 0 otherwise $0-1, \forall i, j \in I, i \neq j$;
y_{ij}	Binary, set to 1 if the handling of the vessel i ends not later than the handling of vessel j starts, 0 otherwise, $\forall i, j \in I, i \neq j$;
w_{it}	Binary, set to 1 if at least one QC is assigned to the vessel i at time t , 0 otherwise, $\forall i \in I, t \in T$;
p_{itk}	Binary, set to 1 if QC k is assigned to the vessel i at time t , 0 otherwise, $\forall i \in I, t \in T, k \in K$;
θ_i	Buffer time of the i^{th} vessel;
$\theta_{i\tau q}$	Binary, set to 1 if exactly q QCs are assigned to the vessel i and the buffer time is τ , 0 otherwise;
θ_{itq}	Binary, set to 1 if exactly q QCs are assigned to the vessel i at time t , 0 otherwise, $\forall i \in I, t \in T, q \in K$;
$s_i^+(\omega), s_i^-(\omega)$	Increment (or decrement) relative to s_i in scenario ω ;
$s_i^0(\omega)$	Increment of berthing time, increment of s_i^+ relative to the buffer time θ_i in scenario ω ;
$b_i^+(\omega), b_i^-(\omega)$	Increment (or decrement) relative to b_i in scenario ω ;
$q_i^+(\omega), q_i^-(\omega)$	Increment (or decrement) relative to q_i in scenario ω .

References

1. Zhen, L.; Zhuge, D.; Wang, S.; Wang, K. Integrated berth and yard space allocation under uncertainty. *Transp. Res. Part B Methodol.* **2022**, *162*, 1–27. [\[CrossRef\]](#)
2. Park, Y.M.; Kim, K.H. A Scheduling Method for Berth and Quay Cranes. *OR Spectr.* **2003**, *25*, 1–23. [\[CrossRef\]](#)
3. Bierwirth, C.; Meisel, F. A follow-up survey of berth allocation and quay crane scheduling problems in container terminals. *Eur. J. Oper. Res.* **2015**, *244*, 675–689. [\[CrossRef\]](#)
4. Carlo, H.J.; Vis, I.F.A.; Roodbergen, K.J. Seaside operations in container terminals: Literature overview, trends, and research directions. *Flex. Serv. Manuf. J.* **2015**, *27*, 224–262. [\[CrossRef\]](#)
5. Iris, Ç.; Pacino, D.; Ropke, S.; Larsen, A. Integrated berth allocation and quay crane assignment problem: Set partitioning models and computational results. *Transp. Res. Part E Logist. Transp. Rev.* **2015**, *81*, 75–97. [\[CrossRef\]](#)
6. Salhi, A.; Alsoufi, G.; Yang, X. An evolutionary approach to a combined mixed integer programming model of seaside operations as arise in container ports. *Ann. Oper. Res.* **2019**, *272*, 69–98.
7. Zheng, F.F.; Li, Y.; Chu, F.; Liu, M.; Xu, Y.F. Integrated Berth Allocation and Quay Crane Assignment with Maintenance Activities. *Int. J. Prod. Res.* **2019**, *57*, 3478–3503. [\[CrossRef\]](#)
8. Chiu, Y.F.; Shih, C.J. Rescheduling Strategies for Integrating Rush Orders with Preventive Maintenance in a Two-Machine Flow Shop. *Int. J. Prod. Res.* **2012**, *50*, 5783–5794. [\[CrossRef\]](#)
9. Zhen, L. Tactical berth allocation under uncertainty. *Eur. J. Oper. Res.* **2015**, *247*, 928–944. [\[CrossRef\]](#)
10. Shang, X.T.; Cao, J.X.; Ren, J. A Robust Optimization Approach to the Integrated Berth Allocation and Quay Crane Assignment Problem. *Transp. Res. Part E Logist. Transp. Rev.* **2016**, *94*, 44–65. [\[CrossRef\]](#)
11. Liu, C.C.; Xiang, X.; Zheng, L. A two-stage robust optimization approach for the berth allocation problem under uncertainty. *Flex. Serv. Manuf. J.* **2020**, *32*, 425–452. [\[CrossRef\]](#)
12. Xiang, X.; Liu, C.C. An almost robust optimization model for integrated berth allocation and quay crane assignment problem. *Omega-Int. J. Manag. Sci.* **2021**, *104*, 102455. [\[CrossRef\]](#)
13. Zhen, L.; Lee, L.H.; Chew, E.P. A decision model for berth allocation under uncertainty. *Eur. J. Oper. Res.* **2011**, *212*, 54–68. [\[CrossRef\]](#)
14. Iris, Ç. *Exact and Heuristic Methods for Integrated Container Terminal Problems*; DTU Management Engineering: Kongens Lyngby, Denmark, 2016; 231p.

15. Zhen, L.; Sun, Q.; Zhang, W.; Wang, K.; Yi, W. Column generation for low carbon berth allocation under uncertainty. *J. Oper. Res. Soc.* **2021**, *72*, 2225–2240. [[CrossRef](#)]
16. Park, H.J.; Cho, S.W.; Lee, C. Particle swarm optimization algorithm with time buffer insertion for robust berth scheduling. *Comput. Ind. Eng.* **2021**, *160*, 107585. [[CrossRef](#)]
17. Agra, A.; Rodrigues, F. Distributionally robust optimization for the berth allocation problem under uncertainty. *Transp. Res. Part B Methodol.* **2022**, *164*, 1–24.
18. Chargui, K.; Zouadi, T.; Sreedharan, V.R. Berth and quay crane allocation and scheduling problem with renewable energy uncertainty: A robust exact decomposition. *Comput. Oper. Res.* **2023**, *256*, 106251.
19. Li, M.Z.; Jin, J.G.; Lu, C.X. Real-Time Disruption Recovery for Integrated Berth Allocation and Crane Assignment in Container Terminals. *Transp. Res. Rec.* **2015**, *2479*, 49–59. [[CrossRef](#)]
20. Umang, N.; Bierlaire, M.; Erera, A.L. Real-Time Management of Berth Allocation with Stochastic Arrival and Handling Times. *J. Sched.* **2017**, *20*, 67–83. [[CrossRef](#)]
21. Xiang, X.; Liu, C.C.; Miao, L.X. Reactive Strategy for Discrete Berth Allocation and Quay Crane Assignment Problems under Uncertainty. *Comput. Ind. Eng.* **2018**, *126*, 196–216.
22. Lv, X.H.; Jin, J.G.; Hu, H. Berth allocation recovery for container transshipment terminals. *Marit. Policy Manag.* **2020**, *47*, 558–574. [[CrossRef](#)]
23. Liu, C.C.; Xiang, X.; Zheng, L. Two decision models for berth allocation problem under uncertainty considering service level. *Flex. Serv. Manuf. J.* **2017**, *29*, 312–344.
24. Iris, Ç.; Lam, S.L.J. Recoverable robustness in weekly berth and quay crane planning. *Transp. Res. Part B Methodol.* **2019**, *122*, 365–389. [[CrossRef](#)]
25. Iris, Ç.; Pacino, D.; Ropke, S. Improved formulations and an adaptive large neighborhood search heuristic for the integrated berth allocation and quay crane assignment problem. *Transp. Res. Part E Logist. Transp. Rev.* **2017**, *105*, 123–147. [[CrossRef](#)]
26. Tan, C.M.; He, J. Integrated proactive and reactive strategies for sustainable berth allocation and quay crane assignment under uncertainty. *Ann. Oper. Res.* **2021**; advanced online publication. [[CrossRef](#)]
27. Dai, H.; Ma, J.; Yang, Y.; Sun, J.; Dai, Y. A bi-layer model for berth allocation problem based on proactive-reactive strategy. *Comput. Ind. Eng.* **2023**, *179*, 109200. [[CrossRef](#)]
28. Li, Y.; Chu, F.; Zheng, F.; Kacem, I. Integrated Berth Allocation and Quay Crane Assignment with Uncertain Maintenance Activities. In Proceedings of the 2019 International Conference on Industrial Engineering and Systems Management (IESM), Shanghai, China, 25–27 September 2019; pp. 1–6.
29. Li, Y.; Chu, F.; Zheng, F.; Liu, M. A Bi-Objective Optimization for Integrated Berth Allocation and Quay Crane Assignment with Preventive Maintenance Activities. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 2938–2955.
30. Joslin, D.E.; Clements, D.P. Squeaky wheel optimization. *J. Artif. Intell. Res.* **1999**, *10*, 353–373.
31. Meisel, F.; Bierwirth, C. Heuristics for The Integration of Crane Productivity in The Berth Allocation Problem. *Transp. Res. Part E Logist. Transp. Rev.* **2009**, *45*, 196–209. [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.