TRANSPORT
ISSN 1648-4142 / eISSN 1648-3480
2021 Volume 36 Issue 6: 444–462
https://doi.org/10.3846/transport.2021.16169

OPTIMIZING THE CONTAINER TRUCK PATHS WITH UNCERTAIN TRAVEL TIME IN CONTAINER PORTS

Jiaming LIU[1], Bin YU[1,2], Wenxuan SHAN[1], Baozhen YAO[3*], Yao SUN[4]

[1]School of Transportation Science and Engineering, Beihang University, Beijing, China
[2]Advanced Innovation Center for Big Data and Brain Computing, Beihang University, Beijing, China
[3]State Key Laboratory of Structural Analysis for Industrial Equipment, School of Automotive Engineering, Dalian University of Technology, Dalian, China
[4]Tianjin Municipal Engineering Design and Research Institute, Tianjin, China

Submitted 26 September 2019; resubmitted 25 November 2019, 7 April 2020; accepted 5 July 2020

Abstract. The yard template problem in container ports determines the assignment of space to store containers for the vessels, which could impact container truck paths. Actually, the travel time of container truck paths is uncertain. This paper considers the uncertainty from two perspectives: (1) the yard congestion in the context of yard truck interruptions, (2) the correlation among adjacent road sections (links). A mixed-integer programming model is proposed to minimize the travel time of container trucks. The reliable shortest path, which takes the correlation among links into account is firstly discussed. To settle the problem, a Shuffled Complex Evolution Approach (SCE-UA) algorithm is designed to work out the assignment of yard template, and the A* algorithm is presented to find the reliable shortest path according to the port operator's attitude. In our case study, one yard in Dalian (China) container port is chosen to test the applicability of the model. The result shows the proposed model can save 9% of the travel time of container trucks, compared with the model without considering the correlation among adjacent links.

Keywords: container port, yard template, reliable shortest path, SCE-UA algorithm.

Abbreviations

AGV – automatic guided vehicle;
AVG – average result;
BR – best result;
CCE – competitive complex evolution;
CDF – cumulative distribution function;
CPLEX – IBM ILOG CPLEX optimizer;
CPU – central processing unit;
GDP – gross domestic product;
NON-CORR – non-correlation;
OCTP–UTT – optimizing container truck paths with uncertain travel time;
QC – quay crane;
RGS – route guidance system;
RSPP – reliable shortest path problem;
SCE-UA – shuffled complex evolution approach;
SD – standard deviation;
SWO – squeaky wheel optimization;
TEU – twenty-foot equivalent unit.

Introduction

Background

Compared with the growth rate of world GDP, the amount of container transportation has increased about three times (Meng *et al.* 2014). As lots of containers are stored and transhipped among ports every day, the turnaround time of containers is influenced by the operation efficiency of ports. It is essential to improve the operation efficiency and maximize the throughput because the profit of a port is relevant to the number of handled containers (Chang *et al.* 2010). There are many factors impacting the operation efficiency, such as the berth and yard operation (Jin *et al.* 2015). In the past decades, the operation efficiency of ports in the berth side has been improved significantly because of the advanced technologies (e.g., indented berths and the double forty-foot QCs) and management (berth allocation, QC assignment and scheduling) – Lee and Jin (2013), Zhen (2015). While the yard side may become a bottleneck that hinders the operation efficiency, especially

*Corresponding author. E-mail: *yaobaozhen@dlut.edu.cn*

in the ports that have a large number of QCs. Therefore, the management of the yard side is crucial to promote the port's competitiveness on global shipping market.

The yard template planning is a concept of planning the yard in container ports (Moorthy, Teo 2006), which is concerned with the assignment of yard storage locations (subblocks). It aims to minimize the total cost of moving containers from berth or gate to subblocks and vice versa. One way is to minimize the total distance of moving containers, but it could hardly improve the situation in practice. Although the travel distance is minimized, the container trucks may waste lots of time when there has congestion along the path. Actually, the yard congestion is common in reality. Nowadays, multi-level stacking is universal in yards with heavy traffic. It may lead to high concentration of activities within a small area and cause yard traffic congestion (Han *et al.* 2008). When there are too many container trucks running along a link or passing through a cross at the same time, the cruising speed of these trucks will be affected by each other. In this condition, the trucks have to slow down or even stop when they come across yard congestion. Moreover, the congestion on a link could easily transfer to adjacent links according to the traffic flow theory. That is, there exists correlation among adjacent links. The travel time of a container truck is uncertain with the consideration of yard congestion on some link and the impact of adjacent links. This is the reason why the optimized distance is not equivalent to the optimized travel time. In this paper, our motivation is to minimize the travel time and find a reliable shortest path for container trucks considering the uncertain travel time.

Literature review

There are abundant studies on container allocation, berth allocation and crane assignment (Zhang *et al.* 2003; Fan *et al.* 2012; Maloni, Paul 2013; Peng *et al.* 2016). Kim and Bae (1998) discussed to reallocate export containers to the best organization for loading vessels. They used a hierarchical approach to divide the problem into three sub-problems, bay matching, move planning and task sequencing. Zhang *et al.* (2003) studied how to allocate storage space in the yards considering the mixture of import and export containers. They divided the allocation problem into two steps: (1) all of the containers were placed in a determined storage block, (2) all the containers were allocated to minimize the total travel distance. Kim, K. H. and Kim, K. Y. (2007) presented a method to determine the minimum price for storing the containers in a yard. In addition, the storage charge urged the customers to store their containers only for a short time so as to relieve congestion. Zhen *et al.* (2011) proposed an integrated model, which considered berth allocation and yard template planning simultaneously, these two problems fit well with each other. Following the study in 2011, Zhen (2015) formulated a robust problem of berth allocation under uncertain environment. The factor of periodicity had been explicitly considered in the stochastic programming formulation model and the robust formulation model. Jin *et al.* (2015) proposed the yard crane profile that was used in an optimization model on storage deployment and management. Zhen *et al.* (2019) studied an integrated optimization problem on QC and yard truck scheduling in container terminals, which showed good results.

Besides, the transhipment tends to be increasingly important, both in contemporary and forthcoming future. Many researchers have studied transhipment management in container ports. Lee and Jin (2013) settled three tactical decision problems simultaneously for a container transhipment terminal considering the quayside congestion and the cost of container movements. Moccia *et al.* (2009) came up with a method based on column generation for allocating containers in transhipment ports. Nishimura *et al.* (2009) developed an optimization model that aimed to minimize the time of moving containers and dwell time. Zhen (2013) proposed a mixed-integer programming model to minimize the expected route length of containers flows considering the uncertain berthing time and position. A heuristic algorithm was developed to solve the large-scale instance. Wang *et al.* (2015) proposed the container assignment model based on profit maximization, considering transhipment under liner shipping networks, a segmentation procedure was developed to accelerate the algorithm.

The container allocation in the yard affects the travel time of the container truck as different assignments result in different container truck paths. Some researchers have investigated the container truck routing and scheduling problem (Vis, De Koster 2003; Kaveshgar, Huynh 2015; He *et al.* 2015; Chen *et al.* 2019; Shan *et al.* 2019). Vis and De Koster (2003) reviewed early works of container transportation from ship to yard and vice versa. Each truck was assigned to a path to complete the transportation task. Nishimura *et al.* (2005) focused on the trailer routing problem. The dynamic routing was proposed to reduce the travel distance. Cao *et al.* (2010) proposed an integrated model for yard truck and yard crane operation. The Bender's decomposition was used to solve the model. Chen *et al.* (2011) studied the truck transportation in the container terminal. The multiple truck routing problem was solved based on the transportation tasks. Chen *et al.* (2013) proposed a nonlinear programming model to analyse time-dependent truck queuing process in which stochastic service time distributions at gates and yards of a container terminal were considered. Lu and Le (2014) studied the integration of yard carne, QC and yard truck scheduling problem with uncertain factors. They assumed that the yard truck driving time was subject to normal distribution according to the statistics of Shanghai port, which showed good results. Kaveshgar and Huynh (2015) considered the yard truck scheduling problem from the real-world operational instances such as precedence degree of containers and QC safety margin. A mixed-integer programming model was formulated to work out the problem. He *et al.* (2015) integrated three factors that would impact the yard efficiency, QC, yard truck and yard crane. Shan *et al.* (2019) considered a facility location and truck routing problem from the supply chain perspective.

A heuristic algorithm was used to solve the problem because the model was complicated.

In recent years, scholars try to develop effective shortest path algorithms for RGS (Huang *et al.* 2007; Zeng, Church 2009; Yu *et al.* 2018; Peng *et al.* 2020). Most RGSs assume the travel time on the link is deterministic. However, link travel time seems to be highly stochastic in yard networks due to the complex port operation and spatial correlation (Chan *et al.* 2009; Yao *et al.* 2019a, 2019b). For instance, a yard congestion happening on a link may cause travel delays on adjacent links as well. Besides, yard cranes move from side to side on a link may also cause the travel delays on other links. Therefore, researchers have developed methods to solve the RSPP. Shao *et al.* (2004) presented a metaheuristic algorithm to find the reliable shortest path considering the travel time and variance. Chen and Ji (2005) proposed the alpha-shortest path that aimed to minimize the total travel time in a certain confidence level. Nikolova (2009) developed a quantification method to find the reliable shortest path for passengers with different objectives. Nie and Wu (2009) proposed a novel model, which could generate nondominated paths in which the path could not be replaced by others.

Previous works have made great efforts to improve the yard operation efficiency. Zhen (2016) first discussed the notion of yard congestion in the context of yard truck interruptions and developed a combination of probabilistic and physics-based models for truck interruptions. Inspired by his work on truck interruptions, we propose a model to optimize the reliable container truck paths with uncertain travel time.

Contribution

The contribution of this paper are as follows:

- *first*, this paper develops a model to optimize the path travel time for container trucks and find a reliable shortest path by considering the yard congestion and the impact of adjacent links. The variance and covariance among links are introduced to describe the uncertainty of travel time. From our case study, the yard congestion and impact among adjacent links truly affect the actual travel time of container trucks in ports;
- *second*, this paper considers the probability of arriving at the destination within the expected travel time. In different situations, the attitude of port operators toward the risk of being late is changeable. In this paper, the on-time arrival probability α which represents port operators' attitude toward risk of being late is presented. Port operators characterized by risk-averse, risk-neutral, and risk-seeking could choose the routing strategy freely.

The remainder of this paper is organized as follows:

- section 1 describes the problem;
- section 2 is divided into 2 subsections: subsection 2.1 describes the basic work for mathematical formulation and subsection 2.2 formulates the model;
- a solution method is developed in section 3;
- results of the cases are shown in section 4;
- some conclusions are summarized in the last section.

1. Problem description

In the yard side, the container truck path is influenced by the yard template planning, which assigns the container flows between vessels and subblocks. Without considering the container truck path, the yard template planning is a general assignment problem that can be well settled. However, because a large number of container trucks running along the path may cause traffic congestion, the travel time of each truck path is uncertain in reality, especially in the cross that two paths intersect or in the link that two container trucks merge. Here, a typical circumstance in a transhipment port is taken as an example.

In Figure 1, Vessel 1 performs the loading and unloading process. The unloading path is illustrated by the dashed line. The containers to be loaded on other vessels (i.e., Vessel 2, Vessel 3) in future are sent to the subblocks, i.e., S51, S69, S121 for vessel 2 and S21, S74, S109, S114 for Vessel 3. The solid lines refer to the loading paths. The containers from Vessel 2 reserved in the subblocks (i.e., S82, S115, S138) are loaded on Vessel 1. In practice, the travel time of each path is influenced by traffic flow in the yard. In addition, traffic congestion happens commonly, especially at the cross or around the yard crane. Moreover, the link with traffic congestion may also affect the travel time of the adjacent links. In this paper, we consider the influence of traffic flow by: (1) a congestion model in the context of truck interruptions that formulate the travel time of a single link, and (2) the correlation among adjacent links that formulate the travel time of a whole path.

Note that the working range of a QC is limited. When the coming vessel is large, there are probably more than one QC serving the vessel. Thus, the loading/unloading path from different QCs may be different. For simplicity, we assume the berth position of the vessel, which is used to determine the loading/unloading path, is the vessel's middle point.

2. Optimizing container truck paths with uncertain travel time

2.1. Basic work for model formulation

To formulate the mathematical model, some basic work including formulating the travel time of a single link in the congestion model (subsection 2.1.2), considering the correlation among adjacent links (subsection 2.1.3), finding the reliable shortest paths (subsection 2.1.4), balancing the workload protocol (subsection 2.1.5) should be done in advance.

2.1.1. The yard network

In Figure 2, the yard network could be denoted by a directed graph with nodes and links, which is referred to Zhen (2016). Let $G=(N, A, \Psi)$ be a directed graph consisting of a set of nodes N, a set of links A, and a set of movements Ψ. Figure 2 has three kinds of node, which are berth node, cross node, and subblock node. Berth node and subblock node respectively represents the berth position and subblock, which is denoted by $o \in N$ or $d \in N$ according to the direction of container flow. Cross node represents the crossing in practice. Container trucks at cross node could turn or pass through. A link $a = (i, j) \in A$ has a predecessor node $i \in N$ and a successor node $j \in N$. The container truck path is made up of a series of consecutive links from node o to node d. In this paper, index o and d specially refer to the origin and destination of a path, while (i, j) refers to a link in the path. $\psi_{i,j,k} = \left((i, j), (j, k)\right) \in \Psi$ denotes an allowed movement (e.g., turn or pass through movement) at node j. $\psi_{i,j,k} \in \Psi$ means that the movement has to be carried out at the middle node j. Here we take the unloading process as an instruction. In Figure 2, the blue line represents one of the paths from the berth o to the subblock d, let $\Omega_u^{o,d} = \left\{a_1, ..., a_m, ..., a_\lambda\right\}$ be the u path from o to d, consisting of λ consecutive links. We define the path travel time as $t_u^{o,d}$, which is a sum of link travel time, as is shown in Equation (1). Considering the uncertainty of path travel time, $t_u^{o,d}$ is a random variable that we will talk detail in section 2.1.3:

$$t_u^{o,d} = \sum_{m=1}^{\lambda} t_{a_m}, \tag{1}$$

where: t_{a_m} is the travel time of a_m (the mth link of path u).

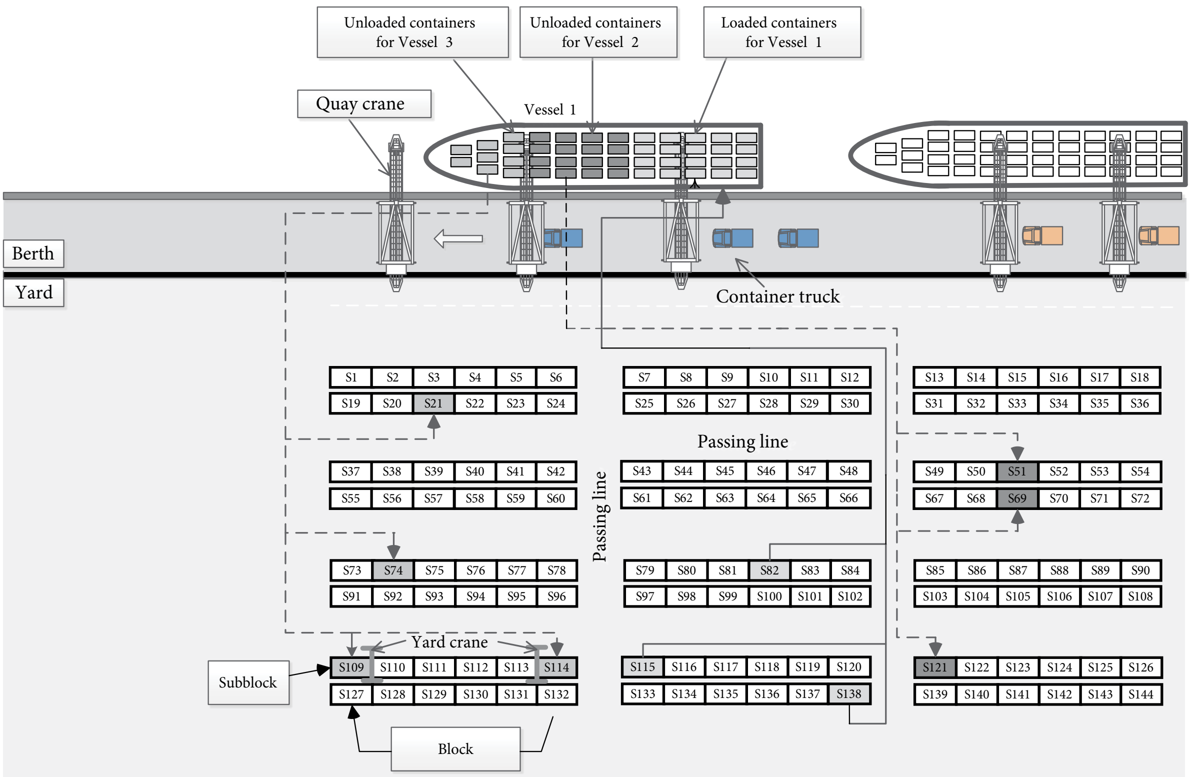


Figure 1. A typical working process of container ports

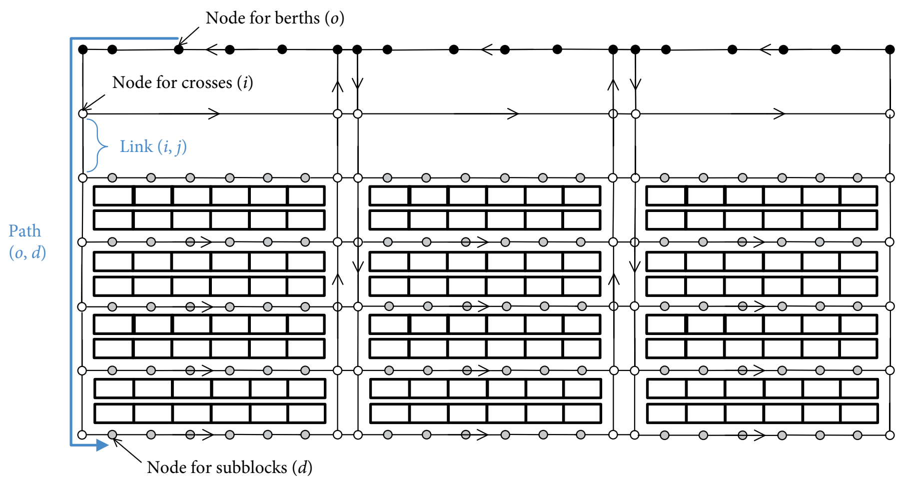


Figure 2. The yard network denoted by nodes and links

There are probably more than one path between the origin and destination, but it should be noted that paths in the yard network should follow the traffic rule. In this paper, the truck is guided by an anticlockwise direction, as shown in Figure 2. There are two truck lanes between every two adjacent blocks.

2.1.2. Congestion model of link travel time

As above mentioned, the link travel time is uncertain considering the yard congestion. The yard congestion prevents container trucks from traveling freely or prevents AGVs from running freely in the ports (Roy *et al.* 2016). If excessive container trucks are passing through the crossroad or running along a narrow lane at the same time, the normal speed of container trucks will be affected by each other. On this condition, the container trucks are forced to slow down or even stop when they are interrupted during the transportation (Zhang *et al.* 2009).

Figure 3 shows two common examples of the truck interruption happened in the yard, which is referred to Zhen (2016). As shown in Figure 3a, when a container truck (Truck 1) is running along the link, at the same time some other container trucks are running from the inside lane to the main lane, Truck 1 may slow down or stop to avoid collisions. The other type of interruption is shown in Figure 3b. When Truck 1 is running along the link, at the same time some other container trucks are running from passing line 1, 2 or 3 to passing line 4, Truck 1 may slow down or stop to avoid collisions.

Here we employ Zhen (2016) to formulate the influence of traffic flow to the link travel time. The travel time is affected by the number of truck interruptions on a link. Let $t_{i,j}(r)$ be the expected travel time of the link (i, j) given r interruptions, and let $P(r)_{i,j}$ be the probability of occurring r interruptions on link (i, j). The expected travel time of passing through link (i, j) can be calculated as:

$$t_{i,j} = \sum_{r=0}^{+\infty} P(r)_{i,j} \cdot t_{i,j}(r). \tag{2}$$

Referring to the interruption model on a link (Zhang *et al.* 2009), the probability $P(r)_{i,j}$ is formulated as:

$$P(r)_{i,j} = \frac{\left(\overline{R}\right)^r \cdot e^{-\overline{R}}}{r!}, \tag{3}$$

where: $\overline{R}$ is a mean number of the interruptions that would influence the transportation of container trucks.

Parameter $\overline{R}$ is calculated by the formula:

$$\overline{R} = \frac{s_{i,j} \cdot t_{i,j} \cdot v^2}{4 \cdot a \cdot d_{i,j} \cdot e_{i,j} \cdot h_{YC}},$$

where: $s_{i,j}$ denotes the number of the working subblocks on link (i, j); v denotes the average speed of container trucks; a denotes the acceleration (deceleration) of container trucks; $d_{i,j}$ denotes the length of the link (i, j); $e_{i,j}$ denotes the number of lanes on link (i, j); h_{YC} denotes the average handling time of a yard crane for a container.

Then, $t_{i,j}(r)$ can be calculated by:

$$t_{i,j}(r) = \frac{d_{i,j}}{v} + \frac{v_{i,j}'^2 + v_{i,j}''^2}{2a \cdot v} + \frac{v - v_{i,j}' - v_{i,j}'}{a} + \frac{r \cdot v}{8 \cdot a}, \tag{4}$$

where: $v_{i,j}'$, $v_{i,j}''$ are the speed of trucks at the beginning and end of link (i, j).

The expected travel time of link (i, j) is calculated by:

$$t_{i,j} = \sum_{r=0}^{+\infty} P(r)_{i,j} \cdot t_{i,j}(r) = \frac{8 \cdot \Gamma}{8 \cdot a - s_{i,j} \cdot \Delta}, \tag{5}$$

where:

$$\Gamma = \frac{2 \cdot a \cdot d_{i,j} + v_{i,j}'^2 + v_{i,j}''^2}{2 \cdot v} + v - v_{i,j}' - v_{i,j}'';$$

$$\Delta = \frac{v^3}{4 \cdot a \cdot d_{i.j} \cdot e_{i,j} \cdot h_{YC}}.$$

Equation (5) indicates that the link travel time is related to $s_{i,j}$ considering the yard congestion. While $s_{i,j}$ is a decision variable, which can be determined by the yard template planning, the travel time of link (i, j) in the yard template planning is correlated with $s_{i,j}$. As $t_{i,j}$ depends on $s_{i,j}$, it can be denoted as $t_{i,j,s}$. Here, $t_{i,j,s}$ means the travel time of link (i, j) when there are s working subblocks on link (i, j). The formulation of link travel time considering yard congestion is validated by a large number of simulation

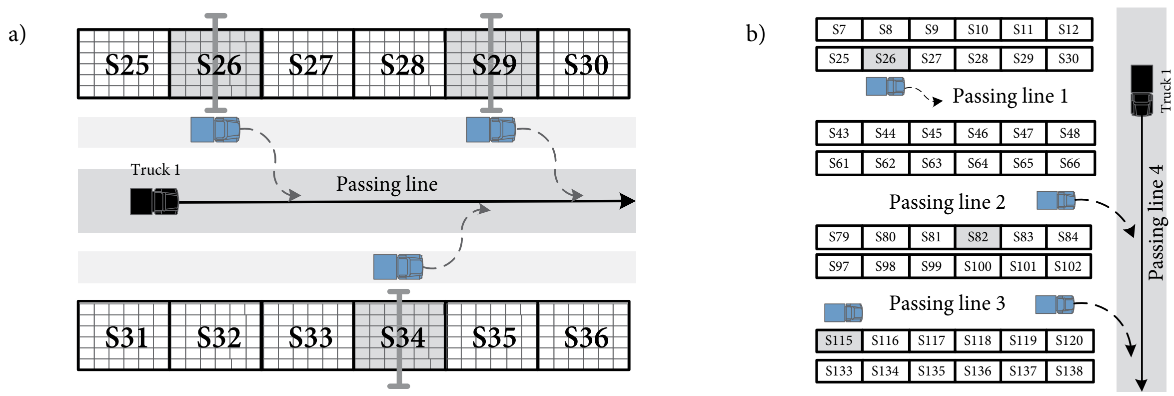


Figure 3. Interruptions happened in the travel path

runs, all the average relative deviation between simulated results and theoretical results calculated by Equation (5) is lower than 1.5%. For more proof information about Equations (2)–(5), we suggest the readers refer to Zhen (2016).

2.1.3. Congestion model considering the correlation among adjacent links

In fact, the link that is congested may have an influence on the adjacent links. We take Figure 3b for an example, any congestion happened in passing line (1~3) may induce the congestion in passing line 4 as well. In addition, the drivers may change their path to the destination so as to save the travel time. To formulate the correlation among adjacent links, we use *k*-neighbouring links of link (*i*, *j*), which means the travel time of link (*i*, *j*) is related to *k*-neighbouring links from a spatial perspective. Let $X_{i,j}^{q,w}$ be the topological distance (a value that reflects the closeness of the two links) between link (*i*, *j*) and link (*g*, *w*), e.g. the topological distance of two directly connected links is 1. A link (*g*, *w*) is said to be a *k*-neighbouring link of link (*i*, *j*) if and only if $X_{i,j}^{q,w} = k$. k can be valued through a sensitive analysis.

In this paper, we assume that the link travel time follows normal distribution, which is common to see in the studies of RSPP (Chang *et al.* 2005; Chen *et al.* 2012, 2014). The expectation of link travel time $t_{i,j,s}$ can be calculated by the congestion model in subsection 2.1.2. And the SD of the link σ_m can be calculated by statistical data of the yard. In practice, σ_m means the dispersion of the *m*th link travel time. The larger σ_m is, the greater it impacts the adjacent links. Therefore, the expectation of the path travel time $t_u^{o,d}$ and the SD of the path travel time $\sigma_u^{o,d}$ can be calculated as:

$$t_u^{o,d} = \sum_{(i,j)\in u} t_{i,j,s}; \tag{6}$$

$$\sigma_u^{o,d} = \sqrt{\sum_{m=1}^{\lambda}\sigma_m^2 + \sum_{n=1}^{k}\sum_{m=1}^{\lambda-n} 2\cdot \operatorname{cov}\left(a_m, a_{m+n}\right)}, \tag{7}$$

where: $\operatorname{cov}\left(a_m, a_{m+n}\right)$ is the covariance of travel time between link a_m and a_{m+n}; *k* is the closeness degree.

Let $F_u^{o,d}(\alpha)$ be the CDF of path travel time $t_u^{o,d}$ at α confidence level. So the path travel time can be expressed by the inverse of CDF:

$$F_u^{o,d}(\alpha)^{-1} = t_u^{o,d} + z_\alpha \cdot \sigma_u^{o,d}, \tag{8}$$

where: z_α is the value of standard normal distribution at α confidence level.

Equations (6)–(8) are referred to Chen *et al.* (2012), which proves the effectiveness of the formulas. Here, the confidence level α ∈ (0,1) is the probability that the container truck arrives at the destination within the travel time $F_u^{o,d}(\alpha)^{-1}$. The on-time arrival probability represents port operators' attitudes toward the risk of being late (α < 0.5, α = 0.5 and α > 0.5 indicate risk-seeking, risk-neutral, and risk-averse attitudes, respectively). The value of α can be determined based on port operator's purpose.

2.1.4. Find the reliable shortest path

Now, given the origin *o*, destination *d*, and on-time arrival probability α, we can find the reliable shortest path according to Chen and Ji (2005).

$$\min \sum_{o\in N}\sum_{d\in N} F_u^{o,d}(\alpha)^{-1} \tag{9}$$

subject to Equation (8), and

$$t_u^{o,d} = \sum_{(i,j)\in A} t_{i,j,s}\cdot x_{i,j}^{o,d}; \tag{10}$$

$$\sum_{j\in N} x_{i,j}^{o,d} - \sum_{k\in N} x_{k,i}^{o,d} = \begin{cases} 1, & \forall i = o; \\ 0, & \forall i \neq o, i \neq d; \\ -1 & \forall i = d; \end{cases} \tag{11}$$

$$x_{i,j}^{o,d} \in \{0,1\},\ \forall a_{ij} \in A; \tag{12}$$

$$\psi_{i,j,k} \in \Psi,\ \forall (i,j) \in \Omega_u^{o,d},\ \forall (j,k) \in \Omega_u^{o,d}, \tag{13}$$

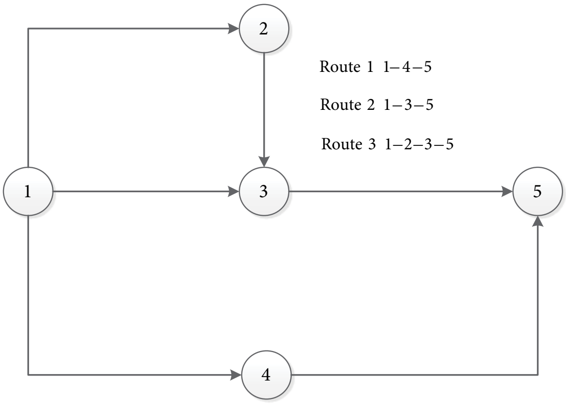
where: the decision variable $x_{i,j}^{o,d}$ is regarded as the relationship between link-path; $x_{i,j}^{o,d} = 1$ denotes that a_{ij} link (*i,j*) is in the path Ω_u^{od}, otherwise $x_{i,j}^{o,d} = 0$. Equation (9) is to minimize the travel time of all feasible paths. Equations (8) and (10) define the path travel time. Equation (11) guarantees the feasibility of the path. Constraint (12) should be a binary variable concerned with the link-path. Constraint (13) ensures the feasibility of all the movements in the reliable shortest path.

Here a small example is given to show the influence of correlation links to the shortest path. In Figure 4, the travel time of the links obeys normal distribution. The number on the link denotes the average link travel time. The variance and covariance of the link travel time are shown in the matrix. In the variance-covariance matrix, elements on the diagonal line are the variance of the link travel time and off-diagonal elements are the covariance of the travel time between two links. Considering the matrix is symmetric, only a triangular matrix is shown.

In Figure 4, there are three paths $\Omega_1^{15} = a_{14} \cup a_{45}$, $\Omega_2^{15} = a_{13} \cup a_{35}$, and $\Omega_3^{15} = a_{12} \cup a_{23} \cup a_{35}$ from the Node 1 to the Node 5. ∪ is a path connector $\left(\Omega_1^{15} = a_{14} \cup a_{45}\right.$ indicates Ω_1^{15} passes a_{14} and $\left.a_{45}\right)$. From Table 1, when α = $0.1\left(Z_\alpha = -1.28\right)$, the port operators would like to choose path Ω_3^{15}, in which the travel time variation is large, to get a small travel time $F_3^{15}(\alpha)^{-1} = 5.02$. When α = 0.5 $\left(Z_\alpha = 0\right)$, the port operators tend to choose path Ω_2^{15}, which has the smallest mean travel time $t_2^{15} = 10$. When α = $0.9\left(Z_\alpha = 1.28\right)$, the port operators are risk-averse. They prefer to use the more reliable path Ω_1^{15} with a small travel time SD and a larger travel time $F_1^{15}(\alpha)^{-1} = 14.22$. In results, the optimal solution of the reliable shortest path depends on the attitudes of port operators.

Table 1. The results of illustrative example

α	$F_1^{15}(\alpha)^{-1}$	$F_2^{15}(\alpha)^{-1}$	$F_3^{15}(\alpha)^{-1}$
0.1	9.78	5.46	**5.02**
0.5	12	**10**	11
0.9	14.22	14.54	16.98



$$
\begin{array}{c|cccccc|}
 & a_{12} & a_{13} & a_{14} & a_{23} & a_{35} & a_{45} \\
a_{12} & 4 & & & & & \\
a_{13} & -2 & 5 & & & & \\
a_{14} & -1 & -0.2 & 1 & & & \\
a_{23} & 2 & -1 & -0.7 & 3 & & \\
a_{35} & 1.4 & 1.3 & 0.5 & 1.5 & 5 & \\
a_{45} & -2 & -0.6 & 0.5 & -0.4 & -1.5 & 1
\end{array}
$$

Figure 4. An illustrative example:

$F_1^{15}(\alpha)^{-1} = F^{a_{14} \cup a_{45}}(\alpha)^{-1} = 5+7+Z_\alpha \cdot \sqrt{1+1+2\cdot 0.5}$;

$F_2^{15}(\alpha)^{-1} = F^{a_{13} \cup a_{35}}(\alpha)^{-1} = 4+6+Z_\alpha \cdot \sqrt{5+5+2\cdot 1.3}$;

$F_3^{15}(\alpha)^{-1} = F^{a_{12} \cup a_{23} \cup a_{35}}(\alpha)^{-1} = 2+3+6+Z_\alpha \cdot \sqrt{4+3+5+2\cdot 2+2\cdot 1.5+2\cdot 1.4}$

2.1.5. Workload balancing protocol

In container yard, if two neighbouring subblocks simultaneously have loading or unloading activities, they may accumulate a large number of container trucks in the same lane, which could easily cause traffic congestion. To mitigate the congestion in the planning period, we employ a commonly used high–low workload protocol (Lee *et al.* 2006; Han *et al.* 2008; Jiang *et al.* 2012). Workload means the number of containers handled by a yard crane in a time unit. The high workload is defined as a range, e.g. $[10, 20)$, which could not contain the range of low workload, e.g. $[0, 10)$. The idea is that two neighbouring subblocks should not simultaneously be in high workload. The judgement of neighbourhood between two subblocks is done by a vicinity matrix used in Lee *et al.* (2006). Here, a subblock is the neighbour of another one only if they are adjacent and share the same lane. For example, S56 and S57 are neighbours, but S56 and S38 are not neighbours even they are back to back, as shown in Figure 1.

2.2. Model formulation

Before building the mathematical model, we first clarify some assumptions:

- number of transhipment containers are considered to be available and deterministic within the planning horizon;
- the berth position of the ship and the berthing time are deterministic;
- operational level decisions, for example, the work order of yard cranes is not considered;
- the path travel time of container trucks follows normal distribution;
- the waiting time at yard cranes and QCs is same for every container truck;
- trucks in the yard are guided by an anticlockwise direction;
- the congestion model is suitable for all the truck lanes.

2.2.1. Notations

Parameters:

A – set of all the links in the yard network indexed by $a=(i, j)$, $a=(i, j)\in A$;

$A_{s,v}^L$ – subset of links in the loading path from subblock s to vessel v, $A_{s,v}^L \subseteq A$;

$A_{v,s}^U$ – subset of links in the unloading path from vessel v to subblock s, $A_{v,s}^U \subseteq A$;

N – set of all the nodes in the yard network; note that node $o \in N$ refers to origin and $d \in N$ refers to destination;

N_s – subset of subblock nodes in the yard network, $N_s \subseteq N$;

N_v – subset of berth nodes in the yard network, $N_v \subseteq N$;

P – set of the whole time periods indexed by p;

P_v – subset of periods when vessel v moors, $P_v \subseteq P$;

S – set of all the subblocks indexed by s, note that N_s refer to the corresponding node set of S according to the yard network;

S_v – subset of candidate subblocks that are assigned to vessel v, $S_v \subseteq S$;

$S_{i,j}$ – subset of subblocks that may have influence on the traffic in link (i, j), $S_{i,j} \subseteq S$;

S_g – the group of subblocks, which belongs to block g, $S_g \subseteq S$;

S_{neigh} – the pair of neighbour subblocks, e.g. $S_{neigh} = \{21, 39\}$ means subblock 21 and subblock 39 are neighbours, $S_{neigh} \in \mathbb{S}$, $\mathbb{S}$ is set of all the neighbour pairs;

V – set of vessels indexed by v; note that N_v refer to the corresponding node set of V according to the yard network;

V_v – subset of vessels that will load the containers that unloaded from vessel v, $V_v \subseteq V$;

K – set of possible number of subblocks that are taking loading or unloading activities (indexed by k); $K=\{0, 1, \ldots, |K|\}$;

- Ω – set of all the possible paths, and $\Omega_u^{o,d}$ means the u path from o to d; note that $u \subseteq A_{s,v}^L$ for loading path and $u \subseteq A_{v,s}^U$ for unloading path;
- m – number of vessels in the planning period;
- l_v – time length when vessel v moors;
- n_v – number of subblocks that are assigned to vessel v;
- $q_{\overline{v},v}$ – number of containers, which are unloaded from vessel $\overline{v}$, stored in the yard, and then loaded on to vessel v in future;
- $t_{i,j,k}$ – the expected travel time when there are k subblocks performing loading or unloading activities on link (i, j);
- $\varepsilon_{i,j,p}^{o,d}$ – equals to 1 if link (i, j) is on the path $\Omega^{o,d}$ in period p;
- W_{LB} – lower bound of the minimum workload;
- W_{UB} – upper bound of the maximum workload;
- W_{YC} – the maximum workload of a yard crane in a period;
- Y_{YC} – the maximum number of yard cranes, which can work simultaneously in a block.

Decision variables:

- $x_u^{o,d} \in \{0,1\}$ – set to one if u path is selected from original o to destination d;
- $\beta_s^v \in \{0,1\}$ – set to one if subblock s is assigned to vessel v; and zero otherwise;
- $\eta_{s,p}^L \in \{0,1\}$ – set to one if subblock s has loading activities in period p; zero otherwise;
- $\eta_{s,p}^U \in \{0,1\}$ – set to one if subblock s has unloading activities in period p; zero otherwise;
- $\gamma_{i,j,k,p} \in \{0,1\}$ – set to one if there are k subblocks that have loading or unloading activities along link (i, j) in period p; and zero otherwise;
- $\mu_{s,p} \in \{0,1\}$ – set to one if workload of subblock v is high in period p; and zero otherwise;
- $\lambda_{i,j,p} \ge 0$ – number of subblocks that have loading or unloading activities along link (i, j) in period p;
- $n_{o,d,p}^L \ge 0$ – number of loaded containers that go through path $\Omega^{o,d}$ in period p;
- $n_{o,d,p}^U \ge 0$ – number of unloaded containers that go through path $\Omega^{o,d}$ in period p;
- $\delta_{s,p}^L \ge 0$ – number of containers loaded from subblock s in period p;
- $\delta_{s,p}^U \ge 0$ – number of containers unloaded to subblock s in period p.

2.2.2. Model for optimizing container truck paths with uncertain travel time

$M_{\text{OCTP-UTT}}$:

$$\min \sum_{o \in N_s,\, d \in N_v,\, p \in P} \sum_{u \in \Omega^{o,d}} \left(t_u^{o,d} + z_\alpha \sigma_u^{o,d}\right) \cdot x_u^{o,d} \cdot n_{o,d,p}^L +$$

$$\sum_{o \in N_v,\, d \in N_s,\, p \in P} \sum_{u \in \Omega^{o,d}} \left(t_u^{o,d} + z_\alpha \sigma_u^{o,d}\right) \cdot x_u^{o,d} \cdot n_{o,d,p}^U \quad (14)$$

subject to:

$$\sum_{v \in V} \beta_s^v \le 1, \ \forall s \in S; \quad (15)$$

$$\sum_{s \in S_v} \beta_s^v = n_v, \ \forall v \in V; \quad (16)$$

$$\sum_{s \in S \backslash S_v} \beta_s^v = 0, \ \forall v \in V; \quad (17)$$

$$\sum_{u \subseteq A_{s,v}^L} x_u^{o,d} = 1, \ \forall o \in N_s, \ d \in N_v; \quad (18)$$

$$\sum_{u \subseteq A_{v,s}^U} x_u^{o,d} = 1, \ \forall o \in N_v, \ d \in N_s; \quad (19)$$

$$n_{i,j,p}^L = \sum_{v \in V,\, s \in S: (i,j) \in A_{s,v}^L,\, p \in P_v} \beta_s^v \cdot \frac{\sum_{\overline{v} \in V} q_{\overline{v},v}}{n_v \cdot l_v},$$
$$\forall (i,j) \in A, \ \forall p \in P; \quad (20)$$

$$n_{i,j,p}^U = \sum_{\overline{v} \in V,\, s \in S: (i,j) \in A_{\overline{v},s}^U,\, p \in P_{\overline{v}}} \sum_{v \in V} \beta_s^v \cdot \frac{q_{\overline{v},v}}{n_v \cdot l_v},$$
$$\forall (i,j) \in A, \ \forall p \in P; \quad (21)$$

$$n_{o,d,p}^L = \sum_{i=o,\, j \in N} n_{i,j,p}^L \cdot \varepsilon_{i,j,p}^{o,d},$$
$$\forall o \in N_s, \ \forall d \in N_v, \ \forall p \in P_v; \quad (22)$$

$$n_{o,d,p}^U = \sum_{i \in N,\, j=d} n_{i,j,p}^U \cdot \varepsilon_{i,j,p}^{o,d},$$
$$\forall o \in N_v, \ \forall d \in N_s, \ \forall p \in P_v; \quad (23)$$

$$\eta_{s,p}^L = \sum_{v \in V: p \in P_v} \beta_s^v, \ \forall s \in S, \ \forall p \in P; \quad (24)$$

$$\eta_{s,p}^U \ge \frac{\sum_{v \in V: p \in P_v} \sum_{\overline{v} \in V_v} \beta_s^{\overline{v}}}{m}, \ \forall s \in S, \ \forall p \in P; \quad (25)$$

$$\sum_{k \in K} \gamma_{i,j,k,p} = 1, \ \forall (i,j) \in A, \ \forall p \in P; \quad (26)$$

$$\sum_{k \in K} k \cdot \gamma_{i,j,k,p} = \lambda_{i,j,p}, \ \forall (i,j) \in A, \ \forall p \in P; \quad (27)$$

$$\lambda_{i,j,p} = \sum_{s \in S_{i,j}} \left(\eta_{s,p}^L + \eta_{s,p}^U\right) \ \forall (i,j) \in A, \ \forall p \in P; \quad (28)$$

$$t_u^{od} = \sum_{(i,j) \in u} \sum_{p \in P} \sum_{k \in K} t_{i,j,k} \cdot \gamma_{i,j,k,p},$$
$$\forall o \in N_s, \ \forall d \in N_v \text{ or } \forall o \in N_v, \ \forall d \in N_s; \quad (29)$$

$$\delta_{s,p}^L = \sum_{v \in V: p \in P_v} \beta_s^v \cdot \frac{\sum_{\overline{v} \in V} q_{\overline{v},v}}{n_v \cdot l_v}, \ \forall s \in S, \ \forall p \in P; \quad (30)$$

$$\delta_{s,p}^{U} = \sum_{\bar{v}\in V:p\in P_{\bar{v}}} \sum_{v\in V} \beta_s^v \cdot \frac{q_{\bar{v},v}}{n_v \cdot l_v}, \forall s\in S,\ \forall p\in P; \quad (31)$$

$$\mu_{s,p} \cdot W_{LB} \le \delta_{s,p}^{L} + \delta_{s,p}^{U} \le W_{LB} + \mu_{s,p} \cdot \left(W_{UB} - W_{LB}\right),$$

$$\forall s\in S,\ \forall p\in P; \quad (32)$$

$$\sum_{s\in S_{neigh}} \mu_{s,p} \le 1,\ \forall S_{neigh} \in \mathbb{S},\ \forall p\in P; \quad (33)$$

$$\sum_{s\in S_g} \left(\delta_{s,p}^{L} + \delta_{s,p}^{U}\right) \le W_{YC} \cdot Y_{YC},\ \forall S_g \subseteq S,\ \forall p\in P; \quad (34)$$

$$x_u^{o,d} \in \{0,1\},$$

$$\forall o\in N_s,\ \forall d\in N_v \text{ or } \forall o\in N_v,\ \forall d\in N_s; \quad (35)$$

$$\beta_s^v \in \{0,1\},\ \forall v\in V,\ \forall s\in S; \quad (36)$$

$$\eta_{s,p}^{L}, \eta_{s,p}^{U}, \mu_{s,p} \in \{0,1\},\ \forall s\in S,\ p\in P; \quad (37)$$

$$\delta_{s,p}^{L}, \delta_{s,p}^{U} \ge 0,\ \forall s\in S,\ \forall p\in P; \quad (38)$$

$$\lambda_{i,j,p}, n_{i,j,p}^{L}, n_{i,j,p}^{U} \ge 0\ \ \forall (i,j)\in A,\ \forall p\in P; \quad (39)$$

$$n_{o,d,p}^{L}, n_{o,d,p}^{U} \ge 0,\ \forall o\in N_s,\ \forall d\in N_v,\ p\in P \text{ or}$$

$$\forall o\in N_v,\ \forall d\in N_s,\ p\in P; \quad (40)$$

$$\gamma_{i,j,k,p} \in \{0,1\}, \forall (i,j)\in A,\ \forall p\in P,\ \forall s\in S. \quad (41)$$

Objective (14) is to minimize the total travel time of containers going through the paths, which include the loading and unloading paths between vessels and subblocks. Constraints (15) express that each subblock is allocated to no more than one vessel. Constraints (16) and (17) ensure the number of subblocks allocated to vessel *v*. Constraints (18) and (19) ensure that only one path is selected for each loading/unloading process. Constraints (20) and (21) indicate the number of loaded and unloaded containers passing through link (*i*, *j*) in period c. Constraints (22) and (23) define the number of loaded and unloaded containers through path $\Omega^{o,d}$ in period *p*. Constraints (24) and (25) define the binary variable to ensure that whether a subblock has loading or unloading activities in a period. In the right side of Constraint (25), $\sum_{v\in V:p\in P_v} \sum_{\bar{v}\in V_v} \beta_s^{\bar{v}}$ means the number of all the vessels that use subblock *s* to unload containers. As $\eta_{s,p}^{U}$ is binary variable, $\sum_{v\in V:p\in P_v} \sum_{\bar{v}\in V_v} \beta_s^{\bar{v}}$ should be divided by *m*. If $\eta_{s,p}^{U}$ is positive, $\eta_{s,p}^{U}$ equals to one; otherwise, it is zero. Constraints (26) and (27) combine the binary variables $\gamma_{i,j,k,p}$ with the integer variables $\lambda_{i,j,p}$, which denotes the number of working subblocks on link (*i*, *j*) in period (*i*, *j*). Constraint (28) define the relationship between $\lambda_{i,j,p}$ and $\eta_{s,p}^{L}, \eta_{s,p}^{U}$. Constraint (29) express that the expectation of path travel time is the sum of link travel time considering yard congestion. Constraints (30) and (31) respectively calculate the number of containers, which are loaded from and unloaded to a subblock during a time period. Constraint (32) ensure the workload activities in each subblock is either high (i.e., $\mu_{s,p} = 1$) or low (i.e., $\mu_{s,p} = 0$). Constraint (33) guarantee that high-workload activity should not happen between two neighbour subblocks simultaneously. Constraints (32) and (33) are derived from a common workload balancing protocol to mitigating congestion (see subsection 2.1.5). Constraint (34) restrict that the workload activities within a block should not exceed the number of yard cranes. Usually, there are two yard cranes in each block. Constraints (35)–(41) define decision variables.

2.2.3. Linearization for the model

Objective (14) is nonlinear. To linearize the objective so that it could be solved by commercial solvers, some auxiliary decision variables and constraints are added. The new variables $\omega_{o,d,u,p}^{L}$ and $\omega_{o,d,u,p}^{U}$ are defined to take the place of $x_u^{o,d} n_{o,d,p}^{L}$ and $x_u^{o,d} n_{o,d,p}^{U}$, respectively:

- »» $\omega_{o,d,u,p}^{L} \ge 0$ the number of loaded containers from origin *o* to destination *d* in period *p*, if *u* path is selected, $\omega_{o,d,u,p}^{L} = 0$ otherwise;
- »» $\omega_{o,d,u,p}^{U} \ge 0$ the number of unloaded containers from origin *o* to destination *d* in period *p*, if *u* path is selected, $\omega_{o,d,u,p}^{L} = 0$ otherwise.

The model could be changed to:

$M_{\text{OCTP-UTT}}$:

$$\min \sum_{o\in N_s,\, d\in N_v,\, p\in P} \left(t_u^{o,d} + z_\alpha \sigma_u^{o,d}\right) \cdot \omega_{o,d,u,p}^{L} + \sum_{o\in N_v,\, d\in N_s,\, p\in P} \left(t_u^{o,d} + z_\alpha \sigma_u^{o,d}\right) \cdot \omega_{o,d,u,p}^{U} \quad (42)$$

subject to Constraints (16)–(42).

$$\omega_{o,d,u,p}^{L} \ge n_{o,d,p}^{L} + \left(x_u^{o,d} - 1\right) \cdot M,$$

$$\forall o\in N_s,\ \forall d\in N_v,\ \forall u\in \Omega^{o,d},\ \forall p\in P; \quad (43)$$

$$\omega_{o,d,u,p}^{U} \ge n_{o,d,p}^{U} + \left(x_u^{o,d} - 1\right) \cdot M,$$

$$\forall o\in N_v,\ \forall d\in N_s,\ \forall u\in \Omega^{o,d},\ \forall p\in P; \quad (44)$$

$$\omega_{o,d,u,p}^{L}, \omega_{o,d,u,p}^{U} \ge 0,$$

$$\forall o\in N_s,\ \forall d\in N_v,\ \forall u\in \Omega^{o,d},\ p\in P \text{ or}$$

$$\forall o\in N_v,\ \forall d\in N_s,\ \forall u\in \Omega^{o,d},\ \forall p\in P. \quad (45)$$

In Constraints (43) and (44), *M* is a large number to guarantee the linearization that:

$$\omega_{o,d,u,p}^{L} = \begin{cases} n_{o,d,p}^{L}, & \text{if } x_u^{o,d} = 1; \\ 0, & \text{if } x_u^{o,d} = 0, \end{cases}$$

and

$$\omega_{o,d,u,p}^{U} = \begin{cases} n_{o,d,p}^{U}, & \text{if } x_u^{o,d} = 1; \\ 0, & \text{if } x_u^{o,d} = 0. \end{cases}$$

3. Model solution

Considering the proposed $M_{OCTP-UTT}$ model is complex, it cannot be solved by analytical algorithm methods such as column generation, branch and price algorithm efficiently, because it is difficult to define the objective value of columns. Other widely used analytical algorithm method such as the dynamic programming is also not valid to solve the model because the decision process is hard to be divided into stages. Therefore, in this paper, the SCE-UA and A^* algorithm are introduced to solve the proposed $M_{OCTP-UTT}$ model. The SCE-UA algorithm is an effective evolution algorithm, which is similar to genetic algorithm. This method was firstly applied to optimize the parameters of the hydrologic models (Duan *et al.* 1994). Because of the advantage of global optimization, it was applied to other areas. Yu *et al.* (2020) used SCE-UA algorithm on the Subordinate Net Points Layout Optimization of Express Enterprise. The results were feasible and had excellent robustness. While A^* algorithm is an efficient algorithm in finding shortest path between any two given nodes, it is also a heuristic algorithm firstly applied in Hart *et al.* (1968). In our problem, we consider the uncertain travel time talked in subsection 2.1 and propose a reliable shortest path algorithm based on A^*, called RSPP–A^*.

3.1. SCE-UA algorithm to work out the yard template planning problem

The SCE-UA algorithm is a global optimization algorithm that integrates the advantages of deterministic search, random search, and competition evolution. It performs well in global search performance and efficiency of multi-parameter combination. In this section, we use SCE-UA algorithm to work out the yard template planning problem, in which the algorithm decides the assignment of containers between vessels and subblocks. The pseudo-code of SCE-UA algorithm is presented in Algorithm 1. Step 1 to step 10 is initial process, which defines the parameters (complex p and points in p) and find a feasible solution. Step 11 to step 17 is heuristic searching process. In the process, the genetic and mutation step is done in a CCE algorithm (see Algorithm 2) to generate new solutions. The algorithm stops until the solution satisfies a check convergence procedure.

The CCE algorithm is the crucial part in SCE-UA. The main purpose of CCE is to generate better solutions than that in last iteration. The pseudo-code of CCE algorithm is presented in Algorithm 2. Step 1 is initial process to set the parameter (the iteration σ and evolution τ). Step 3 to step 5 define the probability of being selected from parent solution. Step 7 to step 11 generate offspring solution from the selected parent solution. Step 12 to 27 check the dominance rules to see if the offspring solution could dominance the parent solution.

3.2. A^* algorithm to find reliable shortest path

This section presents a multicriteria A^* algorithm, named RSPP–A^*, to find the reliable shortest path in the proposed yard nsetwork. Given the origin o and destination d of the containers obtained from subsection 3.1 and the on-time probability α, the RSPP–A^* algorithm could find a reli-

Algorithm 1: *SCE-UA( ) // Shuffled Complex Evolution Algorithm*	
	//**initialization**
1	Set n for the number of complex, $P=\{p_1, p_2, p_3, ..., p_n\}$, $p_i=\phi$;
2	Set m for the number of points in each complex;
3	For each complex i, $i \in P$
4	$p(i)=\left\{\underbrace{\{0\},\{0\},...,\{0\}}_{m}\right\}$
5	For each point k, $k \in p_i$
6	Generate an initial solution x in the feasible space $\Omega \in R^n$
7	Use A^* algorithm to calculate the objective function $f(x)$ *//see Algorithm 3.*
8	End
9	End
10	Set $D=\{[x, f(x)] \mid x=1,2,3,...,n\cdot m\} \leftarrow rank[f(x)]$ *//rank the* $n\cdot m$ *points in ascending Order of objective function* $f(x)$*, and store them in D.*
	//**searching process**
11	While *check_convergence* $()=false$, do
12	For each complex i, $i \in P$
13	$p_i=\{(x_j^i, f_j^i) \mid x_j^i=x_{i+p\cdot(j-1)}, f_j^i=f_{i+p\cdot(j-1)}, j=1,2,...,m\}$
14	$CCE(p_i)$ *//Do a competitive complex evolution algorithm, see Algorithm 2.*
15	$D \leftarrow rank(P=\{p_1, p_2, ..., p_n\})$ *// rank the n complexes in ascending order of objective function and store them in D*
16	End
17	End
18	Return $[x, f(x)]$ *//find the near-optimal solution.*

Algorithm 2: *CCE( ) // competitive complex evolution algorithm*

```
   //initialization
1  Set q, σ, τ, iter = 0, evo = 0 (2 ≤ q ≤ m, σ ≥ 1, τ ≥ 1) //where q is the number of subcomplexes, σ is the target number of generations,
   t is the target number of evolutions of each complex, iter and evo are iteration factors.
   //heuristic process
2  While iter ≤ σ, do
3      For each point k, k ∈ p_i // using a trapezoidal probability distribution to produce weights w_k
4          w_k = 2·(m+1−k) / (m·(m+1))  //the point k = 1 has the high probability w_1 = 2/(m+1); the point k = m has the lowest probability
           w_m = 2 / (m·(m+1))
5      End
6      While evo ≤ τ, do
7          Set B = {(u_k, f_k) | i = 1, 2, ..., q} ← random(q)  //randomly select q different points u_1, u, u_q from p_i according to
           the probability distribution, and store them in B, where f_k is the objective function of u_k.
8   Set L = {l_k | k = u_1, u_2, ..., u_q} ← location(u_k)  //store the relative location of u_k in p_i
9          Rank (B); Rank (L)  //rank B and L in ascending order of function value
10         Set centroid g = 1/(q−1) · Σ_{j=1}^{q−1} u_j
11         Generate new point r = 2 · g − u_q
12         If r ∈ Ω  //the dominance rules
13             If f_r < f_q
14                 u_q ← r
15             Else
16                 c = (g + u_q) / 2
17                 if f_c < f_q
18                     u_q ← c
19                 Else
20                     Compute the smallest hypercube H ∈ R^n that contains p_i
21                     Randomly generate z ∈ H, u_q ← z
22                 End
23             End
24         Else
25             Compute the smallest hypercube H ∈ R^n that contains p_i
26             Randomly generate z ∈ H , u_q ← z
27         End
28     End
29     p_i ← B, according to L //update p_i with B in the relative location according to L
30     Rank(p_i) //rank p_i in ascending order of objective function
31 End
32 Return p_i
```

able shortest path for container trucks. The pseudo-code of RSPP–A* algorithm is given in Algorithm 3. Step 1 to step 10 initialize the parameters by setting an open list for unexamined nodes and a close list for examined nodes. The idea of RSPP–A* is consistently transforming nodes from open list to close list until finding the destination d. Step 11 to step 14 calculate the estimated distance function $F(i) = x(i) + y(i)$. Here $x(i)$ is the reliable distance from origin o to node i considering the yard congestion and correlation between adjacent links, while $y(i)$ is an estimated distance from node i to destination d. We use Manhattan distance to calculate $y(i)$. Manhattan distance only considers the horizontal and vertical moves to estimate a distance from node i to destination d. $y(i)$ may not be a real distance but could give a direction for path extension. Step 15 to step 16 decide the node that could be transformed to the close list. Step 17 to step 28 check the dominance. Node with smaller distance could dominate the same node with larger distance. For example, $F^1(i) \le F^2(i)$, $F^1(i)$ dominates $F^2(i)$.

Algorithm 3: A* algorithm

//**initialization**
set $P = \phi$, $T = \{\{o\}\}$ *// T is an open list that stores nodes will be examined, P is a close list that will not be examined.*
For each node i, $i \in N$
    If $neighbour(o, i)$ *// $neighbour(i, j)$ is used to judge if node i and node j are neighbours*
        If $i = d$
            Return $t^{o,d}$; brake *//find the shortest path*
        else
            $T \leftarrow T \cup \{i\}$
        End
    End
End
//**path extension**
While $T \neq \phi$, do
    For each node i, $i \in T$
        Calculate $F^1(i) = x(i) + y(i)$ *// $F(i)$ is an estimated distance through node i, $x(i)$ is the distance from origin o to node i, $y(i)$ is an estimated distance from node i to destination d. Here we use Manhattan distance to calculate $y(i)$*
    End
    Select $\min\_i \leftarrow i$, where $F^1(i)$ is minimum, $i \in T$ *//find the minimum $F(i)$, and delivery i to* min_i
    $P \leftarrow P \cup \{\min\_i\}$, $T \leftarrow T \setminus \{\min\_i\}$
    For each node i, $i \in N$
        If $neighbour(\min\_i, i)$
            If $i = d$
                Return $F(i)$; brake *//find the shortest path*
            Else
                $T \leftarrow T \cup \{i\}$
                Calculate $F^2(i) = x(i) + y(i)$
                Dominance$(F^1(i), F^2(i))$ *//check the dominance between $F^1(i)$ and $F^2(i)$, select the smaller value and node i.*
            End
        End
    End
End

4. Case study

In case study, one yard of Dalian container port is used to demonstrate the applicability of the proposed algorithm. In addition, several experiments are performed to validate the effectiveness of the proposed model.

4.1. The specification of case study

The planning horizon is 1 week considering the cycle of container liner transportation typically measures in weeks. The planning horizon is divided into 168 time periods and each time period is set to be 1 h. It is assumed that each vessel has specific arrival time made in advance, so the subset periods for each vessel with loading/unloading activities at port could be determined. In this case, the yard is considered to have 144 subblocks at most, and the layout of one yard in Dalian container port is shown in Figure 5.

Each block is 6 containers (TEU) deep and 36 containers long. There are at most two yard cranes in one block, which is coincident with the real situations in ports. The block is further divided into six subblocks. Each subblock is six containers long and five containers high. The capacity of each subblock is 180 (= $6 \cdot 6 \cdot 5$) TEUs. The width of horizontal and vertical passing lanes in the yard are set to be 30 and 70 m respectively. Each vessel is set to have 4 or 5 subblocks to store its containers. The transhipped containers in each vessel can be loaded to at most 5 other vessels. Then according to the data of Dalian container port, the number of containers to be unloaded and loaded by a yard crane in one hour is 24 TEUs. When calculating the travel time for each link by Equation (5), the parameters are set as follow: the average speed of trucks is 8 m/s; the acceleration or deceleration rate of trucks is 2 m/s^2. The turning speed at the cross is 4 m/s. The handing time for a container is 150 s. The parameters are estimated by Dalian container port. For the yard cranes' capacity in the port, the maximum value during one period is set to 30. For mitigating the yard congestions, the condition of high workload is set to a range of [15, 30), while the low workload is [0, 15).

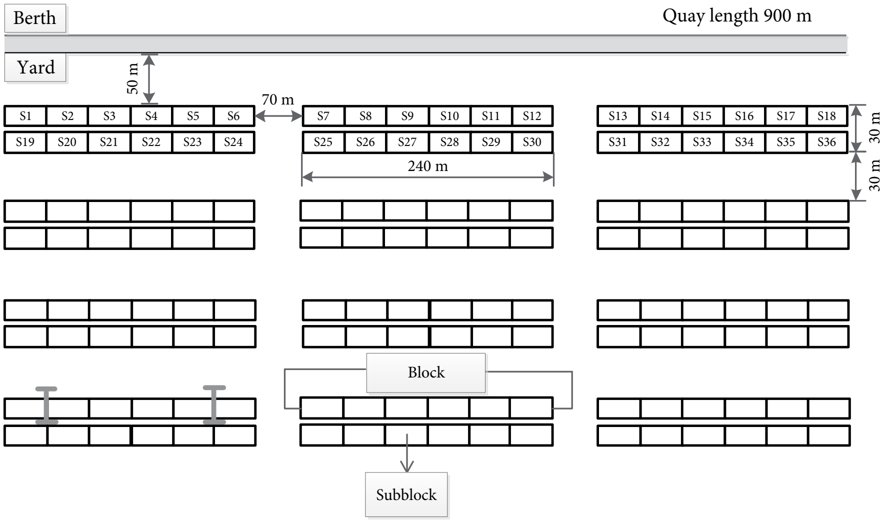


Figure 5. The layout of one yard in Dalian container port

4.2. Evaluating the efficiency of the proposed solution method

4.2.1. Comparing with the optimal results of small-scale instances

In order to evaluate the performance of the SCE-UA algorithm, we compare the results of the SCE-UA algorithm with the CPLEX. Table 2 shows the comparison results between two methods on objective value and computation time. As we can see, the gap is small. Moreover, the solution time of the SCE-UA algorithm is much shorter than the CPLEX. The results in Table 2 validate the efficiency of the SCE-UA algorithm on small-scale instances.

4.2.2. Comparing with the situation of not considering the correlation among adjacent links

The previous studies mainly minimize the total length or time of traffic routes. In other words, it is assumed that the expected travel time of each (i, j) link is determined before the optimization, and is not influenced by the number of container trucks on each link (Zhen *et al.* 2011; Jiang *et al.* 2012). One contribution in this study is the consideration of the correlation among adjacent links when finding the shortest path of container trucks. We would like to explore the influence of considering and not considering the correlation among adjacent links in Table 3. According to Equation (8), σ is evaluated by the data we researched in Dalian port. We investigated the travel time of 15 sequential container trucks at each link and then got the triangular matrix, which contains the variance and covariance of adjacent links. Table 3 shows the results of the two situations, in which column 2 is the results of $M_{OCTP-UTT}$ model considering the link correlation, and column 3 is the results of $M_{OCTP-UTT}$ model not considering link correlation.

In Table 3, the average gap value varies from 3 to 9%. The results show that a yard with the size of 144 subblocks can save the total truck travel time by 9% when its yard template is optimized by considering the correlation among adjacent links. Furthermore, some interesting results could be observed. Figure 6 presents that the gap increases with the growth of subblocks. While in Figure 7, the gap decreases with the increased number of vessels, because the feasible solution decreases and the shortest paths are relatively fixed. As a result, the gap increases with the number of subblocks and the number of vessels increased simultaneously, which can be seen in Figure 8. The three figures indicate that the influence of increased subblocks to the shortest path plays a more important role than that of increased vessels.

4.2.3. Testing the performance of the proposed algorithm by benchmark

In this section, the cases about 20 subblocks and 4 vessels are also solved by the algorithm used in Zhen (2016), which is taken as a benchmark. In Zhen (2016), the authors applied a SWO for changing the sequence of vessels so as to improve the quality of solutions. As the yard template problem is to assign subblocks to vessels, the sequence of the vessel is important for the assignment. In their algorithm, they firstly calculated the path travel time relate to each vessel, and then swap two consecutive vessels if the path travel time of the former is lower than the latter. After swap operation, they will get new solutions. The algorithm stops after observing a number of unchanged solutions. The results of the proposed algorithm and the SWO based algorithm are compared in Table 4. The performance differences between these two heuristics are small. It is observed that the proposed SCE-UA algorithm has longer CPU time (20.35 s on average, compared to16.30 s) but near optimal solution (305530 on average, compared to 305579).

4.2.4. Sensitivity analysis for the closeness degree of *k*-neighbouring links

The correlation among adjacent links is considered by building a k-neighbouring network for link (i, j), denoted as $G_{ij}^k = \left(N_{ij}^k, A_{ij}^k, \Psi_{ij}^k\right)$, satisfying $X_{ij}^{qw} \le k$, $\left(q, w\right) \in A_{ij}^k$. X_{ij}^{qw} is the topological distance between link (i, j) and link (q, w). The sensitivity analysis of the closeness degree is shown in Table 5.

Table 2. Comparison of the optimal solution for the proposed solution method

Case	Best results by CPLEX		SCE-UA algorithm		Gap [%]
	BR_{CPLEX}	CPU time [s]	$BR_{SCE\text{-}UA}$	CPU time [s]	
1	306551	1061	306551	15	0.00
2	310415	1525	310415	25	0.00
3	314279	1558	314279	23	0.00
4	316211	1479	316211	15	0.00
5	302687	2199	302687	20	0.00
6	304560	1413	304681	15	0.04
7	307371	1638	307647	21	0.09
8	308638	2903	308730	19	0.03
9	301502	3773	301562	27	0.02
10	305231	3915	305322	21	0.03
11	300609	3355	300639	22	0.01
12	304444	2001	304778	15	0.11
13	310181	1736	310646	19	0.15
14	312992	1631	313649	17	0.21
15	309531	3059	309654	18	0.04
16	306568	5255	306843	20	0.09
17	307753	5573	307968	18	0.07
18	296315	2845	296492	23	0.06
19	291341	2304	291457	24	0.04
20	290142	2786	290401	16	0.09

Notes:
»» the case consists of 24 subblocks and 4 vessels;
»» $\text{Gap} = \frac{BR_{SCE\text{-}UA} - BR_{CPLEX}}{BR_{CPLEX}} \cdot 100\%$.

Table 3. Comparison of situations with and without considering the correlation among links

Case scale	$M_{OCTP\text{–}UTT}$ model considering link correlation	$M_{OCTP\text{–}UTT}$ model not considering link correlation	Gap [%]
	AVG $BR_{SCE\text{-}UA}$	AVG $BR_{NON\text{-}CORR}$	
24–4	305714	316842	3.64
24–6	354242	366475	3.45
24–8	409534	422578	3.19
36–6	465278	483146	3.84
36–8	501403	519941	3.70
36–9	541576	560624	3.52
36–12	642887	664124	3.30
48–8	631477	660145	4.54
48–12	726632	757140	4.20
48–16	835864	867654	3.80
72–12	957921	1014547	5.91
72–18	1117636	1175655	5.19
72–24	1342180	1406533	4.79
96–16	1304893	1401697	7.42
96–24	1493546	1593462	6.69
96–32	1762462	1864421	5.79
144–24	1943267	2127440	9.48
144–36	2304521	2510462	8.94

Note: The case scale 24–4 means the cases consists of 24 sub-blocks and 4 vessels, etc.

Table 4. Performance of $BR_{SCE\text{-}UA}$ in benchmark instances

Case	SWO based algorithm		SCE-UA algorithm	
	BR_{SWO}	CPU time [s]	$BR_{SCE\text{-}UA}$	CPU time [s]
1	306551	13	306551	17
2	310415	23	310415	25
3	314279	12	314279	16
4	316211	13	316211	23
5	302687	11	302687	20
6	304660	18	304681	17
7	307371	19	307647	23
8	308638	19	308730	19
9	301554	18	301562	27
10	305279	22	305322	21
11	300652	18	300639	22
12	304563	11	304778	15
13	310796	12	310646	26
14	312774	14	313649	19
15	309531	16	309654	18
16	306959	19	306843	19
17	308412	23	307968	21
18	297043	15	296492	23
19	291834	14	291457	17
20	291387	16	290401	19
AVG	305579	16.30	305530	20.35

Note: the case consists of 20 subblocks and 4 vessels.

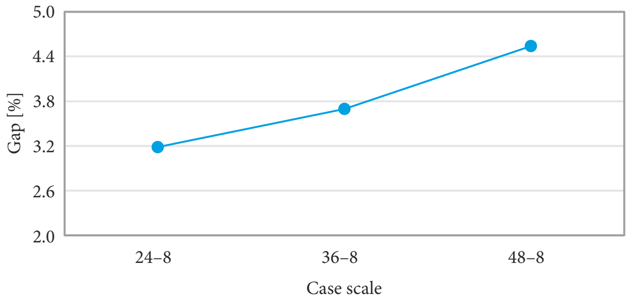


Figure 6. The trend of gap with the increasing subblocks

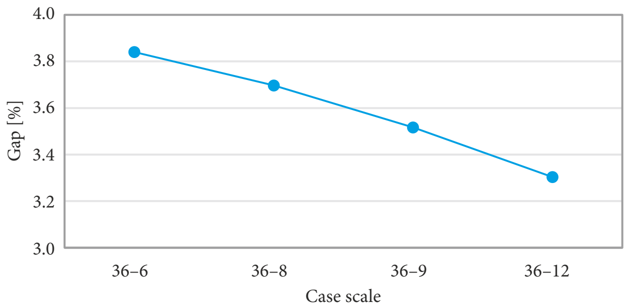


Figure 7. The trend of gap with the increasing vessels

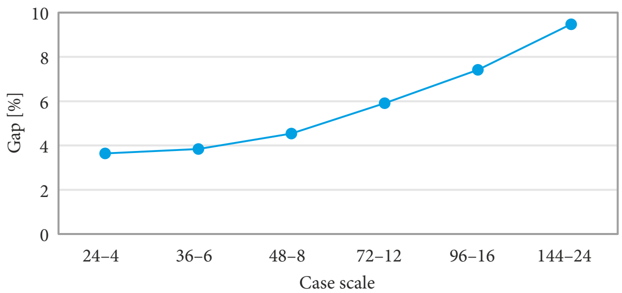


Figure 8. The trend of gap with the increasing subblocks and vessels

As can be seen in Table 5, with the increase of the closeness degree, the average results decreases. In case 24–4 and case 48–8, the results decrease sharply when k = 1, 2, 3, but when k = 4, 5, the results hardly decrease. k = 3 is the break point. Similar phenomena can be seen in case 72–12 and case 144–24, but the break point becomes k = 4. The reason is, in small scale cases, the correlation among adjacent links is not quite obvious, and the covariance nearly equals 0 when the closeness k = 4 or k = 5. Therefore, the results in small scale cases decrease a little when k = 4, 5. While in large scale cases such as 72–12 and 144–24, the path becomes complicated and easy to be influenced by adjacent links, so the results decrease sharply when k = 1, 2, 3, 4, and hardly decrease when k = 5. For simplicity, we choose k = 3 in this paper.

4.2.5. Sensitivity analysis for yard scales and number of yard trucks

In this section, cases with different yard scales and number of yard trucks are designed. Each case is done by 5 simulation runs. If the yard scale changes, the travel distance of container trucks changes accordingly. The results are shown in Table 6. The gap of results decreases with

Table 5. Results of cases with different closeness degree

Case scale	Closeness degree k	AVG BR_{SCE-UA}
24–4	1	313277
	2	309651
	3	**305714**
	4	**305710**
	5	**305707**
48–8	1	643576
	2	638549
	3	**631477**
	4	**631470**
	5	**631461**
72–12	1	981251
	2	964339
	3	957921
	4	**949387**
	5	**949374**
144–24	1	2118936
	2	2105894
	3	1943267
	4	**1942763**
	5	**1942751**

Note: the case scale 24–4 means the cases consists of 24 subblocks and 4 vessels, etc.

Table 6. Results of cases with different yard scales

Case scale	Number of subblocks in one block	Mean value	Min value	Max value	Gap [%]
4–4	4	249453	243824	256577	5.23
	5	279546	274356	286630	4.47
	6	305714	301338	311843	3.49
	8	368943	361356	371854	2.91
	10	435585	431320	439379	1.87
	12	463246	459111	465407	1.37
	15	527520	525894	530619	0.90
8–4	4	312493	308203	324221	5.20
	5	368905	361894	376965	4.16
	6	429534	420439	434218	3.28
	8	499543	495655	508230	2.54
	10	553940	550283	556932	1.21
	12	593063	590893	596220	0.90
	15	624952	622956	627018	0.65
8–8	4	522371	517303	529209	2.30
	5	584910	579281	590366	1.91
	6	631477	625683	638524	2.05
	8	714859	710235	718090	1.11
	10	750922	747128	754006	0.92
	12	793274	790283	796212	0.75
	15	852038	850173	854293	0.48

Notes:
»» the case scale 4–4 means the cases consists of 4 blocks and 4 vessels, etc.;
»» $\text{Gap} = \frac{\text{max value} - \text{min value}}{\text{min value}} \cdot 100\%$.

the increase of subblocks in one block. The reason is that less truck interruptions happen on the path if the length of block increases, because the containers are dispersed in the subblocks and avoid congestion. Therefore, the travel time of container trucks would become reliable. Furthermore, the number of blocks seems to have no apparent effects to the gap when comparing the case 4 blocks and 4 vessels to the case 8 blocks and 4 vessels. The number of vessels could reduce the gap apparently when comparing the case 8 blocks and 4 vessels to the case 8 blocks and 8 vessels. The port operators could consider to enlarge the number of subblocks in one block.

Next, cases with different number of container trucks entering into yard are designed. The results are shown in Table 7. It seems that the optimal number of container trucks is different in different case scale. The optimal number of container truck is 125 vel/h, 150 vel/h and 225 vel/h for the case scale of 24 subblocks 4 vessels, 48 subblocks and 8 vessels and 72 subblocks and 12 vessels, respectively.

Table 7. Results of cases with different number of container trucks

Case scale	Number of container trucks	AVG $BR_{SCE\text{-}UA}$
24–4	50	341680
	75	319034
	100	308482
	125	**304952**
	150	305714
	175	310368
	200	329487
	225	351082
	250	398495
48–8	50	771464
	75	720633
	100	689051
	125	652482
	150	**631477**
	175	646392
	200	661543
	225	680372
	250	703189
72–12	50	1540330
	75	1284954
	100	1103943
	125	1028492
	150	957921
	175	913782
	200	874859
	225	**842924**
	250	850922

Note: the case scale 24–4 means the cases consists of 24 subblocks and 4 vessels, etc.

4.2.6. Sensitivity analysis for confidence level α

The other contribution of this study is considering the probability of arriving at the destination within the expected travel time. As the yard operation is complex, e.g. the yard crane moves from one side to the other side at uncertain time, container trucks may arrive at the destination early or late. The confidence level $\alpha \in (0,1)$ is the probability that container trucks arrive at the destination within the expected travel time. The on-time arrival probability α represents port operator's attitude towards risks of being late ($\alpha > 0.5$, $\alpha = 0.5$ and $\alpha < 0.5$ for risk-averse, risk-neutral, and risk-seeking attitudes, respectively). The value of α can be predetermined based on port operators' purpose. $\alpha = 0.5$ means the operators take no account of potential risk in the yard. In addition, $\alpha > 0.5$ means the operators pay attention to the potential risk, which often happens in heavy workload ports. $\alpha < 0.5$ means the operators ignore the potential risk, which often happens in low workload ports. As shown in Table 8, the $BR_{SCE\text{-}UA}$ is different with different confidence level α. When $\alpha = 0.1$, the case 11 obtains the minimum travel time cost. But the case 20 obtains the minimum travel time cost when $\alpha = 0.5$.

Table 8. Results of cases with different confidence level α

Case	$BR_{SCE\text{-}UA}$		
	$\alpha = 0.1$	$\alpha = 0.5$	$\alpha = 0.9$
1	295576	306551	317526
2	305526	310415	315304
3	311883	314279	316675
4	311178	316211	321244
5	290139	302687	315235
6	303358	304681	305762
7	304285	307647	310457
8	304875	308730	312401
9	300534	301562	302470
10	298049	305322	312413
11	**284561**	300639	316657
12	304206	304778	304682
13	309107	310646	311255
14	311970	313649	314014
15	305014	309654	314048
16	303381	306843	309755
17	296939	307968	318567
18	295083	296492	297547
19	292395	291457	295463
20	294081	**290401**	294721

Note: the case consists of 24 subblocks and 4 vessels.

Conclusions

The paper studies the optimization of the container truck paths with uncertain travel time in container ports. In the proposed model, the link travel time influenced by the yard congestion is formulated as a basic work. In addition, the reliable shortest path related with the correlation among adjacent links is discussed in our problem. A small illustrative example shows that the shortest path may change considering the correlation among adjacent links. It is necessary to emphasize that the container truck path optimization should not be separately treated. The problem is influenced by the yard template planning.

Considering the intricate workload in a yard, this paper proposes the confidence level α to cover different situations in container ports. In the real application, it could provide different strategies in different conditions (peak season or low season), which is helpful for container port operation and schedule. A mixed-integer programming model is proposed to minimize the total travel time of container trucks in the yard. The combination of SCE-UA and A^* algorithm is developed to solve the model. The cases are presented to validate the availability of the model. The results show that 9% of the travel time of container trucks can be saved when the correlation among adjacent links is considered. It may raise an inspired idea to yard management and equipment scheduling in ports, especially in transhipment ports.

However, there are limitations on the current model. For example, the container truck drivers may have different driving behaviour. It may be better to use car following model to calculate the expected travel time on each link. Besides, the arrival time of each vessel is determined before the yard template planning. In practice, the arrival time and operation time of vessels are stochastic. These limitations will be further studied in our future researches.

Funding

This research was supported by National Natural Science Foundation of China (Grants No U1811463, 71961137008), the State Key Laboratory of Structural Analysis for Industrial Equipment (Grant No S18307), and partially funded by Beijing Advanced Innovation Center for Big Data and Brain Computing of Beihang Universtity.

Author contributions

Jiaming Liu and *Bin Yu* conceived the study and were responsible for the design and development of the data analysis.

Jiaming Liu, *Baozhen Yao* were responsible for data collection and analysis.

Wenxuan Shan and *Baozhen Yao* were responsible for data interpretation.

Yao Sun wrote the first draft of the article.

Disclosure statement

Authors are required to include a statement at the end of their article to declare whether or not they have any competing financial, professional, or personal interests from other parties.

References

Cao, J. X.; Lee, D.-H.; Chen, J. H.; Shi, Q. 2010. The integrated yard truck and yard crane scheduling problem: Benders' decomposition-based methods, *Transportation Research Part E: Logistics and Transportation Review* 46(3): 344–353. https://doi.org/10.1016/j.tre.2009.08.012

Chan, K. S.; Lam, W. H. K.; Tam, M. L. 2009. Real-time estimation of arterial travel times with spatial travel time covariance relationships, *Transportation Research Record: Journal of the Transportation Research Board* 2121: 102–109. https://doi.org/10.3141/2121-11

Chang, D.; Jiang, Z.; Yan, W.; He, J. 2010. Integrating berth allocation and quay crane assignments, *Transportation Research Part E: Logistics and Transportation Review* 46(6): 975–990. https://doi.org/10.1016/j.tre.2010.05.008

Chang, T.-S.; Nozick, L. K.; Turnquist, M. A. 2005. Multiobjective path finding in stochastic dynamic networks, with application to routing hazardous materials shipments, *Transportation Science* 39(3): 383–399. https://doi.org/10.1287/trsc.1040.0094

Chen, A.; Ji, Z. 2005. Path finding under uncertainty, *Journal of Advanced Transportation* 39(1): 19–37. https://doi.org/10.1002/atr.5670390104

Chen, B. Y.; Lam, W. H. K.; Sumalee, A.; Li, Z.-L. 2012. Reliable shortest path finding in stochastic networks with spatial correlated link travel times, *International Journal of Geographical Information Science* 26(2): 365–386. https://doi.org/10.1080/13658816.2011.598133

Chen, B. Y.; Lam, W. H. K.; Sumalee, A.; Li, Q.; Tam, M. L. 2014. Reliable shortest path problems in stochastic time-dependent networks, *Journal of Intelligent Transportation Systems: Technology, Planning, and Operations* 18(2): 177–189. https://doi.org/10.1080/15472450.2013.806851

Chen, C.; Tian, Z.; Yao, B. 2019. Optimization of two-stage location–routing–inventory problem with time-windows in food distribution network, *Annals of Operations Research* 273(1–2): 111–134. https://doi.org/10.1007/s10479-017-2514-3

Chen, L.; Langevin, A.; Lu, Z. 2013. Integrated scheduling of crane handling and truck transportation in a maritime container terminal, *European Journal of Operational Research* 225(1): 142–152. https://doi.org/10.1016/j.ejor.2012.09.019

Chen, X.; Zhou, X.; List, G. F. 2011. Using time-varying tolls to optimize truck arrivals at ports, *Transportation Research Part E: Logistics and Transportation Review* 47(6): 965–982. https://doi.org/10.1016/j.tre.2011.04.001

Duan, Q.; Sorooshian, S.; Gupta, V. K. 1994. Optimal use of the SCE-UA global optimization method for calibrating watershed models, *Journal of Hydrology* 158(3–4): 265–284. https://doi.org/10.1016/0022-1694(94)90057-4

Fan, L.; Wilson, W. W.; Dahl, B. 2012. Congestion, port expansion and spatial competition for US container imports, *Transportation Research Part E: Logistics and Transportation Review* 48(6): 1121–1136. https://doi.org/10.1016/j.tre.2012.04.006

Han, Y.; Lee, L. H.; Chew, E. P.; Tan, K. C. 2008. A yard storage strategy for minimizing traffic congestion in a marine container transshipment hub, *OR Spectrum* 30(4): 697–720. https://doi.org/10.1007/s00291-008-0127-6

Hart, P. E.; Nilsson, N. J.; Raphael, B. 1968. A formal basis for the heuristic determination of minimum cost paths, *IEEE Transactions on Systems Science and Cybernetics* 4(2): 100–107. https://doi.org/10.1109/TSSC.1968.300136

He, J.; Huang, Y.; Yan, W.; Wang, S. 2015. Integrated internal truck, yard crane and quay crane scheduling in a container terminal considering energy consumption, *Expert Systems with Applications* 42(5): 2464–2487. https://doi.org/10.1016/j.eswa.2014.11.016

Huang, B.; Wu, Q.; Zhan, F. B. 2007. A shortest path algorithm with novel heuristics for dynamic transportation networks, *International Journal of Geographical Information Science* 21(6): 625–644. https://doi.org/10.1080/13658810601079759

Jiang, X.; Lee, L. H.; Chew, E. P.; Han, Y.; Tan, K. C. 2012. A container yard storage strategy for improving land utilization and operation efficiency in a transshipment hub port, *European Journal of Operational Research* 221(1): 64–73. https://doi.org/10.1016/j.ejor.2012.03.011

Jin, J. G.; Lee, D.-H.; Hu, H. 2015. Tactical berth and yard template design at container transshipment terminals: A column generation based approach, *Transportation Research Part E: Logistics and Transportation Review* 73: 168–184. https://doi.org/10.1016/j.tre.2014.11.009

Kaveshgar, N.; Huynh, N. 2015. Integrated quay crane and yard truck scheduling for unloading inbound containers, *International Journal of Production Economics* 159: 168–177. https://doi.org/10.1016/j.ijpe.2014.09.028

Kim, K. H.; Bae, J. W. 1998. Re-marshaling export containers in port container terminals, *Computers & Industrial Engineering* 35(3–4): 655–658. https://doi.org/10.1016/S0360-8352(98)00182-X

Kim, K. H.; Kim, K. Y. 2007. Optimal price schedules for storage of inbound containers, *Transportation Research Part B: Methodological* 41(8): 892–905. https://doi.org/10.1016/j.trb.2007.02.001

Lee, D.-H.; Jin, J. G. 2013. Feeder vessel management at container transshipment terminals, *Transportation Research Part E: Logistics and Transportation Review* 49(1): 201–216. https://doi.org/10.1016/j.tre.2012.08.006

Lee, H. L.; Chew, E. P.; Tan, K. C.; Han, Y. 2006. An optimization model for storage yard management in transshipment hubs, *OR Spectrum* 28(4): 539–561. https://doi.org/10.1007/s00291-006-0045-4

Lu, Y.; Le, M. 2014. The integrated optimization of container terminal scheduling with uncertain factors, *Computers & Industrial Engineering* 75: 209–216. https://doi.org/10.1016/j.cie.2014.06.018

Maloni, M.; Paul, J. A. 2013. Evaluating capacity utilization options for US west coast container ports, *Transportation Journal* 52(1): 52–79. https://doi.org/10.5325/transportationj.52.1.0052

Meng, Q.; Wang, S.; Andersson, H.; Thun, K. 2014. Containership routing and scheduling in liner shipping: overview and future research directions, *Transportation Science* 48(2): 265–280. https://doi.org/10.1287/trsc.2013.0461

Moccia, L.; Cordeau, J.-F.; Monaco, M. F.; Sammarra, M. 2009. A column generation heuristic for a dynamic generalized assignment problem, *Computers & Operations Research* 36(9): 2670–2681. https://doi.org/10.1016/j.cor.2008.11.022

Moorthy, R.; Teo, C.-P. 2006. Berth management in container terminal: the template design problem, *OR Spectrum* 28(4): 495–518. https://doi.org/10.1007/s00291-006-0036-5

Nie, Y.; Wu, X. 2009. Shortest path problem considering on-time arrival probability, *Transportation Research Part B: Methodological* 43(6): 597–613. https://doi.org/10.1016/j.trb.2009.01.008

Nikolova, E. 2009. *Strategic Algorithms*. PhD Thesis. Massachusetts Institute of Technology, MA, US. 212 p. Available from Internet: https://dspace.mit.edu/handle/1721.1/54673

Nishimura, E.; Imai, A.; Janssens, G. K.; Papadimitriou, S. 2009. Container storage and transshipment marine terminals, *Transportation Research Part E: Logistics and Transportation Review* 45(5): 771–786. https://doi.org/10.1016/j.tre.2009.03.003

Nishimura, E.; Imai, A.; Papadimitriou, S. 2005. Yard trailer routing at a maritime container terminal, *Transportation Research Part E: Logistics and Transportation Review* 41(1): 53–76. https://doi.org/10.1016/j.tre.2003.12.002

Peng, Z.; Shan, W.; Jia, P.; Yu, B.; Jiang, Y.; Yao, B. 2020. Stable ride-sharing matching for the commuters with payment design, *Transportation* 47(1): 1–21. https://doi.org/10.1007/s11116-018-9960-x

Peng, Z.; Shan, W.; Guan, F.; Yu, B. 2016. Stable vessel-cargo matching in dry bulk shipping market with price game mechanism, *Transportation Research Part E: Logistics and Transportation Review* 95: 76–94. https://doi.org/10.1016/j.tre.2016.08.007

Roy, D.; Gupta, A.; De Koster, R. B. M. 2016. A non-linear traffic flow-based queuing model to estimate container terminal throughput with AGVs, *International Journal of Production Research* 54(2): 472–493. https://doi.org/10.1080/00207543.2015.1056321

Shan, W.; Yan, Q.; Chen, C.; Zhang, M.; Yao, B.; Fu, X. 2019. Optimization of competitive facility location for chain stores, *Annals of Operations Research* 273(1–2): 187–205. https://doi.org/10.1007/s10479-017-2579-z

Shao, H.; Lam, W. H. K.; Chan, K. S. 2004. The problem of searching the reliable path for transportation networks with uncertainty, in *Proceeding of 9th Conference of the Hong Kong Society for Transportation Studies*, Hong Kong, China, 226–234.

Vis, I. F. A.; De Koster, R. 2003. Transshipment of containers at a container terminal: an overview, *European Journal of Operational Research* 147(1): 1–16. https://doi.org/10.1016/S0377-2217(02)00293-X

Wang, S.; Liu, Z.; Bell, M. G. H. 2015. Profit-based maritime container assignment models for liner shipping networks, *Transportation Research Part B: Methodological* 72: 59–76. https://doi.org/10.1016/j.trb.2014.11.006

Yao, B.; Chen, C.; Song, X.; Yang, X. 2019a. Fresh seafood delivery routing problem using an improved ant colony optimization, *Annals of Operations Research* 273(1): 163–186. https://doi.org/10.1007/s10479-017-2531-2

Yao, B.; Chen, C.; Zhang, L.; Yu, B.; Wang, Y. 2019b. Allocation method for transit lines considering the user equilibrium for operators, *Transportation Research Part C: Emerging Technologies* 105: 666–682. https://doi.org/10.1016/j.trc.2018.09.019

Yu, B.; Wang, H.; Shan, W.; Yao, B. 2018. Prediction of bus travel time using random forests based on near neighbors, *Computer-Aided Civil and Infrastructure Engineering* 33(4): 333–350. https://doi.org/10.1111/mice.12315

Yu, B.; Wang, H.; Song, X.; Zhao, Z.; Tian, Z.; Yao, B. 2020. Optimising subordinate net points layout of express enterprise with SCE-UA algorithm, *Proceedings of the Institution of Civil Engineers – Transport* 173(1): 51–58. https://doi.org/10.1680/jtran.16.00092

Zeng, W.; Church, R. L. 2009. Finding shortest paths on real road networks: the case for A*, *International Journal of Geographical Information Science* 23(4): 531–543. https://doi.org/10.1080/13658810801949850

Zhang, C.; Liu, J.; Wan, Y.-W.; Murty, K. G.; Linn, R. J. 2003. Storage space allocation in container terminals, *Transportation Research Part B: Methodological* 37(10): 883–903. https://doi.org/10.1016/S0191-2615(02)00089-9

Zhang, M.; Batta, R.; Nagi, R. 2009. Modeling of workflow congestion and optimization of flow routing in a manufacturing/warehouse facility, *Management Science* 55(2): 267–280. https://doi.org/10.1287/mnsc.1080.0916

Zhen, L. 2016. Modeling of yard congestion and optimization of yard template in container ports, *Transportation Research Part B: Methodological* 90: 83–104. https://doi.org/10.1016/j.trb.2016.04.011

Zhen, L. 2015. Tactical berth allocation under uncertainty, *European Journal of Operational Research* 247(3): 928–944. https://doi.org/10.1016/j.ejor.2015.05.079

Zhen, L. 2013. Yard template planning in transshipment hubs under uncertain berthing time and position, *Journal of the Operational Research Society* 64(9): 1418–1428. https://doi.org/10.1057/jors.2012.108

Zhen, L.; Chew, E. P.; Lee, L. H. 2011. An integrated model for berth template and yard template planning in transshipment hubs, *Transportation Science* 45(4): 483–504. https://doi.org/10.1287/trsc.1100.0364

Zhen, L.; Yu, S.; Wang, S.; Sun, Z. 2019. Scheduling quay cranes and yard trucks for unloading operations in container ports, *Annals of Operations Research* 273(1–2): 455–478. https://doi.org/10.1007/s10479-016-2335-9