ISSN: 2664-9640

Robust optimization method for integrated berth and quay crane scheduling in automated terminals

Enjie Wu, Jin Zhu*

l Institute of Logistics Science and Engineering, Shanghai Maritime University, Shanghai 201306, China

*1594597346@qq.com

Abstract

Automated terminals play a vital role as a representative logistics facility of contemporary trade, due to land scarcity, labor cost and limited high-tech equipment, the increase in the number of containers and vessels brings new challenges to port management and resource scheduling. In this study, the integrated berthing and quay-crane scheduling problem is studied from the tactical level. Considering the uncertainties of vessel arrival time and operation time, a data-driven robust optimization model is established based on historical data to minimize the total cost of the deviation between the planned berthing time and the expected berthing time. In order to solve the model, we first use K-means clustering to construct the uncertainty set and then propose an adaptive large neighborhood search algorithm to solve the model. A large number of numerical experiments verify the validity of the model and algorithm.

Keywords

Automated terminal, integrated berth and quay crane scheduling, robust optimization model, K-means clus-tering, adaptive large neighborhood search.

1. Introduction

With the increasing throughput of automated container terminals worldwide, management of the port is facing new challenges. Therefore, it is very important to seek higher terminal operation efficiency through some effective dispatching strategies. In addition, how to use the limited resources of the port to improve efficiency and productivity, has also received more and more scholars' attention. Scholars pay attention to berth and quayside bridge integrated scheduling problems, yard distribution problems, empty container relocation problems, container premarshalling problems and so on. In these studies, berth and quayside bridge integrated scheduling (BACAP) plays an important role in container terminal management. Among wharf resources, berth quayside bridge is closely related to technical equipment storage yard. In general, the goal of the BACAP is to determine the berthing location and time of each ship arriving at the dock. The BACAP can be divided into three variants at the plan level: strategic and tactical and operational.

In the past few decades, most scholars have focused on the integrated berthing and quayside Bridge scheduling problem (BACAP) at the operational level. That is, only the shortest time range is considered, including decisions from one day to several days. Liang et al. considered the berthing and quayside bridge scheduling problem of ship-to-ship transfer, and established a mixed integer programming model aiming at minimizing the sum of delay time and waiting time of ship in port [1]. Wang et al. studied BACAP on the operational level, taking into account the interference of QC and the increased QC capacity requirement due to deviation from the ideal anchorage position [2]. At the strategic level, it looks at time horizons ranging from one year to several years, including decisions on establishing contractual arrangements for shared and dedicated berths and co-operation between terminals and shipping lines. Imai et al. proposed a strategic berth model problem, that is, to select the ships that need service from the ships requesting service and arrange their berth Windows within a limited planning scope [3]. Tactical decisions include tactical berthing and yard allocation quaybridge operations, ranging from one week to several months. Moorthy et al. first mentioned the concept of BACAP at the tactical level and proposed a new concept of integrated scheduling for tactical discrete berth allocation and quay bridge scheduling [4]. Vacca et al. proposed an accurate branch pricing algorithm for tactical BACAP [5]. Xie et al. subsequently studied a similar problem and proposed a branch pricing algorithm based on Dantzig Wolfe decomposition framework to obtain the optimal solution [6]. Zhen et al. combined berth allocation and yard planning at the tactical level [7]. Meisel et al. proposed an integrated framework to coordinate all decisions on berth allocation quayside bridge allocation and quayside bridge scheduling in an integrated manner [8]. Lalla-ruiz et al. established a mixed integer programming model for BACAP at the tactical level, aiming at minimizing the management cost of inter-ship transshipment containers [9]. Yang et al. continued to build a dynamic berthing and quayside bridge cooperative scheduling model based on discrete berth layout at the tactical level, aiming at minimizing the total service cost of container ships at the port, and designed a genetic algorithm to solve the model [10]. Jiao et al. studied BACAP in the case of berth dredging in view of the actual characteristics of container terminal berths requiring regular maintenance [11].

Most of these articles have studied BACAP under deterministic conditions. However, the operation of container terminal is faced with complicated environment such as typhoon, earthquake and rainstorm. Therefore, in recent years, some scholars began to study BACAP under uncertainty. From the operational perspective,Zhen et al. developed a dual-objective mixed integer programming model to find the tradeoff between system cost and robustness under the consideration of uncertain arrival time and running time [12]. Rodriguez-molins et al. proposed a robust model and an active strategy that proportioned buffer time between tasks without prior knowledge of uncertainty [13]. At the tactical level,Iris et al. integrated active and passive strategies to resolve BACAP under uncertainty, and accordingly proposed an active baseline plan with passive recovery costs to achieve resilience [14]. Xiang et al. introduced the concept of robustness indicator for tactical BACAP and established a model with strong robustness aiming at minimizing total cost [15]. These scholars only studied tactical BACAP under one uncertainty factor. In this study, we focused on the integrated scheduling problem of tactical berth and quayside bridge under two uncertainties. In tactical BACAP, port planners must decide, in consultation with shipping lines, when and where ships will berth. The liner first informs the port of their expected berthing time, and the port makes plans to meet their requirements as best as possible. In addition, the allocated berthing position also affects the ship's operation time. In order to shorten the distance between allocated berths and storage locations, ships tend to berth at preferred berths, thus minimizing the operation time.

Figure 1 shows an example of tactical BACAP for tactical BACAP, ships arrive periodically. As stated in [4,7], most ships have a one-week arrival mode. For such ships, the liner tends to stay as close as possible to the expected berthing time in order to reduce energy consumption. Therefore, port planners must make an integrated scheduling plan for tactical berths and quayside crane to minimize the deviation between expected berthing time and planned berthing time. In this study, our goal was to develop a tactical plan for ships arriving in mode each week. According to our strategy, port planners can adjust this tactical plan at an operational level to achieve weekly operational plans with accurate information.

Prior data is widely used in container operation management. Most studies assume that previous information, namely the actual time of arrival and the number of containers, is accurate and reliable. In fact, more than 40% of ships on the international container ship route are at least one day behind their expected arrival time [16], most ships don't make it to port on

ISSN: 2664-9640

time. In addition, a ship's operating time depends on the number of containers to be handled. Ships visit ports regularly and their total number of containers fluctuates over time due to unpredictable factors. And the unproductive recombination caused by container stacking position also has an impact on operation time. With the development of information technology, port planners can collect historical data and make decisions based on objective data. Therefore, we study tactical BACAP given uncertain arrival time and operation time based on a data-driven approach.

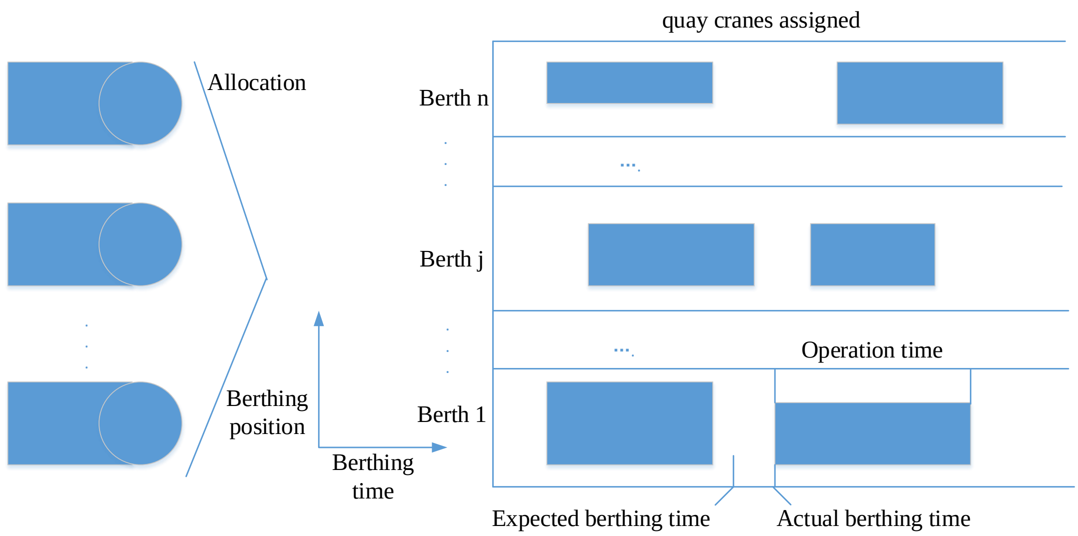

Figure 1: An illustration of the tactical BACAP

In recent years, data-driven methods have been well studied. The data-driven approach used in this study was proposed by Bertsimas et al. [17], who provided a data-driven robust optimization (RO) approach. The main idea is to design the uncertainty set of RO with limited data. In this framework, it is assumed that planners only have historical data information about the number of containers. In the previous literature,Bertsimas assumed that uncertainty sets were defined by specific structures and sizes in terms of available data points. Although the structure of the uncertainty set is not predefined, we consider the uncertainty set constructed from historical data. Then, we develop a BACAP data-driven RO framework for a random number of ships. In order to obtain the uncertainty set based on the collected historical data, and how to effectively solve the RO problem given the uncertainty set, we first propose a data-driven uncertainty set construction method based on K-means clustering, and an adaptive large neighborhood search algorithm (ALNS) is proposed to solve the model.

2. Problem Description

This article examines a tactical problem, and we use Figure 2 to illustrate the planning process for creating a tactical BACAP. In the first step, we don't know the exact information for the next few weeks. So we use the historical data we collect to make tactical plans. In this tactical plan, there is the same schedule every week. Based on this scheme, the terminal can make some resource preparations. Over time, we can get almost accurate information. Using this information, we adjust our original tactical plans and develop plans. In this paper, we focus only on how to generate tactical plans based on historical data. The questions considered in this study have two important characteristics: random fluctuations in the number of ships and containers arriving each week. Next, we'll cover these two features. We begin by describing the characteristics of the weekly ship arrival pattern, as shown in Figure 2. Ships arrive at the dock once a week, leading to periodic changes in BACAP. In this article, our goal is to develop a tactical plan for BACAP for the coming weeks, leaving aside the problem of developing an operational plan for each week. For example, we consider a group of ships $|V|$ serving within a cycle $|H|$, in which time is discrete into a cycle (24h), which means that we divide each cycle into 4 time periods. [8,15] and other literatures have adopted similar Settings. The berth is limited in the discrete case we need to determine the time and location of the ship berthing, as well as the number of quayside Bridges allocated to each ship for each time period. The objective is to minimize the total cost of deviation from the expected berthing position and time.

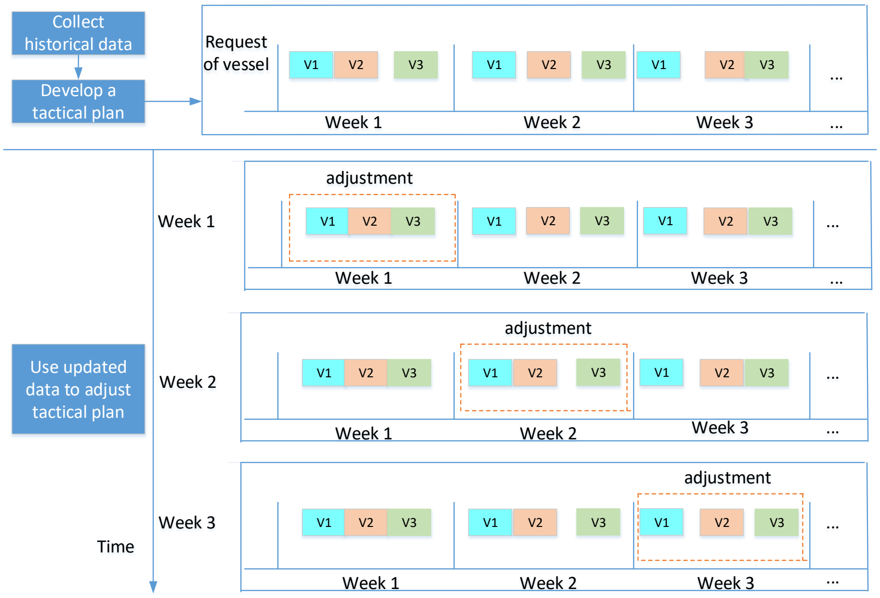

Figure 1: Tactical BACAP applies to ships that arrive weekly and the number of containers fluctuates randomly

Another feature is the random fluctuation of the number of ships. BACAP creates uncertainty because of the unpredictable number of ships. As shown in Figure 2, the ship informs the port of the exact number of containers prior to the tactical plan. However, because ships visit the port regularly, the total number of containers fluctuates from period to period. Although information technology is highly developed, it is difficult to accurately predict fluctuations, and random fluctuations of container quantity will lead to uncertainty of operation time [15]. Therefore, assume that the operation time is random. In addition, the known ship arrival time deviation is also one of the main uncertainties in ship scheduling. ξ_1 and ξ_2 are defined to represent the deviation of the ship's arrival time relative to the expected arrival time and the extra operation time required, respectively. The random parameter $\xi_i = \xi_1 + \xi_2$ represents the sum of the deviation of the ship's arrival time and operation time. Anyway, instead of using the deterministic operation time vector t, we use the $t+\xi$ notation for random operation time.

In this study, we use time buffering to characterize the robustness of scheduling. The time buffer of ship i is represented by η_i, which is defined as the start time of subsequent ships minus the end time of ship i. If $(\xi_i - \eta_i) > 0$, there will be a conflict between ship i and its successor. To resolve this conflict, terminals must increase productivity, resulting in penalty costs.

3. Model Formulation

The symbols used in this article are shown below:

Main notations:

V set of ships, indexed by i, j

ISSN: 2664-9640

G total number of ships

M set of berths, indexed by m

B total number of berths

H set of time segments, indexed by t

W set of scenes

T time planning horizon

EA_i estimated arrival time of ship i

EC_i estimated number of containers required to be handled of ship i

ED_i expected departure time of ship i

t_{im} the operation time when ship i berthed at berth

L_m the length of berth m

B_i the expected berthing position of the ship i, that is, the most advantageous position to reduce the horizontal transport distance of container handling

C_{im}^1 estimated unit transport cost of a ship between berth B_i and berth m

C_i^2 unit penalty cost of ship i leaving later than ED_i

C_i^3 dock increases productivity, resulting in penalty costs

ξ_{is} suppose there are S scenarios, and ξ_{is} is represented as different scenarios ξ_i, where $s \in W$, $s = 1, \ldots, S$.

MI_i the minimum number of quay Bridges that can be assigned to ship i

MA_i the maximum number of quayside Bridges that can be assigned to ship i

AQ the total number of quayside Bridges available in each time period

AP the average number of ships that a quay can serve over a period of time

u_{is}^+, u_{is}^- two non-negative auxiliary variables

M a big enough positive value

Decision variables:

α_i start time of berthing of ship i

β_i the departure time of ship i

θ_{im} 1 when ship i is assigned to berth m, 0 otherwise

ζ_{it} the number of quay crane allocated to ship i during time period t

ε_{it} 1 if $\zeta_{it} > 0$, 0 otherwise

φ_{ijm} when ship i and j are berthed at the same berth, the departure time β_i of ship i is earlier than the berthing time α_j of ship j, that is, 1 when $\theta_{im} = \theta_{jm} = 1$; 0 otherwise

δ_{imt} 1 when berth m is allocated to ship i in time period t, that is, 1 when $\theta_{im} = \varepsilon_{it} = 1$, 0 otherwise

Objective functions and constraints:

$$\min \sum_{i \in V} \left[\sum_{m \in M} C_{im}^1 \theta_{im} + C_i^2 (\beta_i - ED_i)^+ \right] + E\left[\sum_{i \in V} C_i^3 (\xi_{is} - \eta_i)^+ \right] \tag{1}$$

$$\sum_{m \in M} \theta_{im} = 1, \quad \forall i \in V \tag{2}$$

$$\sum_{i\in V}\zeta_{it}\leqslant AQ,\quad \forall t\in H \tag{3}$$

$$\sum_{t\in H}AP\cdot\zeta_{it}\geqslant EC_i,\quad \forall i\in V \tag{4}$$

$$\zeta_{it}-MI_i\cdot\varepsilon_{it}\geqslant 0,\quad \forall i\in V,t\in H \tag{5}$$

$$\zeta_{it}-MA_i\cdot\varepsilon_{it}\leqslant 0,\quad \forall i\in V,t\in H \tag{6}$$

$$\beta_i-(t+1)\cdot\varepsilon_{it}\geqslant 0,\quad \forall i\in V,t\in H \tag{7}$$

$$\alpha_i-t\cdot\varepsilon_{it}-M(1-\varepsilon_{it})\leqslant 0,\quad \forall i\in V,t\in H \tag{8}$$

$$\sum_{t\in H}\varepsilon_{it}=\beta_i-\alpha_i+1,\quad \forall i\in V \tag{9}$$

$$EA_i\leqslant\alpha_i,\quad \forall i\in V \tag{10}$$

$$\beta_i-\alpha_j-M(1-\varphi_{ijm})\leqslant 0,\forall i,j\in V,i\neq j \tag{11}$$

$$\varphi_{ijm}+\varphi_{jim}-\theta_{im}-\theta_{jm}+1\geqslant 0,\forall i,j\in V,i\neq j \tag{12}$$

$$\delta_{imt}-\theta_{im}-\varepsilon_{it}+1\geqslant 0,\forall i\in V,m\in M,t\in H \tag{13}$$

$$\sum_{i\in V}\delta_{imt}\leqslant 1,\ \forall m\in M,t\in H \tag{14}$$

$$\sum_{i\in V}t_{im}\theta_{im}\leqslant T,\ \forall m\in M \tag{15}$$

$$\xi_{is}-\eta_i=u_{is}^{+}-u_{is}^{-},\ \forall i\in V,s\in W \tag{16}$$

$$\sum_{i\in V}\sum_{m\in M}t_{im}\theta_{im}+\sum_{i\in V}\eta_i=TB \tag{17}$$

$$\alpha_i,\beta_i,\zeta_{it},\eta_i\geqslant 0,\ \forall i\in V,t\in H \tag{18}$$

$$u_{is}^{+},u_{is}^{-}\geqslant 0,\ \forall i\in V,s\in W \tag{19}$$

$$\theta_{im},\varepsilon_{it},\varphi_{ij},\delta_{imt}\in\{0,1\},\forall i,j\in V,i\neq j,m\in M \tag{20}$$

Objective (1) is to minimize the total cost, including the penalty cost caused by berthing position and the delay cost of departure time, as well as the penalty cost of conflict resolution. Constraints (2) ensure that each vessel is allocated a berth. Constraint (3) indicates that the total number of quayside Bridges used at a time cannot exceed the number of available quayside Bridges. Constraints (4) ensure that all tasks must be completed. Constraint (5)(6) ensures that when $\varepsilon_{it}=1$, ζ_{it} must be in the range of $[MI_i,MA_i]$, $\zeta_{it}=0$ otherwise. Constraints (7) and (8) relate ε_{it} to the start berthing time α_i and departure time β_i. Constraint (9) ensure that the service operation of the ship is not preempted. Constraint (10) ensure that the ship cannot berth until arrival. Constraints (11)(12) state that if two ships are berthed in the same berth, they cannot overlap. Constraint (13) relates δ_{imt} to θ_{im} and ε_{it}. Constraint (14) means that each berth can be allocated to no more than one ship at any time period. Constraint (15) means that the total operational tasks of ships assigned to each berth must be completed within the planning period. Constraint (16) introduces two nonnegative variables u_{is}^{+} and u_{is}^{-} to linearize $\xi_{is}-\eta_i$. Constraint (17) ensures that the sum of all operations and relaxation times is equal to TB.

Then, we reformulate the objective function using the sample average approximation (SAA) scheme:

$$\min\ \sum_{i\in V}\left[\sum_{m\in M}C_{im}^{1}\theta_{im}+C_i^{2}(\beta_i-ED_i)^{+}\right]+\frac{1}{S}\sum_{s\in W}\sum_{i\in V}C_i^{3}u_{is}^{+} \tag{21}$$

We turn model (21) into a general RO model (TRO). RO model assumes an uncertain set to describe an uncertain factor, and the solution obtained by this method is robust to any interruption within the uncertain set. Assuming that the uncertain set $\Omega=\{\xi^1,\xi^2,\dots,\xi^S\}$ is known, we can obtain the following model:

$$\min\sum_{i\in V}\left[\sum_{m\in M}C_{im}^1\theta_{im}+C_i^2(\beta_i-ED_i)^+\right]+\max_{s\in\Omega}\sum_{i\in V}C_i^3u_{is}^+ \tag{22}$$

RO models have two key characteristics, that is, describing the uncertainty with the uncertainty set and finding solutions in the uncertainty set with the lowest recovery cost in the worst case. Capturing randomness with just one set of uncertainties may not achieve the desired result. A small set of uncertainties is difficult to accurately describe randomness. However, a large set of uncertainties can guarantee reliable randomness, including any possible scenarios, but this tends to yield too conservative solutions. In this study, we divide the uncertain scenarios into multiple uncertain sets, assign a weight to the worst-case performance of each uncertain set, and calculate the weighted sum of the worst-case performance of these uncertain sets. Using multiple uncertainty sets can effectively avoid the solution being too conservative. We introduce the process of estimating the number of ships and time of arrival from historical data and describe it by constructing an uncertainty set, namely k-means clustering, using a data-driven approach. Since we studied the problem of weekly shipping arrival patterns, we could collect the number of containers and time of arrival that fluctuated weekly. Let's assume that the historical data we collect contains S sample paths. Each example path s contains implemented requirement ξ_{is}. In order to achieve k-means clustering, we divide $\{\xi^1,\dots,\xi^S\}$ set into K non-overlapping full-dimensional clusters. The k-means clustering algorithm outputs the center of mass, which we express as μ_k, $\mu_k=\{\mu_{k1},\mu_{k2},\dots,\mu_{kG}\}$, for $k\in\Upsilon=\{1,\dots,K\}$. The formal description of the uncertainty set is:

$$\Omega_k=\left\{\xi\in\left[\underline{\xi},\bar{\xi}\right]|(\xi-\mu_k)'(\xi-\mu_k)\leqslant(\xi-\mu_l)'(\xi-\mu_l),\forall l\in\Upsilon\right\} \tag{23}$$

It can be simplified as:

$$\Omega_k=\left\{\xi\in\left[\underline{\xi},\bar{\xi}\right]|2\xi'(\mu_l-\mu_k)+\mu_k'\mu_k-\mu_l'\mu_l\leqslant 0,\forall l\in\Upsilon\right\} \tag{24}$$

Where $\underline{\xi}$ and $\bar{\xi}$ are upper and lower bounds of ξ respectively.

Now the weights can be determined:

$$w_k=\sum\nolimits_{s=1}^{S}\mathbb{I}(\xi^s\in\Omega_k)/S \tag{25}$$

Where $\mathbb{I}(.)$ is the index function.

This set of uncertainties allows us to model various structural information about ξ, which depends on the number of clusters. According to the constructed uncertainty set, the DATA driven RO model (KRO) can be obtained.

$$\min\sum_{i\in V}\left[\sum_{m\in M}C_{im}^1\theta_{im}+C_i^2(\beta_i-ED_i)^+\right]+\sum_{k\in\Upsilon}w_k\left(\max_{s\in\Omega}\sum_{i\in V}C_i^3u_{is}^+\right) \tag{26}$$

4. Solution Algorithm

We decomposed the problem into two stages and solved it hierarchically. Algorithm 1 gives the framework of the solution the framework requires a baseline plan and parameters for each scenario. In the first phase of the framework (lines 1-2), obtain A solution for each scenario problem A and collect the cost of the scenario solution. In this stage, a heuristic algorithm based on adaptive Large Neighborhood search (ALNS) is adopted. In order to generate a recoverable baseline schedule, each scenario problem requires a solution. In phase 2 (line 3), baseline plan

X_o is generated based on the scenario solution and baseline parameters. The entire framework returns the best baseline timeline and scenario solutions to the problem.

Algorithm 1: Heuristic algorithm framework
Input : baseline parameters, scene options ξ
for $\xi = 1 \dots S$ do
$X_\xi \leftarrow SceneALNS(\xi)$
$X_o \leftarrow BaselineALNS(X_\xi)$
return X_o, X_ξ

ALNS was proposed by Ropke et al. [18]. In ALNS, an initial candidate solution is first obtained. In each iteration, the destruction operator destroys part of the current solution, and then the repair operator reconstructs the remaining part of the solution An important feature of ALNS is that the usage probability of each operator is dynamically adjusted by iteration according to the performance of operators. The overall framework of scenario and baseline ALNS is given in Algorithm 2.

Algorithm 2: Scenario ALNS & baseline ALNS
Input: Initial solution X_i , each operator initial operator weight w_o of o, set of destruction operator N^- , set of repair operators N^+
$X_b \leftarrow X_i$
$X_c \leftarrow X_i$
while do not meet termination criteria do
determine the number of vessels to be removed/inserted ϕ
$X_p \leftarrow DO(N^-, \phi, X_c)$
$X_n \leftarrow RO(N^+, \phi, X_p)$
if $Z(X_n) < Z(X_b)$ then
$X_c \leftarrow X_n$, $X_b \leftarrow X_n$
else
if X_n meet acceptance criteria then
$X_c \leftarrow X_n$
$AOW(N^-, N^+, w_o)$
return X_b

Algorithm 2 starts with the initial solution, which is set to the best solution and the current solution. The initial solution is generated in a greedy manner using a basic greedy insertion method. In each iteration, the number of ships ϕ to be removed from the current solution X_c is randomly determined between $\bar{\phi}/2$ and $\bar{\phi}$, $\bar{\phi}$ is the upper limit of the number of ships that need to be removed in the current solution X_c. The exact method of damage or repair is determined by the roulette wheel mechanism. Function $DO(N^-,\phi,X_c)$ first selects the operator and then executes it. For the destruction operator, the current solution is the input, while the new partial solution is the input for the repair operator. After performing the corresponding repair operations, you get a new solution X_n. The weight of the operator determines the choice of the operator, and the weight is adaptive as the search goes on. The better the performance of the operator, the greater the weight. After each iteration, the performance (that is, weight) of each operator is updated with $AOW(N^-,N^+,w_o)$. The update mechanism uses an equal partition function. If the target value $Z(X_n)$ of the new solution is better than the best target value $Z(X_b)$, The best solution X_b and the current solution X_c are updated to X_n. If the target value of X_n is not better than the best target value, the acceptance criteria based on simulated annealing are checked. If the criteria meet the fix, update the current solution X_c to X_n. These iterations will continue until a fixed number of iterative algorithms 2 are performed to return the optimal solution after the termination condition is satisfied.

5. Solution Algorithm

We generate problem instances in the following way, with a planned cycle of four hours per week. Therefore, $|H|=42$. Three different scales and parameter Settings are shown in Table 1. The expected time of arrival EA_i and the expected berthing position B_i are randomly generated from the uniform distribution of $U[0,36]$ and $U[1,|M|]$. Ships are divided into three classes: small, medium and large, as shown in Table 2. In previous literatures, the actual arrival time and random operation time of ships are usually evenly distributed. Therefore, we assume that the random parameter ξ obeys a uniform distribution $[\zeta_i^l.\zeta_i^u]$ and that they are generated independently of each other.

Table 1: Scale of Instance Groups

Group	G	B	$\|H\|$	AQ
S1	15	2	42	5
S2	30	4	42	11
S3	50	7	42	18

Table 2: Vessels' technical specifications

Class	Proportion	MI_i	MA_i	EC_i
Small	1/3	1	3	[280,700]
Medium	1/3	2	4	[700,1960]
Large	1/3	3	6	[1960,2800]

5.1. Algorithm Efficiency

In order to verify the effectiveness of ALNS algorithm, we compared it with CPLEX Adaptive Genetic Algorithm (AGA)[19]. In this experiment, we set instance combinations at three scales, and in order to study the influence of different number of scenarios on algorithm performance, we set Z to 5,15,30,60,120,240. For each combination, 5 instances will be generated. Table 3 shows the CPU time, target value and corresponding Gap value of each algorithm. It can be seen from Table 3 that ALNS algorithm is very effective in solving the model. The average Gap of

ISSN: 2664-9640

ALNS is 1.95% 3.04% and 5.04%, respectively, and the average target value is 4.98% optimized. With the increase of the example size, the average Gap difference between ALNS and AGA increases gradually. It is shown that the quality of ALNS solution increases with the increase of instance size. In a single instance, the Gap results of ALNS are better than AGA in most cases. In addition, ALNS is less time-consuming than CPLEX. For 180 instances,ALNS can get the optimal solution of the instance in 2 hours, with a maximum CPU time of only 1.74h. The average CPU time of CPLEX is greater than that of ALNS. Moreover, the average CPU time of CPLEX increases dramatically as the number of scenarios increases. In contrast, the number of scenarios only has a small effect on the average CPU time of ALNS.

Figure 3 shows the average Gap(a) and average calculation time (b) of different iterations. The results are summarized based on each instance size. After about 440,000 iterations, the average Gap convergence effect of all instance sizes is good. With the increase of iterations, the average calculation time increases linearly. In large, medium and small cases, the best average Gap is about 4.70%, 2.68% and 1.58%. Figure 3 shows that it is best to choose 260,000 iterations as the termination criterion, because Gap improvement is minimal after this value and calculation time is reasonable, which is applicable in practice.

Table 3: Comparisons between ALNS, AGA and CPLEX

Ins		ALNS			AGA			CPLEX	
	S	T1(s)	Obj1	Gap1(%)	T2(s)	Obj2	Gap2(%)	T3(s)	Obj3
	5	438	243.1	0.00	365	243.1	0.00	735	243.1
	15	465	245.5	0.32	402	245.3	0.26	1275	244.7
L1	30	511	249.3	1.38	428	249.8	1.57	3250	245.9
	60	538	251.0	2.35	462	254.5	3.78	7373	245.2
	120	557	253.2	3.28	487	257.0	4.81	11389	245.2
	240	584	254.7	4.39	516	257.6	5.57	16965	244.0
	5	2872	1253.2	0.73	2409	1260.5	1.32	4297	1244.1
	15	3013	1261.7	1.31	2519	1276.7	2.51	7655	1245.4
L2	30	3396	1282.0	2.59	2957	1306.8	4.58	11691	1249.6
	60	3804	1294.2	3.54	3328	1329.1	6.33	16230	1250.0
	120	4175	1297.5	4.61	3737	1337.5	7.84	21600	1240.3
	240	4468	1301.4	5.45	4011	1359.4	10.15	21600	1234.1
	5	4718	1593.0	2.04	4452	1625.5	4.12	6225	1561.2
	15	5149	1611.3	3.19	4861	1676.1	7.34	10948	1561.5
L3	30	5447	1624.2	4.36	5128	1721.0	10.58	16341	1556.3
	60	5741	1632.7	5.51	5446	1774.1	14.65	21600	1547.4
	120	6105	1648.0	6.80	5791	1847.7	19.74	21600	1543.1
	240	6273	1657.5	8.34	5948	1925.2	25.84	21600	1529.9

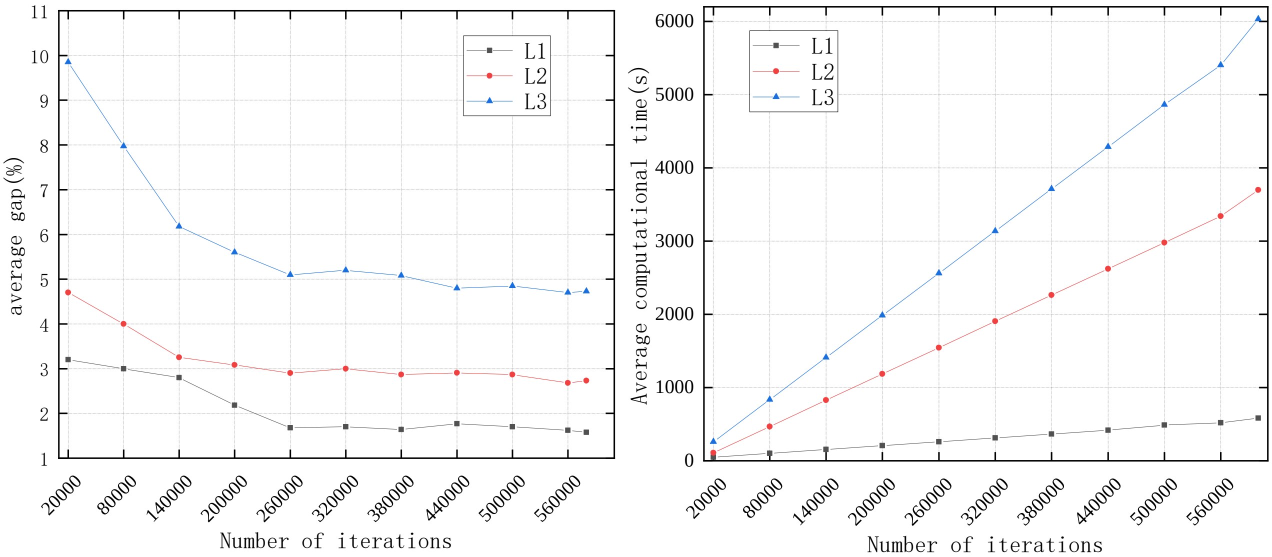

Figure 3: Analysis of number of iterations.

5.2. Model Performance

In this section, we will verify the validity of the proposed model by comparing it with other mathematical models and actual policies. For each model and policy, we perform the following procedures in each case.

Step 1: Generate 60 scenarios to obtain baseline scheduling, the baseline cost is denoted by Z^1.

Step 2: Generate 1000 new scenarios and use them to simulate scheduling, record the earliest arrival time EA^s of all ships in scenario s.

Step 3: To simulate all scenarios, we adopt the right shift strategy [] to adjust the interrupt. According to the actual scheduling table, we can get the actual berthing start α_i^{so} and end time β_i^{so}. We record the latest departure time LD^s of all ships. We also recorded the number of quay cranes TQ_t^s and berths TB_t^s broken at each time in scene s. Recovery cost in scenario s is defined RC_s.

Step 4: Calculate the four indicators used to evaluate the above method.

•Total expected cost Z, including baseline cost and recovery cost for dealing with uncertainties: $Z = Z^1 + \frac{1}{1000}\sum_{s=}^{1000} RC_s$.

• Total vessel delays VD compared with the original schedule: $VD = \frac{1}{1000}\sum_{s=1}^{1000}\sum_{i\in V}(\beta_i^{so} - \beta_i^s)^+$.

• Utilization rate UB of berth: $UB = \frac{1}{1000}\sum_{s=1}^{1000}\frac{\sum_{t=EA^s}^{LD^s} TB_t^s}{(LD^s - EA^s + 1)\cdot B}$.

• Utilization rate of quay crane: $UQ = \frac{1}{1000}\sum_{s=1}^{1000}\frac{\sum_{t=EA^s}^{LD^s} TQ_t^s}{(LD^s - EA^s + 1)\cdot AQ}$

Table 4: Comparisons between KRO model with deterministic model and TRO model

model	L1				L2				L3			
	Z	VD	UB (%)	UQ (%)	Z	VD	UB (%)	UQ (%)	Z	VD	UB (%)	UQ (%)

KRO	265	4	76	93	1392	17	74	85	1796	22	70	88
Determini	490	36	73	87	2486	173	72	80	2924	204	68	83
TRO	354	2	72	86	1835	6	71	78	2346	10	67	80

From Table 4, we can see that KRO performs best in terms of total expected cost. In L1,KRO can save 45.9% and 25.1% of expected total costs compared to the other two methods. In terms of ship delay index, VD of KRO method is larger than THAT of TRO model, but their values are smaller than that of deterministic model. From the perspective of resource utilization, the KRO model performs best. However, the deterministic model does not consider the uncertainty in the design of dispatching, which leads to the delay of many ships and a large latest departure time. In addition, the berth utilization rate of KRO model in the three groups of examples is 75.8%, 73.9% and 69.7% respectively, which are higher than the other two models. The reason is that the ship stays in port for a long time due to the limited quay resources In general, our results show that THE KRO method performs best in terms of total expected cost and resource utilization. In addition, the model can solve the ship delay problem well and has good practical application value.

6. Conclusion

This study discusses the berth shore bridge integrated scheduling problem, to assign to the terminals of each time period regularly berthing time position and number of shore bridge a ship regular access to a port, the container number and the time of arrival in every period have volatility, this demand fluctuations are responsible for a lot of uncertainty And the ship's operation time depends on the container number is going to deal with Therefore, we consider the uncertainty of vessel arrival time and operation time We first established a uncertainty model, including the expected arrival time and operation time Then, by introducing uncertainty set to describe uncertainty, the expanded into a robust model, including all the implementation must be for all constraints is feasible In order to obtain the conservative solution, we first divide the uncertain set into K non-overlapping full-dimensional clusters by k-means clustering. Then, we build the BACAP KRO model by means of the mean goal and weighted maximum penalty function. In order to solve this problem, an adaptive large neighborhood search algorithm is proposed Experimental results show that the algorithm is convergent under limited iterations, and the model can significantly increase the expected cost within an acceptable computing time, and provide the utilization rate of berth and quayside resources. In addition, our model is more suitable for solving the problems with high uncertainty.

In the future, we will consider other types of uncertainty, such as equipment failures, unscheduled ship arrivals and other unforeseen events. In addition, there are many clustering methods that divide uncertain sets into multiple clusters. In future studies, the application of other clustering methods in terminal scheduling will also be explored.

Acknowledgements

This project is sponsored by Shanghai Pujiang Program (16PJC043).

References

[1] LIANG C, HWANG H, GEN M.A berth allocation planning problem with direct transshipment consideration[J].Journal of Intel-ligent Manufacturing,2012,23(6):2207-2214.

[2] WANG, K., ZHEN, L., WANG, S., & LAPORTE, G. Column generation for the integrated berth allocation, quay crane assignment, and yard assignment problem. Transportation Science, 2018: 52(4), 812–834.

[3] IMAI A, YAMAKAWA Y, HUANG K. The strategic berth template problem. Transportation Reserch Part E: Logistics and Transportation Review. 2014;72:77-100.J. Liu, E.L. Chen and Z.T. He: Journal of Shi Jia Zhuang Railway Institute (Natural Science), Vol. 22 (2009) No. 4, p.40-42.

[4] MOORTHY R, TEO C-P. Berh management in container terminal: the temolate design problem. OR Spectrum 2006;28(4):495-518.

[5] VACCA I, SALANI M, BIERLAIRE M. An exact algorithm for the integrated planning of berth allocation and quay crane as-signment. Transportation Science. 2013;47(2):148-61.

[6] XIE F, WU T, ZHANG C. A branch-and-price algorithm for the integrated berth allocationand quay crane assignment problems under uncertainty. Computers & Industrial Engineering.2018;126:196-216.

[7] ZHEN L, CHEW EP, LEE LH. Ann integrated model for berth template and yard template planning in transshipment hubs. Transportation Science. 2011;45(4):483-504.

[8] MEISEL F, BIERWIRTH C, A framework for integrated berth allocation and quay crane operations planning in seaport container terminals. Transportation Science.2013;47(2):131-47.

[9] LALLA-RUIZ E, GONZALEZ-VELARDE J L, MELIAN- BATISTA B, et al. Biased random key genetic algorithm for the tactical berth allocation problem[J].Applied Soft Computing2014;22:60-76.

[10] YANG J, GAO H, LIU W. Dynamic Coordination Scheduling of Berth and Quayside Bridge under Discrete Berth Layout. Computer Engineering and Applications, 2018, 54(3):6.

[11] JIAO XG, ZHENG F, Xu YF. Joint scheduling of continuous berth and Dynamic Quay bridge considering berth dredging. Operations Research and Management, 2020, 29(2):11.

[12] ZHEN L, CHANG DF. A bi-objective model for robust berth allocation scheduling. Computers and Industrial Engineering. 2012;63(1):262-73.

[13] RODRIGUEZ-MOLINS M, SALIDO MA, BARBER F. Robust scheduling for berth allocation and quay crane assignment problem. Mathematical Problems in Engineering 2014;6:1–17.

[14] IRIS C , LAM J . Recoverable robustness in weekly berth and quay crane planning[J]. Transportation Research Part B: Meth-odological, 2019, 122(APR.):365-389.

[15] XIANG X , LIU C . An almost robust optimization model for integrated berth allocation and quay crane assignment problem[J]. Omega, 2021(9):102455.

[16] Tan C, He J. Integrated proactive and reactive strategies for sustainable berth allocation and quay crane assignment un-der uncertainty[J]. Annals of Operations Research 2021(3):1-32.

[17] BERTSIMAS D , GUPTA V , KALLUS N . Data-driven robust optimization[J]. Mathematical Programming, 2017.

[18] ROPKE S , PISINGER D . An Adaptive Large Neighborhood Search Heuristic for the Pickup and Delivery Problem with Time Windows[J]. Transportation ence, 2006, 40.

[19] LIANG CJ, WU Y. Joint Berth and Quayside Scheduling of Container Terminals under Uncertain Environment. Computer Engineering and Applications, 2017, 53(7):8.