

The Integrated Optimization of Quay Crane-Yard Truck Scheduling in the Container Terminal with Uncertain Factors

Yiqin Lu^{1,2}, Youfang Huang¹ and Bin Yang¹

¹*Logistics Research Center, Shanghai Maritime University, Shanghai 200135, China;*

²*School of Economics & Management, Shanghai University of Electric Power, Shanghai 200090, China
lu_yiqin@163.com*

Abstract

The key to improve the container terminal efficiency is the integrated optimization of the quay crane (QC) and the yard truck (YT) scheduling, which is normally settled separately and considered in a certain condition in classical literatures. To improve the operation efficiency and simulate the practical operation, a PSO-based integrated QC-YT scheduling optimization model with uncertain factors is established in this research, considering two uncertain factors of YT travel speed and unit time of QC loading/discharging operation that affect the operating efficiency of the terminal greatly. The goal is the minimal operated time of QCs with the coordination of YTs. To solve this difficult combinatorial problem, the PSO algorithm is developed. PSO is evaluated for combinatorial problems with uncertain factors, which represents a new application of PSO. Numerical experiments show that the model of this research gives systemic simulation for the scheduling process with uncertain factors. And the results are better than model without uncertainties in terms of the accuracy and stability.

Key words: *integrated optimization of QC-YT scheduling, model with uncertain factors (MUF), the Particle Swarm Optimization (PSO) algorithm*

1. Introduction

Container transportation has gradually become the main direction of the development of shipping today. With the ship's large-scale, specialization and modernization, the container terminal enterprises are facing competition from the global market and the challenge of shortening operation time, satisfying the demand of the customers, reducing cost, etc., [1] (Yan, Zhu and He, 2014). Increased reliance on maritime trade of the world necessitates operation efficiency growth in container terminals and other container port related equipment such as yard cranes (YCs), QCs and YTs. Thus, any operational efficiency is constrained by the bottlenecks residing at the various points of the terminal operation, particularly at the quayside operation. So the integration of QC-YT is of great importance to the efficiency improvement in the container terminal.

The remaining parts of this research are as follows. Section 2 presents a brief review of the literature on the problem of the integrated optimization of QC-YT scheduling in the container terminal with uncertain factors. Problem definition and formulation are described in Section 3 in detail. The PSO algorithm is developed to solve the problem in Section 4, and tested by numerical experiments in Section 5. Finally, the research is concluded in Section 6.

2. Literature Review

Most researches were conducted on the optimization of QCs and YTs independently, including: QCs allocation and dispatch optimization [2-7] (Chen, *et al.*, 2011; Legato, *et al.*, 2012; Chung and Choy 2012; Lee, *et al.*, 2008; Meisel and Bierwirth 2011; Moghaddam, *et al.*, 2009) and YTs dispatch and route optimization [8-12] (Yan and Tao 2006; Zeng and Yang 2008; Zhang, *et al.*, 2009; Lee, *et al.*, 2009; Guo, *et al.*, 2001). Since these two steps influence each other, it is necessary to study integrated optimization of QCs and YTs in container terminal operation. Therefore, some researchers started to study the integrated optimization of these two steps. Bish (2003) [13] considered QCs operation sequence and YTs dispatch at the same time; Chen, *et al.*, (2008) [14] established the integrated optimization model and designed a tabu search algorithm; Zeng and Yang (2010) [15] built an integrated dispatch model and designed a hybrid optimization algorithm; Ji and Jin (2007) [16] considered the transportation time of YTs and operation time of QCs and established the optimization model with the solution of evolution algorithm.

All these researches are a determined one without uncertain factors. However, there are a lot of uncertainties in the integrated optimization of QC-YT scheduling, such as YT speed, QC loading/discharging time, waiting time of YTs at QCs, *etc.* Only a few researches considered uncertain factors on the integrated scheduling of QCs and YTs. Zhou and Kang [17] (2008) established a berth-QC allocation model under the random environment according to the randomness of ship arriving time and operation time with the object to minimize the average waiting time of ships. Yang, C.X. (2011) [18] studied the berth and quay crane scheduling problem under uncertainty environments. The berth and quay crane scheduling approach, the rescheduling policies, the generative rescheduling approach, and the repair-based rescheduling approach were discussed. However, in these researches decision makers' attitude towards risks and decision-making preference were brought into all stages of container operation, based on which the uncertain model of the integrated scheduling was established. Lu and Le (2014) [19] considered the problem of the integrated optimization of container terminal scheduling with uncertain factors, but the research focused on the integrated optimization of YTs and YCs. However, it is still an unsolved problem to effectively coordinate the operation of QCs and YTs to improve the container terminal operation efficiency.

Particle swarm optimization (PSO) is a population (called a swarm) based stochastic optimization technique that optimizes a problem by iteratively trying to improve a candidate solution (called particles) with regard to a given measure of quality. It is originally attributed to Eberhart and Kennedy (1995) [20], inspired by social behavior of bird flocking or fish schooling. The algorithm was simplified and observed to be performing optimization. They described many philosophical aspects of PSO and swarm intelligence. As an optimization technique, PSO has obtained much attention during the past decade. It is gaining popularity, especially because of the speed of convergence and the fact that it is easy to realize [21] (Li, *et al.*, 2008). An extensive survey of PSO applications was made by Poli (2008) [22]. Research of applications for combinatorial optimization problems can be found in the recent literature [23-25] (Wang, *et al.*, 2011; Souza and Goldberg, 2006; Hu, 2011). In recent years, PSO has been successfully applied in many research and application areas. It was demonstrated that PSO gets better results in a faster, cheaper way compared with other methods (Hu, 2011) [26]. Another reason that PSO is attractive is that there are rare parameters to adjust. With slight variations, PSO has been used for approaches that can be used across a wide range of applications, such as nonlinear optimization, artificial neural network training, fuzzy control, *etc.*

The main innovations of this research are as follows: (1) Modeling. This includes: (i) developing the model with the object to minimize the operation time of QC and YT operation. The model of this research significantly improves the solution quality compared with the models that consider QC operation and YT operation separately; (ii) the integration of two types of equipment (QC and YT) that incorporate in uncertain circumstances, which is considered by few literature, is taken into account for quayside scheduling. More realistic operational factors are considered. The uncertain factors of YT speed and unit time of QC loading/discharging operation are considered in this research for the first time as they have been ignored in the determined context. The model with the aforementioned uncertain factors is robust and obviously closer to the real work environment; (2) Algorithm. The PSO algorithm is applied to solve the integrated optimization of QC-YT scheduling with uncertain factors and the results can be got in permitted time with stability and satisfactory. Compared with the results in the determined context, the results of uncertain factors are better and more stable in allowable CPU time.

3. Problem Description and Proposed Model

3.1 Problem Description

The container terminal is the frontier of the port. The main functions of the container terminal include loading and discharging containers, temporarily stacking and storing, containers acceptance and picking up and other basic functions. Figure 1 is the illustration of the integrated QC-YT operation in the container terminal. When ships to be loaded and discharged arrive at the container terminal at the same time, the QCs deliver the containers to be discharged to the waiting YTs; the YTs deliver the containers to the yard and then the containers will be discharged by the yard cranes. The YTs fetch containers to the ship to be loaded and then queue at QCs for service. Finally the empty YTs go back to ships to be discharged. All of these constitute a complete QC-YT operation process. Ensuring that all the loading and discharging tasks are finished, how to minimize the operation time is the object of this research.

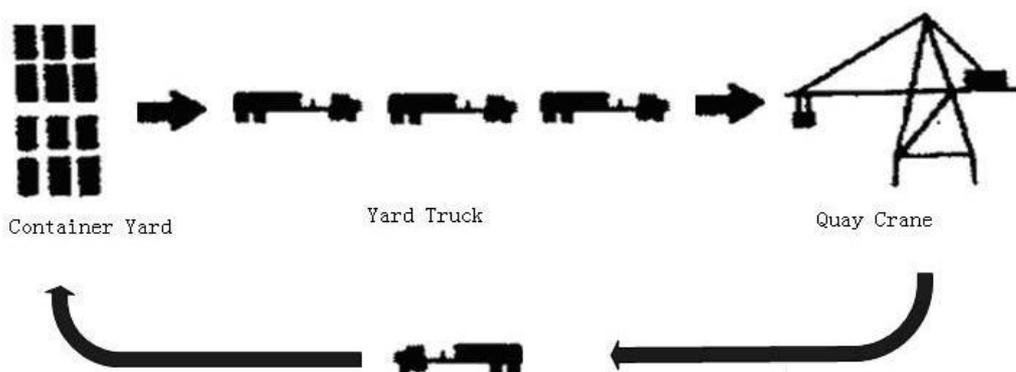


Figure 1. Illustration of the Integrated QC-YT Operation

3.2 Assumptions and Notations

In this Section, we first list three main assumptions of our model: (1) the containers are all 40ft standard containers. Loading/discharging tasks are known. There are K QCs and the locations of QCs are known. There are n_1 loading tasks and n_2 discharging tasks. The locations of the container yard according to all the tasks are known, not considering the waiting time of the YTs in the yard; (2) all YTS select their optimal path to complete their tasks, without considering the traffic problem, *i.e.*, the driving

distance for each YT to complete the assigned task is certain. All tasks are assigned to N YTs, and the driving speeds of loading and empty of YT l are the random variables η_l and ξ_l . The unit time of QC loading/discharging operation is also a random variable, denoted by τ ; (3) the operation time for each QC is calculated from the first task started to the last task completed. The total time of QCs is calculated from the time when the task is started by the first QC to the time when the tasks are completed by the last QC.

The problem in this research is abstracted as a network. We assume that there are n instructions of tasks allocated to QCs and YTs. Each instruction is corresponding to the loading/discharging task from the container yard to QCs. We denote CH as sets of loading tasks and UL as discharging tasks. They are corresponding to n nodes $V = \{1, 2, \dots, n\}$ in the network.

In order to indicate the loading and discharging tasks in the network, we introduce the concept of virtual task points. A virtual task point represents a command. It can be seen that it is the empty YT that runs between any two virtual task points. In order that the YT loading/discharging tasks can be formed as a loop, we assume there is a virtual task point O . The distance from O to any point is 0, that is, all YTs start the tasks from the virtual task point O . For more details, please refer to Lu and Le [19].

d_{ij} : The distance between any two virtual nodes i, j , i.e., the distance from the discharging virtual task point to the loading virtual task point, and generally $d_{ij} \neq d_{ji}$.

l : The order of YTs, $l \in \langle N \rangle$ i.e., $1 \leq l \leq N$.

k : The order of QCs, $k \in \langle K \rangle$ i.e., $1 \leq k \leq K$.

J_l : The task set of YT l , obviously $\prod_{l=1}^N J_l = V$;

Q_k : The task set of QC k , obviously $\prod_{k=1}^K Q_k = V$;

3.3 Parameters

η_l : Random variable, speed of loading YT l ;

ξ_l : Random variable, speed of empty YT l ;

τ : Random variable, unit time of QC loading/discharging operation;

T_i^l : Random variable, the driving time of YT l which can be got through distance divided by η_l ;

S_k, F_k : The time when QC k starts and finishes task;

s_k^i, f_k^i : The time when QC k starts and finishes task i , $i \in Q_k$;

g_l^i : The time when YT l arrives at QCs after being allocated task i to;

$\tilde{s}_l^i, \tilde{f}_l^i$: The time when YT l starts and finishes task i .

3.4 Decision Variables

X_{ij}^l : X_{ij}^l states whether YT l runs from the discharging point of the task number

i to the loading point of the task number j , *i.e.*:

$$X_{ij}^l = \begin{cases} 1, & \text{YT } l \text{ completes task } j \text{ after it completes } i \\ 0, & \text{otherwise} \end{cases}$$

When $X_{ii}^l = 1$, YT l completes task i only and will not accept another task.

Y_{ij}^k : Y_{ij}^k states whether QC k completes task j after completing task i , *i.e.* :

$$Y_{ij}^k = \begin{cases} 1, & \text{QC } k \text{ completes task } j \text{ after it completes } i \\ 0, & \text{otherwise} \end{cases}$$

When $Y_{ii}^k = 1$, QC k completes task i only and will not accept another task.

3.5 Model

The object of the model is to minimize the total time for QCs to complete all the tasks with the coordination of YTs. The whole model is written as the following:

$$\min \bar{t} = \max_{k \in \langle K \rangle} \{ F_k - S_k \} \quad (1)$$

s. t.

$$\sum_{l \in \langle N \rangle} \sum_{j \in V} (X_{ij}^l + X_{ji}^l) \geq 1 \quad \forall i \in V, \quad (2)$$

$$\sum_{l \in \langle N \rangle} \sum_{j \in V} X_{ij}^l \leq 1 \quad \forall i \in V, \quad (3)$$

$$\sum_{l \in \langle N \rangle} \sum_{i \in V} X_{ij}^l \leq 1 \quad \forall j \in V, \quad (4)$$

$$\sum_{l \in \langle N \rangle} \sum_{j \in V} (Y_{ij}^k + Y_{ji}^k) \geq 1 \quad \forall i \in Q_k, \quad (5)$$

$$\sum_{l \in \langle N \rangle} \sum_{j \in V} Y_{ij}^k \leq 1 \quad \forall i \in Q_k, \quad (6)$$

$$\sum_{l \in \langle N \rangle} \sum_{i \in V} Y_{ij}^k \leq 1 \quad \forall j \in Q_k, \quad (7)$$

$$X_{ij}^l, Y_{ij}^k \in \{0, 1\} \quad \forall i, j \in V, l \in \langle N \rangle, k \in \langle K \rangle \quad (8)$$

$$F_k = \max_{i \in Q_k} \{ f_k^i \} \quad \forall k \in \langle K \rangle, \quad (9)$$

$$S_k = \min_{i \in Q_k} \{ s_k^i \} \quad \forall k \in \langle K \rangle, \quad (10)$$

$$f_k^i = s_k^i + \tau \quad \forall k \in \langle K \rangle, i \in Q_k, \quad (11)$$

$$s_k^j = \max \{ f_k^i, g_l^j \} \quad \forall k \in \langle K \rangle, \text{ when } Y_{ij}^k = 1, \sum_{i \in V} X_{ij}^l = 1, \quad (12)$$

$$\tilde{f}_l^i = \tilde{s}_l^i + T_l^i + (s_k^i - g_l^i) + \tau \quad \forall k \in \langle K \rangle, l \in \langle N \rangle, i \in Q_k \cap J_l, \quad (13)$$

$$\tilde{s}_l^j = \tilde{f}_l^i + \frac{d_{ij}}{\xi_l}, \quad \forall l \in \langle N \rangle, \text{ when } X_{ij}^l = 1, \quad (14)$$

$$g_l^i = \tilde{f}_l^i + T_l^i \quad \forall i \in CH, \quad g_l^i = \tilde{s}_l^i \quad \forall i \in UL. \quad (15)$$

Objective function (1) is means the total time of QCs completing all tasks, and the objective is to minimize the value. At this time, the entire loading/discharging process may not been completed. The YT is still delivering the last container to the container yard, but the quayside tasks are completed. Constraint (2) makes sure that all tasks are completed, *i.e.*, YTs complete one virtual task once at least. (3) and (4) restrict that each task is completed by one truck once, and a task isn't repeated. (3) shows the YT has one following task at most after completing task i . (4) shows the YT has one previous task at most before completing task i . Constraints (5) to (7) ensure that each task is allocated to a QC once. Constraint (6) shows the QC has one following task at most after completing the task i . (7) shows the QC has one previous task at most before completing the task i . Constraint (8) indicates that the value range of independent variable is discrete variables from 0 to 1. Constraint (9) defines the relationship between the time for a QC to finish work and finish each work. Constraint (10) defines the relationship between the time for a QC to start work and start each work; Constraint (11) defines the relationship between the time for a QC to start and finish each task. Constraint (12) means the time for a QC to start the next task after completing a task. Constraint (13) represents the time of a YT to complete a loading/discharging task. It includes the sum of the driving time of YT (T_i^l) + YT waiting time at QCs ($s_k^i - g_l^i$) + QC operation time (τ). Therefore, the relationship between the time for a YT to start and finish a task is obtained.

$\frac{d_{ij}}{\xi_1}$ in constraint (14) represents the time of the empty YT running between two virtual task points. So the relationship between the time when a task is finished and when the next task is started can be got. Constraint (15) represents the relationship between g_l^i and \tilde{s}_l^i .

4 PSO Algorithms

4.1 Model Assumption

In the models of classical literatures, the uncertain values in this research were set by exact values (usually average). In this research, the driving time for each YT to complete the task basically follows normal distribution $N(\mu, \sigma)$ according to the statistics of Shanghai Port practical operation. Its mean value μ is calculated from distance divided by average speed which is estimated from statistical sampling. The statistics also indicate that there are no major differences between loading and empty YTs in average speed. However, the loading truck is more susceptible to external impact and thus the standard deviation σ is higher. In addition, the unit time of QC loading/discharging operation basically follows normal distribution and the standard deviation is relatively higher.

According to the operation characteristics of the container terminal, the locations of the QCs are normally fixed at the quayside and will not move widely. So QCs are limited in selecting tasks. Currently, the QC operation sequence is subject to the sequence of loading/discharging operation of YTs waiting at the QCs. In another word, the QC will load/dischARGE the container delivered by the first YT in the queue. According to the fundamentals of the queuing theory, each QC is regarded as a service counter. The YTs queue in arriving order when QCs are not available. When a QC completes a task, it will serve the first YT in the queue.

4.2 Parameters

Let S be the particle swarm population size. Let n be the dimension of each particle. $x = (x_{k1}, x_{k2}, \dots, x_{kn})$ indicates the position of each particle. The position range for each component is an integer from 1 to N , indicating the number of the tasks assigned to YTs. For each YT, it takes the task of the first number as the starting task. After completing each task, the YT takes the closest task as the next one based on greedy rule. The speed of each particle is denoted as $v = (v_{k1}, v_{k2}, \dots, v_{kn})$. The initial state is given by a random value. For more details, please refer to Lu and Le [19].

4.3 Calculating Process

The whole process of PSO is illustrated in Figure 2. The initial position of the particles is given in a random way; the fitness function value is calculated according to Figure 3; the update and decoding of the speed and position of the particles are respectively described in step 2 and step 3; the calculation usually stops when the objective function value doesn't change obviously.

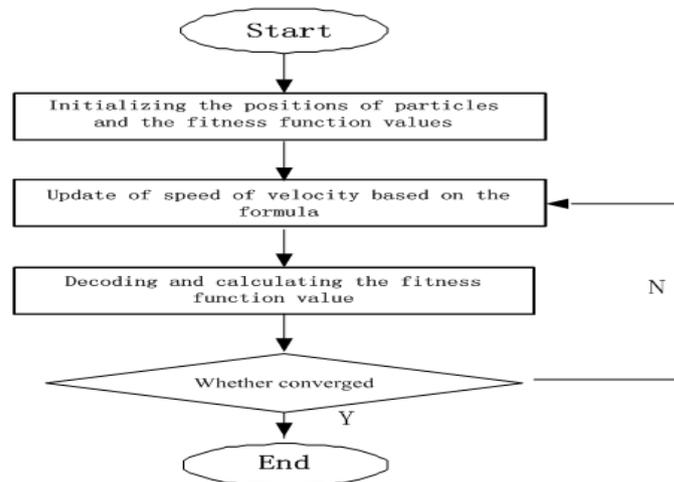


Figure 2. PSO Process

Step 1: Set the initial statement of PSO process. We choose the positions of particles at random and calculate the fitness function values.

Step 2: The update of the velocity and position. A basic variant of the PSO algorithm works by having a population (called a swarm) of candidate solutions (called particles). These particles are moved around in the search-space according to a few simple formulae. The movements of the particles are guided by their own best known position in the search-space as well as the entire swarm's best known position. After being discovered, the improved positions will then come to guide the movements of the swarm. The process is repeated and by doing so it is hoped, but not guaranteed, that a satisfactory solution will eventually be discovered.

During each step of iterative process, the optimal value of the particles searched by their own best known position in the search-space is denoted as $f_k = (f_{k1}, f_{k2}, \dots, f_{kn})$, $k = 1, 2, \dots, K$. The optimal value searched by the entire swarm's best known position is denoted as $f_{best} = (f_{b1}, f_{b2}, \dots, f_{bn})$. f_k and f_{best} decide the speed and velocity of the next step. We update the position and velocity of particles as follows:

$$v_{kd}^{i+1} = wv_{kd}^i + c_1r_1(f_{kd}^i - x_{kd}^i) + c_2r_2(f_{bd}^i - x_{kd}^i)$$

$$x_{kd}^{i+1} = x_{kd}^i + \rho v_{kd}^{i+1}$$

The coefficients w , c_1 and c_2 define how much a particle trust in its previous movement, own history and in the whole set of particles, respectively (Engelbrecht, 2007) [25]. w is the inertia weight coefficient which maintains the original speed. The value of w is determined by linear reduction strategy, *i.e.*

$$w = w_{\max} - (w_{\max} - w_{\min}) \frac{\text{Current iteration number}}{\text{Total iteration number}}$$

Let $c_1=c_2=2$. The terms r_1 and r_2 are random numbers drawn from a uniform distribution during $[0,1]$. ρ indicates the step length, which is usually called as the restriction factor and set to 1.

Step 3: Decoding process and calculating the fitness function value. Because the position of each particle stands for the number of the tasks allocated to YTs which must be an integer between 1 to N, the decoding process is required after the position is updated by each step of iteration in PSO algorithm. We decode each component of particle positions to the nearest integer [19].

After each step of iteration, the fitness function value of each particle is calculated according to the new position of each particle. The model of this research is based on the fundamentals of queuing theory to calculate the loading/discharging time of QCs and the driving and waiting time of YTs. Figure 3 indicates the calculation process of fitness function value.

The YT scheduling is got after being given the position of each particle. We calculate the total time of QCs by using the Monte Carlo method .The total time is calculated from the time when the first YT arrives at the quayside to the time when the QCs complete all the loading/unloading tasks. As the driving speed of the YTs and the loading/unloading time of the QCs are uncertain variables, the simulation method is applied to estimate the function value during the calculation.

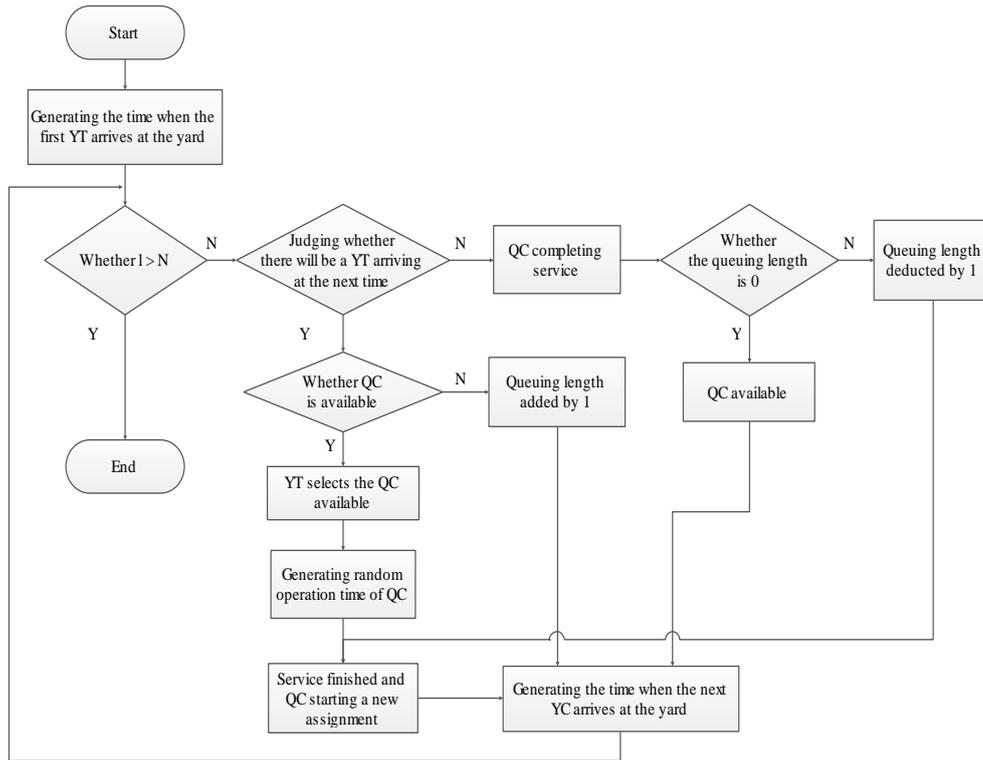


Figure 3. Fitness Function Value Calculation

5. Numerical experiments and Results

In this section, we simulate the practical operation of the terminal. The numerical experiment involves 2 QCs, 100 loading/discharging tasks and 10 YTs. We suppose that the 2 QCs serve for the same ship at the same time and they are very close to each other. The distance between them is ignored. All the YTs have the same capacity. Uncertain variables are: speed of loading and empty YTs and unit time of QC loading/discharging operation. The YTs will queue up in a line when arriving at the QCs and choose the QC available for loading/discharging tasks. According to the actual operation data of Shanghai Port, the average speed of YTs $v=10\text{m/s}$ is selected and the average driving time $t = \frac{S}{v}$ follows the normal distribution. Standard deviation of loading YT is 2% of the driving time. Standard deviation of empty YT is 1% of the driving time. Average unit time for loading/discharging operation of QCs is 100 seconds with standard deviation of 3%. We allocate the 100 tasks to the 2 QCs and 10 YTs.

The larger the scale of the population is selected, the higher the calculation accuracy can be got, but more time the calculation will spend. We use MATLAB software and CPU Intel(R) Core(TM) 2 Duo to test the numerical experiments. Hundreds of tests show that the algorithm accuracy will be maintained at a comparatively high level within acceptable time with the population size $S=10$. The tests also show that selection of parameters $w_{\max}=1$, $w_{\min}=0.5$ gives relatively ideal convergent precision and speed. We observe 1000-step iterative objective function values of MUF and model without uncertainties, shown in Figure4. The optimal results in Figure 4 show that the change of objective function of MUF is not strictly decreased. The reason is that its fitness function is random, but there is an overall trend of decreasing. After about 900 steps of iterations, variation of objective

function value is obviously reduced, and approaches to the optimal value with algorithm convergence gradually. The convergence of the PSO algorithm is not affected by the uncertain factors. Thus, it is very effective in solving integrated scheduling of QC-YT for the shortest operation time with uncertain factors, and the model is obviously closer to the actual operation. Optimal operation time decreases from the initial state of about 250 minutes to finally about 81.2 minutes, greatly improving the efficiency. Therefore, the assignment can always get nearly optimal results while considering uncertain factors.

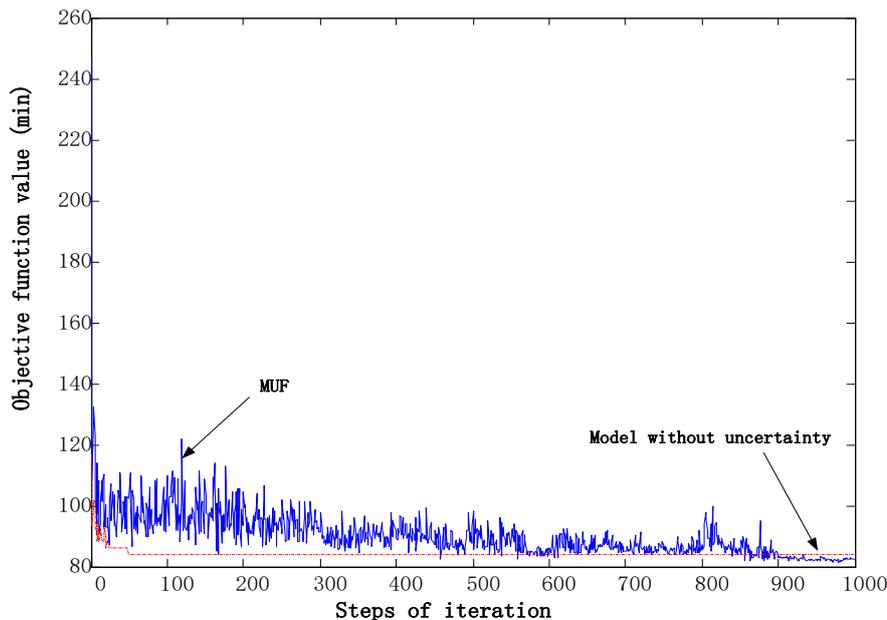


Figure 4. Comparison of Model Objective Function Values between MUF and the Model without Uncertainty

Figure 4 also shows the comparison of model objective function values between MUF and the model without uncertainty. The optimal value of MUF is about 81.2 minutes and 83.9 minutes of the model without uncertainty. Although the optimal objective function values of the two models are very close, the optimal task assignments in Table 1 are different completely. In MUF, as the objective function values of the last 100 steps of iteration do not change obviously, the corresponding optimal task assignment won't change. Therefore, MUF can get a more stable solution and guarantee the objective function to achieve optimal results in the uncertain circumstance in this research. And the solution of the model without uncertainty can only ensure the optimal value in determined circumstances. However, when some parameters change in the actual operation, the model without uncertainty may lose efficiency.

Table 1. Comparison of Task Assignments between MUF and the Model without Uncertainty

Container Number	Model	Task Assignment																										
1	the model without uncertainty	5	6	4	7	1	9	9	1	2	8	1	1	2	5	1	6											
	MUF	6	1	7	2	8	2	9	5	1	0	9	1	6	3	3	5	0										
2	the model without uncertainty	7	1	9	3	9	1	9	3	9	4	2	7	3	0	4	3	8	4	8	8	3						
	MUF	5	2	6	7	3	7	1	7	1	9	9	2	6	2	1	7	0	8	3	2	6	7	3	6			
3	the model without uncertainty	6	5	7	1	8	1	8	2	9	4	3	8	0	9	1	1	7	6	2	5	8						
	MUF	9	1	4	1	2	5	1	0	4	6	4	14	7	6	7	0	9	4	0	2	9	5	3	0	2	5	3
4	the model without uncertainty	6	4	8	23	3	7	4	15	5	7	2	8	1	5	9	0	3	6	3	5	3	5	5	7	1	2	9
	MUF	5	4	6	1	4	3	3	4	15	5	7	2	8	1	5	9	0	7	1	9	3	7	5	5	7	1	2
5	the model without uncertainty	3	6	3	5	3	5	5	3	4	5	3	4	71	1	9	3	7	5	5	7	1	2	9				
	MUF	7	3	6	4	6	5	5	8	8	7	2	77	6	4	4	9	5	5	8	8	7	2					
6	the model without uncertainty	7	2	7	4	7	6	1	5	3	4	85	5	8	5	5	4	3	6	5	0	6	6					
	MUF	3	6	3	5	90	3	3	1	1																		
7	the model without uncertainty	3	8	4	7	2	6	6	1	5	4	5	97	9	9	5	0	8	6	4	0	2	8	6	4			
	MUF	2	8	2	8	3	7	2	6	4	5	1	5	94	8	6	0	1	7	8	7	6	8	4	1	3		
8	the model without uncertainty	2	8	4	7	2	7	1	6	99	9	2	2	6	4	4	6	4										
	MUF	4	8	8	7	1	6	6	97	6	7	4	4	9	4	8	7											
9	the model without uncertainty	10	3	9	4	8	2	8	2	8	2	5	1	5	0	7	2	0	8	0	3	1	0	7	9	9	3	
	MUF	10	2	9	3	9	2	8	8	2	8	3	5	0	9	8	2	8	2	8	3	5						
10	the model without uncertainty	0	9	8	2	2	5	9	3	8	1	2	6	9														
	MUF	0	9	8	2	2	5	9	3	8	1	2	6	9														

6. Conclusions

1. With the progress of economic globalization and transport containerization, the container terminals are now facing an increasing competitiveness. The scale economy in ship sizes also results in the challenge of shortening time for ships in terminals. Container terminal operation system is a complex system with the cooperation of QCs and YTs at the quayside. Effective integration of QCs and YTs can greatly improve the efficiency and shorten time for loading and discharging so that the container terminal will be more competitive. As there are many uncertainties in the terminal operation, the integrated optimization of QCs and YTs scheduling with uncertain factors can help the terminal to survive in a new economic environment.

2. This research presents a PSO-based integrated QC-YT scheduling optimization model with uncertain factors which consist of YT travel speed and unit time of QC loading/discharging operation. PSO is evaluated for combinatorial problems with

uncertain factors, which represents a new application of PSO. PSO gets better results in a faster, cheaper way compared with other methods. Another reason that PSO is attractive is that there are rare parameters to adjust. Numerical experiments show that the MUF gives systemic simulation for the scheduling process with uncertain factors, and the results are better than the certain model in terms of the accuracy and stability. Our results suggest potential uses in the integrated optimization of QC-YT scheduling and the results can be applied into the operation of Shanghai port in practice. For example, the work efficiency of the container terminals can be improved and the waiting time of the YTs can be shortened.

3. However, the yard cranes are very crucial to improve the yard operation efficiency and also of great importance to improve the overall performance of the container terminal. Therefore, the study on the integrated optimization of QC-YT-YC scheduling will direct future research.

Acknowledgements

We acknowledge the editor and the anonymous reviewers for insightful suggestions on this work.

This study was supported by the following fund projects:

National Natural Science Foundation of China (71171129).

Key Project of Shanghai Science and Technology Committee (12510501600).

Ability-improved Project of Engineering Center of Shanghai Science and Technology Committee (14DZ2280200).

Doctoral innovation fund project of Shanghai Maritime University (yc2012062).

References

- [1] W. Yan, Y. S. Zhu and J. L. He, "Performance analysis of a new type of automated container terminal", *International Journal of Hybrid Information Technology*, vol. 2, no. 7, (2014), pp. 237-248.
- [2] J. H. Chen, Lee and J. X. Cao, "Heuristics for quay crane scheduling at indented berth", *Transportation Research*, vol. 6, no. 47, (2011), pp. 1005-1020.
- [3] P. Legato, R. Trunfio and F. Meisel, "Modeling and solving rich quay crane scheduling problems", *Computers and Operations Research*, vol. 9, no. 39, (2012), pp. 263-278.
- [4] S. H. Chung and K. L. Choy, "A modified genetic algorithm for quay crane scheduling operations", *Expert Systems with Applications*, vol. 4, no. 239, (2012), pp. 4213-4221.
- [5] D. H. Lee, H. Q. Wang and L. X. Miao, "Quay crane scheduling with non-interference constraints in port container terminals", *Transportation Research Part E*, vol. 1, no. 44, (2008), pp. 124-135.
- [6] F. Meisel and C. Bierwirth, "A unified approach for the evaluation of quay crane scheduling models and algorithms", *Computer & Operations Research*, vol. 3, no. 38, (2011), pp. 683-693.
- [7] T. Moghaddam, A. Makui and S. Salahi, "An efficient algorithm for solving a new mathematical model for a quay crane scheduling problem in container ports", *Computers and Industrial Engineering*, vol. 1, no. 56, (2009), pp. 241-248.
- [8] Z. Yan and D. X. Tao, "Technology of truck process system of container terminals based on dynamic optimized combination", (2006).
- [9] Q. C. Zeng and Z. Z. Yang, "A scheduling model and Q-learning algorithm for yard trailers at container terminals", *Journal of Harbin Engineering University*, vol. 1, no. 29, (2008), pp. 1-4.
- [10] R. Y. Zhang, Y. Y. Won and M. Ilkyeong, "A Reactive Tabu search algorithm for the multi-depot container truck transportation", *Transportation Research Part E*, vol. 6, no. 45, (2009), pp. 904-914.
- [11] D. H. Lee, J. X. Cao, Q. X. Shi and J. H. Chen, "A heuristic algorithm for yard truck scheduling and storage allocation problems", *Transportation Research Part E*, vol. 5, no. 45, (2009), pp. 810-820.
- [12] X. Guo, S. Y. Huang, W. J. Hsu and M. Y. H. Low, "Dynamic yard crane dispatching in container terminals with predicted vehicle arrival information", *Advanced Engineering Informatics*, vol. 3, no. 25, (2011), pp. 472-484.
- [13] E. K. Bish, "A multiple-crane-constrained scheduling problem in a container terminal", *European Journal of Operational Research*, vol. 1, no. 144, (2003), pp. 83-107.
- [14] L. Chen, N. Bostel, P. Dejax, J. Cai and L. Xi, "A tabu search algorithm for the integrated scheduling problem of container handling systems in a maritime terminal", *European Journal of Operational Research*, vol. 1, no. 181, (2008), pp. 40-58.
- [15] Q. C. Zeng and Z. Z. Yang, "Integrating scheduling model and hybrid optimization algorithm for

- container terminals”, *Journal of systems engineering*, vol. 2, no. 25, (2010), pp. 264-270.
- [16] M. J. Ji and Z. H. Jin, “A united optimization of crane scheduling and yard trailer routing in a container terminal”, *Journal of Fudan University (Natural Science)*, vol. 4, no. 46, (2007), pp. 476-480.
- [17] P. F. Zhou and H. G. Kang, “Study on berth and quay crane allocation under stochastic environments in container terminal”, *Theory and Practice of Engineering System*, vol. 1, no. 28, (2008), pp. 161-169.
- [18] C. X. Yang, “Optimization of berth-quay crane scheduling under uncertainty environments in container terminals”, China: Dalian Maritime University, (2011).
- [19] Y. Q. Lu and M. L. Le, “The integrated optimization of container terminal scheduling with uncertain factors”, *Computers and Industrial Engineering*, vol. 3, no. 75, (2014), pp. 209-216.
- [20] R. C. Eberhart and J. Kennedy, “Particle Swarm Optimization”, *Proceedings of IEEE International Conference on Neural Networks IV*, (1995), 1942-1948.
- [21] H. Q. Li, L. Li, T. H. Kim and S. L. Xie, “An improved PSO-based of harmony search for complicated optimization problems”, *International Journal of Hybrid Information Technology*, vol. 1, no. 3, (2008), pp. 91-98.
- [22] R. Poli, “Analysis of the publications on the applications of particle swarm optimization”, *Journal of Artificial Evolution and Applications*, (2008), pp. 1-10.
- [23] J. Wang, Y. Cai, Y. Zhou, R. Wang and C. Li, “Discrete particle swarm optimization based on estimation of distribution for terminal assignment problems”, *Computers & Industrial Engineering*, vol. 4, no. 60, (2011), pp. 566-575.
- [24] G. Souza, E. F. Goldberg and M. C. Goldberg, “Particle swarm for the traveling salesman problem”, *Lecture Notes in Computer Science*, vol. 3906, (2006), pp. 99-110.
- [25] X. Hu, “PSO tutorial”, (2011), <http://www.swarmintelligence.org/tutorials.php>> Last accessed 20.06.11.
- [26] A. P. Engelbrecht, “Computational intelligence: An introduction (2nd.)”, John Wiley and Sons, (2007).

Authors



Yiqin Lu, Female, Chinese, Shanghai Maritime University doctoral student, teacher of Shanghai University of Electric Power, the research direction is the management information system of container terminal operation.

