Advanced Engineering Informatics 39 (2019) 14–24

Contents lists available at ScienceDirect

Advanced Engineering Informatics

journal homepage: www.elsevier.com/locate/aei

Full length article

Yard crane scheduling problem in a container terminal considering risk caused by uncertainty

Junliang He[a,b,*], Caimao Tan[a,*], Yuting Zhang[c]

[a] China Institute of FTZ Supply Chain, Shanghai Maritime University, Shanghai, PR China
[b] Engineering Research Center of Container Supply Chain Technology, Ministry of Education, Shanghai Maritime University, Shanghai, PR China
[c] Smart Solutions of ZPMC, Shanghai Zhenhua Heavy Industries Co., Ltd., Shanghai, PR China

ARTICLE INFO

Keywords:
Container port operation
Yard crane scheduling
Uncertainty
Three-stage algorithm
GA-based framework

ABSTRACT

In a container terminal, the arriving times and handling volumes of the vessels are uncertain. The arriving times of the external trucks and the number of containers which are needed to be brought into or retrieved from a container terminal by external trucks within a period are also uncertain. Yard crane (YC) scheduling is under uncertainty. This paper addresses a YC scheduling problem with uncertainty of the task groups' arriving times and handling volumes. We do not only optimize the efficiency of YC operations, but also optimize the extra loss caused by uncertainty for reducing risk of adjusting schedule as the result of the task groups' arriving times and handling volumes deviating from their plan. A mathematical model is proposed for optimizing the total delay to the estimated ending time of all task groups without uncertainty and the extra loss under all uncertain scenarios. Furthermore, a GA-based framework combined with three-stage algorithm is proposed to solve the problem. Finally, the proposed mathematical model and approach are validated by numerical experiments.

1. Introduction

Accompanied with the slowdown in trade growth, the volume growth of container shipping has been slowed down gradually. Container terminals are facing more challenges, such as fiercer and fiercer competition, attracting freight source, improving the service level, promoting operation efficiency, saving operation cost. The yard crane (YC) is the most important equipment for container yard operation, which can directly impact the entire operation efficiency. YC scheduling is one of the most important operation problems of a container terminal, which mainly refers to the handling sequence and starting service time of each YC for each task group (the concept of task group is referred to the core idea proposed by He et al. [17]. YC scheduling can directly impact the handling efficiency and cost of a container terminal.

As the result of the importance of YC scheduling, the operation efficiency of container terminals is most studied, and a lot of YC scheduling approaches for solely improving the efficiency are developed. To the best of our knowledge, most of the traditional researches on YC scheduling generally consider deterministic environment, where the arriving times and the handling volumes of the task groups are not uncertain. However, YC scheduling must face to various uncertain factors and unknown issues in reality. For example, because of the change of shipping liner's plan and weather reasons, the vessel may arrive at port earlier or later. As the result of handling equipment failure, the total served time of the vessel may be longer. The report of Drewry Shipping Consultants shows that the average arriving time of all ships at East-West route deviated from their estimated arrival time 1.9 days in January and February 2015 [7]. All of the uncertain factors impact the initial schedule. Once the uncertainty occurs, the planners should adjust or reschedule the initial schedule to satisfy the reality. But, this adjustment or rescheduling incur extra cost, and impact the other plans or schedules. If the uncertainty can be considered in the initial schedule, the risk of adjusting schedules will be reduced significantly. Therefore, container terminals need some models and methods for the YC scheduling problem with uncertainty. This paper mainly addresses the YC scheduling problem with uncertain arriving time and uncertain handling volume of task groups, which significantly impact the YC handling sequence . The main contributions of this work are: (1) the extra loss caused by uncertainty is measured, (2) a MIP model is formulated for minimizing the extra loss considering a lot of uncertain scenarios, and (3) a GA embedded three-stage algorithm is developed for obtaining enhanced solution.

The remainder of the paper is organized as follows. Literature review is addressed in Section 2. In Section 3, the YC scheduling problem with uncertain factors is described. In Section 4, the YC scheduling

* Corresponding author.
E-mail addresses: soldierlianglian@163.com (J. He), caimaotan@hotmail.com (C. Tan).

https://doi.org/10.1016/j.aei.2018.11.004
Received 12 July 2018; Received in revised form 29 October 2018; Accepted 22 November 2018

problem with uncertainty is modeled as a mixed integer programming (MIP). A three-stage optimization for solving this problem is proposed in Section 5. Section 6 validates the performance and the effectiveness of the developed model and algorithm by numerical experiments, and the last section concludes this work and gives the future research.

2. Literature review

Up to now, there are abundant studies focused on the operations of container terminals, such as handling equipment scheduling, storage space allocation, berth allocation and integrated scheduling of multi-resource [31,30,8]. Most of these works are devoted to port operation management under deterministic environment, whereas the uncertain factors are involved less. In this section, we mainly review the literatures highly related to port operation management under uncertainty and YC scheduling.

Port operation management under uncertainty have attracted more and more attention among academies, society and port industry. Up to present, most of literatures about uncertain scheduling in container terminals were focused on the berth allocation problem. Golias [9] addressed the berth allocation problem under uncertainty, and integrated exact algorithm, GA and simulation for solving this problem with uncertain vessel operation times. Moorthy and Teo [26] addressed the berth design problem considering data uncertainty and stochastic project scheduling. Zhou and Kang [38] formulated a berth & quay crane allocation model considering uncertain vessel arrival time and handling time to minimize the average waiting time. Han et al. [16] addressed berth scheduling problems with uncertain vessel arrival time and handling time; and a simulation-based optimization procedure was proposed to obtained robust schedule. Similarly, Zhen et al. [37] proposed a MIP model and two-stage meta-heuristic to solve the continuous berth allocation problem under stochastic arrival and operation times. The tactical berth allocation problem under uncertainty is studied more deeply by Zhen [36]. He developed a stochastic programming model considering the arbitrary probability distributions of operation time deviation, and developed a robust model to face limited information about probability distributions.

Golias et al. [10] developed a mathematical model and proposed a hierarchical optimization method for solving the berth scheduling problem under uncertain vessel arrival and handling times. For the other uncertain scheduling problem in container terminals, Angeloudis and Bell [1] proposed an AGV dispatching approach in a container terminal under various conditions of uncertainty such as uncertain tasks' time and space. Cai et al. [3] proposed two rescheduling strategies to solve long-term straddle carriers scheduling problem under the uncertainty of new job arrival at an automated container terminal. Zhen [35] proposed an approach for generating yard template under uncertain numbers of containers that need being loaded onto vessels. Lin and Chiang [25] studied the yard allocation problem, and in which the proposed approach could handle the uncertainty. Moreover, Güven and Türsel [15] proposed an approach for allocating the stack to each container online, which can be used as reactive strategy when some uncertainties occurred. Sislioglu et al. [29] proposed a simulation model and data envelopment analysis approach for the optimum investment alternatives considering different investment scenarios.

Aiming at YC scheduling problem, there are plenty of literatures in this area. For the scheduling of a single YC, Kim and Kim [20] used MIP to address the route and loading sequence problem for a single YC. Narasimhan and Palekar [27] developed a branch-and-bound based enumerative method to solve the similar problem. Furthermore, Kim and Kim [21] proposed a GA and a beam search algorithm to solve the same problem. Similarly, Kim et al. [19] also applied the similar algorithm to solve the handling sequence, travel route and the pickup number problem for a single YC. Guo et al. [12] addressed the handling sequence of a yard crane for delivering and pickup jobs, and proposed A^* search with acceleration approach for solving. Gharehgozli et al. [13] studied the handling sequence problem of a single YC for the container storage and retrieval requests, and proposed a two-phase algorithm to solve the problem. For the scheduling of multiple YCs, Zhang et al. [34] formulated a MIP model for deploying YCs among different blocks in a container yard. However, they only generated the handling range in the entire yard within each period, and the route and handling sequence for the YCs were not generated. Similarly, Cheung et al. [6] also formulated MIP model for the movement of YCs among all blocks, and proposed a Lagrangian decomposition solution procedure for solving this problem. Some realistic operational constraints are very important in the scheduling of multiple YCs, such as the interference of inter-crane, the limited operation ranges of YCs and simultaneous container storing/retrieving, which were taken into account in many YC scheduling models and algorithms [28,24]. In order to manage YC workload and schedule the multiple YCs, Guo et al. [11] proposed a new hierarchical approach and a time and space partitioning algorithm to. He et al. [17] developed a YC scheduling model as a vehicle routing problem (VRP) with soft time windows, and integrated a simulation model and heuristic for solving the problem. For the scheduling of double-rail-mounted gantry crane systems, Cao et al. [4] proposed a greedy heuristic algorithm, a simulated annealing (SA) algorithm and a combined scheduling heuristic. For the integrated scheduling of YCs and other equipment, He et al. [18] addressed the coordinated scheduling problem of YC, yard truck and quay crane, and combined GA, particle swarm optimization (PSO) and simulation model to solve the problem. Similarly, Li et al. [23] proposed a GA based on the heuristic algorithm to solve the integrated scheduling problem of YC, yard truck and quay crane.

According to our previous work [17], VRP with soft time windows is an appropriate and easily understood approach for modeling the YC scheduling problem. In recent years, VRP with soft time windows is widely studied. Bertsimas and Simchi-Levi [2] respectively investigated the models and algorithms for VRPs with static stochastic demand and dynamic stochastic demand. They thought that priori optimization approach (proactive method) for VRP with uncertain factors is better than re-optimization approach (reactive method). Taş et al. [32] addressed time-independent VRP with uncertainty. They mainly optimized the total transportation costs and extra costs caused by uncertain arrival times. Furthermore, Taş et al. [33] addressed a time-dependent VRP with uncertain travel times. Similarly, Gutierrez et al. [14] developed a multi-population evolving algorithm to solve VRP with soft time windows. This paper also employs VRP to formulate the YC scheduling problem under uncertainty.

In summary, few scholars have considered uncertainty in YC scheduling problem. Therefore, YC scheduling under uncertainty is a key problem in reality that has not been addressed in the literature, and is the core of this work. This paper is a supplement on the deterministic YC scheduling problem.

3. Problem description

3.1. Description of YC scheduling under uncertainty

YC scheduling problem mainly refers to two decisions: (1) the served sequence for each task group; and (2) the start served time of each task group. In order to reduce computational complexity, the scheduling object is not each container but task group. The construction approach of a task group is referred to the approach proposed by He et al. [17,18]. Task group means a lot of tasks are belong to the same shipping line or the same carrier at adjacent bays in the same block, which volume cannot exceed the working capacity of one YC in the planning horizon. Fig. 1 shows a typical YC schedule, where the sequence of Yard crane *YC1* is *TG1* → *TG4* → *TG3*, and the sequence of Yard crane *YC2* is *TG5* → *TG6* → *TG2*.YC scheduling could significantly impact the departing time of each vessel and the turnaround time of each external truck at terminal, and promote the entire handling

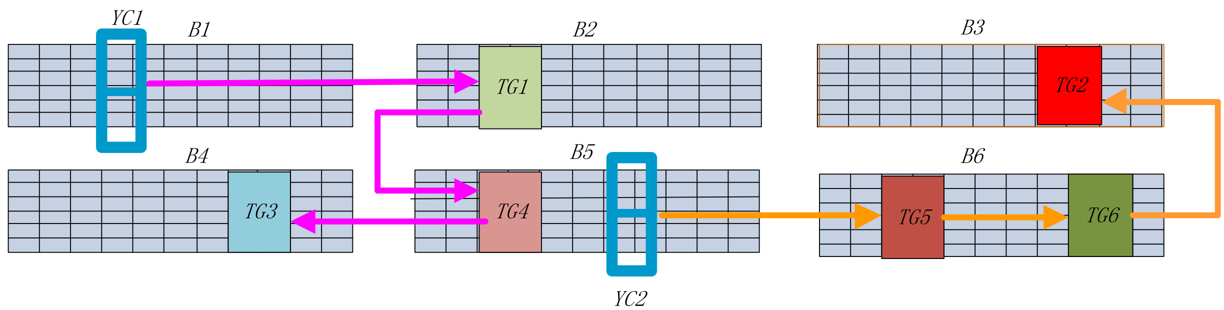

Fig. 1. An illustration of YC scheduling.

efficiency of a container terminal. As such, it is imperative to seek an appropriate YC scheduling approach.

In reality, there are a lot of uncertain factors in operation of a container terminal, such as the arriving time and handling volume of each vessel, the arriving times and number of external trucks. Since the vessels' arriving times may be earlier or later than their planned times, and the external trucks' arriving times may be earlier or later than their appointed times, the task groups' arriving times are fluctuated. Since the vessels' loading/unloading volumes may be more or less than their planned handling volumes, and the number of containers which are needed to be brought into or retrieved from a container terminal may be more or less than their planned handling volumes within a period, the task groups' handling volumes are fluctuated. Therefore, the arriving time and task number of each task group in YC scheduling are uncertain. For example, we suppose the planning arriving sequence of all task groups is $TG5 \to TG6 \to TG1 \to (TG2, TG4) \to TG3$, and the planning arriving times and handling volumes of $TG2$ and $TG4$ are almost the same. There are several YC scheduling scenarios: (1) If $TG4$ actually arrives late, but not impact the operations of $TG3$, Scheme 1: $YC1$ ($TG1 \to TG4 \to TG3$) and $YC2$ ($TG5 \to TG6 \to TG2$) is more appreciate (without extra loss and less transiting time). (2) If the real arrival time of $TG4$ is later than the completion time of $TG3$, Scheme 1: $YC1$ ($TG1 \to TG4 \to TG3$) and $YC2$ ($TG5 \to TG6 \to TG2$) also is more appreciate (with less extra loss and less transiting time). The arrival delay of $TG4$ in Scheme 1 does not impact the handling of other task groups. It can be handled after $TG3$ has been completed. (3) If the real handling time of $TG4$ overlaps with $TG3$, Scheme 2: $YC1$ ($TG1 \to TG2 \to TG3$) and $YC2$ ($TG5 \to TG6 \to TG4$) is more appreciate (with less extra loss) than Scheme 1. The arrival delay of $TG4$ in Scheme 2 does not impact the handling of other task groups, but it in Scheme 1 will cause the handling delay of $TG3$. Thus, it is necessary to develop a YC scheduling model and method to consider the operational efficiency and the adjusting cost caused by uncertainty simultaneously.

As a result of inter-crane interference, the operation ranges of YCs are often limited in reality. But in our work, a major goal is minimizing the adjusting cost caused by uncertainty. If the ranges of YCs are limited, some schemes with less the adjusting cost may be missed. Therefore, we assume that a yard crane can turn to any unvisited task groups in the yard, no matter how far away the new task group is from the previous one. Mover, our model and algorithm also can reduce the occurrence of this situation as much as possible, because the other goal is minimizing the completion delay cost of deterministic scheduling without uncertainty.

3.2. Extra loss caused by uncertainty

Fig. 2 illustrates an example of time axis of YC scheduling under uncertainty. In this example, parameters a_k, st_k, S_k, dp_k respectively denote the arriving time, the start served time, the needed handling time and the planned ending time of Customer k without uncertain factor. In deterministic scenario, we should minimize the completion delay, i.e., minimize$(st_k + S_k - dp_k)$.However, since the arriving times and the handling volumes of some task groups may fluctuate, the start served time and the needed service time are uncertain. Therefore, we should adjust schedule to satisfy the uncertain scenario. The adjustment of schedule may cause extra loss, such as the increment of YCs' moving distances, the increasing of YCs' handling volume unbalance, the adjustment of YCs' operation sequence and the increment of total completion delay of all task groups. All of the extra loss caused by uncertainty can be measured by the increment of total delay. In this example, parameters a_{ks}, st_{ks}, S_{ks}, dp_{ks} respectively denote the arriving time, the start served time, the needed handling time and the planned ending time of Customer k in Scenario s. Under uncertainty, we should minimize the completion delay in deterministic scenario and the expected value of the extra loss under uncertainty. The extra loss means the increment of completion delay of Customer k in Scenario s with respect to it in deterministic scenario. Thus, we should minimize$\Delta_{ks}^{+} = (st_{ks} + S_{ks} - dp_{ks}) - (st_k + S_k - dp_k)$, where Δ_{ks}^{+} means the increment of completion delay.

4. Modeling the YC scheduling problem under uncertainty

4.1. Notation

Parameters	
i	The index of task groups
j	The index of YCs
k	The index of customers
T	The set of all task groups in the planning horizon.
Y	The set of all YCs.
U	$U = T \cup Y \cup \{0\}$, where $\{0\}$denotes virtual depot.
p	The loading and unloading speed of each YC (unit: h/move)
V_k	The task number of Customer k(unit: move). If $k \in Y \cup \{0\}, V_k = 0$.
S_k	The needed handling time of Customer k.$S_k = V_k \cdot p$
a_k	The arriving time of Customer k. If $k \in Y \cup \{0\}, a_k = 0$.
dp_k	The planned ending time of Customer k. If $k \in Y \cup \{0\}$, $dp_k = 0$.
$t_{kk}^{\prime}$	The moving time between Customers k and k' (unit: h). From the depot to any customer, the move time is 0. We use the method proposed by He et al. [17,18] to calculate this parameter, i.e., $tt_{kk'} = \begin{cases} \frac{\text{The distance between Customers } k \text{ and } k'}{\text{The moving speed of a YC}}, \text{ if Customers } k \text{ and } k' \text{ are at the same lane;} \\ \frac{\text{The distance between Customers } k \text{ and } k'}{\text{The moving speed of a YC}} + 2 \cdot (\text{The unit turning time of a YC}), \text{otherwise} \end{cases}$

Cp_k	The unit penalty cost because of the completion delay of Customer k(unit: \$/h). If $k \in Y \cup \{0\}$, $Cp_k = 0$
M	A large enough number.
δ_{0k}^{j}	$\delta_{0k}^{j} = 1$, if Customer k is the initial location of YC j; $\delta_{0k}^{j} = 0$, otherwise. It denotes the first visiting customer of each YC after departing the virtual depot.
Ω	The set of uncertain scenarios (indexed by s)
P_s	The probability of Scenario s
a_{ks}	The actual arriving time of Customer k in Scenario s. The uncertainty is caused by the various arriving time of each task group. If $k \in Y \cup \{0\}$, ${}^{a}_{ks} = 0$.
dp_{ks}	The actual planned ending time of Customer k in Scenario s. The uncertainty is caused by the various arriving time and task number of each task group. If $k \in Y \cup \{0\}$, $dp_{ks} = 0$.
V_{ks}	The task number of Customer k in Scenario s(unit: move). If $k \in Y \cup \{0\}$, $V_{ks} = 0$.
S_{ks}	The actual needed handling time of Customer k in Scenario s. The uncertainty is caused by the various handling volume of each task group.$S_{ks} = V_{ks} \cdot p$
Φ_s	The set of customers that have varied arriving time or needed handling time in Scenario s. If $a_{ks} \neq a_k$, or $S_{ks} \neq S_k$,Then $k \in \Phi_s$.
λ_s	The frozen period point in Scenario s, i.e., the arriving time of a customer which have firstly varied arrival time or needed service time among all customers in Scenarios, $\lambda_s = \min_{k \in \Phi_s}(a_k)$. If a customer arrives before this point, its schedule must not be adjusted; otherwise the schedule may be changed.
Cp_k^+, Cp_k^-	The cost rate (or reward rate), caused by more (or less) completion delay in recovery process, where $Cp_k^- < 0$, $Cp_k^+ > Cp_k > -Cp_k^-$. As the result of the recovery cost of adjusting other schedules, the cost rate and reward rate are different [37]. For $k \notin T$, $Cp_k^-, Cp_k^+ = 0$.
Decision variables	
$\varphi_{kk'}^{j}$	$\varphi_{kk'}^{j} = 1$, if YC j travels from Customer k to Customer k'; $\varphi_{kk'}^{j} = 0$, otherwise
st_k	The start served time of Customer k.
st_{ks}^+, st_{ks}^-	The delay (or earlier time) of start served time of Customer k with respect to st_k in Scenario s
Δ_{ks}^+, Δ_{ks}^-	The increment (or decrement) completion delay of Customer k with respect to$st_k + S_k - dp_k$in Scenario s
$\varphi_{kk's}^{j}$	$\varphi_{kk's}^{j} = 1$, if YC j travels from Customer k to Customer k' in Scenario s; $\varphi_{kk's}^{j} = 0$, otherwise

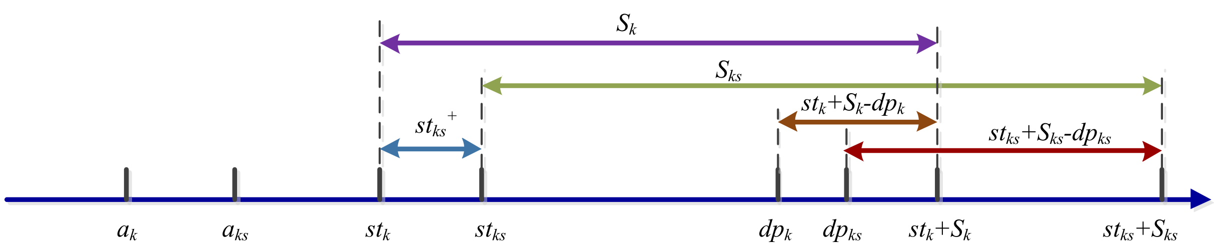

Fig. 2. An illustration of time axis of YC scheduling under uncertainty.

4.2. Mathematical model

In this work, we refer to our previous deterministic model [17]. https://doi.org/10.1016/j.aei.2014.09.003). The YC scheduling problem under uncertainty can be converted into a VRP. As the result of the uncertainty of the arriving time and the task number of each task group, the time windows of the VRP are uncertain. The YC scheduling problem under uncertainty is formulated as follows.

$$\min f = \sum_{k \in U} Cp_k \cdot \max(st_k + S_k - dp_k, 0) + \sum_{s \in \Omega} P_s \cdot \sum_{k \in U} (Cp_k^+ \cdot \Delta_{ks}^+ + Cp_k^- \cdot \Delta_{ks}^-) \tag{1}$$

$$\text{S.T.} \sum_{j \in Y} \sum_{k \in U, k \neq k'} \varphi_{kk'}^{j} = 1, \forall\, k' \in U, k' \neq 0 \tag{2}$$

$$\sum_{j \in Y} \sum_{k' \in U, k' \neq k} \varphi_{kk'}^{j} = 1, \forall\, k \in U, k \neq 0 \tag{3}$$

$$\sum_{j \in Y} \varphi_{0k}^{j} \leqslant 1, \forall\, k \in U, k \neq 0 \tag{4}$$

$$\sum_{k \in U, k \neq 0} \varphi_{0k}^{j} = 1, \forall\, j \in Y \tag{5}$$

$$\sum_{j \in Y} \varphi_{k0}^{j} \leqslant 1, \forall\, k \in U, k \neq 0 \tag{6}$$

$$\sum_{k \in U, k \neq 0} \varphi_{k0}^{j} = 1, \forall\, j \in Y \tag{7}$$

$$\varphi_{kk'}^{j} + \varphi_{k'k}^{j} \leqslant 1, \forall\, k, k' \in U, k \neq k', j \in Y, k, k' \neq 0 \tag{8}$$

$$\sum_{k \in U, k \neq k'} \varphi_{kk'}^{j} = \sum_{k'' \in U, k'' \neq k'} \varphi_{k'k''}^{j}, \forall\, k' \in U, k' \neq 0, j \in Y \tag{9}$$

$$st_k \geqslant a_k, \forall\, k \in U \tag{10}$$

$$st_k + (S_k + t_{kk'}) \leqslant st_{k'} + M \cdot (1 - \varphi_{kk'}^{j}), \forall\, k, k' \in U, k \neq k', j \in Y \tag{11}$$

$$\sum_{k \in \Psi} \sum_{k' \in \Psi, k \neq k'} \varphi_{kk'}^{j} \leqslant \sum_{k \in \Psi} \sum_{k' \in U, k \neq k'} \varphi_{kk'}^{j} - 1, \forall\, \Psi \subseteq U: 2 \leqslant |\Psi| \leqslant |U| - 1, j \in Y \tag{12}$$

$$\varphi_{0k}^{j} = \delta_{0k}^{j}, \forall\, k \in U, k \neq 0, j \in Y \tag{13}$$

$$st_k = 0, \forall\, k \in Y \text{ or } k = 0 \tag{14}$$

Eq. (1) is used to minimize the completion delay cost of deterministic scheduling without uncertainty and the expected value of the extra loss costs under uncertainty. The Formula $\sum_{s \in \Omega} P_s \cdot \sum_{k \in U} (Cp_k^+ \cdot \Delta_{ks}^+ + Cp_k^- \cdot \Delta_{ks}^-)$denotes extra loss costs under all uncertain scenarios. If the completion delay of Customer k with respect to $st_k + S_k - dp_k$ in Scenario s is increased, we use the penalty cost rate to calculate the extra loss costs; If the completion delay of Customer k with respect to$st_k + S_k - dp_k$ in Scenario s is decreased, we use the reward cost rate to calculate the extra loss costs, and the extra loss costs should be negative. It can avoid bringing too much bad impact on the task completion time by adjusting the base schedule under an uncertain scenario. In other words, it provides a scheme that can be easily adjusted to other scenarios.

Eqs. (2)–(11) and (13) and (14) are referred to our another paper [17], which are used to solve deterministic YC scheduling problem without uncertainty. Eqs. (2) and (3) express that each customer can be visited by exactly one YC. Eqs. (4) and (5) express that each YC must depart the virtual depot. Eqs. (6) and (7) express that each YC must back to the virtual depot from its last customer. Eq. (8) specifies that the travel is unidirectional. Eq. (9) defines the YC's route continuity, i.e., a

YC must move to next customer from its current customer. Eq. (10) enforces the start served time cannot be earlier than the arrival time of each customer. Eq. (11) defines the time relationship between two sequential customer served by the same YC, i.e., $st_k + (S_k + t_{kk'}) \leqslant st_{k'}$ when Customer k' is immediately handled by YC j after Customer k is completed. Eq. (12) is subtour elimination constraint. Eqs. (13) and (14) enforce that the first visiting customer of each YC after departing the virtual depot must be its initial location.

$$\sum_{j\in Y}\sum_{k\in U,k\neq k'}\varphi^j_{kk's}=1,\ \forall\ k'\in U,\ k'\neq 0,\ s\in\Omega \tag{15}$$

$$\sum_{j\in Y}\sum_{k'\in U,k'\neq k}\varphi^j_{kk's}=1,\ \forall\ k\in U,\ k\neq 0,\ s\in\Omega \tag{16}$$

$$\sum_{j\in Y}\varphi^j_{0ks}\leqslant 1,\ \forall\ k\in U,\ k\neq 0,\ s\in\Omega \tag{17}$$

$$\sum_{k\in U,k\neq 0}\varphi^j_{0ks}=1,\ \forall\ j\in Y,\ s\in\Omega \tag{18}$$

$$\sum_{j\in Y}\varphi^j_{k0s}\leqslant 1,\ \forall\ k\in U,\ k\neq 0,\ s\in\Omega \tag{19}$$

$$\sum_{k\in U,k\neq 0}\varphi^j_{k0s}=1,\ \forall\ j\in Y,\ s\in\Omega \tag{20}$$

$$\varphi^j_{kk's}+\varphi^j_{k'ks}\leqslant 1,\ \forall\ k,\ k'\in U,\ k\neq k',\ j\in Y,\ k,\ k'\neq 0,\ s\in\Omega \tag{21}$$

$$\sum_{k\in U,k\neq k'}\varphi^j_{kk's}=\sum_{k''\in U,k''\neq k'}\varphi^j_{k'k''s},\ \forall\ k'\in U,\ k'\neq 0,\ j\in Y,\ s\in\Omega \tag{22}$$

$$st_k+st^+_{ks}-st^-_{ks}\geqslant a_{ks},\ \forall\ k\in U,\ s\in\Omega \tag{23}$$

$$\begin{aligned}st_k+st^+_{ks}-st^-_{ks}+(S_{ks}+t_{kk'})\leqslant st_{k'}+st^+_{k's}-st^-_{k's}+M\cdot(1-\varphi^j_{kk's}),\\ \forall\ k,\ k'\in U,\ k\neq k',\ j\in Y,\ s\in\Omega\end{aligned} \tag{24}$$

$$\sum_{k\in\Psi}\sum_{k'\in\Psi,k\neq k'}\varphi^j_{kk's}\leqslant\sum_{k\in\Psi}\sum_{k'\in U,k\neq k'}\varphi^j_{kk's}-1,\ \forall\ \Psi\subseteq U\colon 2\leqslant|\Psi|\leqslant|U|-1,\ j\in Y,\ s\in\Omega \tag{25}$$

$$\varphi^j_{0ks}=\delta^j_{0k},\ \forall\ k\in U,\ k\neq 0,\ j\in Y,\ s\in\Omega \tag{26}$$

$$st_{ks}=0,\ \forall\ k\in Y\ \text{ or }\ k=0,\ s\in\Omega \tag{27}$$

$$\begin{aligned}&\max(st_k+S_k-dp_k,\ 0)+\Delta^+_{ks}-\Delta^-_{ks}\\ &\quad=\max(st_k+st^+_{ks}-st^-_{ks}+S_{ks}-dp_{ks},\ 0),\ \forall\ k\in T,\ s\in\Omega\end{aligned} \tag{28}$$

$$\Delta^+_{ks}+\Delta^-_{ks}=\max(\Delta^+_{ks},\ \Delta^-_{ks}),\ \forall\ k\in T,\ s\in\Omega \tag{29}$$

$$st^+_{ks}+st^-_{ks}=\max(st^+_{ks},\ st^-_{ks}),\ \forall\ k\in T,\ s\in\Omega \tag{30}$$

$$st^+_{ks},\ st^-_{ks}=0,\ \forall\ k\in U,\ s\in\Omega\colon a_{ks}\leqslant\lambda_s \tag{31}$$

$$\varphi^j_{kk's}=\varphi^j_{kk'},\ \forall\ k\in U,\ s\in\Omega\colon a_{ks}\leqslant\lambda_s \tag{32}$$

Eqs. (15)–(27) are used as the VRP constraints for YC scheduling problem under uncertainty, which functions are similar to Eqs. (2)–(14). Eq. (28) defines the relationship between the delay (or earlier time) of start served time (st^+_{ks}, st^-_{ks})and the increment (or decrement) completion delay(Δ^+_{ks}, Δ^-_{ks}). Eq. (29) ensures that there only exits one case, i.e., either more or less completion delay of Customer k in Scenario$s$. Eq. (30) ensures that there only exits one case, i.e., the start service time for Customer k is either delayed or earlier in Scenario$s$. Eqs. (31) and (32) ensure that if the arriving time of a customer is earlier than the frozen period point in Scenario s, its schedule have no change with respect to deterministic scheduling.

$$st_k,\ st^+_{ks},\ st^-_{ks},\ \Delta^+_{ks},\ \Delta^-_{ks}\geqslant 0,\ \forall\ k\in U,\ s\in\Omega \tag{33}$$

$$\varphi^j_{kk'},\ \varphi^j_{kk's}\in\{0,\ 1\},\ \forall\ k,\ k'\in U,\ k\neq k',\ j\in Y,\ s\in\Omega \tag{34}$$

Eqs. (33) and (34) defines the decision variables.

Although Eqs. (1), (28) and (29) contain maximum value forms, the model can be handled by commercial solvers such as ILOG CPLEX.

5. Solution algorithm

5.1. Overall framework

As aforementioned, the YC scheduling problem under uncertainty is a VRP with high complexity caused by uncertainty. Generally, the large-size VRP with deterministic environment is hard to solve in a short CPU time using exact algorithms or commercial solvers such as ILOG CPLEX, not to mention the YC scheduling problem under uncertainty, especially with a lot of scenarios. Thus a GA-based framework combined with three-stage algorithm is proposed to solving the problem, where the GA is used for global search and three-stage algorithm is used for generating YC schedules at each evolve generation. In the first stage, we try to obtain a baseline schedule, i.e., YCs' handling sequences under deterministic environment. In the second stage, the recovery schedules for all uncertain scenarios are generated, i.e., obtaining YCs' handling sequences under uncertain scenarios. Other decision variables related to time are obtained in the third stage. The framework for the proposed solution method is shown in Fig. 3.

5.2. Genetic algorithm for obtaining YCs' handling sequences under deterministic environment

In this section, we mainly referred the proposed GA for deterministic YC scheduling problem in our another paper [17] except fitness evaluation.

The main purpose of the solution method is to give operation sequences of all task groups by YCs. We firstly use the sequence encoding to represent the deterministic YC scheduling problem. Each gene represents a task group, and the sequence of the genes is the order of handling these task groups. Subsequently, the integer encoding is used to represent the YCs. An example of solution encoding is illustrated in Fig. 4, which represents a schedule as follows:

Route $YC1$: $0\rightarrow 3\rightarrow 5\rightarrow 2\rightarrow 4\rightarrow 1\rightarrow 0$

Route $YC2$: $0\rightarrow 6\rightarrow 8\rightarrow 7\rightarrow 0$

Push-Forward Insertion Heuristic (PFIH) is usually used to generate the initial solution of VRP. In this paper, we still used a modified PFIH to obtain the initial YCs' handling sequences [17]. The feasibility of all chromosomes should be checked by Eqs. (2)–(14). The gene repair procedure are the same as our previous proposed method.

We only want to obtain the deterministic schedule at this stage, but the expected value of the extra loss costs of recovery schedules under all uncertain scenarios based on different deterministic schedules are different. Therefore, the fitness value of each deterministic schedule should be based on Function (1), and can be obtained at the third stage. The fitness can be evaluated as:

$$F=1/\left[\sum_{k\in U}Cp_k\cdot\max(st_k+S_k-dp_k,\ 0)+\sum_{s\in\Omega}P_s\cdot\sum_{k\in U}(Cp^+_k\cdot\Delta^+_{ks}+Cp^-_k\cdot\Delta^-_{ks})\right] \tag{35}$$

Aiming at parent selection strategy, a roulette wheel approach [5] are used. To avoid generate unfeasible chromosome, the order crossover is still used for the cross operation of the first vector in a chromosome. The second vector can be obtained by the same procedure as the method described in Section 5.2.5 of our another paper [17]. Aiming at mutation operation, a swap mutation operator is employed for the 'Task group' chromosome [22]. For offspring acceptance, we still keep the size of the population unchanged form a new population in the next generation by ranking all chromosome fitness values in a

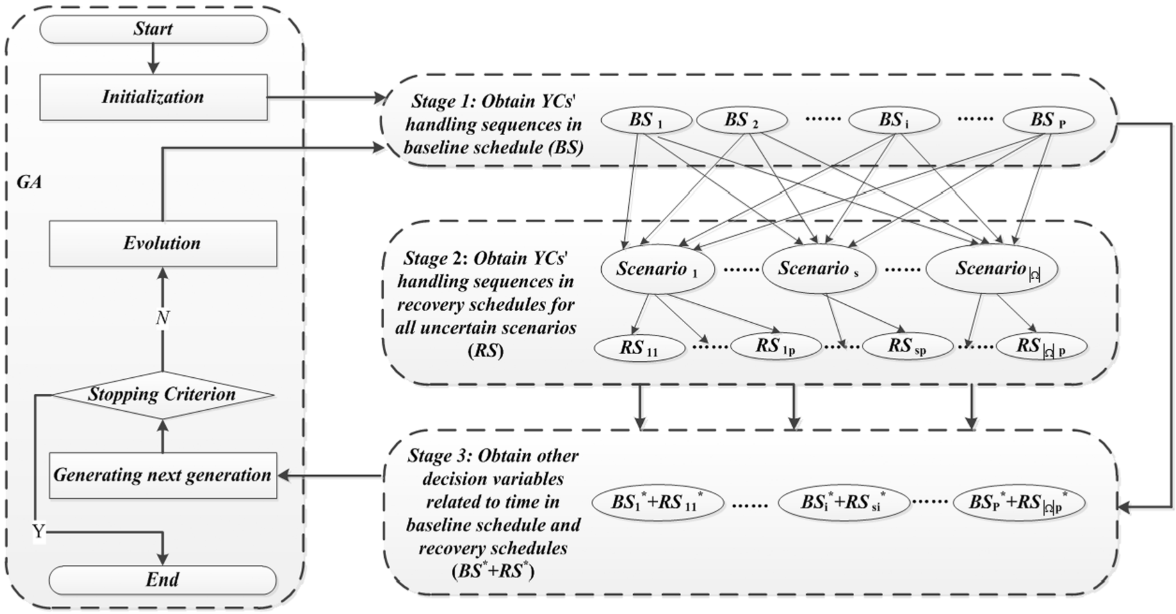

Fig. 3. Framework of the proposed three-stage algorithm.

Task group	3	5	2	4	1	6	8	7
YC	1	1	1	1	1	2	2	2

Fig. 4. Solution encoding.

descending sequence. The maximum elapsed generations experimentally-determined is used as stopping criterion.

5.3. Simulated annealing algorithm for obtaining YCs' handling sequences under uncertain scenarios

From the above, when YCs' handling sequences in a baseline schedule, we should obtain YCs' handling sequences under each uncertain scenario. We employ the framework of simulated annealing (SA) to optimize YCs' handling sequences under a uncertain scenario. As aforementioned the definition of λ_s in Section 4.1, if a task group arrives before the frozen period point in Scenario s, its schedule need not be adjust. Therefore, the proposed SA mainly is used to adjust YC schedules for the task groups which arrive after the frozen period.

(1) The framework of the SA

Notations T, α and R respectively represent the initial temperature, cooling rate ($0 < \alpha < 1$), and the number of nested loops. For the YCs' handling sequences in a baseline schedule BS_i, the framework of the SA to obtain the recovery schedule under Scenario s is as follows.

Let the initial YCs' handling sequences S^o is BS_i, and the uncertain scenario is s.
Set SA parameters T, α, *threshold*;
While $T >$ *threshold* **Do**
    Generate R neighbor sequences of the initial YCs' handling sequences S^o using the later proposed neighborhood generation method, where S_i represents the ith neighbor;
    For $i = 1$ to R **Loop**
        Obtain the schedule of the neighbor S_i. Since the YCs' handling sequences in the neighbor S_i, we only need to determine the starting served times of all task groups which arrive after the frozen period. The starting served time of a task group in Scenario s is maximum value between its arriving time and the time of its assigned YC arriving at its location from the previous task group.
        Calculate the evaluation value $f(S_i)$ of S_i using Equation $\sum_{k\in U}(Cp_k^+\cdot\Delta_{ks}^+ + Cp_k^-\cdot\Delta_{ks}^-)$.
        Let $\Delta = f(S_i) - f(S^o)$;
        If $\Delta < 0$ **Then**
            Let $S^o = S_i$;
        Else
            Generate a random number r between 0 and 1;
            If $\alpha < e^{-\Delta/T}$ **Then**
                Let $S^o = S_i$;
            End if;
        End if;
End for;
Let $T = \alpha \times T$;
End while;

---

(2) Neighborhood generation

To generate new neighbors, the proposed approach selects two task groups from the currently optimal schedule and swaps them in the chromosome. All task groups which arrive before the frozen period point in Scenario s must not be selected for swapping. The SA prefers to select the task groups with more extra loss to swap. If a task group with more extra loss is ahead in the sequence, it will have more probability to be handled early, and the task group would have less extra loss. The details are as follows:

Step 1: Evaluate the extra loss of each task groups which arriving after the frozen period point using equation $Cp_k^+\cdot\Delta_{ks}^+ + Cp_k^-\cdot\Delta_{ks}^-$;
Step 2: Evaluate the gap of extra loss between two adjacent task groups: $\text{Gap}(k) = \text{Loss}(T_k)$ -$\text{Loss}(T_{k+1})$, where T_k denotes the kth task group;
Step 3: Swap Task group T_k with the largest Gap(k)and Task group T_{k+1} to generate a new neighborhood.
Step 4: Generate other neighbors by swapping two adjacent task groups with the second largest, …, the Rth largest Gap(k). Duplicated neighbors with previous iterations should be deleted, and more ones should be generated according to the gap sequence until all neighbors are not repeated.

5.4. Obtaining other decision variables related to time in the proposed model

The YCs' handling sequences under deterministic environment and uncertain scenarios are respectively obtained in the first and second stages described in Sections 5.2 and 5.3. In this section, YCs' handling sequences $\varphi_{kk'}^j$ and $\varphi_{kk's}^j$of the baseline schedule and the recovery

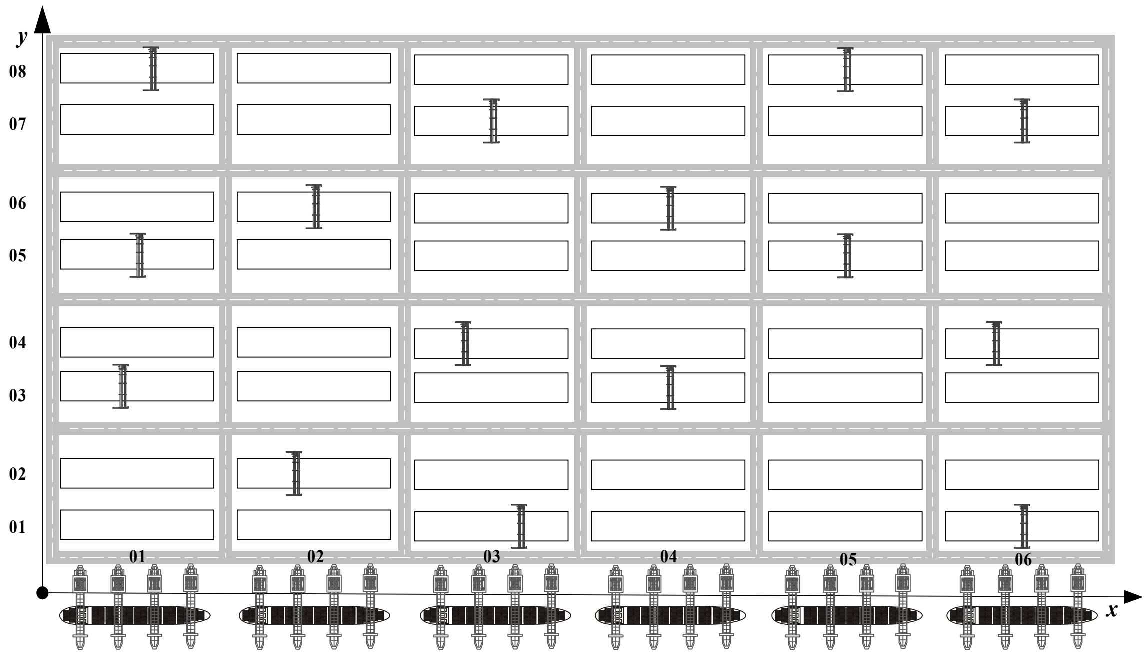

Fig. 5. Configuration of container terminal layout.

schedules in each uncertain scenario are as given parameters, and the other decision variables related to time will be obtained by CPLEX. Since $\varphi_{kk'}^{j}$ and $\varphi_{kk's}^{j}$ are determined, the constraints only related to $\varphi_{kk'}^{j}$ and $\varphi_{kk's}^{j}$will not be exist, and only the time-related constraints will be kept. The constraints related to $\varphi_{kk'}^{j}$ and $\varphi_{kk's}^{j}$ with large number M will be simplified into normal constraints as follows:

When $\varphi_{kk'}^{j}=1$, $t_k+(S_k+t_{kk'})\leqslant st_{k'}$, $\forall\, k,\,k'\in U,\,k\neq k',\,j\in Y$;

When $\varphi_{kk's}^{j}=1$ $st_k+st_{ks}^{+}-st_{ks}^{-}+(S_{ks}+t_{kk'})\leqslant st_{k'}+st_{k's}^{+}-st_{k's}^{-}$, $\forall\, k,\,k'\in U,\,k\neq k',\,j\in Y,\,s\in\Omega$

Therefore, the simplified model is as follows:

[P2] min Objective function (1)

S.T. Constraints (10), (14), (23), (27)-(31) and

$$t_k+(S_k+t_{kk'})\leqslant st_{k'},\ \forall\, k,\,k'\in U,\,k\neq k',\,j\in Y,\,\varphi_{kk'}^{j}=1 \tag{36}$$

$$t_k+st_{ks}^{+}-st_{ks}^{-}+(S_{ks}+t_{kk'})\leqslant st_{k'}+st_{k's}^{+}-st_{k's}^{-},\quad \forall\, k,\,k'\in U,\,k\neq k',\,j\in Y,\,s\in\Omega,\,\varphi_{kk's}^{j}=1 \tag{37}$$

The above model can be solved by CPLEX, and the decision variables related to time can be obtained.

6. Computational experiments

To validate the effectiveness of the proposed approach, we conduct a series of numerical experiments with different sizes. This section consists of three parts: (1) performance analysis, (2) extra loss analysis and (3) sensitivity analysis. In the performance analysis, the CPU time and solution quality between the proposed method and CPLEX are compared. In the extra loss analysis, we compare the completion delay costs and the extra loss costs between schedules obtained by the proposed model and the traditional deterministic scheduling model. Such differences could demonstrate the ability of the proposed approach for handling the uncertainty. Sensitivity analysis are performed to test the varieties of the extra loss cost and delay cost with varied the number of YCs and the number of Task groups. All experiments are run on a PC with Intel Core TM i7-2820QM @ 2.3 GHz processors and 16 GB RAM.

Table 1
The initial position of YCs in the yard.

YC No.	Yard Position	YC No.	Yard Position	YC No.	Yard Position
01	(3,1)	06	(3,7)	11	(3,2)
02	(4,2)	07	(4,4)	12	(1,3)
03	(2,1)	08	(5,3)	13	(6,4)
04	(2,5)	09	(4,8)	14	(6,8)
05	(6,1)	10	(1,6)	15	(5,7)

Table 2
The parameters of each task group.

TG No.	Arriving time	Planned completion time	Handling volume	Yard position
T1	00:00	02:12	55	(3,1)
T2	00:00	02:28	67	(2,2)
T3	00:00	01:46	42	(4,2)
T4	00:32	03:38	63	(2,1)
T5	01:42	03:00	66	(6,1)
T6	00:50	04:06	84	(5,3)
T7	01:20	05:28	112	(3,3)
T8	00:54	04:22	93	(4,3)
T9	01:38	03:48	51	(1,4)
T10	01:08	03:38	57	(1,2)
T11	01:36	05:16	95	(5,4)
T12	01:32	03:56	58	(2,5)
T13	00:32	03:38	74	(6,7)
T14	01:36	04:42	76	(4,1)
T15	01:42	03:00	62	(2,8)
T16	03:54	07:15	98	(3,5)
T17	03:07	05:21	46	(2,6)
T18	01:58	04:25	25	(3,7)
T19	02:45	06:27	70	(1,8)
T20	03:37	06:41	62	(6,3)
T21	04:45	06:32	42	(4,5)
T22	02:27	05:14	32	(3,2)
T23	03:15	06:02	56	(4,4)
T24	05:27	08:51	83	(1,1)
T25	04:29	06:49	59	(3,4)
T26	02:54	05:16	61	(6,3)
T27	03:36	07:01	24	(4,1)
T28	02:12	04:35	30	(6,5)
T29	03:03	06:05	27	(5,2)
T30	03:41	06:27	52	(4,5)

Table 3
The arriving time and task number of each task group in each scenario.

TG No.	Scenario 1		Scenario 2		Scenario 3		Scenario 4		Scenario 5	
	a_k	V_k	a_k	V_k	a_k	V_k	a_k	V_k	a_k	V_k
T1	00:00	55	00:00	55	00:00	47	00:00	55	00:00	55
T2	00:00	67	00:00	67	00:00	67	00:00	82	00:15	67
T3	00:28	42	00:00	42	00:28↑	42	00:00	42	00:00	50
T4	00:32	63	00:24↓	63	00:32	63	00:24↓	63	00:32	63
T5	01:33↓	79	01:42	74↑	01:42	66	01:33↓	66	01:42	66
T6	00:50	84	00:50	84	00:50	84	00:50	84	00:43↓	84
T7	01:03↓	112	01:20	130↑	01:04↓	130↑	01:20	130↑	01:20	112
T8	00:54	93	00:54	93	00:54	93	00:54	93	01:08↑	93
T9	01:38	42	01:24	51	01:38	51	01:24	51	01:38	51
T10	01:08	57	01:08	57	01:08	57	01:08	57	01:08	57
T11	02:01↑	95	01:36	95	01:36	95	01:36	95	01:36	95
T12	01:32	58	01:32	58	01:32	58	01:32	58	01:32	58
T13	00:24↓	74	00:32	74	00:32	74	00:32	74	00:32	74
T14	01:36	76	01:36	76	01:36	76	01:36	76	01:36	76
T15	01:42	62	01:42	62	01:42	62	01:42	62	01:42	62
T16	03:54	98	03:54	98	03:54	98	03:54	98	03:54	98
T17	03:07	46	03:07	46	03:07	46	03:07	46	03:07	46
T18	01:58	25	01:58	25	01:58	25	01:58	25	01:58	25
T19	02:45	70	02:45	70	02:45	70	02:45	70	02:45	70
T20	03:37	62	03:37	62	03:37	62	03:37	62	03:37	62
T21	04:45	42	04:45	42	04:45	42	04:45	42	04:45	42
T22	02:27	32	02:27	32	02:27	32	02:27	32	02:27	32
T23	03:15	56	03:15	56	03:15	56	03:15	56	03:15	56
T24	05:27	83	05:27	83	05:27	83	05:27	83	05:27	83
T25	04:29	59	04:29	59	04:29	59	04:29	59	04:29	59
T26	02:54	61	02:54	61	02:54	61	02:54	61	02:54	61
T27	03:36	24	03:36	24	03:36	24	03:36	24	03:36	24
T28	02:12	30	02:12	30	02:12	30	02:12	30	02:12	30
T29	03:03	27	03:03	27	03:03	27	03:03	27	03:03	27
T30	03:41	52	03:41	52	03:41	52	03:41	52	03:41	52

Note: the shaded cells represent the data have changed compared with the initial data of deterministic scenario.

6.1. Instances generation

For the instances generated in these experiments, we use the configuration of container terminal layout is shown in Fig. 5. As we can see, there are 48 blocks and 15 available YCs in the yard area. Meanwhile, the direction at the front of the yard is defined as the X-axis, the perpendicular direction is the Y-axis. Therefore, the coordinate (x, y) can be used to represent the detailed position of a task group.

All YCs in the container terminal have the same handling performance, the handling efficiency, the moving speed and 90-degree turn time are set as 2 min/move, 120 m/min and 2 min, respectively. The initial positions of all YCs in the yard are given in Table 1.

We generate 30 task groups randomly, whose handling volume obey a uniform distribution U (10,240). Their arriving intervals obey the distribution E (4 h*3600 S/Number of task groups). The planned ending time of each task group is calculated as $dp_k = a_k + S_k{\cdot}U$ (1.05, 1.1). The detailed parameters setting of each task group are shown in Table 2.

Two uncertain factors which affect the result of YC scheduling are selected as the main parameters for creating uncertain environment, which are arrival time and task number of each task group. Based on the above data, 5 uncertain scenarios are generated (Scenarios 1–5). The probabilities of each scenario are set to 0.23, 0.28, 0.17, 0.09 and 0.23 respectively. The yard congestion of Scenarios 1–5 are set to 1.35, 0.96, 1.05, 1.19 and 0.87 respectively. Besides, the arriving time and task number of each task group in Scenarios 1–5 are shown in Table 3.

The unit penalty cost for delay of each task group in deterministic scenarios and uncertain scenarios are set based on its priorities, the detailed parameters are shown in Table 4.

Table 4
Penalty cost of each task group in deterministic scenarios and uncertain scenarios ($/min).

TG No.	Certain situation	Uncertain situation	
	Delay penalty cost	Expended delay cost	Shortened delay cost
T1	10.8	12.4	−8.2
T2	9.3	11.3	−7.9
T3	10.4	11.7	−8.3
T4	4.6	6.2	−3.4
T5	8.1	9.3	−6.8
T6	5.9	6.8	−4.2
T7	6.3	7.4	−5.6
T8	7.1	8.9	−5.1
T9	9.6	10.5	−7.3
T10	5.5	6.7	−3.2
T11	6.2	12.1	−4.6
T12	8.3	11.6	−5.5
T13	11.6	18.4	−6.7
T14	6.1	9.4	−4.8
T15	4.2	7.3	−2.1
T16	13.7	17.3	−4.9
T17	6.9	9.2	−3.4
T18	8.2	10.8	−5.5
T19	4.8	6.7	−2.1
T20	3.7	5.9	−1.4
T21	6.3	8.8	−4.2
T22	5.1	7.9	−3.0
T23	2.8	5.3	−1.0
T24	7.9	10.7	−2.7
T25	8.2	10.9	−3.6
T26	3.8	9.2	−1.8
T27	7.4	9.9	−4.2
T28	3.7	6.3	−1.4
T29	5.1	8.8	−3.3
T30	9.9	14.2	−2.8

Table 5
Parameter setting for solution algorithm.

Parameters	Small-scale	Large-scale
Population size	60	90
Maximum number of generations	80	120
Crossover probability	0.6	0.8
Mutation probability	0.1	0.2
Initial temperature T	40	40
Cooling rate α	0.6	0.6
Number of nested loops R	40	40
Threshold	1	1

Table 7
Comparisons for large size problems.

NO.	Scale	Lower bound	The proposed method		GAP
		f^L($)	f^G($)	CPU (s)	
10	9 YCs and 15 TGs	3526.1	3786.6	173.35	6.9%
11	9 YCs and 18 TGs	6063.2	6613.1	231.84	8.3%
12	9 YCs and 21 TGs	10806.8	11679.5	202.19	7.5%
13	12 YCs and 18 TGs	4275.8	4855.1	327.36	11.9%
14	12 YCs and 22 TGs	7463.9	8342.0	284.32	10.5%
15	12 YCs and 26 TGs	13004.1	14359.1	325.91	9.4%
16	15 YCs and 20 TGs	4263.1	4599.7	425.07	7.3%
17	15 YCs and 25 TGs	6302.7	7226.3	474.83	12.8%
18	15 YCs and 30 TGs	9821.3	11203.8	492.70	12.3%
Avg.					9.7%

Note: GAP = $(f^G - f^L)/f^L \times 100$.

Based on the preliminary tests, different setting parameters are used for the proposed solution algorithm in the computational experiments of small size and large size instances. The detailed parameters are shown in Table 5.

6.2. Performance analysis

6.2.1. Performance analysis of the proposed solution algorithm in small size instances

Due to the complexity of the proposed mathematical model, only the small-size instances (e.g. 5 YCs and 9 TGs) can be solve by CPLEX. Table 6 compared the results between the two approaches. It shows that the optimality gap of the proposed method is a very small (avg. 6.8%) compared with CPLEX. Meanwhile, the computation time of the proposed method is shorter than that of CPLEX. The results validated the efficiency of the proposed method on small-size instances.

6.2.2. Performance analysis of the proposed solution algorithm in large size instances

For large-size instances, Table 7 shows that the average gap between the result obtained by the proposed method and the lower bound obtained by CPLEX is about 9.7%, which implies that the actual optimality gap of the proposed method is less. Moreover, the advantage of the proposed method is even more obvious when solving large-size instances.

6.3. Extra loss analysis

For evaluating the effective of the YC scheduling under uncertain

Table 6
Comparisons for small size problems.

NO.	Scale	CPLEX			The proposed method			GAP	
		$f^C_{certain}$($)	f^C($)	CPU (s)	$f^G_{certain}$($)	f^G($)	CPU (s)	GAP_c	GAP_t
1	3 YCs and 5 TGs	818.1	1582.8	9.68	848.9	1636.76	44.90	3.6%	3.3%
2	3 YCs and 6 TGs	1274.5	2381.2	81.34	1340.7	2522.63	88.27	4.9%	5.6%
3	3 YCs and 7 TGs	2523.7	3986.7	538.43	2693.5	4317.69	112.78	6.3%	7.7%
4	4 YCs and 6 TGs	865.7	1544.6	46.50	884.7	1630.44	102.02	2.1%	5.3%
5	4 YCs and 7 TGs	1464.5	2259.5	186.60	1515.5	2432.24	129.22	3.4%	7.1%
6	4 YCs and 8 TGs	2719.8	3775.3	850.40	2886.3	4137.83	134.78	5.8%	8.8%
7	5 YCs and 7 TGs	816.7	1642.4	196.89	838.1	1749.53	128.17	2.6%	6.1%
8	5 YCs and 8 TGs	1368.0	2396.9	309.57	1431.9	2617.19	158.67	4.5%	8.4%
9	5 YCs and 9 TGs	2719.0	3834.9	966.70	2936.1	4212.83	173.75	7.4%	9.0%
Avg.								4.5%	6.8%

Note: $GAP_c = (f^G_{certain} - f^C_{certain})/f^C_{certain} \times 100$, and $GAP_t = (f^G - f^C)/f^C \times 100$.
$f^C_{certain}$: The delay cost of deterministic scheduling without uncertainty obtained by CPLEX, $f^G_{certain}$: The delay cost of deterministic scheduling without uncertainty obtained by the proposed method, f^C: The total cost considering uncertainty obtained by CPLEX, f^G: The total cost considering uncertainty obtained by the proposed method.

Table 8
Comparison results between TYCS and UYCS in small-scale instance.

NO.	TYCS ($)		UYCS ($)		Optimal cost difference	
	Delay cost f^T_{delay}	Extra loss f^T_{extra}	Delay cost f^R_{delay}	Extra loss f^R_{extra}	Certain situation (GAP_c)	Uncertain situation (GAP_u)
1	614.7	1177.9	848.9	787.9	38.10%	−33.11%
2	1162.3	1377.7	1340.7	1181.9	15.35%	−14.21%
3	2182.3	2073.8	2693.5	1624.2	23.42%	−21.68%
4	614.7	1261.9	884.7	745.7	43.92%	−40.91%
5	969.5	1740.3	1515.5	916.7	56.32%	−47.33%
6	2129.1	2477.5	2886.3	1251.5	35.56%	−49.49%
7	662.4	1473.1	838.1	911.4	26.52%	−38.13%
8	969.5	1908.1	1431.9	1185.3	47.69%	−37.88%
9	1962.7	2526.2	2936.1	1276.7	49.59%	−49.46%
				Avg.	37.39%	−36.91%

Note: $GAP_c = [f^R_{delay} - f^T_{delay}]/f^T_{delay}$, and $GAP_u = (f^R_{extra} - f^T_{extra})/f^T_{extra}$.

Table 9
Comparison results between TYCS and UYCS in large-scale instance.

NO.	TYCS ($)		UYCS ($)		Optimal cost difference	
	Delay cost f^T_{delay}	Extra loss f^T_{extra}	Delay cost f^R_{delay}	Extra loss f^R_{extra}	Certain situation (GAP_c)	Uncertain situation (GAP_u)
10	1435.9	3239.5	1690.3	2096.3	17.7%	−35.3%
11	2749.2	4934.3	3442.8	3170.3	25.2%	−35.7%
12	6807.9	5934.9	8017.4	3662.1	17.8%	−38.3%
13	2222.9	3610.8	2802.7	2052.4	26.1%	−43.2%
14	5218.5	4671.7	5958.9	2383.1	14.2%	−49.0%
15	9967.1	5291.1	11733.2	2625.9	17.7%	−50.4%
16	1985.1	5814.3	2301	2298.7	15.9%	−60.5%
17	3482.6	6073.8	4228.1	2998.2	21.4%	−50.6%
18	6122.5	7069.9	7945.2	3258.6	29.8%	−53.9%
				Avg.	20.6%	−46.3%

Table 10
Sensitivity analysis of small size instances.

No.	Size		UYCS ($)		
	YCs	TGs	Delay cost	Extra loss	Scheduling cost
1	3	5	848.9	787.9	1636.76
2	3	6	1340.7	1181.9	2522.63
3	3	7	2693.5	1624.2	4317.69
4	4	6	884.7	745.7	1630.44
5	4	7	1515.5	916.7	2432.24
6	4	8	2886.3	1251.5	4137.83
7	5	7	838.1	911.4	1749.53
8	5	8	1431.9	1185.3	2617.19
9	5	9	2936.1	1276.7	4212.83

Table 11
Sensitivity analysis of large size instances.

No.	Size		UYCS ($)		
	YCs	TGs	Delay cost	Extra loss	Scheduling cost
1	9	15	1690.3	2096.3	3786.6
2	9	18	3442.8	3170.3	6613.1
3	9	21	8017.4	3662.1	11679.5
4	12	18	2802.7	2052.4	4855.1
5	12	22	5958.9	2383.1	8342.0
6	12	26	11733.2	2625.9	14359.1
7	15	20	2301	2298.7	4599.7
8	15	25	4228.1	2998.2	7226.3
9	15	30	7945.2	3258.6	11203.8

environment proposed in this paper, we compare the results (i.e., the completion delay costs and the extra loss costs) of the proposed YC scheduling approach considering uncertainty and the traditional YC scheduling approach both in small size and large size instances.

Table 8 shows the comparison results between traditional yard crane scheduling (TYCS) and the proposed YC scheduling approach considering uncertainty (UYCS) in small size instance problem instances. As we can observe that the TYCS has a lower delay cost than the UYCS in determined environment (about 37%).

However, the UYCS has a lower extra loss (about 36.9%) than the TYCS under uncertain environment. Such differences could demonstrate the superiority of the proposed approach for handling the uncertainty. Besides, although the delay cost of UYCS are higher, the total scheduling cost still lower than that obtained by TYCS under uncertain environment.

For further evaluating the effectiveness of the proposed UYCS, we compare the results by UYCS and TYCS for solving large-size instances. This same conclusion is reinforced by the comparison results shown in Table 9. The extra loss obtained by UYCS are obviously smaller than those obtained by TYCS (for example, the average gap is −46.3%, the minimum gap is −35.3%, and the maximum gap is up to −60.5%).

6.4. Sensitivity analysis

We use varied the number of task groups with the same number of YCs and varied the number of YCs with the same number of task groups to are analyze sensitivity of the proposed method. Tables 10 and 11 show that when the same YCs with varied TGs (e.g. Instances 1–3, 3 YCs with 5, 6 and 7 TGs respectively), the delay cost and extra loss increase with the rise of TGs. However, when the same TGs with varied YCs (e.g. Instances 2 and 4, 6 TGs with 3 and 4 YCs), the delay cost and extra loss decrease with the increment of TGs.

7. Conclusions

Since there are a lot of uncertain factors, such as the changes of shipping liner's plan, changes of weather, handling equipment failure and variations of requested service time, operation plans and schedules may have to be adjusted. We should reduce the adjustment caused by the uncertain factors. This paper addresses YC scheduling problem in a container terminal under uncertainty in reality. A mixed integer programming (MIP) model and three-stage optimization algorithm are proposed to handle this problem. By comparing with other research in this area, the major contributions are as follows:

(1) In order to evaluate the recovery capacity of a YC schedule under uncertain environment, we should calculate the extra loss as a result of the schedule adjustment. In this paper, extra loss caused by uncertainty is measured by the expected value of total completion delay increments in all scenarios with respect to it in deterministic scenario. The smaller the expected value of the extra loss, the more recovery capacity the YC schedule.
(2) A MIP model under uncertainty is formulated considering a lot of uncertain scenarios. The model includes two parts: the one is formulated to solve deterministic YC scheduling problem without uncertainty and the other is formulated for solving it in uncertain scenarios. The objective function is to minimize the completion delay cost of deterministic scheduling without uncertainty and the expected value of the extra loss costs under uncertainty.
(3) A GA-based framework combined with three-stage algorithm is proposed to solving the problem. GA is used as the first stage algorithm to optimize handling sequence under deterministic environment. SA is used as the second stage algorithm to obtain handling sequences in all uncertain scenarios. In the third stage, other decision variables related to time are obtained by CPLEX.

However, there are limitations in the current model and solution approaches. In this paper, we handle uncertainty based on uncertain scenario. However, a lot of uncertainties can not be represented using scenario, e.g., disruption events, equipment failure and vessels back-logging. Therefore, YC scheduling problem considering emergencies will be a valuable future research direction.

8. Declarations

The authors declare that they have no competing interests, and the first two authors contributed equally to this work.

Acknowledgement

This work is sponsored by National Natural Science Foundation of China (71602114) and Shanghai Science & Technology Committee Research Project (17040501700, 15590501700).

References

[1] P. Angeloudis, M.G.H. Bell, An uncertainty-aware AGV assignment algorithm for automated container terminals, Transp. Res. Part E: Logistics Transp. Rev. 46 (2010) 354–366.
[2] D.J. Bertsimas, D. Simchi-Levi, A new generation of vehicle routing research: robust algorithms, addressing uncertainty, Operations Res. 44 (2) (1996) 286–304.
[3] B. Cai, S. Huang, D. Liu, G. Dissanayake, Rescheduling policies for large-scale task allocation of autonomous straddle carriers under uncertainty at automated container terminals, Robotics Autonomous Syst. 62 (2014) 506–514.
[4] Z. Cao, D.-H. Lee, Q. Meng, Deployment strategies of double-rail-mounted gantry crane systems for loading outbound containers in container terminals, Int. J. Prod. Econ. 115 (1) (2008) 221–228.
[5] M. Gen, R. Cheng, Genetic Algorithms and Engineering Design, John Wiley & Sons Inc, New York, 1997.
[6] R.K. Cheung, C. Li, W. Lin, Interblock crane deployment in container terminals, Transp. Sci. 36 (2002) 79–93.
[7] Drewry. Carrier Performance Insight–First Quarter 2015. Drewry Shipping Consultants, London, 2015.
[8] J.C. Fransoo, C.-Y. Lee, The critical role of ocean container transport in global supply chain performance, Prod. Operat. Manage. 22 (2013) 253–268.
[9] M. Golias, A bi-objective berth allocation formulation to account for vessel handling time uncertainty, J. Maritime Econ. Logistics 13 (2011) 419–441.
[10] M. Golias, I. Portal, D. Konur, E. Kaisar, G. Kolomvos, Robust berth scheduling at marine container terminals via hierarchical optimization, Comput. Operat. Res. 41 (2014) 412–422.
[11] X. Guo, S.Y. Huang, Dynamic space and time partitioning for yard crane workload management in container terminals, Transp. Sci. 46 (1) (2012) 134–148.
[12] X. Guo, S.Y. Huang, W.J. Hsu, M.Y.H. Low, Dynamic yard crane dispatching in container terminals with predicted vehicle arrival information, Adv. Eng. Inform. 25 (3) (2011) 472–484.
[13] A.H. Gharehgozli, Y. Yu, R. de Koster, J.T. Udding, An exact method for scheduling a yard crane, Eur. J. Operat. Res. 235 (2) (2014) 431–447.
[14] A. Gutierrez, L. Dieulle, N. Nacima Labadie, N. Nubia Velasco, A multi-population algorithm to solve the VRP with stochastic service and travel times, Comput. Indust. Eng. 125 (2018) 144–156.
[15] C. Güven, D. Türsel Eliiyi, Modelling and optimisation of online container stacking with operational constraints, Maritime Policy Manage. (2018), https://doi.org/10.1080/03088839.2018.1450529.
[16] X.-L. Han, Z.-Q. Lu, L.-F. Xi, A proactive approach for simultaneous berth and quay crane scheduling problem with stochastic arrival and handling time, Eur. J. Operat. Res. 207 (2010) 1327–1340.
[17] J.L. He, Y.F. Huang, W. Yan, Yard crane scheduling in a container terminal for the trade-off between efficiency and energy consumption, Adv. Eng. Inform. 29 (1) (2015) 59–75, https://doi.org/10.1016/j.aei.2014.09.003.
[18] J.L. He, Y.F. Huang, W. Yan, Integrated internal truck, yard crane and quay crane scheduling in a container terminal considering energy consumption, Expert Syst. Appl. 42 (5) (2015) 2464–2487.
[19] K.H. Kim, J.S. Kang, K.R. Ryu, A beam search algorithm for the load sequencing of outbound containers in port container terminals, OR Spectrum 26 (2004) 93–116.
[20] K.H. Kim, K.Y. Kim, An optimal routing algorithm for a transfer crane in portcontainer terminals, Transp. Sci. 33 (1999) 17–33.
[21] K.Y. Kim, K.H. Kim, Heuristic algorithms for routing yard-side equipment forminimizing loading times in container terminals, Naval Res. Logistics 50 (2003) 498–514.
[22] D.-H. Lee, H.Q. Wang, L.X. Miao, Quay crane scheduling with non-interference constraints in port container terminals, Transp. Res. Part E: Logistics Transp. Rev. 44 (2008) 124–135.
[23] C. Li, Z.Q. Lu, X.L. Han, Y.J. Zhang, L. Wang, Integrated scheduling of a container handling system with simultaneous loading and discharging operations, Eng. Optimiz. 48 (3) (2016) 397–414.
[24] W.K. Li, Y. Wu, M.E.H. Petering, M. Goh, R. de Souza, Discrete time model and algorithms for container yard crane scheduling, Eur. J. Operat. Res. 198 (1) (2009) 165–172.
[25] D.Y. Lin, C.W. Chiang, The storage space allocation problem at a container terminal, Maritime Policy Manage. 44 (6) (2017) 685–704.
[26] R. Moorthy, C.-P. Teo, Berth management in container terminal: the template design problem, OR Spectrum 28 (4) (2006) 495–518.
[27] A. Narasimhan, U.S. Palekar, Analysis and algorithms for the transtainer routing problem in container port operations, Transp. Sci. 36 (2002) 63–78.
[28] W.C. Ng, Crane scheduling in container yards with inter-crane interference, Eur. J. Operat. Res. 164 (1) (2005) 64–78.
[29] M. Sislioglu, M. Celik, S. Ozkaynak, A simulation model proposal to improve the productivity of container terminal operations through investment alternatives, Maritime Policy Manage. (2018), https://doi.org/10.1080/03088839.2018.1481544.
[30] R. Stahlbock, S. Voβ, Operations research at container terminals: a literature update, OR Spectrum 30 (1) (2008) 1–52.
[31] D. Steenken, S. Voβ, R. Stahlbock, Container terminal operation and operations research-a classification and literature review, OR Spectrum 26 (1) (2004) 3–49.
[32] Duygu Taş, N. Dellaert, T. Van Woensel, T. de Kok, Vehicle routing problem with stochastic travel times including soft time windows and service costs, Comput. Operat. Res. 40 (1) (2013) 214–224.
[33] Duygu Taş, N. Dellaert, T. Van Woensel, T. de Kok, The time-dependent vehicle routing problem with soft time windows and stochastic travel times, Transp. Res. Part C: Emerging Technol. 48 (2014) 66–83.
[34] C. Zhang, Y.W. Wan, J.Y. Liu, R. Linn, Dynamic crane deployment in container storage yards, Transp. Res. Part B: Methodol. 36 (6) (2002) 537–555.
[35] L. Zhen, Container yard template planning under uncertain maritime market, Transp. Res. Part E: Logistics Transp. Rev. 69 (2014) 199–217.
[36] L. Zhen, Tactical berth allocation under uncertainty, Eur. J. Operat. Res. 247 (2015) 928–944.
[37] L. Zhen, L.H. Lee, E.P. Chew, A decision model for berth allocation under uncertainty, Eur. J. Operat. Res. 212 (2011) 54–68.
[38] P. Zhou, H. Kang, Study on berth and quay-crane allocation under stochastic environments in container terminal, Syst. Eng.- Theory Practice 28 (1) (2008) 161–169.