The Integrated Rescheduling Problem of Berth Allocation and Quay Crane Assignment with Uncertainty

Hongxing Zheng *, Zhaoyang Wang and Hong Liu

Abstract: The baseline plan of terminals will be impacted to a certain extent after being affected by uncertain events, such as vessel delay and unscheduled vessel arrival, resulting in disorderly terminal operations, wasted resources, and reduced loading and unloading efficiency, which further aggravates terminal congestion. To effectively cope with the disturbance of terminal operations by the above uncertain events and improve the operational efficiency of container terminals, this paper investigates the integrated rescheduling problem of berth allocation and quay crane assignment with vessel delay and unscheduled vessel arrival. Two steps are designed to deal with uncertainty shocks. The first step is to determine the rescheduling moment by using a rolling time-domain approach. The second step is to establish a rescheduling model and design an improved genetic algorithm (IGA) to obtain a rescheduling solution using various rescheduling strategies at the rescheduling moment. Moreover, through scenario experiments, comparisons with commercial solvers and other algorithms, it can be seen that the solution speed of IGA is better than that of commercial solvers and the average gap does not exceed 6%, which verifies the effectiveness and superiority of this algorithm.

Keywords: integrated rescheduling; berth allocation; quay crane assignment; improved genetic algorithm

1. Introduction

Because about 80% of global cargo transportation is by sea, maritime container transportation has become the most important type of transportation for international trade. In recent years, the low rate of container vessels arriving on schedule due to uncertainties, such as epidemics and wars, has had a significant impact on container terminal operations, leading to severe congestion in ports around the world. Berth and quay cranes are important operational resources for terminals. The key to ensuring the terminal’s efficient operation is a suitable berth allocation and quay crane assignment scheme. The integrated rescheduling problem of the berth allocation and quay crane assignment problem with uncertainty is investigated in this paper to enhance the operational efficiency of container terminals by coping with the disturbance of terminal operations caused by events such as vessel delays and unscheduled vessel arrivals.

As the key resource of the container terminal at sea, the berth allocation and quay crane assignment problem (BAQCAP) has been a hot topic of concern for all parties concerned [1]. The integrated scheduling of berths and quay cranes has attracted the attention of scholars, and a wealth of research results have been achieved [2–7]. At present, most of the literature takes the premise that the vessels arrive at the scheduled time and the number of arriving vessels remains unchanged. It also takes all the vessels expected to arrive at a container terminal in a single planning period as the optimization objectives, such as the minimum total waiting time for all vessels or the earliest end time for the last vessel, and finally gives the berth position, handling time, and the number of allocated quay cranes for each expected arriving vessel, which has certain guidance and reference values for the daily operations of the terminal. However, uncertain events such as vessel delays and
unscheduled vessel arrivals often occur in operation. The above uncertain events may lead terminals into such predicaments as berth resource constraints and abnormal blockage during some hours. The study of integrated rescheduling of berths and quay cranes under the influence of such events is valuable to the application of terminals and has received much attention.

In earlier studies, scholars set buffers to deal with the disturbance of uncertain events before terminal operations. For example, Rodriguez-Molins et al. [8] established multi-objective robust optimization by introducing buffer times to maximize the absorption of possible uncertainty events and to solve them by a genetic algorithm. Zhen and Chang [9] proposed a bi-objective optimization model with the objectives of minimizing operation costs and maximizing operation plan robustness by considering the uncertainties (e.g., vessel arrival time and handling time), and designed a heuristic algorithm to solve the large-scale problem. Zhen et al. [10] considered the uncertainty of vessel arrival time and operation time to establish a stochastic model, considered a robust model with limited information on probability distribution, analyzed the relationship between the two models, and proposed a variety of heuristic algorithms for the solution. Xu et al. [11] considered the uncertainty of vessel delay and handling time, inserted a time buffer between two vessels at the same berth to improve the robustness of the scheme, established a model intending to minimize the total delay time of all vessels and maximize the robustness of the scheme, and designed a robust berth scheduling algorithm (RBSA) solution combining simulated annealing and branch delimitation algorithms.

Following this research line, Du [12] considered, for the first time, the specific impact of uncertainty scenarios in the plan implementation process, introduced a feedback mechanism into the berth allocation problem for vessel delays, and emphasized the use of real-time operational progress to guide the time buffer settings for different phases. Zhen et al. [13] studied the berth allocation problem with uncertainty in vessel arrival time and handling time, considered a certain degree of uncertainty in the baseline plan, proposed a proactive recovery strategy in the plan execution, and established a two-stage model under uncertainty. Liu et al. [14] developed a baseline plan cost minimization decision model, including delay cost and offset preference berth cost, by using discrete scenarios to explicitly deal with uncertainty, considering uncertainty in vessel arrival time and handling time. Xiang et al. [15] studied the BAP problem with uncertainty in vessel arrival time and handling time, dealt with the uncertainty linearly using a discretized scenario, developed a bi-objective robust optimization model focusing on economic efficiency and customer satisfaction, and designed an adaptive gray wolf algorithm to solve it. Sheikholeslami and Ilat [16] considered the effect of tides on the port berth allocation scheme and the uncertainty of vessel arrival time. They also proposed a rigorous mathematical model that presented the sample average approximation method to generate an efficient berth allocation scheme. Liu et al. [17] used the uncertainty set to describe the possible scenarios without using probability distributions and designed a two-stage robust model that gave the baseline plan first and the rescheduling plan after the uncertainty scenario was explicit. Rodrigues and Agra [18] considered the berth allocation and quay cranes assignment problem with uncertain vessel arrival time and developed a two-stage robust mixed-integer programming model where berth allocation occurred before the exact arrival time and the adjusted crane assignment operations, according to the arrival time. Xiang and Liu [19] considered the integrated berth allocation and quay crane assignment problem with uncertain vessel delays and container quantities, developed a robust model to minimize the total cost, and proposed a decomposition method to solve the problem.

Both of the above two types of studies set aside a time buffer or resource buffer in the plan, and once the uncertainty event has not occurred during the operation, the reserved resources are wasted, so the operation scheme given by the above studies is relatively conservative. Therefore, some scholars have proposed a reactive approach in recent years, which first gives a baseline plan to minimize costs or vessel handling time, and then adjusts the baseline plan when an uncertainty event occurs. Liu et al. [20]
focused on the problem of rescheduling quay cranes in the event of a breakdown during plan execution and rescheduling the system to minimize negative deviations from the baseline plan. Xiang et al. [21] studied the problem of container terminals in uncertain berth allocation and quay crane assignments and developed a mixed-integer programming model considering practical constraints to obtain a baseline plan to minimize the baseline cost. Considering disruptions (e.g., vessel arrival time deviation, vessel handling time deviation, unscheduled vessel call, and quay crane breakdown) when executing the baseline plan, a response strategy with the baseline plan as a reference and the objective of minimizing the recovery cost was proposed. Li et al. [22] studied the disruption recovery optimization problem in the integrated berth allocation and quay crane assignment problem of container terminals and developed a multi-objective programming model to maximize the service quality while minimizing the recovery cost. They also designed a heuristic method based on squeaky wheel optimization to solve the model. Al-Refaie and Abedalqader [23] studied the berth allocation problem under unscheduled vessel arrivals, proposed an optimization method for container ports when an unscheduled vessel arrives, and developed three sequential models to maximize the number of unscheduled vessels, served with minimal disruption to the regular vessel service schedule. Kim [24] proposed a method of berth and quay crane rescheduling caused by the updated information on vessel arrival time to solve the problem of frequent fluctuations in vessel arrival time at container terminals, established a mixed-integer linear programming model for the berth allocation and quay crane allocation problem, and solved it using a rolling-horizon method.

From an application point of view, the reactive approach is divided into two steps in the application process. The first step is the identification of uncertain events. Xiang et al. [21] proposed a reactive strategy (a rolling-horizon heuristic method), which takes the baseline schedule as a reference and aims to minimize the recovery cost. The second step is to design the algorithm to solve the model and obtain the rescheduling scheme. Tan et al. [25] developed a two-stage metaheuristic framework based on GA to solve this problem. Li et al. [26] proposed an efficient optimization algorithm that is a hybrid of the iterated greedy and simulated annealing algorithms to solve the flexible job shop scheduling problem with crane transportation processes. Rodrigues et al. [7] counted the algorithms used in all relevant studies from 2006 to 2021, with metaheuristic algorithms accounting for 55% and genetic algorithms for 20% of all studies. It shows that GA is an effective algorithm for solving the BAQCAP, especially for continuous berth. Some scholars have also started to use machine-learning methods to solve such problems. Du et al. [27] proposed a deep Q-network model to solve a multi-objective flexible job shop scheduling problem with crane transportation and setup times, and in the following research, a flexible job shop scheduling problem with a time-of-use electricity price constraint is considered [28].

In summary, most of the existing literature on the BAQCAP for coping with the effects of various uncertain events initially used robust optimization or two-stage disturbance recovery models. Due to the high number and irregularity of various uncertain events in recent years and the high impact leading to the near impossibility of restoring the baseline plan, there has been a gradual increase in the literature on constructing rescheduling models [29–32], which can provide a reference for terminals when developing contingency plans. Additionally, there are still the following problems: the above article only considers some uncertain events in vessel delay, handling time uncertainty, and unscheduled vessels and does not include all three types of uncertain events. The time of rescheduling is determined in the reactive strategy, but the vessels that operate in the port are not considered to be moved to improve the utilization of berth.

2. Problem Formulation

During the planning period, vessels inform the terminal of their arrival information in advance, and the terminal develops a baseline plan based on the information and the terminal’s conditions. However, some vessels are affected by uncertain events during terminal operations that can cause vessel delays. At the same time, there are be unscheduled vessel
arrivals during the planning period. The baseline plan needs to be adjusted after vessel delays and unplanned vessel arrivals to avoid disorderly terminal operations, resulting in wasted resources and reduced operational efficiency.

In this study, the berth type refers to the continuous berth, where the vessel can berth at any position as long as the space allows. Quay cranes interfere with each other in operation, resulting in a decrease in operational efficiency. The marginal productivity loss index of shore bridges is denoted by $\alpha$. In addition, the operational efficiency is also affected by the transport time of land-side trucks. The closer a vessel berth is to its yard, the shorter the transport time of trucks, and these locations are usually the best berthing position for vessels.

A specific description is given in a two-dimensional coordinate diagram in Figure 1 to visualize the disruptions to the baseline plan caused by vessel delays and unplanned vessel arrivals. The horizontal axis represents the length of the shoreline, the vertical axis represents the time, the rectangles represent the vessels, and the length and width of the rectangle represent the length (e.g., $l_1$ of vessel 1) and the handling time (e.g., $h_1$ of vessel 1) of the vessel, respectively. The distribution of the quay cranes on the same track can be panned but not crossed. The number of quay cranes that can be allocated at the same time varies from vessel to vessel, and the longer the vessel, the higher the number of quay cranes that can be allocated at the same time (e.g., quay cranes [8,9] for vessel 1, quay cranes [5–7] for vessel 2). Because the vessels must ensure spatial and temporal independence between them, i.e., there can be no overlap between the two rectangles, when a vessel is delaying, it might overlap with others (e.g., vessel 4 and delayed vessel 3 overlapped). By the same token, unscheduled vessels (e.g., vessel 6) can cause overlap between vessels. Both of these situations can lead to confusion in the baseline plan and therefore require timely adjustments to the baseline plan. The optimization objective of rescheduling is to minimize the delay between the actual departure time of the original vessels and the expected departure time in the baseline plan, as well as the waiting time of the unscheduled vessels.

The rescheduling strategy includes moving the vessels operating in the terminal, increasing the number of quay cranes, adjusting the berthing time of vessels, and reallocating berth positions, as shown in Figure 2. The red rectangle in the figure shows the vessels in the baseline plan and the black rectangle shows the vessels in the rescheduling plan. In Figure 2, vessel 1 and vessel 2, which are operating, are shifted to the sides so that space can be freed up for the unscheduled vessel 9 to berth. Because vessel 3 is delayed, the quay cranes are added to vessel 3 to reduce the handling time and avoid overlapping
with vessel 4, which is berthed later. However, not all vessels can add quay cranes in the baseline plan, so vessels that need to berth in the back order adjust their berthing time to avoid overlap (e.g., vessel 6 adjusts its berthing time backward to avoid overlapping with vessel 5). If overlap cannot be avoided by adding quay cranes and adjusting berthing times, the berths need to be reallocated (e.g., vessel 7 is delayed so the berthing positions of vessel 7 and vessel 8 are adjusted). Again, the above adjustment strategies are all applicable to overlaps caused by unscheduled vessels, and a combination of rescheduling strategies can be implemented for the same vessel.

![Rescheduling strategies diagram](image)

**Figure 2.** Rescheduling strategies diagram.

### 3. Model Formulation

For the problem, the following reasonable assumptions are made in this paper.

1. All quay cranes are of the same type and quality and do not break down during operation.
2. Handling operations will not cause an impact on the balance of the vessel.
3. The waiting time for trucks and tugboats is negligible.
4. The channel and berth water depths do not affect the vessels.
5. The terminal is informed immediately after the vessel has been delayed.

### 3.1. Notations

Parameters and Sets:
- **V**: Set of vessels, the number of vessels \( |V| \).
- **l_i**: The length of the vessel \( i \).
- **S_i**: QCs capacity demand of vessel \( i \) was given as a number of QC-hours.
- **K**: Set of QCs, \( k = 1, 2, \ldots, |K| \), where \( |K| \) denotes the total number of QCs.
- **a_k, b_k**: The movement range of QC \( k \).
- **ω**: QC interference exponent.
- **q_i^{min}**: The minimum number of QCs that must be assigned to vessel \( i \).
- **q_i^{max}**: The maximum number of QCs that can be assigned to vessel \( i \).
- **t**: Set of one-hour periods, \( t = 1, 2, 3, \ldots T \), \( T \) is the planning horizon.
- **L**: Length of berth.
- **B_i**: Best berthing position of vessel \( i \).
- **σ**: Berth deviation factor, which refers to the increased proportion in QC-hours demand caused by per unit the distance between the vessel’s berthing position and the best berthing position.
- **SP_i**: Arrival time of vessel \( i \).
- **EB_i**: Original departure time.
\( \lambda_i \): Vessel \( i \) handling at berth at the time of rescheduling is 1, and 0 otherwise.

\( \varphi_i \): Equal to 1 if vessel \( i \) is unscheduled, 0 otherwise.

\( \delta_i \): Equal to 1 if vessel \( i \) has been moved before this rescheduling time, 0 otherwise.

\( m \): The maximum distance a vessel can be moved.

\( M \): A large enough positive number.

Variables:

- \( R_i \): Integer, number of quay cranes assigned to vessel \( i \).
- \( D_i \): Continuous, handling time of vessel \( i \).
- \( K_i \): Continuous, wait time of vessel \( i \).
- \( EB_i \): Continuous, departure time of vessel \( i \).
- \( \Delta_1 \): Continuous, time of vessel \( i \) later than the schedule.
- \( \Delta_2 \): Continuous, time of vessel \( i \) earlier than the schedule.
- \( \beta_i \): Integer, berthing position of vessel \( i \).
- \( y_{ij} \): Binary, equal to 1 if vessel \( i \) departs before the berthing start time of vessel \( i \), and 0 otherwise.
- \( x_{ij} \): Binary, equal to 1 if vessel \( i \) is berthed on the left of vessel \( i \), and 0 otherwise.
- \( \chi_{itk} \): Binary, equal to 1 if QC \( k \) is assigned to vessel \( i \) in a continuous \( t \) period, and 0 otherwise.
- \( r_{it} \): Binary, equal to 1 if vessel \( i \) is handling at \( t \) period, and 0 otherwise.
- \( \theta_i \): Binary, equal to 0 if vessel \( i \) leaves before the scheduled departure time, and 1 otherwise.
- \( \eta_i \): Binary, equal to 0 if vessel \( i \) moves the position, and 1 otherwise.

### 3.2. Rescheduling Model

This paper proposes a mixed integer linear programming model and employs linearization to facilitate solutions with commercial solvers. To this end, various scheduling strategies are integrated into the model, with the following objectives and constraints stated accordingly.

\[
\text{min} \quad z = \sum_{i \in V} \Delta_1_i + \Delta_2_i + \varphi_i D_i 
\]  
(1)

Equation (1) is the objective function that minimizes the sum of the vessel schedule deviation time during the planning period and the unscheduled vessel operation time.

\[
\beta_i + \frac{I_i + I_j}{2} \leq \beta_j + M(1 - x_{ij}), \forall i \in V, \forall j \in V, i \neq j
\]  
(2)

\[
SP_i + KB_i + D_i \leq SP_j + KB_j + M(1 - y_{ij}), \forall i \in V, \forall j \in V, i \neq j
\]  
(3)

\[
y_{ij} + y_{ji} + x_{ij} + x_{ji} \geq 1, \forall i \in V, \forall j \in V, i \neq j
\]  
(4)

Constraints (2)–(4) indicate the temporal and spatial independence between vessels.

\[
\beta_i + \frac{M_i}{2} \leq L, \forall i \in V
\]  
(5)

\[
\beta_i - \frac{M_i}{2} \geq 0, \forall i \in V
\]  
(6)

Constraints (5) and (6) indicate that the vessel must berth within the berth boundaries.

\[
EB_i = SP_i + KB_i + D_i, \forall i \in V
\]  
(7)

\[
EB_0^i - EB_i \leq M(1 - \theta_i), \forall i \in V
\]  
(8)

\[
(EB_0^i - EB_i)(1 - \theta_i) \geq 0, \forall i \in V
\]  
(9)

Constraints (7) indicates the actual departure time of the vessel, and the vessel’s time in port mainly includes the waiting time for berthing and the handling time. Constraints (8) and (9) are
used to judge the sequential relationship between the actual departure time of the vessel and the original departure time.

\[ \Delta 1_i = -\theta_i(EB^0_i - EB_i), \forall i \in V \]  
\[ \Delta 2_i = (1 - \theta_i)(EB^0_i - EB_i), \forall i \in V \]  

Constraints (10) and (11) indicate the delayed departure time and early departure time, respectively.

\[ R_i - \sum_{k \in K} \chi_{itk} \leq M(1 - r_{it}), \forall i \in V, \forall t \in T \]  
\[ \sum_{k \in K} \chi_{itk} = R_ir_{it}, \forall i \in V, \forall t \in T \]  

Constraints (12) and (13) indicate that the number of quay cranes is not variable when the vessel is handling.

\[ \sum_{i \in V} \chi_{itk} \leq 1, \forall t \in T, \forall k \in K \]  
\[ \sum_{k \in K} \chi_{itk} \leq q_i, \forall t \in T \]  
\[ \sum_{i \in V} \sum_{k \in K} \chi_{itk} \leq k_{max}, \forall t \in T \]  

Constraint (14) indicates that the quay crane can only provide loading and unloading services for one vessel in a period, and Constraint (15) indicates that the total number of quay cranes operating for a vessel in the same period cannot exceed the maximum number of quay cranes that can be allocated to the vessel, and Constraint (16) indicates that the number of quay cranes operating in the same period cannot exceed the total number of quay cranes.

\[ D_{it} = \sum_{i \in V} r_{it} \omega_i R_i \geq S_i(1 + \Delta^1_i), \forall i \in V, \forall t \in T \]  

Constraint (17) indicates the length of time that the vessel is handling.

\[ SP_i + KB_i - t \leq M(1 - \chi_{itk}), \forall i \in V, \forall t \in T, \forall k \in K \]  
\[ t - (SP_i + KB_i + D_i) \leq M(1 - \chi_{itk}), \forall i \in V, \forall t \in T, \forall k \in K \]  
\[ -1 \leq \chi_{itk} + \chi_{(it+1)k} - \chi_{(it+1)k} \leq 1, \forall i \in V, \forall t \in T, \forall k \in K \]  
\[ -1 \leq \chi_{itk} + \chi_{(it+2)k} - \chi_{(it+1)k} \leq 1, \forall i \in V, \forall t \in T, \forall k \in K \]  

Constraints (18) and (19) indicate that the vessel has berthed during the operation period of the quay crane, and Constraints (20) and (21) indicate that the spatial position and operation period of the quay crane are continuous.

\[ B_i + \frac{I_i}{2} \geq \chi_{itk} \omega_k, \forall i \in V, \forall t \in T, \forall k \in K \]  
\[ B_i - \frac{I_i}{2} - b_k \leq M(1 - \chi_{itk}), \forall i \in V, \forall t \in T, \forall k \in K \]  

Constraints (22) and (23) limit the movement range of the quay crane. The quay crane can only serve the vessel if the vessel is within the movement range of the quay crane.

\[ \eta_i \leq M(1 - \delta_i), \forall i \in V \]  
\[ \lambda_i \Delta_{1i} \leq m, \forall i \in V \]  
\[ \lambda_i \Delta_{1i} \leq M(1 - \eta_i), \forall i \in V \]
Constraints (24)–(26) indicate the constraints related to the berth shifting of vessels in the port. Constraint (24) ensures that the vessels that have been moved position during the planning period cannot be moved again to limit interference to the land transportation system caused by a vessel moving multiple times at multiple rescheduling decision points. Constraint (25) indicates that the vessel can only be moved within a certain distance to limit the horizontal moving distance of the vessel and to avoid change in the spatial relationship between two vessels causing the vessel to leave the berth before berthing. Additionally, Constraint (26) ensures that only the vessels operating in the port can move the position.

4. Solution Method

The process of dealing with uncertainty has two steps. The first step is to determine the rescheduling moment using the rolling-horizon method, and the second is to obtain the rescheduling scheme using the algorithm solution at the rescheduling moment, as shown in the following flowchart in Figure 3. In Section 4.1, the working principle of the rolling-horizon method is explained in detail and in Section 4.2, the detailed solution steps of the algorithm are described in detail.

![Rescheduling framework diagram.](image)

4.1. Rolling-Horizon Optimization Strategy

In this study, we use a rolling-horizon method to determine the moment of rescheduling. There are three ways to respond to uncertainty. The first is event-driven, where an uncertain event is rescheduled immediately. The second is the fixed-length rolling type, where rescheduling is performed only at the end of each fixed time window. The third type is the hybrid type, in which the impact of an uncertainty event in the time window is minor. Only this event is recorded and rescheduled at the end of the time window, and when the
impact of the uncertainty event is serious, rescheduling occurs immediately. The hybrid approach combines the advantages of both [29]. Therefore, this paper adopts a hybrid approach to determine the rescheduling moment. Execute the baseline plan (as shown in Figure 4a) under normal circumstances. After receiving the vessel change information, if there is an overlap between vessels in the baseline plan (as shown in Figure 4b), it is defined as a serious interference, and if there are no overlap (as shown in Figure 4c), it is regarded as a slight interference. For severe disturbances, the vessels are rescheduled immediately, and for slight interference, the vessels are rescheduled at the end of the current time window. This is shown in the figure below.

4.2. Rescheduling Algorithm

After determining the rescheduling moment, the rescheduling schema is solved based on the current operation status in the terminal and the latest arrival information of the affected vessels. At each moment during rescheduling, the problem can be categorized as a continuous berth and quay crane integrated scheduling problem [24]. The model of this problem takes a long time to solve by a commercial solver (GUROBI) after exceeding a certain number of vessels, so an improved genetic algorithm (IGA) is proposed to solve the problem in this paper.

4.2.1. Initial Population Generation

The chromosomal code of any individual is required for a complete representation of a feasible scheduling scheme with the following variables: \( \beta_i \), \( \eta_i \), and \( R_i \). These variables are able to determine the berthing position and handling time of the vessels by adjusting these variables to avoid the overlapping of vessels due to uncertainty events. (This process is actually the process of applying multiple strategies.) The above parameters can be coded in multiple layers and the coding form is shown below.

\[
X_n = \begin{bmatrix}
\beta_1 \beta_2 \ldots \beta_{|V|} \\
\eta_1 \eta_2 \ldots \eta_{|V|} \\
R_1 R_2 \ldots R_{|V|}
\end{bmatrix}
\] (27)

The initial population can be generated by Algorithm 1 according to the chromosome coding rules described above.
Algorithm 1 Generate initial population

Require: Known information about the vessels at the moment of rescheduling
Ensure: Initial population

1: Set population size = \( N, n = 1 \), the arrival time of the vessels in operation as the rescheduling moment
2: While \( n \leq N \)
3: Sort vessel numbers according to arrival time (random arrangement with the same time)
4: for \( i \) in range(\(|V|\))
5: Select berthing position \( \beta_i \)
6: if \( \beta_i = B_i \) and the vessel in operation and \( \delta_i = 0 \); \( \eta_i = 1 \) else \( \eta_i = 0 \)
7: Generate random integers \( R_i(q_i^{\text{min}} \leq R_i \leq q_i^{\text{max}}) \), assign \( R_i \) quay cranes for vessel
8: if any two vessels do not overlap
9: Keep this individual, \( n = n + 1 \); else \( n = n \)
10: return initial population

4.2.2. Calculate Fitness and Select Offspring Population

After obtaining each generation of the population, it is necessary to screen and retain the offspring according to the fitness value of individuals. The fitness value is the inverse of the objective function:

\[
F(X_n) = \frac{1}{\sum_{i \in V} \Delta_1_i + \Delta_2_i + \varphi_i D_i} \tag{28}
\]

In Equation (28), \( n \) is the individual serial number, which is decoded to obtain a scenario represented by an individual by calculating the objective function value according to the parameters in the objective function value of the model \((\Delta_1_i, \Delta_2_i, \varphi_i, D_i)\), taking the inverse to obtain the fitness value \( F(X_n) \), and aggregating. The steps of the screening are as follows.

Step 1: Sort the fitness values in descending order, correspond the order of individual numbers to the above sort, count the number of individuals corresponding to each fitness value separately, and calculate the probability of each category of the individuals being selected, before moving to step 2.

Step 2: Randomly generate a random number obeying uniform distribution between [0, 1], select the individuals with corresponding fitness values according to the interval corresponding to the random number, put them into the offspring population, and move to step 3.

Step 3: If the number of individuals in the offspring population reaches the population size, the selection of offspring is completed. Otherwise, go to step 2 and repeat the above operation.

4.2.3. Neighborhood Search Strategy

The problem studied in this paper needs to ensure independence among vessels, and the random search may destroy the initial feasible solution. Therefore, a search strategy is designed to eliminate some deferred vessels from the feasible solution code and reassign the berth and quay crane to these vessels as a neighborhood search process. The steps of the neighbor search are as Algorithm 2.
Algorithm 2 Neighborhood search process

Require: Paternal population, population
Ensure: Offspring population

1: Set the neighborhood search probability (NSP) to 0.3
2: for $n \leq N$
3: NSP’ = random number [0,1]
4: if NSP’ $\leq$ NSP
5: Remove a delayed vessel (unscheduled vessel), update the scheme, and put the vessel back in
6: else:
7: pass
8: if any two vessels do not overlap and $F(X_n)' \leq F(X_n)$
9: Replace the current solution with the new solution
10: return offspring population

4.2.4. Stopping Rule

After obtaining the new offspring population and completing the neighborhood search process, repeat the process described in Sections 4.2.2 and 4.2.3. Set the maximum number of iterations $M$ and stop the algorithm when it reaches $M$ generations.

5. Numerical Experiments

A set of problem instances are employed in the experiment. There are 14 QC’s along a 1200-m-long continuous berth. The planning horizon is 168 h. The following Table 1 gives information on the thirty vessels. As shown in Figure 5, the baseline plan can be derived by removing the constraints related to the moving berths and unscheduled vessels from the model in this paper.

Figure 5. Baseline plan diagram.
Table 1. Information of originally scheduled arrival vessels.

<table>
<thead>
<tr>
<th>V</th>
<th>$S^0$</th>
<th>$l$</th>
<th>$B$</th>
<th>$q$</th>
<th>$S$</th>
<th>$wz$</th>
<th>$EB^0$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>280</td>
<td>1020</td>
<td>6</td>
<td>72</td>
<td>0</td>
<td>24</td>
</tr>
<tr>
<td>2</td>
<td>1.5</td>
<td>300</td>
<td>710</td>
<td>7</td>
<td>44</td>
<td>0</td>
<td>23.5</td>
</tr>
<tr>
<td>3</td>
<td>2.75</td>
<td>180</td>
<td>100</td>
<td>4</td>
<td>48</td>
<td>0</td>
<td>26.75</td>
</tr>
<tr>
<td>4</td>
<td>3</td>
<td>140</td>
<td>280</td>
<td>3</td>
<td>44</td>
<td>0</td>
<td>25</td>
</tr>
<tr>
<td>5</td>
<td>3.5</td>
<td>180</td>
<td>450</td>
<td>4</td>
<td>48</td>
<td>0</td>
<td>27.5</td>
</tr>
<tr>
<td>6</td>
<td>28.5</td>
<td>200</td>
<td>400</td>
<td>4</td>
<td>44</td>
<td>0</td>
<td>40.5</td>
</tr>
<tr>
<td>7</td>
<td>32.5</td>
<td>260</td>
<td>140</td>
<td>6</td>
<td>36</td>
<td>0</td>
<td>50.5</td>
</tr>
<tr>
<td>8</td>
<td>28</td>
<td>280</td>
<td>1050</td>
<td>6</td>
<td>40</td>
<td>0</td>
<td>48</td>
</tr>
<tr>
<td>9</td>
<td>27.5</td>
<td>120</td>
<td>580</td>
<td>3</td>
<td>38</td>
<td>0</td>
<td>46.5</td>
</tr>
<tr>
<td>10</td>
<td>25</td>
<td>160</td>
<td>760</td>
<td>4</td>
<td>36</td>
<td>0</td>
<td>43</td>
</tr>
<tr>
<td>11</td>
<td>52</td>
<td>260</td>
<td>700</td>
<td>6</td>
<td>40</td>
<td>0</td>
<td>72</td>
</tr>
<tr>
<td>12</td>
<td>51.5</td>
<td>240</td>
<td>1040</td>
<td>6</td>
<td>52</td>
<td>0</td>
<td>77.5</td>
</tr>
<tr>
<td>13</td>
<td>54.5</td>
<td>200</td>
<td>400</td>
<td>4</td>
<td>51</td>
<td>0</td>
<td>71.5</td>
</tr>
<tr>
<td>14</td>
<td>58</td>
<td>200</td>
<td>120</td>
<td>3</td>
<td>44</td>
<td>0</td>
<td>80</td>
</tr>
<tr>
<td>15</td>
<td>85.5</td>
<td>300</td>
<td>1010</td>
<td>7</td>
<td>42</td>
<td>0</td>
<td>106.5</td>
</tr>
<tr>
<td>16</td>
<td>86</td>
<td>160</td>
<td>80</td>
<td>4</td>
<td>32</td>
<td>0</td>
<td>102</td>
</tr>
<tr>
<td>17</td>
<td>89.5</td>
<td>120</td>
<td>400</td>
<td>3</td>
<td>32</td>
<td>0</td>
<td>105.5</td>
</tr>
<tr>
<td>18</td>
<td>79</td>
<td>180</td>
<td>770</td>
<td>4</td>
<td>44</td>
<td>0</td>
<td>101</td>
</tr>
<tr>
<td>19</td>
<td>78</td>
<td>160</td>
<td>580</td>
<td>4</td>
<td>36</td>
<td>0</td>
<td>96</td>
</tr>
<tr>
<td>20</td>
<td>87</td>
<td>160</td>
<td>240</td>
<td>4</td>
<td>38</td>
<td>0</td>
<td>106</td>
</tr>
<tr>
<td>21</td>
<td>106.5</td>
<td>280</td>
<td>1020</td>
<td>6</td>
<td>68</td>
<td>0</td>
<td>140.5</td>
</tr>
<tr>
<td>22</td>
<td>102</td>
<td>300</td>
<td>710</td>
<td>7</td>
<td>44</td>
<td>0</td>
<td>124</td>
</tr>
<tr>
<td>23</td>
<td>107.75</td>
<td>180</td>
<td>100</td>
<td>4</td>
<td>52</td>
<td>0</td>
<td>133.75</td>
</tr>
<tr>
<td>24</td>
<td>110</td>
<td>140</td>
<td>280</td>
<td>3</td>
<td>54</td>
<td>0</td>
<td>137</td>
</tr>
<tr>
<td>25</td>
<td>110.5</td>
<td>180</td>
<td>450</td>
<td>4</td>
<td>56</td>
<td>0</td>
<td>138.5</td>
</tr>
<tr>
<td>26</td>
<td>148</td>
<td>200</td>
<td>400</td>
<td>4</td>
<td>36</td>
<td>0</td>
<td>166</td>
</tr>
<tr>
<td>27</td>
<td>148</td>
<td>260</td>
<td>140</td>
<td>6</td>
<td>36</td>
<td>0</td>
<td>166</td>
</tr>
<tr>
<td>28</td>
<td>146</td>
<td>280</td>
<td>1050</td>
<td>6</td>
<td>32</td>
<td>0</td>
<td>162</td>
</tr>
<tr>
<td>29</td>
<td>146</td>
<td>120</td>
<td>600</td>
<td>3</td>
<td>28</td>
<td>0</td>
<td>160</td>
</tr>
<tr>
<td>30</td>
<td>147.5</td>
<td>160</td>
<td>760</td>
<td>4</td>
<td>28</td>
<td>0</td>
<td>161.5</td>
</tr>
</tbody>
</table>

5.1. Scenario Experiment

The planning period is set to one week [33] and each fixed-length rolling period is set to 1 h. The entire experimental study is programmed to solve using Python 3.8 on an Intel(R) Core(TM) i5-9300CPU@2.40GHz PC with 16GB of RAM. Here, we show a specific scenario of rescheduling. The scenario is as follows: at 3:00, the terminal receives a 20 h delay for vessel 6 and a 10 h delay for vessel 8, and there are two unscheduled vessel arrivals. Information on unscheduled vessels is presented in the Table 2. According to the rescheduling mechanism, the rescheduling is immediate at 3:00, and the rescheduling algorithm is used to solve the problem. The algorithm parameters set the initial number of individuals in the population to 50, the number of iterations to 200, and the neighborhood search probability to 0.3. The rescheduling scheme is in Figure 6.

Table 2. Information of unscheduled vessels.

<table>
<thead>
<tr>
<th>V</th>
<th>$S^0$</th>
<th>$l$</th>
<th>$B$</th>
<th>$q$</th>
<th>$S$</th>
<th>$wz$</th>
<th>$EB^0$</th>
<th>$\delta_1$</th>
</tr>
</thead>
<tbody>
<tr>
<td>31</td>
<td>5</td>
<td>100</td>
<td>240</td>
<td>6</td>
<td>44</td>
<td>0</td>
<td>27</td>
<td>0</td>
</tr>
<tr>
<td>32</td>
<td>124</td>
<td>200</td>
<td>740</td>
<td>6</td>
<td>44</td>
<td>0</td>
<td>146</td>
<td>0</td>
</tr>
</tbody>
</table>
5.1. Scenario Experiment

The planning period is set to one week [33] and each fixed-length rolling period is set to 1 h. The entire experiment is programmed to solve using Python 3.8 on an Intel(R) Core(TM) i5-9300CPU@2.40GHz PC with 16GB of RAM. Here, we show a specific scenario of rescheduling. The scenario is as follows: at 3:00, the terminal receives a 20 h delay for vessel 6 and a 10 h delay for vessel 8, and there are two unscheduled vessel arrivals. Information on unscheduled vessels is presented in Table 2. According to the rescheduling mechanism, the rescheduling is immediate at 3:00, and the rescheduling algorithm is used to solve the problem. The algorithm parameters set the initial number of individuals in the population to 50, the number of iterations to 200, and the neighborhood search probability to 0.3. The rescheduling scheme is in Figure 6.

Table 2. Information of unscheduled vessels.

<table>
<thead>
<tr>
<th>Vessel</th>
<th>Time</th>
<th>Delay (h)</th>
<th>Handling Time (h)</th>
<th>Scheduled (Y/N)</th>
<th>Arrival Time (h)</th>
<th>Initial Position</th>
</tr>
</thead>
<tbody>
<tr>
<td>31</td>
<td>5</td>
<td>100</td>
<td>240</td>
<td>Y</td>
<td>27</td>
<td>0</td>
</tr>
<tr>
<td>32</td>
<td>124</td>
<td>740</td>
<td>840</td>
<td>Y</td>
<td>146</td>
<td>0</td>
</tr>
</tbody>
</table>

Figure 6. Rescheduling scheme diagram.

5.2. Comparison Experiments with Commercial Solvers

The above arithmetic example is divided into four different sizes with the number of vessels being 15, 20, 25, and 30, respectively. Three scenarios are randomly generated for each scale. Scenario 1, Scenario 2, and Scenario 3 contain 1, 2, and 3 delayed vessel (handling time changed vessel) and 1, 2, and 3 unscheduled vessels, respectively. Then, the IGA can be used to solve the examples five times to obtain the best result, and the comparison of the results is shown in Table 3. From the experimental results, we can find that the gap between the algorithm and the lower bound of the model is small, within 6% on average, and the IGA can solve large-scale cases in less time.

Table 3. Comparison results with commercial solver.

<table>
<thead>
<tr>
<th>V1-Scenario No.</th>
<th>Obj (min/h)</th>
<th>Gap (%)</th>
<th>CPU Time (s)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>IGA</td>
<td>Gurobi</td>
<td>IGA</td>
</tr>
<tr>
<td>15-1</td>
<td>23.50</td>
<td>23.50</td>
<td>2.21</td>
</tr>
<tr>
<td>15-2</td>
<td>35.50</td>
<td>34.75</td>
<td>2.28</td>
</tr>
<tr>
<td>15-3</td>
<td>54.50</td>
<td>50.25</td>
<td>8.46</td>
</tr>
<tr>
<td>20-1</td>
<td>23.43</td>
<td>22.00</td>
<td>13.12</td>
</tr>
<tr>
<td>20-2</td>
<td>38.60</td>
<td>35.40</td>
<td>9.04</td>
</tr>
<tr>
<td>20-3</td>
<td>58.50</td>
<td>52.50</td>
<td>13.57</td>
</tr>
<tr>
<td>25-1</td>
<td>36.80</td>
<td>34.25</td>
<td>24.71</td>
</tr>
<tr>
<td>25-2</td>
<td>41.75</td>
<td>38.60</td>
<td>34.12</td>
</tr>
<tr>
<td>25-3</td>
<td>59.50</td>
<td>-</td>
<td>34.99</td>
</tr>
<tr>
<td>30-1</td>
<td>43.40</td>
<td>-</td>
<td>66.77</td>
</tr>
<tr>
<td>30-2</td>
<td>47.25</td>
<td>-</td>
<td>97.38</td>
</tr>
<tr>
<td>30-3</td>
<td>66.75</td>
<td>-</td>
<td>111.53</td>
</tr>
</tbody>
</table>

5.3. Comparison Experiments with Other Algorithms (GA, PSO)

Using the same example and initial solution, the IGA, SA, and PSO are used to solve the problem and are compared with the convergence time. The results are presented in the following Table 4. Compared with SA and PSO, the IGA has a faster and more advantageous solution.
Table 4. Comparison results with SA and PSO.

<table>
<thead>
<tr>
<th>V1-Scenario No.</th>
<th>Obj (min/h)</th>
<th>Gap (%)</th>
<th>CUP Time (s)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>IGA</td>
<td>SA</td>
<td>PSO</td>
</tr>
<tr>
<td>15-1</td>
<td>23.50</td>
<td>23.50</td>
<td>23.50</td>
</tr>
<tr>
<td>15-2</td>
<td>35.50</td>
<td>36.00</td>
<td>35.50</td>
</tr>
<tr>
<td>15-3</td>
<td>54.50</td>
<td>58.00</td>
<td>56.00</td>
</tr>
<tr>
<td>20-1</td>
<td>23.43</td>
<td>25.25</td>
<td>24.50</td>
</tr>
<tr>
<td>20-2</td>
<td>38.60</td>
<td>41.50</td>
<td>40.25</td>
</tr>
<tr>
<td>20-3</td>
<td>58.50</td>
<td>63.75</td>
<td>61.50</td>
</tr>
<tr>
<td>25-1</td>
<td>36.80</td>
<td>40.60</td>
<td>39.75</td>
</tr>
<tr>
<td>25-2</td>
<td>41.75</td>
<td>45.30</td>
<td>42.40</td>
</tr>
<tr>
<td>25-3</td>
<td>59.50</td>
<td>63.60</td>
<td>62.75</td>
</tr>
<tr>
<td>30-1</td>
<td>43.40</td>
<td>47.75</td>
<td>46.65</td>
</tr>
<tr>
<td>30-2</td>
<td>47.25</td>
<td>50.50</td>
<td>50.00</td>
</tr>
<tr>
<td>30-3</td>
<td>66.75</td>
<td>72.25</td>
<td>70.75</td>
</tr>
</tbody>
</table>

5.4. Sensitivity Analysis

Through the experimental process and in-depth analyses of the calculation results, we find that the delay time, the increased handling time, and the number of unscheduled vessels all have an impact on the results and the scheme. Three sets of experiments were designed for this purpose, using the arithmetic example with a total number of vessels of 30 and varying each of these three factors. The three experimental results are presented in Figure 7.

![Figure 7. Sensitivity experiment results. (a) The sensitivity of delay time; (b) The sensitivity of increased handling time; (c) The sensitivity of unscheduled vessel number.](image)

From Figure 7a, we can find that the objective function value increases uniformly with the increase in delay time in the same scenario, but the increase in delay time causes some vessels to deviate from their best berth position, which increases the objective function value.

From Figure 7b, we can see that the short increase in handling time does not affect the original plan, but beyond a certain threshold, it causes changes in the vessel schedule.

From Figure 7c, it can be seen that the unscheduled vessels have a serious impact on the original scheme. When the number of unscheduled vessels exceeds a certain number, the operation of all vessels cannot be handled in the planning period.

6. Conclusions

This study investigates the optimization problem of integrated rescheduling of berth and quay crane considering the effects of uncertain events. We designed a rescheduling...
mechanism to determine the rescheduling moment, proposed multiple rescheduling strategies, formulated a rescheduling model, and contrived an improved genetic algorithm to solve it. The integrated rescheduling optimization of the berth and quay crane can be performed at the right rescheduling time to give a new rescheduling solution. The results show that the uncertainty should be paid attention to, and different strategies are given for different delayed vessel scenarios to ensure minimum variation in the vessel schedule during the planning period and a minimum handling time of unscheduled vessels for rescheduling.

In the numerical experiments, we gave a baseline plan and then set up a scenario to demonstrate the process of response and rescheduling. Several sets of numerical experiments were set up to demonstrate that the solution speed of this algorithm is better than that of commercial solvers and that the average error does not exceed 6% in different sizes of cases. Finally, compared with other heuristic algorithms (GA, PSO), the algorithm presented in this paper has the fastest solution and the advantage of the scheme.

Future research can be based on this study to include tidal factors [34] and uncertainties, such as quay crane failure.

**Author Contributions:** Conceptualization, H.Z. and Z.W.; methodology, Z.W.; software, Z.W.; validation, H.L.; writing—original draft preparation, Z.W. and H.L.; writing—review and editing, Z.W. and H.L.; supervision, H.Z.; project administration, H.Z.; funding acquisition, H.Z. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research is financially supported by the National Natural Science Foundation of China (No. 71872025).

**Data Availability Statement:** All data generated or analyzed during this study are included in this published article.

**Conflicts of Interest:** The authors declare no conflict of internet.

**References**


19. Xiang, X.; Liu, C. An expanded robust optimization approach for the berth allocation problem considering uncertain operation time. *Omega* **2021**, *103*, 102444. [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.