https://dx.doi.org/10.1016/j.tre.2022.102992

Local Container Drayage Problem with Improved Truck Platooning Operations

Xiaoyuan Yan[a], Min Xu[a], Chi Xie[b]*

[a] Department of Industrial and Systems Engineering, The Hong Kong Polytechnic University, Hung Hom, Hong Kong

[b]Department of Transportation Engineering, Tongji University, Shanghai, China

Abstract

Truck platooning has been identified as a promising technology to save labor force for the local container drayage problem (LCDP). However, existing studies for the platoon-based LCDP assumed that each driver was attached to respective leading truck throughout the service process, and the driverless following trucks dropped off at customer sites cannot further serve the drayage requests until regaining guidance, resulting in low efficiency and utilization of drivers and trucks. To overcome this issue, this paper examines the LCDP under a novel improved platooning operation mode (IPOM), where the drivers are not necessarily attached to their respective leading trucks but can move by alternative transport modes to perform subsequent tasks. The objective is to determine the optimal numbers, routes, and schedules of the drivers and trucks that minimize the total operational cost to complete all the delivery and pickup requests of customers. A mixed-integer linear programming (MILP) model is developed to capture the features pertaining to the new platooning mode. A heuristic construction method incorporating simulated annealing is proposed to solve the problem. Numerical experiments are carried out to evaluate the proposed model and solution method and quantify the advantages of the improved platooning mode. Sensitivity analysis is also conducted to explore the impacts of some major influential factors on the system performance and derive managerial insights.

Keywords: local container drayage, truck platooning, drivers' movement, MILP model, simulated annealing

* Corresponding author

1. Introduction

Global trade largely relies on container transportation, in which the intermodal containers move among seaports, terminals, and local customers by vessels, rails, and trucks (DHL, 2021). In the entire transportation chain, the short-haulage container transportation by trucks between a terminal and the local customers in which carriers provide delivery and pickup services for consignees and shippers, respectively, is referred to as the local container drayage problem (LCDP). Although the travel distance is very short in comparison to that of the ocean-going and land-based long-distance haulage, the drayage operational cost typically accounts for a large portion, i.e., up to 80%, of the total cost of the entire container transportation chain (Macharis and Bontekoning, 2004). Therefore, the cost-efficiency of local container drayage services is of particular importance to the profitability of intermodal container transportation. In traditional container drayage operation mode, each consignee/shipper requires one driver to depart from the terminal with a loaded/an empty truck and wait at the customer site for the unpacking/packing work; after the unpacking/packing work is completed, the driver should haul the emptied/loaded truck back to the terminal. In recent decades, the drayage operation mode has been improved in many ways with the aim to reduce the total operational cost, such as the sharing of empty trucks in which an emptied truck released from a consignee can be reused by a shipper without returning to the terminal (Imai et al., 2007). With the development of connected and autonomous vehicle (CAV) technology, the emerging truck platooning in which a set of autonomous trucks travel together with small headways and only the leading truck requires driver's intervention helps to reduce the labor force in LCDP. Moreover, it has been found by many field experiments that truck platooning can also contribute to fuel consumption reduction via lowered aerodynamic drag, especially for the following trucks in a platoon, which achieve significant fuel savings of about 10~20% (Davila et al., 2013; Lammert et al., 2014; Lu and Shladover, 2014). How to reap the maximum cost efficiency by leveraging the emerging technologies is one of the major problems facing container drayage service providers.

1.1 Literature review

Over the past decades, many studies have been conducted for the LCDP considering various practical constraints and features, such as resource constraints, flexible orders, and uncertainties with respect to travel speeds, etc. (Chen et al., 2022). For example, Zhang et al. (2011) examined the LCDP considering the limited numbers of trucks and empty containers

available at depot. They formulated a mixed-integer programming (MIP) model on a directed graph and proposed a reactive tabu search-based algorithm to solve the problem. Escudero et al. (2013) investigated the daily drayage problem considering travel time uncertainty caused by unexpected events. They formulated a real-time optimization model and proposed a genetic algorithm for the dynamic planning and management of the fleet. Zhang et al. (2014) formulated a mixed-integer nonlinear programming model for dynamic container drayage services with flexible orders of containers and developed four different solution methods to solve the problem. Benantar et al. (2020) tackled a real-life container drayage problem considering the availability of containers. Chen et al. (2021) proposed a new variant of LCDP considering heterogeneous trucks with multiple loads, where a hybrid heuristic method based on the cheapest feasible insertion mechanism and variable neighborhood search scheme were proposed to solve large-scale instances. The LCDP with multi-resource constraints was investigated by Zhang et al. (2020), aiming to minimize the number of trucks in operation and the total working time of trucks. In this work, a determined-activities-on-vertex graph was adopted to reformulate the LCDP as a graph-based bi-level nonlinear mixed-integer mathematical model. Moghaddam et al. (2020) presented a two-layered network formulation for a generalized LCDP, in which a graph structure with two layers was used to deal with the time and capacity constraints in an efficient way. The multi-period multi-trip LCDP was examined by Bruglieri et al. (2021), where the time horizon was divided into multiple discrete periods, and trucks can perform multiple trips within each period. Cui et al. (2022) proposed a new variant of LCDP with trailer reposition that allows the separation of tractors and trailers, where a two-stage solution framework based on a hybrid large neighbourhood search and tabu search heuristic was formulated to solve the LCDP. There are also many studies for the LCDP under new cost-efficient drayage operation modes with high utilization of resources. For example, Imai et al. (2007) addressed an LCDP considering the sharing of empty containers and developed a heuristic algorithm based on Lagrangian relaxation to obtain near-optimal solutions by solving several sub-problems. Afterward, to further improve the drayage efficiency, Xue et al. (2014) proposed the tractor-trailer separable mode where a tractor can be separated from its companion trailer, enabling the drivers to continue to serve the next customer without waiting at the current customer sites for the unpacking/packing work. They formulated this problem as a vehicle routing problem with temporal constraints. A tabu search algorithm was developed to solve the problem. Song et al. (2017) extended the model in Xue et al. (2014) by considering necessary container maintenance at the depots. They treated the problem as an

asymmetric vehicle routing problem and formulated it as a MILP model. A branch-and-price-and-cut algorithm was proposed to solve the problem. Both Song et al. (2017) and Xue et al. (2014) assumed that a tractor could only carry one trailer at a time. Later on, some studies relaxed this assumption by allowing a tractor to haul two or more trailers (containers) at a time (Wang and Zhang, 2019; Zhang et al., 2018; Zhang et al., 2020). Specifically, Zhang et al. (2018) explored the foldable container drayage services in which a truck can carry a loaded container or multiple folded containers at a time. A range-based truck state transition method and an improved reactive tabu search algorithm were developed to address the problem. Wang and Zhang (2019) proposed a double-trailer separable mode for the LCDP, where a tractor can carry two trailers. This was further extended to the multi-trailer drop-and-pull mode by Zhang et al. (2020), in which a mixed-integer nonlinear programming model and a backtracking adaptive threshold accepting algorithm were developed to address the problem.

Quite recently, enabled by the CAV technology, vehicle/truck platooning holds great potential for further cost reduction and efficiency gains in the transportation process (Chen et al., 2021; Engholm et al., 2021; Larsen et al., 2019; Mahdinia et al., 2020; Sivanandham et al., 2020; SCHENKER, 2019). As recently reviewed by Ghosal et al. (2021) and Bhoopalam et al. (2018), truck platooning is capable of reducing resource costs while ensuring a high safety for the drivers via a reliable network architecture and the communication protocols, which demonstrates the feasibility of the platooning techniques in practical applications. For example, She et al. (2022) presented a network design model to optimize the positioning configuration of autonomous and connected truck platoons in dedicated lanes, where the problem was formulated as an integrated bi-level optimization model. Liang et al. (2022) investigated the dynamic resource allocation problem for truck platoon in a vehicle network based on the semi-Markov decision process and deep reinforcement learning, aiming to maximize the total profit considering the resource cost and income balance of the transportation system. The effects of truck platoons on the traffic flow were investigated by Calvert et al. (2019), in which the impact of truck platoons on the traffic states, truck gap settings, platoon sizes, and the share of equipped trucks were examined. In particular, there have been some studies incorporating the emerging truck platooning technology into the LCDP (Xue et al., 2021; You et al., 2020; You et al., 2022). Among the studies, You et al. (2020) were the first to address a general LCDP considering the sharing of empty trucks under the truck platooning operation mode, in which only the leading truck required driver's intervention and was autonomously followed by a set

of unmanned trucks. They also assumed that all the drivers and trucks must return to the terminal after completing all the delivery/pickup requests of the customers within a prescribed maximum planning horizon of the drivers. A heuristic method based on the ant colony algorithm was proposed to solve the problem. Xue et al. (2021) later incorporated the fuel-saving benefits of truck platooning in a special platoon-based LCDP model, in which trucks from different platoons cannot platoon together throughout the service process and the sharing of empty trucks was not allowed. Very recently, You et al. (2022) considered the load-dependent fuel cost and the multi-trip feature of drivers in the LCDP with truck platooning, in which a driver can access the terminal multiple times in a working day, but the sharing of empty trucks among customers was not allowed. They formulated a MIP model and proposed a tailored branch-and-price-and-cut algorithm to solve the problem.

1.2 Objective and contributions

The truck platooning modes considered in previous studies all assumed that the unmanned following trucks dropped off at a customer site for conducting the unpacking/packing work cannot move without a driver. In other words, the dropped-off trucks cannot be utilized to serve the next customer until regaining the guidance of a leading truck with a driver, which results in low utilization of the trucks. Moreover, considering that the unpacking/packing tasks always take a relatively long time (often up to a few hours), the drivers in the leading trucks would experience unnecessary waiting time for the completion of the tasks, which is a waste of labor resources. Therefore, we propose a novel improved platooning operation mode (IPOM), in which trucks are allowed to freely form platoons with each other using autonomous driving technologies subject to a maximum platoon size constraint; and notably, each driver is not necessarily always attached to one leading truck throughout the service process, i.e., the driver does not have to wait at a customer site for the packing/unpacking task of the leading truck, but can move to drive other (leading) trucks at different customer sites by available alternative transport modes, e.g., motorcycle, taxi, and shared vehicle, for subsequent tasks. The proposed IPOM unlocks the flexibility and improve the utilization of the drivers and trucks so as to improve the operation efficiency and reduce the cost. We also consider the sharing of empty trucks and the fuel-saving benefit of truck platooning. In particular, different from Xue et al. (2021) which did not allow trucks from different platoons to platoon together, we model the fuel savings by flexible platoons formed during the service process by trucks from different platoons upon departure from the terminal.

The objective of this study is to determine the optimal numbers, routes, and schedules of the drivers and trucks that minimize the total operational cost, including the driver employment cost, the truck deployment cost, the fuel consumption cost of trucks, and the travel cost incurred by using alternative transport modes, in order to complete all the delivery and pickup requests for a group of customers without violating the daily maximum planning horizon.

To achieve the objective, we will formulate a MILP model for the proposed problem. Due to the complicated structure of the problem, even for a small-sized instance, the proposed MILP model is computationally intractable for the commercial solvers. Therefore, a customized heuristic construction method incorporating simulated annealing is proposed to obtain near-optimal solutions for practical-sized problem instances. Extensive numerical experiments are conducted to demonstrate the efficacy of the proposed model and solution method and quantify the advantages of the IPOM for the LCDP. Sensitivity analysis is also conducted to examine the effects of several influential factors on the system performance and derive managerial insights for relevant stakeholders. To the best of our knowledge, no one has ever considered the proposed IPOM in the platoon-based container drayage services. The aforementioned literature review validates the novelty and necessity of this study.

The remainder of this paper is organized as follows. Assumptions, notations, and problem statement are elaborated in Section 2. A MILP model for the proposed problem is formulated in Section 3. A simulated annealing-based heuristic construction method is developed in Section 4, followed by the numerical experiments and sensitivity analyses in Section 5. Finally, Section 6 presents conclusions and future research directions.

2. Assumptions, notations, and problem statement

We consider a drayage firm who schedules a group of drivers and an autonomous truck fleet to serve a set of delivery and pickup customers distributed in the local area around a terminal. A fully loaded truck will depart from the terminal to a delivery customer and wait at the customer site for unpacking. After the unpacking work is completed, the empty truck will then go back to the terminal or a pickup customer for reuse. In other words, each pickup customer will be served either by an empty truck from the terminal or the one released from a delivery customer served previously. After the packing work at the pickup customer is done, the loaded truck will then transport back to the terminal. In summary, each customer corresponds to two tasks in different stages before and after the packing/unpacking work of the service request and should be visited twice by drivers, who are not necessarily the same. The

service duration time between the two tasks of a customer must not be less than the corresponding container unpacking/packing time. We consider an improved platooning operation mode (IPOM) as follows: during the transportation process, trucks can freely form platoons with each other to save fuel subject to the maximum platoon size constraint, in which only the leading truck requires a driver and is autonomously followed by several driverless trucks. Kindly note that trucks from different platoons upon departure from the terminal are allowed to platoon together. Each driver is not necessarily attached to a leading truck but can move by alternative transport modes to other customer sites to drive other trucks. In addition, each driver will undertake at most one route. All the drivers and trucks should return to the terminal within a pre-specified maximum planning horizon.

We define the LCDP over a graph $\mathbf{G}=(\mathbf{N},\mathbf{A})$, where $\mathbf{N}$ is the node set and $\mathbf{A}=\{(i,j)\,|\,i,j\in\mathbf{N},i\neq j\}$ is the arc set. Note that $\mathbf{N}=\mathbf{D}\bigcup\mathbf{P}\bigcup\{0\}$, where $\mathbf{D}$ and $\mathbf{P}$ denote the set of delivery and pickup customer nodes, respectively, and 0 represents the terminal, which gathers drivers in set $\mathbf{V}$ and parks homogeneous autonomous trucks with single-container capacity in set $\mathbf{K}$. In the proposed IPOM, drivers may move by trucks or alternative transport modes from one to another customer site. Hence, there will be two travel times t_{ij} and t'_{ij} associated with each arc $(i,j)\in\mathbf{A}$, representing the required travel time of arc (i,j) by taking trucks or alternative transport modes, respectively. The fuel consumption cost of the trucks and the travel cost of the alternative transport modes to traverse arc $(i,j)\in\mathbf{A}$ are denoted by c_{ij} and c'_{ij}, respectively. During the transportation process, trucks can freely platoon together for more fuel savings and the number of trucks traveling together on each arc cannot exceed the maximum platoon size $_L$ for safety concerns. Regarding the fuel-saving effect by platooning, we assume that the following trucks in the platoons can save fuel by a ratio of η, while the leading trucks experience no fuel savings. For example, the total fuel consumption cost incurred by a platoon made up by m trucks traversing arc (i,j) will be calculated by $c_{ij}[1+(1-\eta)(m-1)]$. In addition, the daily fixed costs of a driver and a truck are denoted by λ_1 and λ_2, respectively. To fully present the LCDP of our interest, we will elaborate on the operational characteristics and truck movement tracking in the following subsections.

2.1 The improved platooning operation mode

In the existing platooning modes for LCDP, the unmanned following trucks dropped

off at a customer site for conducting the unpacking/packing work cannot move without a driver. In other words, the dropped-off trucks cannot be utilized to serve the next customer until regaining the guidance of a leading truck with a driver, which results in low utilization of the trucks. Moreover, considering that the unpacking/packing tasks always take a relatively long time (often up to a few hours), the drivers in the leading truck would experience unnecessary waiting time for the completion of the tasks, which is a waste of labor resources. Therefore, it is of paramount importance to develop an improved platooning operation mode that can unlock the flexibility of the drivers and the trucks so as to improve the operation efficiency and reduce the cost. Motivated by this, in this study, we propose the IPOM for the LCDP to improve the utilization of drivers and trucks. Under this mode, besides the truck, a driver can use alternative transport modes available nearby to head for other customer sites to conduct service tasks, i.e., drive back the trucks dropped off previously. In nowadays' modern transportation network, there are a variety of transport modes, such as taxis, shared vehicles, ride-hailing, motorcycles, and buses, which widely cover the urban areas, suburbs, and industrial zones. In addition, with the rapid development of "Internet+ Travel solutions", various types of new internet-based travel modes enabled by the easy-to-use apps, i.e., ride-sharing platforms, carpool software, taxi-hailing apps, have already achieved a high market penetration, making the movement between different customer sites much easier and more convenient. In summary, thanks to the extensive transportation network and internet-based new travel modes, it may not be difficult for the drivers to find an available travel mode to move to the next customer (APTA, 2021).

We will use the example shown in Figure 1 to illustrate the operational characteristics and benefits of the proposed IPOM. In this example, four delivery customers, including D1, D2, D3, and D4, and two pickup customers, i.e., P1 and P2, are to be served. It shows that driver 1 departs from the terminal with a platoon comprising of three loaded trucks. He/she drops off the loaded trucks at D1, D2, and D3, respectively, and waits at D3 until the unpacking work is completed. Then, this driver revisits D2 to form a platoon of two emptied trucks, while the emptied truck at D1 will be hauled back to the terminal by another driver (driver 3 in this example). Since the sharing of the empty trucks is allowed, driver 1 will send the two empty trucks released from the delivery customers D2 and D3 to P1 and P2 for reuse, respectively. After the packing work at P2 is completed, driver 1 will return to the terminal with a loaded truck, whereas the loaded truck at P1 will be transported back to the terminal by another driver thereafter. As for driver 2, he/she departs from the terminal with one loaded truck and drops

off the truck at customer D4 for the unpacking work. Thanks to the IPOM that allows the use of alternative transport modes after dropping off the truck, the driver can move to P1 by, e.g., taxi, to drive back the loaded truck without waiting at D4 for the completion of the unpacking task, and then revisit D4 to form a platoon of two trucks. Since driver 2 may have extra time within the maximum planning horizon after revisiting D4, he/she can head for D1 to form a platoon of totally three trucks and finally returns to the terminal. However, if IPOM cannot be implemented, driver 2 has to wait at D4 until the unpacking work is completed so that he/she may not have enough time to pick up the truck at P1 without violating the maximum planning horizon, let alone serve D1. In this case, another one or two drivers would be required to drive the truck(s) to haul the trucks at P1 and D1 back to the terminal. As indicated by the example, we find that the IPOM holds great potential to improve the drayage efficiency by labor and truck savings.

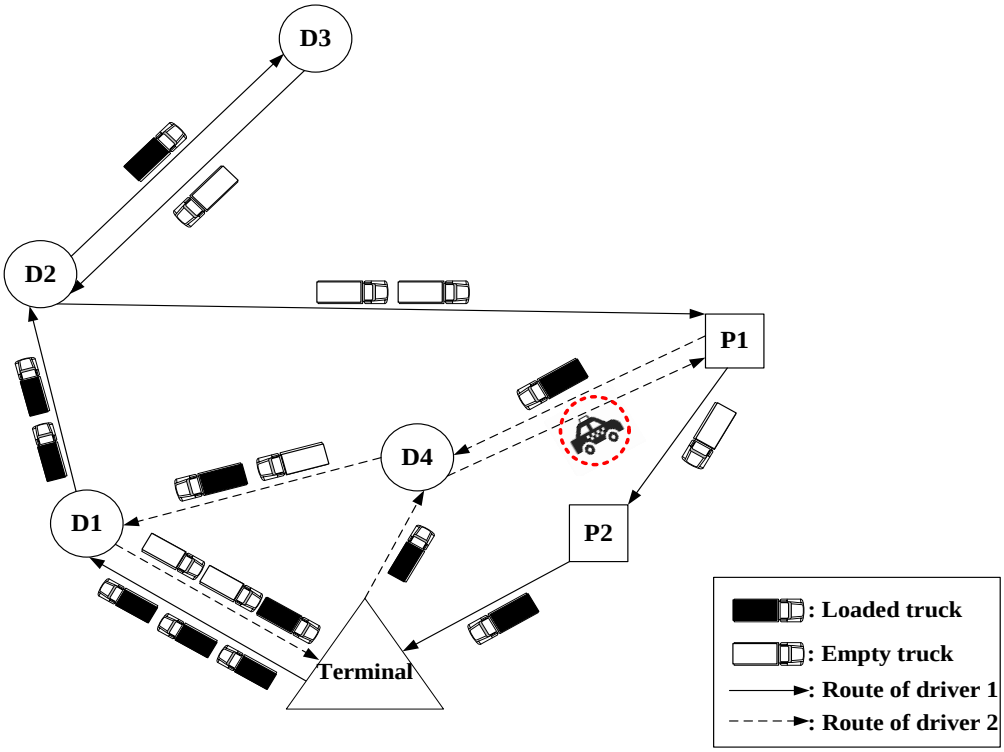


Figure 1. An example of the improved platooning operation mode

2.2 Truck movement tracking

For the considered LCDP problem, an empty truck for a pickup customer will be either from the terminal or the one released from a delivery customer after the unpacking work is completed. Besides, the platooning mode that multiple trucks traveling together in the form of a platoon would force the trucks to traverse many interim customer sites on the way from their

respective starting points to their respective target customers. The integration of the sharing of empty trucks among customers and the platooning mode creates challenges in modeling the specific movements of the trucks and obtaining the number of needed trucks to complete all the drayage requests. For example, even if the route of each truck is known, it is difficult to know whether the truck for a pickup customer is from the terminal or a previously served delivery customer, leading to the confusion regarding the matching between trucks and customers and accordingly the number of actually needed trucks to fulfill all the tasks. To address this issue, we will follow You et al. (2020) to define the OD pairs to track the movements of the trucks when they are moving among customers.

As aforementioned, each customer is associated with two different tasks in two stages before and after the packing/unpacking work. For modeling purposes, we virtually split each customer node into two task nodes to represent the two stages of the tasks for a customer. Specifically, let $\mathbf{C}_1$ denote the set of the first-stage task nodes, including the set $\mathbf{D}_1$ of loaded delivery task nodes and the set $\mathbf{P}_1$ of empty pickup task nodes. Similarly, let $\mathbf{C}_2$ represent the set of the second-stage task nodes, consisting of the set $\mathbf{D}_2$ of emptied delivery task nodes and the set $\mathbf{P}_2$ of loaded pickup task nodes. Let $\mathbf{C}=\mathbf{C_1}\cup\mathbf{C_2}$ denote the set of all task nodes, where $\mathbf{C}_1=\mathbf{D}_1\cup\mathbf{P}_1$ and $\mathbf{C}_2=\mathbf{D}_2\cup\mathbf{P}_2$. The customer node set $\mathbf{N}$ can then be translated into $\mathbf{N}=\mathbf{D}_1\cup\mathbf{D}_2\cup\mathbf{P}_1\cup\mathbf{P}_2\cup\{0\}=\mathbf{C}\cup\{0\}$. Note that the corresponding second-stage task node of a customer's first-stage task node $i\in\mathbf{C}_1$ is denoted by i', and vice versa. Based on the definition of the two-stage task nodes of a customer node, an OD pair can be defined as follows: an OD pair consisting of one origin node and one destination node is utilized to indicate that a truck departing from the origin node is dedicated to serving the destination task node. For a particular OD pair, a set of task nodes sequentially traversed by a truck on the route from the origin to the destination are considered as transfer nodes. The terminal node cannot serve as a transfer node in the route connecting an OD pair. Note each task node can be either an origin or a destination only once, and thus the number of OD pairs is finite in our concerned problem.

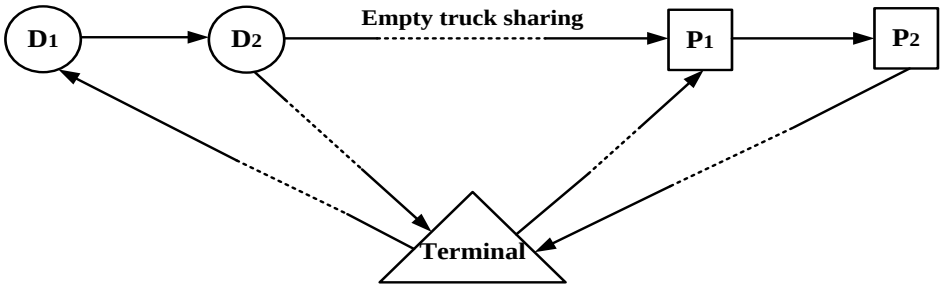


Figure 2. Illustration of all possible OD pairs

Combined with the requirements of the movements of trucks and the splitting strategy for the customer nodes that have been elaborated above in this section, we first list all the possible OD pairs that may occur in the proposed problem in Table 1. We further present Figure 2 to illustrate all the seven types of possible OD pairs, in which the dotted lines represent several task nodes that may be traversed on the routes from the origins to the destinations for the OD pairs. We then apply Constraints (1)-(8) introduced by You et al. (2020) to explicitly track the trucks' movements. Kindly note that to facilitate the reference to an OD pair in the mathematical formulation, we define $\mathbf{R}$ as the set of all possible OD pairs, where $\mathbf{R} = \{(\{0\}, \mathbf{D_1}), (\{0\}, \mathbf{P_1}), (\mathbf{D_2}, \{0\}), (\mathbf{P_2}, \{0\}), \quad \{(d_1, d_2) \mid d_1 \in \mathbf{D_1}, \quad d_2 = d_1'\}, \{(p_1, p_2) \mid \quad p_1 \in \mathbf{P_1}, \quad p_2 = p_1'\}, (\mathbf{D_2}, \mathbf{P_1})\}$.

Table 1. List of all the possible OD pairs

(O, D)	The number of possible OD pairs
$(\{0\}, \mathbf{D_1})$	$\lvert \mathbf{D_1} \rvert$
$(\{0\}, \mathbf{P_1})$	$\lvert \mathbf{P_1} \rvert$
$(\mathbf{D_2}, \{0\})$	$\lvert \mathbf{D_2} \rvert$
$(\mathbf{P_2}, \{0\})$	$\lvert \mathbf{P_2} \rvert$
$\{(d_1, d_2) \mid d_1 \in \mathbf{D_1}, d_2 = d_1'\}$	$\lvert \mathbf{D_1} \rvert$
$\{(p_1, p_2) \mid p_1 \in \mathbf{P_1}, p_2 = p_1'\}$	$\lvert \mathbf{P_1} \rvert$
$(\mathbf{D_2}, \mathbf{P_1})$	$\lvert \mathbf{D_2} \rvert \cdot \lvert \mathbf{P_1} \rvert$

$$\sum_{i \in \mathbf{N}} \alpha_{id}^{0d} = 1, \forall d \in \mathbf{D_1} \tag{1}$$

$$\sum_{j \in \mathbf{N} \setminus d} \left(\sum_{p \in \mathbf{P_1}} \alpha_{dj}^{dp} + \alpha_{dj}^{d0} \right) = 1, \forall d \in \mathbf{D_2} \tag{2}$$

$$\sum_{i \in \mathbf{N} \setminus p} \left(\sum_{d \in \mathbf{D_2}} \alpha_{ip}^{dp} + \alpha_{ip}^{0p} \right) = 1, \forall p \in \mathbf{P_1} \tag{3}$$

$$\sum_{j \in \mathbf{N}} \alpha_{pj}^{p0} = 1, \forall p \in \mathbf{P_2} \tag{4}$$

$$\alpha_{0i}^{od}=0, \forall i \in \mathbf{C}, o \in \mathbf{C}, d \in \mathbf{N}, (o,d) \in \mathbf{R} \quad (5)$$

$$\alpha_{i0}^{od}=0, \forall i \in \mathbf{C}, o \in \mathbf{N}, d \in \mathbf{C}, (o,d) \in \mathbf{R} \quad (6)$$

$$\sum_{i \in \mathbf{N} \setminus o} \alpha_{io}^{od}=0, \forall (o,d) \in \mathbf{R} \quad (7)$$

$$\sum_{i \in \mathbf{N} \setminus d} \alpha_{di}^{od}=0, \forall (o,d) \in \mathbf{R} \quad (8)$$

where the binary decision variables $\alpha_{ij}^{od}, \forall (i,j) \in \mathbf{A}, (o,d) \in \mathbf{R}$ denote whether arc (i,j) is traversed in a trip that connects the (o,d) pair. Constraints (1)-(4) express the requirements of the movements of trucks. To be more specific, Constraints (1) impose that for the first-stage delivery task node, a loaded truck must be transported to it from the terminal. Constraint (2) represents that for the second-stage delivery task node, its released emptied truck can either be transported to the terminal or to a first-stage pickup task node for reuse; Similarly, Constraint (3) expresses that for the first-stage pickup task node, its requirement of an empty truck can be either satisfied by the terminal or a second-stage delivery task node. Constraint (4) ensures that for the second-stage pickup task node, the loaded truck after the completion of the packing task must be hauled back to the terminal. Constraints (5)-(8) are restrictions pertaining to the features of the OD pairs. Specifically, Constraints (5) and (6) ensure that the terminal node cannot be used to connect an OD pair. Constraints (7) and (8) enforce that the origin node and destination node of an OD pair cannot act as transfer nodes to connect this OD pair.

3. Optimization model building

To formulate the LCDP under the IPOM that optimizes the numbers, traveling routes and time schedules of the drivers and trucks, in addition to the binary route variables $x_{ij}^{v}, \forall (i,j) \in \mathbf{A}, v \in \mathbf{V}$ for the drivers and the binary route variables $y_{ij}^{v}, \forall (i,j) \in \mathbf{A}, k \in \mathbf{K}$ for the trucks, indicating whether the driver $v \in \mathbf{V}$ travels along arc (i,j) on his/her trip and whether the truck $k \in \mathbf{K}$ traverses from node $i \in \mathbf{N}$ to node $j \in \mathbf{N}$, respectively, we also need define binary decision variables $\beta_{ij}, \forall (i,j) \in \mathbf{A}$ and continuous time variable $s_i, \forall i \in \mathbf{N}$, to denote whether there is at least one truck traversing arc (i,j) and the service beginning time of task node $i \in \mathbf{C}$, respectively. Besides, let $p_i, i \in \mathbf{C}_1$ denote the required container unpacking/packing time of each customer. The notations used throughout this study are provided in the appendix for readers' convenience. With the aforementioned notations, the LCDP under the IPOM can be formulated by:

[LCDP]

$$\min_{x,y,\alpha,\beta,s} \quad \lambda_1 \sum_{v\in\mathbf{V}}\sum_{j\in\mathbf{C}} x_{0j}^v + \lambda_2 \sum_{k\in\mathbf{K}}\sum_{j\in\mathbf{C}} y_{0j}^k + c_{ij} \sum_{i\in\mathbf{N},i\neq j}\sum_{j\in\mathbf{N}} [\beta_{ij} + (1-\eta)(\sum_{k\in\mathbf{K}} y_{ij}^k - \beta_{ij})] + c'_{ij} \sum_{i\in\mathbf{N},i\neq j}\sum_{j\in\mathbf{N}} (\sum_{v\in\mathbf{V}} x_{ij}^v - \beta_{ij}) \tag{9}$$

subject to Eqs. (1)-(8), and

$$\sum_{i\in\mathbf{C}} x_{0i}^v \le 1, \forall v \in \mathbf{V} \tag{10}$$

$$\sum_{v\in\mathbf{V}}\sum_{i\in\mathbf{N}\setminus j} x_{ij}^v = 1, \forall j \in \mathbf{C} \tag{11}$$

$$\sum_{i\in\mathbf{N}\setminus j} x_{ij}^v = \sum_{i\in\mathbf{N}\setminus j} x_{ji}^v, \forall j \in \mathbf{C}, v \in \mathbf{V} \tag{12}$$

$$\sum_{i\in\mathbf{C}} y_{0i}^k \le 1, \forall k \in \mathbf{K} \tag{13}$$

$$\sum_{k\in\mathbf{K}} y_{ij}^k \ge 1, \forall i \in \mathbf{C_1}, j = i' \tag{14}$$

$$\sum_{i\in\mathbf{N}\setminus j} y_{ij}^k = \sum_{i\in\mathbf{N}\setminus j} y_{ji}^k, \forall j \in \mathbf{C}, k \in \mathbf{K} \tag{15}$$

$$1 \le \sum_{k\in\mathbf{K}}\sum_{i\in\mathbf{N}\setminus\{j,j'\}} y_{ij}^k \le L, \forall j \in \mathbf{C} \tag{16}$$

$$y_{ij}^k \le \sum_{v\in\mathbf{V}} x_{ij}^v, \forall i, j \in \mathbf{N}, j \neq i, i', \forall k \in \mathbf{K} \tag{17}$$

$$\sum_{k\in\mathbf{K}} y_{ij}^k / L \le \beta_{ij} \le \sum_{k\in\mathbf{K}} y_{ij}^k, \forall i, j \in \mathbf{N}, j \neq i \tag{18}$$

$$\sum_{v\in\mathbf{V}} x_{ij}^v \ge \beta_{ij}, \forall i, j \in \mathbf{N}, j \neq i \tag{19}$$

$$\sum_{i\in\mathbf{C}\setminus\{g,g'\}} \alpha_{ig}^{od} = \sum_{j\in\mathbf{C}\setminus\{g,g'\}} \alpha_{gj}^{od}, \forall (o,d) \in \mathbf{R}, g \in \mathbf{C}, g \neq o \neq d \tag{20}$$

$$\sum_{i\in\mathbf{N}} \alpha_{oi}^{od} = \sum_{j\in\mathbf{N}} \alpha_{jd}^{od}, \forall (o,d) \in \mathbf{R} \tag{21}$$

$$\sum_{(o,d)\in\mathbf{R}}\sum_{i\in\mathbf{N}\setminus\{j,j'\}} \alpha_{ij}^{od} = \sum_{i\in\mathbf{N}\setminus\{j,j'\}}\sum_{k\in\mathbf{K}} y_{ij}^k, \forall j \in \mathbf{C} \tag{22}$$

$$\sum_{(o,d)\in\mathbf{R}} \alpha_{ij}^{od} \le \sum_{k\in\mathbf{K}} y_{ij}^k, \forall i, j \in \mathbf{C}, i \neq j \tag{23}$$

$$s_i + p_i \le s_{i'}, \forall i \in \mathbf{C_1} \tag{24}$$

$$s_i \ge t_{0i} \cdot \beta_{0i} + (1-\beta_{0i}) \cdot t'_{0i}, \forall i \in \mathbf{C} \tag{25}$$

$$s_i + t_{i0} \le T, \forall i \in \mathbf{C_2} \tag{26}$$

$$s_j \ge s_i + t_{ij} \cdot \beta_{ij} + (1-\beta_{ij}) \cdot t'_{ij} - M \cdot (\sum_{v\in\mathbf{V}} x_{ij}^v - 1), \forall i, j \in \mathbf{C}, i \neq j \tag{27}$$

$$x_{ij}^v, y_{ij}^k, \alpha_{ij}^{od}, \beta_{ij} \in \{0,1\}, \forall i, j \in \mathbf{N}, i \neq j, v \in \mathbf{V}, k \in \mathbf{K}, (o,d) \in \mathbf{R} \tag{28}$$

$$s_i \ge 0, \forall i \in \mathbf{N} \tag{29}$$

where M is a sufficiently large number. The objective function shown by Eq. (9) is the sum of the total operational costs, including the driver employment cost, the truck deployment cost, the total fuel consumption cost of trucks considering fuel savings of platooning, and the total

travel cost incurred by alternative transport modes. Constraints (10)-(12) construct the drivers' routes. Specifically, Constraint (10) enforces that each driver can only depart from the terminal once at most. Constraint (11) stipulates that each task node should be visited by one driver exactly once. Constraint (12) ensures the flow conservation for each driver. Constraints (13)-(15) construct the traveling routes for trucks. Constraint (13) restricts that each truck cannot be deployed repeatedly. Constraint (14) ensures that there is at least one truck staying at the customer's location until the (un)packing work is completed. Constraint (15) enforces the flow conservation for each truck. Constraint (16) limits the maximum platoon length. Constraints (17)-(19) specify the relationship between drivers' routes and trucks' movements among task nodes. Specifically, Constraint (17) imposes that a truck cannot move without a driver's guidance. Constraint (18) determines the value of the binary variables β_{ij} to indicate whether there is at least a truck traversing arc (i, j). Constraint (19) implies that a driver can traverse an arc without trucks. Constraints (20)-(23) track the specific flow of trucks. Constraint (20) is the flow conservation for each OD pair. Constraint (21) ensures the flow balance of each OD pair. Constraint (22) guarantees that for each task node, the flow of arcs used to connect the OD pairs through the task node is equal to the flow of trucks. Constraint (23) indicates that an arc used to connect the OD pairs is traversed by at least one truck. A graphic illustration for Constraints (20)-(23) is provided in Figure 3, in which (Terminal, D2), (D2, P1), and (P1, Terminal) are the three considered OD pairs. Constraints (24)-(27) are the constraints for the time schedules of task nodes. Constraint (24) makes sure that the required unpacking/packing time must be satisfied. Constraints (25) and (26) impose that each task must be completed within the maximum planning horizon. Constraint (27) specifies the relationships of the service starting time between two customers that the same driver consecutively visits. Constraints (28)-(29) define the domains of the decision variables.

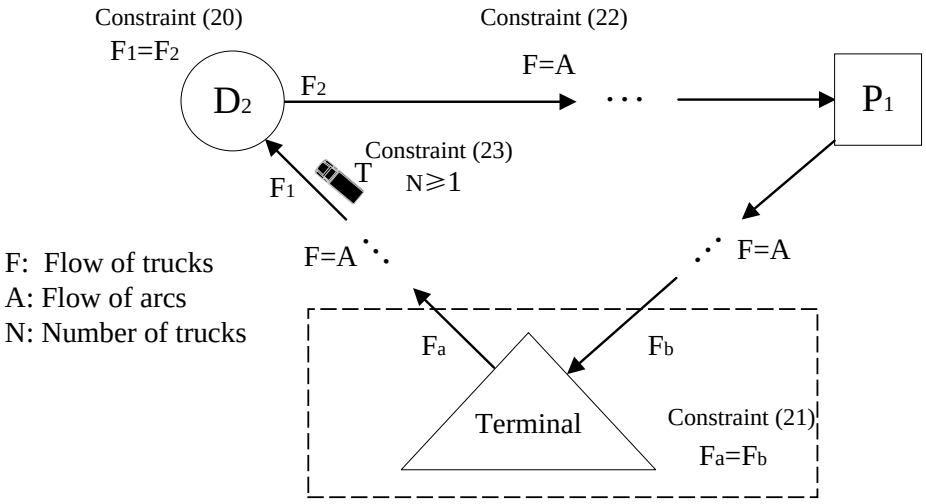


Figure 3. Graphic illustration for Constraints (20)-(23)

The major differences of our study and You et al. (2020) lie in the consideration of fuel saving of platooning and the movement of drivers. The two new features are reflected in variables, terms, or constraints related to the fuel cost and drivers' movement as well as the coupling constraints linking the trucks' and drivers' movements. In fact, if the platoon fuel saving ratio is set to 0 and the unit travel costs or the travel speeds of the alternative transport modes are set to be infinite, the MILP model of this study will reduce to that of You et al. (2020), which indicates that the proposed model is more general.

4. Solution method design

The proposed general LCDP model is NP-hard since its special case, such as the model in You et al. (2020), has already proven to be NP-hard. Simulated annealing (SA) has been widely used to solve this type of complicated problem with demonstrated good performance (Braekers et al., 2014; Escudero et al., 2019; Kuo et al., 2010; Oudani et al., 2021; Wei et al., 2018; Xiao et al., 2012). Therefore, we propose a customized heuristic construction method incorporating simulated annealing (HCSA) to solve the proposed problem.

4.1 Framework of the solution method

As discussed above, we can see that the proposed IPOM is more complicated than the existing operation modes for the LCDP. In particular, the sharing of empty trucks among customers, truck platooning, and drivers' movements by alternative transport modes would make it difficult to calculate the actual number of required trucks, the specific movements of the engaged drivers and trucks among customers, the exact number of trucks traveling (together)

over each arc, as well as the time spent moving between customers of the drivers. Therefore, we develop a customized HCSA to solve the proposed problem by designing tailored procedures devoted to addressing these challenges.

The framework of the HCSA is presented in Figure 4. In the framework, a tailored solution encoding scheme considering the characteristics of the proposed problem is presented in Subsection 4.2, the heuristic construction process for the initial solution is provided in Subsection 4.3, and the SA algorithm specially designed to iteratively improve the solutions for the considered problem are elaborated in Subsections 4.4-4.6. Note that during the heuristic construction process, the main challenge to generate an initial solution is to maintain the platoon size constraint feasibility for each driver since the number of trucks traveling together in a platoon led by a driver will change dynamically during the service process. Therefore, we specify effective rules for updating the platoon size in each driver's traveling route in Subsection 4.3. In addition, the objective function value calculated by Eq. (9), i.e., the total operational cost, will be used as the solution evaluation criteria for the proposed HCSA. However, the calculations for the truck deployment cost, the total fuel consumption cost of trucks, and the total travel cost of alternative transport modes are not trivial since the number of needed trucks, the specific truck flow on the route, and the driver's movement modes (trucks or alternative transport means) among customers over each arc cannot be directly derived from the solutions. Hence, a series of customized calculation rules are designed to precisely evaluate the solutions based on the implicit information provided by a solution, which will be elaborated in Subsection 4.5.2 and Subsection 4.5.3, respectively. Furthermore, in view of the maximum planning horizon constraint imposed on the drivers, it is necessary to guarantee the temporal feasibility for all the task nodes. Nevertheless, the unique strategy in the IPOM that the drivers can move by trucks or alternative transport modes create difficulties in calculating the actual travel time between two consecutively visited customers. Therefore, we design customized procedures for the time feasibility check devoted to the LCDP with IPOM in Subsection 4.5.1 and Algorithm 2 (lines 12-20). Besides, the tailor-made neighborhood search operators and the scheme of SA are presented in Subsections 4.4 and 4.6, respectively.

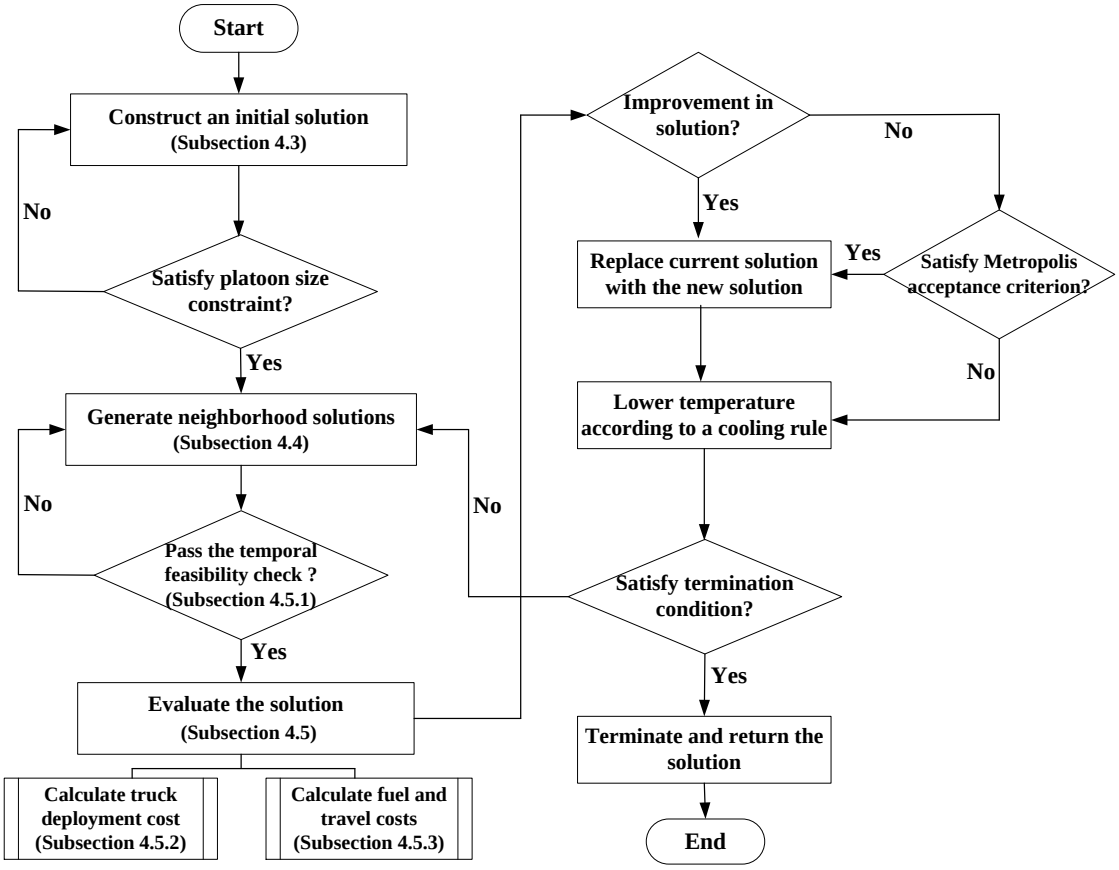


Figure 4. The framework of the HCSA

4.2 Solution encoding

In the proposed HCSA, a set of natural number strings in $\mathbf{S}$ represents a solution to the proposed problem and each string denotes the routes of each driver. Since all drivers will start and end at the terminals, each string $S \in \mathbf{S}$ will have the terminal encoded as '0' at both ends. The task nodes are encoded as positive integer numbers, and the numerical order of each string represents the specific visiting sequence of task nodes for each driver. For example, a string set $\mathbf{S}=\{[0,1,2,4,0],[0,3,5,6,0]\}$ means that two drivers are employed to serve all the 6 task nodes (3 customers) according to the sequences displayed in the strings.

4.3 Initial solution construction

Many experiments and studies have demonstrated that SA is robust; that is, the quality of the final solution is not very dependent on the quality of the initial solution, and hence any feasible solution can serve as an initial solution for the SA (Bernardo et al., 2018; Mousavi et al., 2017; Wang et al., 2015). However, finding an initial solution for the LCDP under the IPOM requires non-trivial efforts due to its complicated operational characteristics and

modeling considerations. In this study, we utilize a greedy algorithm considering the IPOM characteristics to construct the initial solution. To construct a string for each driver, an unserved task node is selected based on the nearest distance to the last inserted node starting from the terminal. If no unserved task node can be inserted without violating the maximum platoon size constraint, the current string will be ended by the terminal and the generation process starts again from the terminal and continues to construct the next string until all task nodes have been inserted into strings in place. During the generation process for the initial solution, it is challenging to maintain the platoon size constraint feasibility because the actual number of trucks occupied in each string (i.e., the fleet size of each string) will change dynamically depending on the type of the next task node attempted to be inserted into the string in the following ways:

(1) For each task node $i \in \mathbf{D}_1$, its required loaded truck can only be satisfied by the terminal. Therefore, the fleet size increases by one.

(2) For each task node $i \in \mathbf{D}_2$, if $i' \in \mathbf{D_1}$ is already included in the string, the fleet size remains unchanged; otherwise, the fleet size increases by one.

(3) For each task node $i \in \mathbf{P}_1$, its requirement can be either satisfied by the terminal or an emptied truck previously released from a second-stage delivery task node. Therefore, if the current string contains at least an available empty truck, the fleet size remains unchanged; otherwise, the fleet size increases by one.

(4) For each task node $i \in \mathbf{P}_2$, if $i' \in \mathbf{P_1}$ is already included in the string, the fleet size remains unchanged; otherwise, the fleet size increases by one.

4.4 Neighborhood structures

In simulated annealing, the current solution is continuously replaced by its neighboring solutions generated in the neighborhoods identified by the predefined search operators with the purpose of finding a global optimum. In this study, six different neighborhood search operators are employed. The first three, i.e., *swap*, *reversion*, and *relocation*, are simple operators that have been commonly adopted in the SA scheme (Braekers et al., 2014; Escudero et al., 2019; Kuo et al., 2010; Oudani et al., 2021; Wei et al., 2018; Xiao et al., 2012), while the other three, named *2-swap*, *remove*, and *split*, are specifically designed operators for the underlying problem to further diversify the solution structures that can be investigated during the search

process. Specifically, the *swap* move is carried out by exchanging the positions of two randomly selected nodes in a solution string. The *reversion* is performed by randomly selecting a pair of nodes in a string and then reversing the order of the substring in-between them. In the *relocation* move, a randomly chosen node is removed and reinserted into another position in the same string. The *2-swap* is to exchange the places of two randomly selected segments of nodes belonging to two different strings. The *remove* operator is executed by removing all nodes of a short string consisting of fewer nodes and inserting them into another long string consisting of more nodes, in which the insertion position is randomly selected. This operator aims to reduce the number of strings (i.e., the number of employed drivers) by integrating the nodes of short strings into other longer ones as long as the resultant fleet sizes of the new strings do not exceed the maximum allowable platoon size. Contrary to the *remove* operator, the *split* operator is implemented by dividing an overlong string into two shorter strings. It will remove a segment of nodes from a long string, and the remaining two parts of the string are combined to generate a new string, while the removed part will build another new one by introducing the terminal at both ends. This operator aims to prevent the number of the task nodes served by the same driver from being unreasonably large; otherwise, longer detours and accordingly prohibitive travel costs may be incurred. Note that SA explores one neighborhood selected from the aforementioned six search operators at each iteration for better solutions. Detailed movement strategies of the operators on the current solutions are also visualized in Figure 5 (a)-(f), with each dedicated to one operator.

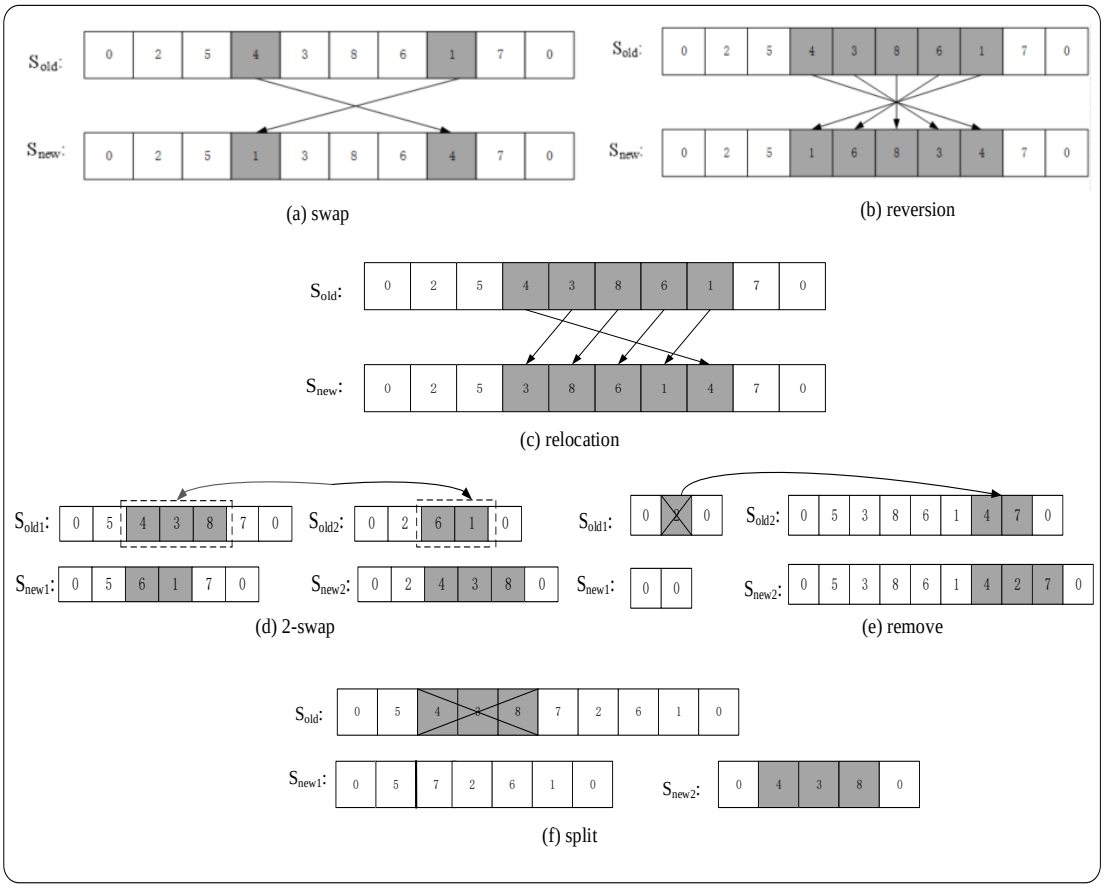


Figure 5. Specific movement strategies for the neighborhood search operators

4.5 Solution evaluation

The objective function value calculated by Eq. (9) is directly used as the evaluation criteria for the proposed HCSA. The objective function is to minimize the sum of the driver employment cost, the truck deployment cost, the fuel consumption cost incurred by trucks and the travel cost incurred by alternative transport modes. According to the encoding structure of the solution, the driver employment cost is evidently associated with the number of strings in the solution, whereas the calculation for the truck deployment cost, the total fuel consumption cost and the total travel cost becomes a difficult issue. In fact, though the visiting order of task nodes is determined in a solution string, the number of the engaged trucks and the practical number of trucks traveling (together) on each arc, as well as the travel times between customers are still unknown to us since the specific truck flow on the route and the driver's movement modes (trucks or alternative transport means) over each arc are not clear. Therefore, we design a customized approach to evaluate the solutions by exactly calculating the truck deployment cost, the fuel consumption cost of trucks, and the travel cost of alternative transport modes, which will be elaborated in the following subsections.

4.5.1 Temporal feasibility check for task nodes

The initial solution constructed by the fast heuristic described in Subsection 4.3 and the neighboring solutions generated according to the search operators may not be guaranteed to be feasible with respect to the planning horizon constraint for all the task nodes imposed by Eq. (26). To reduce unnecessary calculation time, we will only evaluate feasible solutions. Hence, before the solution evaluation process, we should have the following constraints for each task node to go through the temporal feasibility check:

$$[\sigma_i, \tau_i] = [t_{0i}, T - t_{i'0} - p_i], \forall i \in \mathbf{C}_1 \tag{30}$$

$$[\sigma_i, \tau_i] = [t_{0i'} + p_{i'}, T - t_{i0}], \forall i \in \mathbf{C}_2 \tag{31}$$

where σ_i and τ_i denote the earliest and latest service starting time for task $i \in \mathbf{C}$, respectively. The combination of Eq. (30) and Eq. (31) enforces that the service of each task node must be completed within the maximum planning horizon and each second-stage task node cannot be served before the completion of its corresponding first-stage task node. Given the visiting sequences of all task nodes in the solution strings, once the service starting time for a task node is not within its hard time window specified by Eqs. (30)-(31), the string where this node is placed is infeasible. Then, for a solution that contains such a string that violates the maximum planning horizon constraint, we will directly assign an infinite value to its objective function value (i.e., evaluation result). Note that the strategy that drivers can be transferred by trucks or alternative transport modes results in some confusion about the actual travel times between each two consecutively visiting nodes, making the calculation of the service starting times of task nodes become much more complicated. Therefore, we present specific procedures for time feasibility check devoted to the underlying problem in Algorithm 2 (lines 12-20).

4.5.2 Calculation of the truck deployment cost

Given a feasible solution, one major challenge is to calculate the total number of needed trucks. The key to overcoming this challenge is to judge whether the request of a first-stage pickup task node can be satisfied by an empty truck previously released from a second-stage delivery task node. Therefore, we present the pseudocode in Algorithm 1 to explicitly obtain the total number of engaged trucks and the associated truck deployment cost. For each string $S \in \mathbf{S}$, we first locate all the second-stage delivery task nodes that release empty trucks. After that, for each pair of two adjacent second-stage delivery task nodes, we will count the number q of the first-stage pickup task nodes between them. Kindly note that the number ℓ of available

empty trucks for reuse can be accumulated by one in case of $q = 0$. The current number π_s of required trucks of string S will then be updated according to the relative size between q and e.

Algorithm 1. Calculation of the truck deployment cost

For string $S \in \mathbf{S}$
Initialize the number of first-stage delivery task node in string S: $n^{d_1} \leftarrow 0$; the number of second-stage delivery task node in string S: $n^{d_2} \leftarrow 0$; $e \leftarrow 0$; $q \leftarrow 0$; $\pi_s \leftarrow 0$.
  For each task node $i \in S$ do
    If $i \in \mathbf{D_1}$, then
      $n^{d_1} \leftarrow n^{d_1} + 1$; $\pi_s \leftarrow \pi_s + 1$;
    Else if $i \in \mathbf{D_2}$, then
      $n^{d_2} \leftarrow n^{d_2} + 1$;
    End If
  End For
  Find the locations (indices) $l_1, l_2, ..., l_{n^{d_2}}$ of the second-stage delivery task nodes in the string;
  For each task node i located before l_1 do
    If $i \in \mathbf{P_1}$, then
      $q \leftarrow q + 1$;
      $\pi_s \leftarrow \pi_s + q$;
    End If
  End For
  For $l_1 \le l_z < l_{n^{d_2}}$ do
    For each task node i located between l_z and $l_{z+1} (1 \le z < n^{d_2})$ do
      If $i \in \mathbf{P_1}$, then
        $q \leftarrow q + 1$;
      End If
    End for
    If $q = 0$, then
      $e \leftarrow e + 1$;

25 Else if $0 < q \le e$, then

26 $e \leftarrow e - q + 1$;

27 Else if $q > e$, then

28 $\pi_s \leftarrow \pi_s + q - e - 1$; $e = 0$;

29 End If

30 End For

31 For each task node i located behind $l_{n^{d_2}}$ do

32 If $i \in \mathbf{P_1}$, then

33 $q \leftarrow q + 1$;

34 End If

35 End for

36 If $q > e$, then

37 $\pi_s \leftarrow \pi_s + q - e - 1$;

38 End If

39 End For

40 Return the total truck deployment cost: $\lambda_2 \sum_{S \in \mathbf{S}} \pi_s$

4.5.3 Calculation of the fuel and travel cost

To obtain the fuel consumption cost of trucks considering the fuel savings by platooning and the travel cost incurred by alternative transport modes, the number of trucks traveling (together) on each arc has to be exactly calculated. Therefore, we present the pseudocode in Algorithm 2 to capture the explicit truck flow en-route and calculate the associated fuel consumption cost and travel cost. More specifically, for each string, we define $w_{i,i+1}$ to denote the truck flow over the arc between the i^{th} and the $(i+1)^{th}$ node of the string. Note that the truck flow on the first arc, i.e., w_{12}, of string $S \in \mathbf{S}$ evidently equals to the corresponding value of π_s calculated by Algorithm 1. Then, we will update $w_{i,i+1}$ according to the following intuitive rule: $w_{i,i+1}$ will decrease by one only if the i^{th} node belongs to C_1, and, at the same time, the $(i+1)^{th}$ node is not the second stage of the i^{th} node, and will increase by one once traversing a second-stage task node (see lines 3-11). Kindly note that $w_{i,i+1} = 0$ indicates that the driver uses alternative transport modes rather than the truck to come to the $(i+1)^{th}$ node

from the i^{th} node. As such, indicated by whether $w_{i,i+1}$ equals 0 or not, the time required to traverse the arc will be determined, and thus the service starting time of task nodes can be explicitly updated, enabling the temporal feasible check for the task nodes (lines 12-20). In addition, the fuel consumption cost of trucks or the travel cost of alternative transport modes on each traversal arc of the considered string can also be exactly calculated (see lines 21-29). Finally, the sum of the total fuel consumption cost and the total travel cost for the solution can thus be obtained (see line 31).

Algorithm 2. Calculation of the fuel and travel cost

---

1 For string $S \in \mathbf{S}$

2 Initialize the fuel consumption cost of each traversal arc of string S : $f^{S}_{i,i+1} \leftarrow 0, i = 1, 2, ..., \text{length}(S) - 1$; $w_{1,2} \leftarrow \pi_s$.

3 For the i^{th} $(2 \le i \le \text{length}(S) - 1)$ node in string S (denoted by $S(i)$) do

4 If $S(i) \in \mathbf{C}_1$ and $S(i+1) \ne (S(i))'$, then

5 $w_{i,i+1} \leftarrow w_{i,i+1} - 1$;

6 Else if $S(i) \in \mathbf{C}_1$ and $S(i+1) = (S(i))'$, then

7 $w_{i,i+1} \leftarrow w_{i,i+1}$;

8 Else if $S(i) \in \mathbf{C}_2$, then

9 $w_{i,i+1} \leftarrow w_{i,i+1} + 1$;

10 End If

11 End For

12 For the i^{th} $(2 \le i \le \text{length}(S) - 1)$ node in string S do

13 If $S(i) \in \mathbf{C}_1$, then

14 $s_{S(i)} = s_{S(i-1)} + t_{S(i-1),S(i)}$;

15 Else if $S(i) \in \mathbf{C}_2$ and $w_{i-1,i} > 0$, then

16 $s_{S(i)} = \max\{s_{(S(i))'} + p_{(S(i))'},\ s_{S(i-1)} + t_{S(i-1),S(i)}\}$;

17 Else if $S(i) \in \mathbf{C}_2$ and $w_{i-1,i} = 0$, then

18 $s_{S(i)} = \max\{s_{(S(i))'} + p_{(S(i))'},\ s_{S(i-1)} + t'_{S(i-1),S(i)}\}$;

19 End If

20 End For

21 For the i^{th} $(1 \le i \le \text{length}(S)-1)$ node in string S do

22 If ($0 < w_{i,i+1} \le L$) and ($s_{S(i)} \ge \sigma_{S(i)}$) and ($s_{S(i)} \le \tau_{S(i)}$), then

23 $f^{S}_{i,i+1} = c_{S(i),S(i+1)}[1+(1-\eta)(w_{i,i+1}-1)]$;

24 Else if ($w_{i,i+1} = 0$) and ($s_{S(i)} \ge \sigma_{S(i)}$) and ($s_{S(i)} \le \tau_{S(i)}$), then

25 $f^{S}_{i,i+1} = c'_{S(i),S(i+1)}$;

26 Else then

27 $f^{S}_{i,i+1} = +\infty$;

28 End If

29 End For

30 End For

31 Return the total fuel and travel cost: $\sum_{S \in \mathbf{S}} \sum_{1 \le i \le \text{length(S)}} f^{S}_{i,i+1}$

4.6 Simulated annealing mechanism

The heuristic construction method can obtain a feasible solution but may not be a good-quality one to the proposed problem. Simulated annealing (SA) can be easily implemented for a wide range of optimization problems without relying on the restrictive properties of the model, which is a suitable solution method for our proposed platooning problem. Therefore, we incorporate the SA mechanism into the construction heuristic to diversify the solution structures and iteratively guide the search for more satisfactory solutions in an efficient manner. The procedure of the SA scheme is presented in Algorithm 3. As shown in the pseudocode, the SA algorithm starts with a heuristically constructed initial solution $\mathbf{S}^0$. Then, the incumbent solution $\mathbf{S}$ (begins with the initial solution) is iteratively transformed to its neighbor solutions $\mathbf{S}'$ through the predefined neighborhood structures. If the objective function value of $\mathbf{S}'$ is less than that of $\mathbf{S}$, we will replace the incumbent solution $\mathbf{S}$ with the new solution $\mathbf{S}'$; otherwise, we will accept $\mathbf{S}'$ as the incumbent solution $\mathbf{S}$ with the Metropolis probability, i.e., $e^{\frac{-\Delta H}{T}}$, where ΔH is the deviation of the objective function values between $\mathbf{S}'$ and $\mathbf{S}$ (i.e., $\text{obj}(\mathbf{S}') - \text{obj}(\mathbf{S})$), and T is the current temperature. As can be seen, the SA algorithm allows accepting the changes that worsen the solution with a small probability, which effectively reduces the likelihood of getting trapped in the local minimum. This search process is repeated until the stopping condition, i.e., the pre-specified temperature threshold T^{end} or the maximum

number $B_{\max}$ of continuous iterations without improvement on the current solution is satisfied. Finally, the best solution $\mathbf{S}^*$ so far will be updated if found. Note that the essence of the SA scheme is to gradually lower the temperature T from a given initial temperature T^0 to a specified terminating temperature T^{end}. The current temperature T will decrease after a number of consecutive attempts, denoted by MaxIter, for better solutions according to the cooling rule $T \leftarrow T \cdot \varepsilon$, where ε is the annealing rate, i.e., a constant between 0 and 1, to control the cooling speed.

Algorithm 3. Scheme of simulated annealing

1 Initialization: Generate an initial solution $\mathbf{S}^0$ by the construction heuristic; The non-improvement counter: $B \leftarrow 0$; The iteration index: iter $\leftarrow 0$; $\mathbf{S} \leftarrow \mathbf{S^0}$; $\mathbf{S}^* \leftarrow \mathbf{S^0}$; $T \leftarrow T^0$.
2 While ($T > T^{\text{end}}$) do
3 If $B > B_{\max}$, then
4 Break and jump out of the loop;
5 End If
6 For iter<MaxIter do
7 Generate a new neighboring solution $\mathbf{S}'$ of $\mathbf{S}$ in a randomly selected neighborhood predefined by one of the six operators.
8 If $\text{obj}(\mathbf{S}') < \text{obj}(\mathbf{S})$, then
9 $\mathbf{S} \leftarrow \mathbf{S}'$; $B \leftarrow 0$;
10 Else if $\text{obj}(\mathbf{S}') \geq \text{obj}(\mathbf{S})$, then
11 $B \leftarrow B+1$; $\Delta H = \text{obj}(\mathbf{S}') - \text{obj}(\mathbf{S})$; Accept or reject $\mathbf{S}'$ according to the Metropolis probability: $\exp(-\Delta H / T)$; if accepted: $\mathbf{S} \leftarrow \mathbf{S}'$.
12 End If
13 End For
14 If $\text{obj}(\mathbf{S}) < \text{obj}(\mathbf{S}^*)$, then
15 $\mathbf{S}^* \leftarrow \mathbf{S}$;
16 End If
17 $T = T \cdot \varepsilon$;
18 End While

19 Return the best solution ever found: $\mathbf{S}^*$

5. Numerical experiments

This section presents the results of a series of computational experiments on randomly generated instances. First, we will elaborate on the test instances used for our experiments. Then, we will evaluate the proposed model and HCSA algorithm. Finally, sensitivity analysis is conducted to explore the impacts of several major influential factors on the system performance and derive practical insights. The mathematical models are coded in AMPL/CPLEX, whereas the proposed HCSA and the involved competing heuristic algorithms are all implemented in MATLAB R2020b. All numerical experiments are executed on a personal computer with Intel (R) Core (TM) i7 3.4GHz CPUT with 16GB RAM.

5.1 Test instances and parameter settings

We will use randomly generated instances to test the model and algorithm. Customers are randomly selected in a square region with a side length of 200 km, while the terminal is located at the center of the region with coordinates (100, 100). Each customer has only one delivery or pickup request, of which the required unpacking or packing time, i.e., p_i, measured by hour, is uniformly drawn from the set $\{2,3, \ldots, 5\}$. We specify motorcycles as the available alternative transport mode that can be used by drivers at any customer sites. Given the travel speed of 60 km/h for the truck and 35 km/h for the motorcycle, the travel time between two customers can be obtained readily according to their Euclidean distance.

Following the existing parameter settings for the platoon-based LCDP in previous studies, we set the model parameters as follows: the fuel-saving factor η by platooning, the maximum platoon size L, and the maximum planning horizon T is set to be 0.1, 6, and 16 hr, respectively. As for the cost-related parameters, the daily fixed costs to deploy a driver λ_1 and a truck λ_2 are set to be 100 and 50, respectively. The unit fuel consumption cost for the truck and the unit travel cost for the motorcycle are set to be 1 and 0.5, respectively.

Based on some preliminary experiments and the guidelines for the SA, we set the following algorithm parameters: T^0, T^{end}, ε, and MaxIter in the experiments are set to be 20, 0.0001, 0.9999, and 300, respectively. Besides, we use $T^{\text{end}} = 0.0001$ and $B_{\max} = 3000$ as the stopping conditions of the HCSA. Based on these parameter settings, we perform numerical experiments on multiple groups of random instances with different customer sizes ranging from

4 to 400. Each group of instances is named by 'a+b' where a and b represent the number of delivery and pickup customers, respectively. For each group, five random instances will be generated and the average results will be reported.

5.2 Algorithm performance

To evaluate the performance of the proposed solution method for solving the LCDP with IPOM, we will compare the results of the HCSA with those obtained by solving the proposed model directly using CPLEX and the other two well-known metaheuristics including genetic algorithm (GA) and particle swarm optimization algorithm (PSO). Note that the HCSA, GA, and PSO algorithms will be independently run three times for each instance and the average results will be reported to ensure the reliability of the results and to mitigate the uncertainty brought by the usage of each of the above heuristics. The maximum runtime for CPLEX to solve each instance is set to 1 hr. Table 3 presents the comparison results of the proposed HCSA with the other three methods for various test instances, with the best values of each indicator highlighted in bold. For all methods, we report the objective function values obtained within the time limit (Obj) and the elapsed CPU runtimes (Time). We also report the gaps (Gap) of the average objective function values between the HCSA and each of the other three methods, i.e., GA, PSO, and CPLEX, respectively.

We will first assess the performance of the proposed algorithm by comparing it with CPLEX based on the obtained exact solutions. As presented in Table 2, we can see that CPLEX fails to identify even a feasible solution for the instances with more than 30 customers within 1 hr and only the smallest instances can be solved by CPLEX. In contrast, the proposed HCSA can obtain good-quality solutions for all the instances within 8 min on average. More specifically, for most of the instances that can be solved by CPLEX within the time limit, the HCSA can obtain the same or better solutions in a significantly much shorter time than CPLEX, i.e., averagely 69s v.s. 3,004s. For the other instances that are intractable for CPLEX, the HCSA manages to find good-quality solutions within only 0.3 hr on average. In particular, the HCSA can obtain good solutions for instances with up to 400 customers (200+200) within 43 min. The above findings demonstrate the considerable computational complexity of the proposed problem and the effectiveness and efficiency of the proposed algorithm for solving the underlying problem. Hence, we will use the HCSA to conduct the impact analysis of the IPOM and the sensitivity analysis in the following sections. To further examine the computational performance of the HCSA, we visualize the variations of the CPU runtimes of the HCSA with

respect to the customer size in Figure 6. One notable observation is that the CPU time follows a linearly increasing trend with the increase of the customer size. This is consistent with our expectation that the proposed simulated annealing-based heuristic method takes advantage of the metaheuristics that can find quite good solutions within rational time, indicating the great potential of the HCSA to be implemented in real-life applications for the LCDP.

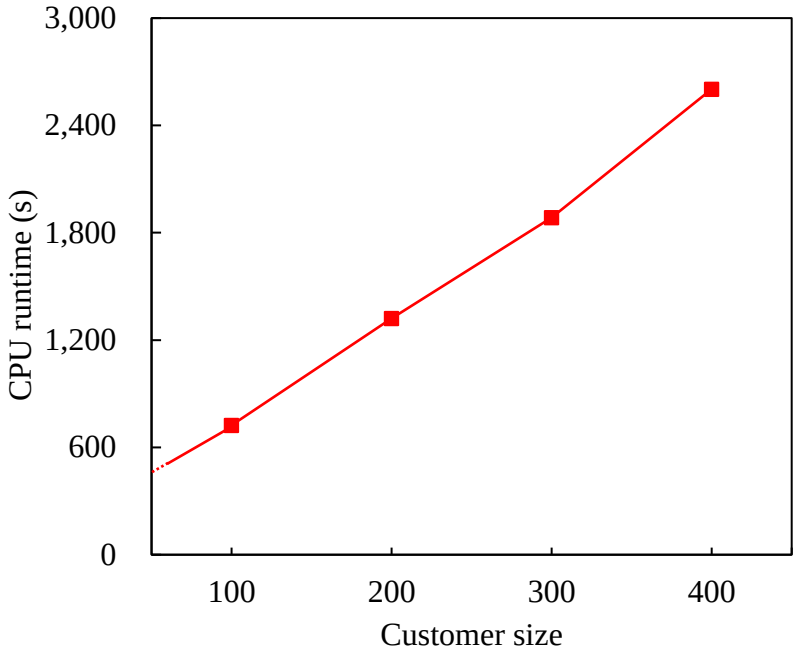


Figure 6. The variations of the CPU runtimes with the increase of customer size

We then compare the solutions obtained by the proposed HCSA algorithm with those of GA and PSO. As shown in Table 3, for almost all the instances, we observe that the HCSA performs significantly better than the other two algorithms, with lower average objective values and less CPU runtime. Specifically, compared with those obtained by GA, objective function values and the CPU runtimes of HCSA have reduced by 12.62% and 44.43% on average, respectively. As for the PSO, HCSA obtains averagely 9.55% smaller objective function values in 34.61% less CPU runtime. The results have indicated that averagely speaking, the HCSA achieves the best performance for solving the proposed IPOM model, followed by PSO and GA. However, we also find that our algorithm slightly underperforms the competing algorithms in a few small-sized problem instances, such as the '2+2', '6+2', '2+8' instance groups, but the differences between the comparing indicator values are marginal. To sum up, the proposed HCSA can produce better solutions with less computational effort compared with GA and PSO, especially for middle and large instances. Moreover, we also observe that the runtimes of GA

and PSO increase much faster than that of the proposed HCSA as the problem size grows, which demonstrates that the efficiency of HCSA is less sensitive to problem size and thus it has good potential to be implemented to solve large-scale LCDP. The satisfactory performance of HCSA should be attributed to the customized framework, the tailor-made neighborhood search operators, as well as the high-quality initial solution construction rules detailed in Section 4. Overall, the proposed HCSA outperforms GA and PSO in terms of both solution quality and runtime for solving the LCDP with IPOM.

We proceed to perform more numerical experiments to examine the performance of the HCSA for solving the proposed model [IPOM]. Specifically, we numerically verify the stability of the HCSA and investigate the impact of the locations of customers (i.e., randomly distributed customers and clustered customers) on the algorithm performance in Subsections 5.2.1 and 5.2.2, respectively.

5.2.1 Stability of the HCSA

For a given number of customers, the performance of the HCSA may vary according to the parameter settings and the randomness of the instances. To numerically verify the stability of the HCSA, we will test 8 groups of random instances with a small to a large number of customers under different values of the annealing rate ε, i.e., $\varepsilon \in \{0.99, 0.999, 0.9999\}$. The results are summarized in Table 3, in which the minimum, the average, and the standard deviation values of the objective function, as well as the average CPU runtimes for each group of instances, are reported in the column Min, Ave, Std and T, respectively.

According to Table 3, we find that the average standard deviation of the objective function values for all groups of randomly generated instances under different values of ε are relatively small, i.e., 62.76 on average, indicating that the HCSA algorithm in general terms maintains satisfactory stability in solving both small and large-sized random instances of our proposed IPOM model. It is also worthwhile to note that the value of ε does influences the performance of the HCSA algorithm. Specifically, compared with the other two settings of ε, the HCSA with $\varepsilon = 0.9999$ produces better solutions with an average of 24.95% lower objective values and 34.76% higher stability, though accompanied by a 17.65% increase in computational time. This observation also suggests that the trade-off between solution quality and computational time should thus be well balanced by fine toning the value of ε in real applications. To sum up, the HCSA exhibits acceptable stability for solving random problem

instances of our proposed problem.

5.2.2 The impact of the locations of customers on algorithm performance

One factor that is expected to affect the algorithm performance is the location of customers. Therefore, we proceed to assess the impact of the location of customers on the algorithm performance by generating two types of instances with different distributions of customers, i.e., randomly distributed customers and clustered customers, and comparing the HCSA with CPLEX. Let $a+bR$ and $a+bC$ represent the randomly distributed (R) and clustered (C) and instances, respectively. Randomly distributed customers are those that are randomly distributed in a square region with a side length of 200 km. For clustered customers, we generated customers in a small square region with the customers' coordinates randomly distributed over [100,120]. Since CPLEX can only solve small instances for the proposed problem, we only conduct experiments on small-sized instances in this subsection. The comparison results between HCSA and CPLEX on small-scale instances under different distributions of customers are reported in Table 4. We can see that the distribution of customers has little influence on the performance of the HCSA for solving the proposed problem since both the values of *Gap* and the CPU runtimes vary little with the distribution types of customers for all groups of instances with different customer sizes. To some extent, this finding also verifies the stability of the HCSA for solving the proposed problem.

Table 2. Performance comparison of HCSA, GA, PSO, and CPLEX

a+b	HCSA		GA		PSO		CPLEX		Gap^1	Gap^2	Gap^3
	Obj	Time (s)	Obj^1	Time (s)	Obj^2	Time (s)	Obj^3	Time (s)			
2+2	**254.41**	84	**254.41**	121	**254.41**	62	**254.41**	**45**	0.00%	0.00%	0.00%
3+3	257.14	**74**	259.15	155	259.33	81	**256.23**	3,600	-0.78%	-0.84%	0.36%
2+4	208.01	**64**	208.13	164	208.02	68	**208.00**	3,600	-0.06%	0.00%	0.00%
4+2	310.96	69	345.10	157	343.79	**62**	**310.80**	3,600	-9.89%	-9.55%	0.05%
4+4	**310.42**	81	371.26	154	358.31	**76**	311.93	3,600	-16.39%	-13.37%	-0.48%
2+6	**387.14**	**61**	412.64	148	419.30	70	410.73	3,600	-6.18%	-7.67%	-5.74%
6+2	501.71	**72**	522.61	124	**501.37**	84	515.53	3,600	-4.00%	0.07%	-2.68%
5+5	**460.42**	79	465.68	121	471.23	**76**	460.73	3,600	-1.13%	-2.29%	-0.07%
2+8	422.45	**78**	**420.30**	182	453.35	88	514.39	3,600	0.51%	-6.82%	-17.87%
8+2	**601.97**	81	673.28	164	645.10	**72**	712.58	3,600	-10.59%	-6.69%	-15.52%
4+6	**378.68**	**83**	422.68	160	401.32	88	410.99	3,600	-10.41%	-5.64%	-7.86%
6+4	**472.14**	91	544.62	212	512.46	**89**	511.06	3,600	-13.31%	-7.87%	-7.62%
10+10	**1,049.43**	**195**	1,792.30	392	1,460.22	226	2,968.36	3,600	-41.45%	-28.13%	-64.65%
15+15	**1,515.36**	**224**	2,231.69	411	1,816.44	435	3,515.69	3,600	-32.10%	-16.58%	-56.90%
20+20	**2,130.23**	**358**	2,691.42	406	2,677.21	464	-	3,600	-20.85%	-20.43%	-
30+30	**2,970.02**	**494**	3,602.55	812	3,381.66	659	-	3,600	-17.56%	-12.17%	-
40+40	**4,583.23**	**603**	5,529.87	1,005	5,254.60	1,216	-	3,600	-17.12%	-12.78%	-
50+50	**5,180.36**	**724**	6,043.72	1,542	5,821.08	1,347	-	3,600	-14.29%	-11.01%	-
100+100	**10,269.27**	**1,321**	12,048.12	2,436	11,724.78	2,131	-	3,600	-14.76%	-12.41%	-
150+150	**19,114.89**	**1,885**	22,756.77	3,574	21,826.19	2,981	-	3,600	-16.00%	-12.42%	-
200+200	**25,834.83**	**2,603**	31,749.20	4,349	30,046.10	3,875	-	3,600	-18.63%	-14.02%	-
Average	**3,676.81**	**444**	4,445.02	799	4,230.30	679	-	-	-12.62%	-9.55%	-

Remarks: $Gap^1=(Obj-Obj^1)/Obj^1\times100\%$; $Gap^2=(Obj-Obj^2)/Obj^2\times100\%$; $Gap^3=(Obj-Obj^3)/Obj^3\times100\%$

Table 3. Comparison results under different values of ε for random instances

$n-m$	$\varepsilon=0.99$				$\varepsilon=0.999$				$\varepsilon=0.9999$			
	Min	*Ave*	*Std*	$T(s)$	*Min*	*Ave*	*Std*	$T(s)$	*Min*	*Ave*	*Std*	$T(s)$
5-5	447.44	461.35	6.45	79	443.48	461.44	4.10	79	447.44	460.78	5.21	77
10-10	1014.87	1035.98	6.26	174	1006.87	1031.98	6.75	195	1004.23	1031.46	5.89	204
20-20	2022.02	2154.56	10.14	311	2013.11	2069.22	8.96	358	2007.96	2054.78	8.12	399
30-30	2994.65	3007.82	11.07	375	2969.05	2974.32	10.87	496	2961.65	2955.89	8.08	494
40-40	4588.30	4624.23	36.47	570	4571.41	4588.64	30.66	622	4566.63	4524.69	21.45	651
50-50	5174.94	5193.45	76.41	614	5171.96	5186.85	54.54	701	5169.58	5178.23	36.63	748
100-100	12436.13	12887.55	185.22	954	12019.13	12241.25	109.12	1158	11945.13	12107.55	55.45	1355
200-200	24838.63	25711.56	464.44	2187	24836.42	25450.56	382.12	2503	24653.63	25032.49	304.44	2667
Average	5852.29	6025.57	80.96	498.62	5812.15	5918.62	61.45	642.00	5755.84	5822.59	45.58	692.54

Table 4. Comparison results between HCSA and CPLEX under different distributions of customers on small-scale instances

Instances	HCSA		CPLEX		Gap
	Obj^1	Time (s)	Obj^2	Time (s)	
2+2R	254.41	84	254.41	45	0.00%
3+3R	257.14	74	256.23	3,600	0.36%
4+4R	310.42	81	311.93	3,600	-0.48%
5+5R	460.42	79	460.73	3,600	-0.07%
10+10R	1,049.43	195	2,968.36	3,600	-64.65%
2+2C	215.46	84	215.46	41	0.00%
3+3C	238.23	72	238.25	3,600	-0.01%
4+4C	257.56	79	257.38	3,600	0.07%
5+5C	342.60	78	386.23	3,600	-1.05%
10+10C	835.02	192	2874.43	3,600	-70.95%
Average	-	102	-	2888	-

Remark: $Gap=(Obj^1-Obj^2)/Obj^2\times100\%$

5.3 Impact analysis of the IPOM

To validate the benefits of the proposed IPOM, we will compare the results with those of the LCDP under the YPOM on test instances, in which YPOM refers to the platooning operation mode considered by You et al. (2020). Kindly note that You et al. (2020) did not consider the fuel-saving benefits of truck platooning. For a fair comparison, we have incorporated the fuel-saving effects to obtain the results under YPOM. More specifically, the YPOM restricts that the drivers are attached to their respective trucks once departing from the terminal, and they cannot move between customers without a truck and have to wait at some customer locations for packing or unpacking. The IPOM considered in this study relaxes the aforementioned assumptions and allows drivers to move among customers by alternative transport modes for subsequent drayage operations since each driver is not necessarily attached to the truck leaving the terminal with himself/herself. In fact, if the unit travel costs or the travel speeds of the alternative transport modes are set to be infinite, the MILP model for IPOM will reduce to that of YPOM, which indicates that the YPOM is a special case of the IPOM. Theoretically speaking, when other conditions remain the same, the IPOM will lead to an optimal objective value better than (put it more exactly, not worse than) that of the YPOM since the MILP model for IPOM is a more general LCDP model and has larger search space than that of the YPOM. To illustrate the differences of the detailed solution results under the IPOM and YPOM, we provide a toy example (150 km × 100 km) of LCDP with 6 customers in Figure 7, where the coordinate beside each node represents the corresponding location information. Specifically, we consider four delivery customers, i.e., D1, D2, D3, D4, and two pickup customers, i.e., P1 and P2. The detailed results under IPOM and YPOM are summarized in Table 5. We can observe that fewer drivers and trucks are employed, and the total operational cost is reduced under the proposed IPOM.

We further present the comparison results of various random instances with different customer sizes under the two different drayage operation modes in Table 6, including the objective function values (Obj), the gaps between the IPOM and YPOM on the objective values (Gap), and the detailed quantitative comparison results in terms of the number of employed drivers (Dr), the number of deployed trucks (Tr), and the total fuel and travel cost (Fc). Again, the minimum value for each indicator in each instance group is highlighted in bold. We can see from Table 6 that the proposed IPOM can effectively reduce both the number of employed drivers and the number of deployed trucks for all groups of instances when compared with the

YPOM. This is consistent with our expectation that fewer drivers and trucks will be needed for the LCDP under the IPOM since drivers can move to other customer sites using alternative transport modes, e.g., motorcycles in the experiment, to drive back the trucks after dropping off their leading trucks, leading to better scheduling and utilization of the drivers and trucks. In addition, we also observe that the IPOM can achieve the lowest fuel and travel costs on average. This implies that some unnecessary deployment or unfavorable detours of trucks traveling in the form of a platoon can be avoided in the IPOM. In total, the proposed IPOM can, on average, save the total operational cost by 5.68% in comparison with the YPOM. This outcome reveals the advantages of our proposed IPOM over the YPOM to reduce the total operational cost. In addition, we find that the average gap values for the middle and large-sized instances (i.e., $a+b>10$) are larger than that for the small-scale (i.e., $a+b\leq 10$) instances, which indicates that the cost-saving benefits of the IPOM are more competitive in larger instances. Moreover, if we dig further, we can find that for a given number of customers, the composition ratio of the pickup and delivery customers does greatly affect the cost-saving performance of the IPOM. In particular, the IPOM is increasingly competitive over the YPOM in the problem instances with a higher composition ratio of pickup to delivery customers than those with lower ones as reflected by the much larger gap values. In fact, the number of loaded trucks required from the terminal and the number of empty trucks that can be shared among customers are largely dependent on the composition ratio of the two customer types.

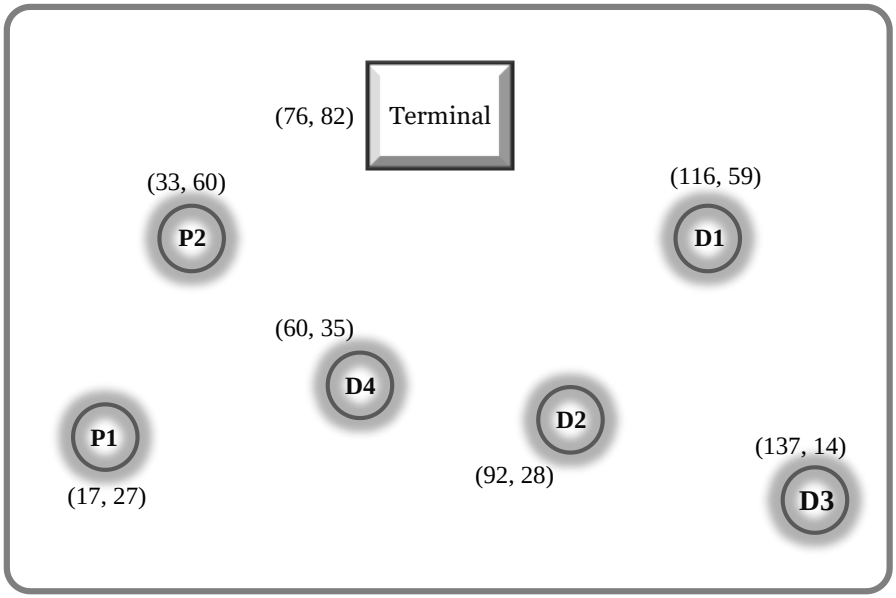


Figure 7. Node locations of the toy example

Table 5. Detailed results of the LCDP under IPOM and YPOM in the toy example

Results	**IPOM**	**YPOM**
Optimal objective value	**412.26**	562.78
Number of drivers	**2**	3
Number of trucks	**4**	5
Driver routes	Driver 1: Terminal-D1-D2-D3-D2-P1-P2-Terminal Driver 2: Terminal-D4-P1-D4-D1-Terminal	Driver 1: Terminal-D1-D2-D3-D2-P1-P2-Terminal Driver 2: Terminal-D4-P1-Terminal Driver 3: Terminal-D1-Terminal
Truck routes	Truck 1: Terminal-D1-Terminal Truck 2: Terminal-D1-D2-P1-P2-Terminal Truck 3: Terminal-D1-D2-D3-D2-P1-D4-D1-Terminal Truck 4: Terminal-D4-D1-Terminal	Truck 1: Terminal-D1-Terminal Truck 2: Terminal-D1-D2-P1-P2-Terminal Truck 3: Terminal-D1-D2-D3-D2-P1-Terminal Truck 4: Terminal-D4-P1-Terminal Truck 5: Terminal-D1-Terminal
Platoon length at each arc	Terminal-D1: 3 Terminal-D4: 1 D1-D2: 2 D2-D3: 1 D3-D2: 1 D2-P1:2 P1-P2: 1 P2-Terminal: 1 D4-P1: 0 P1-D4:1 D4-D1:2 D1-Terminal: 3	Terminal-D1: 3 Terminal D4: 1 D1-D2: 2 D2-D3: 1 D3-D2:1 D2-P1:2 P1-P2: 1 P2-Terminal: 1 D4-P1: 1 P1-D4: 2 P1-Terminal: 2 Terminal-D1: 1 D1-Terminal: 2

Table 6. Comparisons results for the two different operation modes on LCDP instance

a+b	YPOM				IPOM				Gap
	Obj^1	Dr	Tr	Fc	Obj^2	Dr	Tr	Fc	
2+2	**254.41**	**1**	**3**	**4.41**	**254.41**	**1**	**3**	**4.41**	0.00%
3+3	256.64	**1**	**3**	6.64	**256.14**	**1**	**3**	**6.14**	0.19%
2+4	208.56	**1**	3	8.56	**208.01**	**1**	**2**	**8.01**	0.26%
4+2	311.34	**1**	**4**	11.34	**310.80**	**1**	**4**	**10.80**	0.17%
4+4	361.71	**1**	5	12.70	**311.93**	**1**	**4**	**11.93**	13.76%
2+6	460.33	**2**	5	**10.33**	**410.73**	**2**	**4**	10.73	10.77%
6+2	515.27	**2**	**6**	15.27	**512.38**	**2**	**6**	**12.38**	0.56%
5+5	512.03	**2**	6	12.03	**460.42**	**2**	**5**	**10.42**	10.08%
2+8	567.28	**2**	7	17.28	**514.39**	**2**	**6**	**14.39**	9.32%
8+2	713.13	**3**	**8**	13.13	**712.58**	**3**	**8**	**12.58**	0.08%
4+6	512.83	3	6	**12.83**	**413.99**	**2**	**4**	13.99	19.27%
6+4	563.28	3	7	**13.28**	**514.06**	**2**	**6**	14.06	8.74%
10+10	1172.69	**4**	14	72.69	**1,049.43**	**4**	**13**	**68.33**	10.51%
30+30	3,758.94	18	34	258.94	**2,970.02**	**12**	**31**	220.02	20.99%
50+50	5,315.33	23	51	**465.33**	**5,180.36**	**22**	**50**	480.36	2.54%
100+100	11124.35	42	113	1274.35	**10,269.27**	**38**	**107**	**1119.27**	7.69%
200+200	26596.77	109	251	**3446.77**	**25,834.83**	**101**	**246**	3434.83	2.86%
Average	3129.70	13	31	332.70	**2951.99**	**12**	**30**	**320.74**	5.68%

Remark: $\text{Gap} = (\text{Obj}^1 - \text{Obj}^2) / \text{Obj}^1 \cdot 100\%$

5.4 Sensitivity analysis

In this subsection, we will explore the impacts of several major influential factors, i.e., the labor cost, the maximum platoon size, and different types of available transport modes, on the system performance in terms of the total operational cost (Tc), the number of employed drivers (Dr), the number of deployed trucks (Tr) and the total fuel and travel cost (Fc), including fuel consumption cost of trucks (Fc^1) and the travel cost incurred by corresponding alternative transport mode (Fc^2). The sensitivity analysis will be carried out in the instance groups '5+5', '15+15', '30+30', and '50+50'. Kindly note that the proposed HCSA has proven to be robust enough to deliver good-quality results as validated by Subsection 5.2 of this study. Therefore, the conclusions obtained from the sensitivity analysis are reliable.

Impact of labor cost

To explore the impact of the labor cost on the system performance, we will compare the solutions to the proposed problem under different labor costs by setting $\lambda_1 \in \{5, 50, 100, 200\}$ and the results are tabulated in Table 7. We also report the fuel consumption cost of trucks (Fc^1) and the travel cost of the motorcycles (Fc^2), as well as the ratio of Fc^2 / Fc in Table 7. The variations of the number of employed drivers, the number of deployed trucks, and the total fuel and travel cost against different labor costs are further illustrated in Figure 8 for visualization. It can be seen from Table 7 and Figure that the increase in the labor cost will reduce the number of employed drivers, but, at the same time, it will cause the number of deployed trucks and the total fuel and travel cost to increase. Notably, by factoring in the labor cost at the price of 200, the number of employed drivers can be reduced by 20% (calculated by (12.5-10)/12.5), whereas the number of trucks and the total fuel cost will increase by 11.41% (calculated by (26.3-23.3)/26.3) and 31.13% (calculated by (239.43-164.89)/239.43) on average, respectively, when compared with the results for the model with the labor cost set at 5. Nevertheless, it is also worthwhile to mention that the reduction in the number of required drivers is limited. For example, for the '15+15' instance group (30 customers), the number of employed drivers remains unchanged when the labor cost increases from 100 to 200. Similar examples can be found in other instance groups as well. This may be attributed to the strict restriction imposed by Eq. (16), making some more cost-saving employment plans or routing strategies unfeasible for the maximum platoon size constraint.

We dig deeper into the results of the fuel and travel costs in Table 8. It shows that the

travel cost for motorcycles accounts for an increasingly larger part of the total fuel and travel cost (reflected by an increasing value of Fc^2/Fc) with the increase in the labor cost. This suggests that a higher labor cost can facilitate the drivers to make efficient use of motorcycles to serve as many customers as possible. It is also worthwhile to mention that the increase of Fc^2/Fc is not uniform with respect to the change of the labor cost. It shows no obvious increase under small and large labor costs but a fast increase under a moderate labor cost. This finding, on the one hand, indicates that the potential effect of the labor cost on promoting the drivers' use of motorcycles will not work much unless the labor cost becomes comparatively large, in which case the high labor wage level will render the engagement of another driver unfavorable. On the other hand, we should caution that there may exist a threshold beyond which further increase in the labor cost would take little effect on the promotion of using motorcycles since the use of motorcycles would make drivers spend more time on the routes due to its comparatively lower travel speed, resulting in some more labor-saving routing and scheduling plans infeasible in the maximum planning horizon constraint. We further depict the changes of the average values of Fc^2/Fc with respect to the customer size in Figure 9. We find that the value of Fc^2/Fc follows an upward trend with an increasing increment rate, indicating that the change of labor cost can have a larger positive impact on the drivers' decisions regarding the use of motorcycles in larger problem instances. In fact, customers are distributed relatively more densely in larger-scale instances, and in this case, it is more convenient and practicable for the drivers to frequently move among customers by motorcycles with fewer detours than in smaller instances.

Table 7. Effect of the labor cost on the system performance

a+b	$\lambda_1 = 5$				$\lambda_1 = 50$				$\lambda_1 = 100$				$\lambda_1 = 200$			
	Dr	Tr	Fc	Tc	Dr	Tr	Fc	Tc	Dr	Tr	Fc	Tc	Dr	Tr	Fc	Tc
5+5	4	5	25.42	295.42	2	5	30.69	380.69	2	5	10.42	460.42	2	5	32.55	682.55
15+15	9	11	100.74	695.74	7	14	139.45	1,189.45	6	15	165.36	1,515.36	6	15	178.12	2,128.12
30+30	13	32	185.99	1,850.99	13	33	202.14	2,502.14	12	31	220.02	2,970.02	11	34	241.23	4,141.23
50+50	24	45	347.44	2,717.44	22	46	371.25	3,771.25	22	50	480.36	5,180.36	21	51	505.82	7,255.82
Average	12.5	23.3	164.89	1,389.89	11.00	24.5	185.88	1,960.88	10.50	25.3	224.11	2,536.61	10.0	26.3	239.43	3,551.93

Table 8. Detailed solution results for the fuel and travel cost under different labor costs

a+b	$\lambda_1 = 5$			$\lambda_1 = 50$			$\lambda_1 = 100$			$\lambda_1 = 200$		
	Fc^1	Fc^2	Fc^2 / Fc	Fc^1	Fc^2	Fc^2 / Fc	Fc^1	Fc^2	Fc^2 / Fc	Fc^1	Fc^2	Fc^2 / Fc
5+5	20.12	5.30	0.21	24.29	6.40	0.21	8.10	2.32	0.22	25.07	7.48	0.23
15+15	80.04	20.70	0.21	110.17	29.28	0.21	127.33	38.03	0.23	137.16	40.96	0.23
30+30	148.89	37.10	0.20	158.54	43.60	0.22	160.62	59.40	0.27	176.83	64.40	0.27
50+50	236.64	110.80	0.29	246.55	124.70	0.34	345.16	135.20	0.28	365.12	140.70	0.28
Average	121.42	43.48	0.23	136.48	49.40	0.23	165.49	58.63	0.25	176.36	63.08	0.26

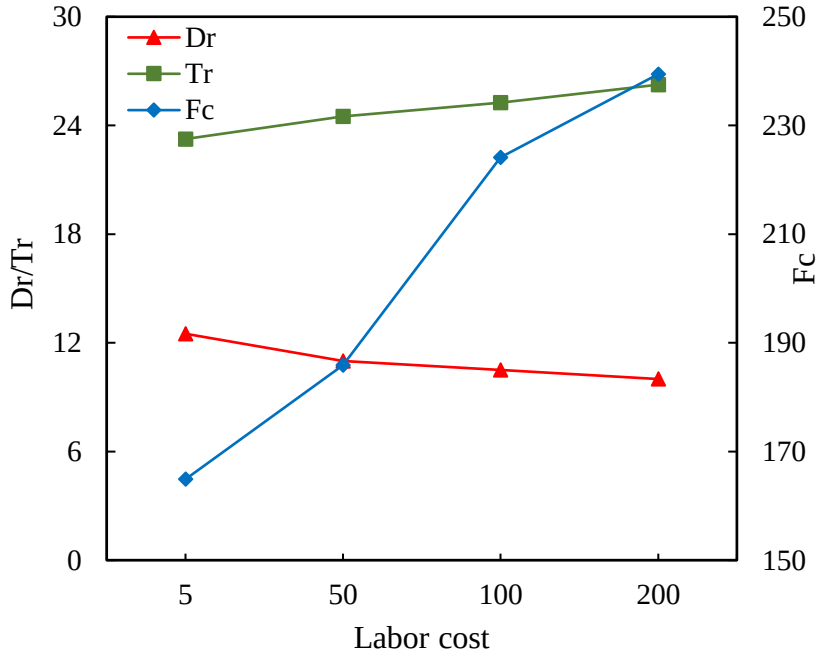


Figure 8. Variations of the number of drivers, number of trucks, and total fuel and travel cost with the increase of labor cost

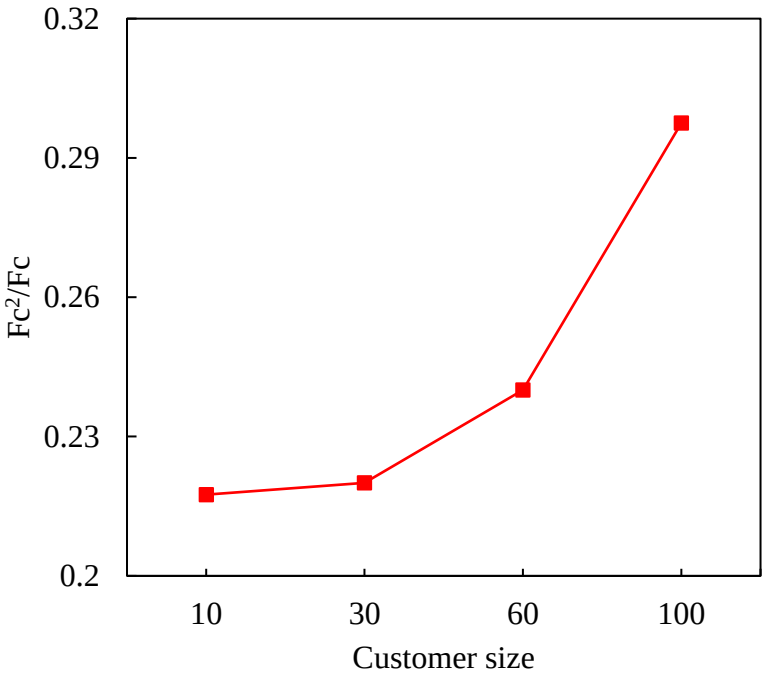


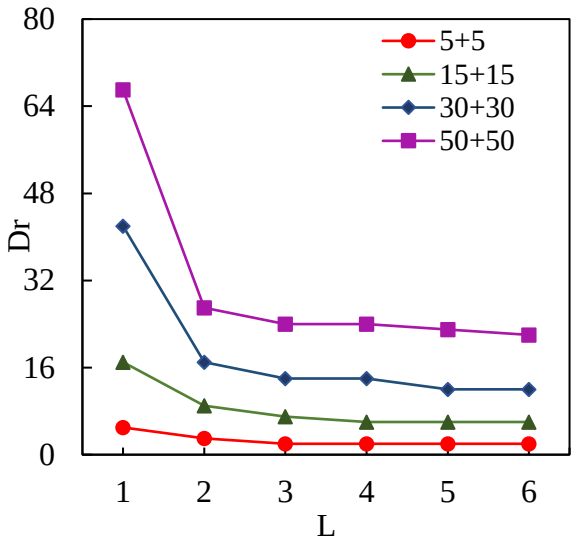
Figure 9. Variations of the values of Fc^2 / Fc with the increase of customer size

Impact of maximum platoon size

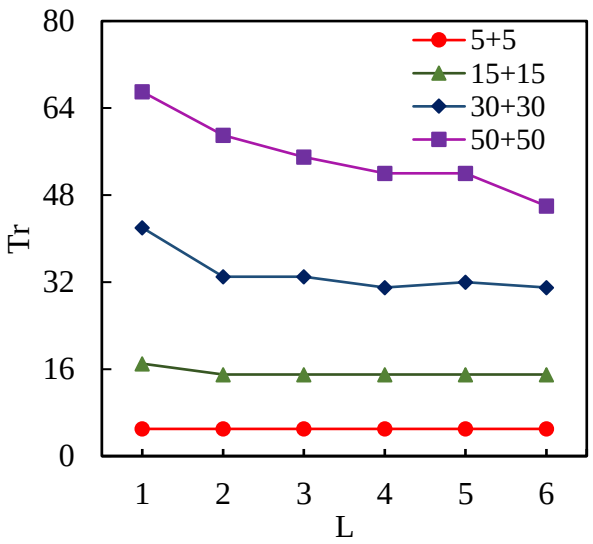
To explore the impact of the maximum platoon size on the system performance, we will compare the solutions to the proposed model under different maximum platoon sizes by setting $L = \{1, 2, 3, 4, 5, 6\}$ in Eq. (16). The results are summarized in Table 9. We also present the detailed solution information related to the fuel and travel cost in Table 10. The variations of the number of employed drivers, the number of deployed trucks, the total fuel and travel cost, and the total operational cost with the increase of the maximum platoon size are further visualized in Figure 10 (a), (b), (c) and (d), respectively.

Overall, it can be seen from Table 9, Table 10, and Figure 10 that the maximum platoon size largely affects the system performance since distinctive indicator values are obtained under different maximum platoon sizes. More specifically, the number of employed drivers decreases with the increase of the maximum platoon size. Notably, on average the number of engaged drivers under $L = 6$ can be reduced by almost two-thirds of that under $L = 1$. This aligns with our expectation: the larger the maximum platoon size, the more the trucks can be operated simultaneously in a platoon using one driver, and hence the fewer drivers are needed. In addition, we also see that the number of deployed trucks, by and large, follows a downward trend with the increase in L. Nevertheless, the reduction in the number of engaged drivers and trucks causes the total fuel and travel costs to increase. For example, for the '50+50' instance group (100 customers), the value of Fc almost doubles when the maximum platoon size increases from 1 to 6. This can be explained by the fact that longer travel distances and larger detours of trucks and motorcycles would be required when fewer drivers and trucks are employed to serve the same number of customers. Moreover, as for the components of the total fuel and travel cost, we can see from Table 10 that the travel cost incurred by motorcycles accounts for an increasing proportion, i.e., from 0.06 to 0.25 on average, of the total fuel and travel cost with the increase in platoon size. This result suggests that a larger allowable platoon size could stimulate a driver to make the most of the motorcycles to serve as many customers as possible in his/her own route. Fortunately, it is encouraging to find that the total operational cost declines with the increase of the maximum platoon size in general. This may be attributed to a higher possibility of finding better routing and scheduling plans for both drivers and trucks due to the larger solution space under a larger platoon size. However, it is also worthwhile to mention that lengthening the allowable platoon size will not always contribute much to the overall performance in terms of labor/truck reduction and cost savings as the variations of the

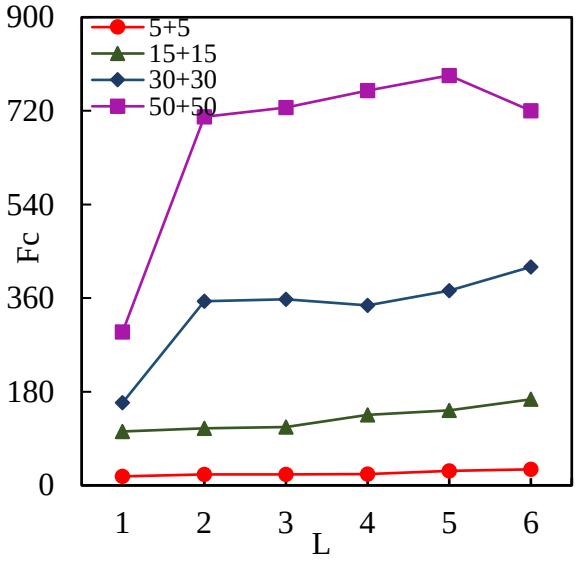
values of the four considered indicators are marginal for most instances when the maximum platoon size exceeds 4, especially for small instances ('5+5' and '15+15' instance groups). In fact, given a certain number of customers and the composition ratio of the customer types, the numbers of required drivers and trucks should be beyond their certain respective thresholds; otherwise, the packing/unpacking requests of the customers may not be completed within the maximum planning horizon. As a result, an overlong maximum platoon size would have nothing to do with the operational efficiency improvement.



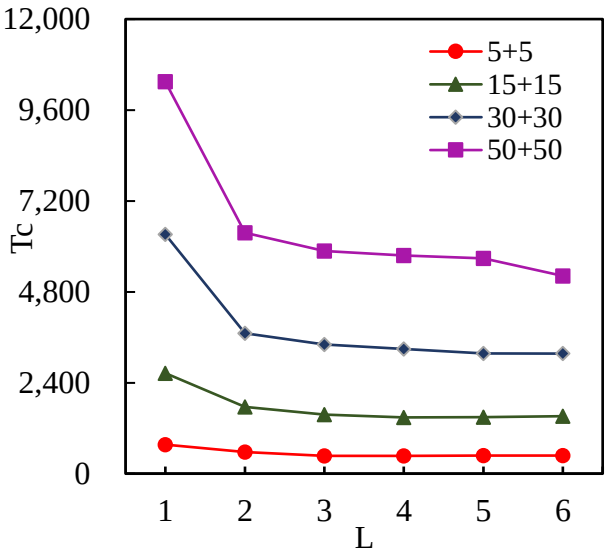
(a) Variations of Dr with the increase of L



(b) Variations of Tr with the increase of L



(c) Variations of Fc with the increase of L



(d) Variations of Tc with the increase of L

Figure 10. Variations of four indicator values with the increase of the maximum platoon size

Table 9. Effect of the maximum platoon size on the system performance

L	5+5				15+15				30+30				50+50			
	Dr	Tr	Fc	Tc	Dr	Tr	Fc	Tc	Dr	Tr	Fc	Tc	Dr	Tr	Fc	Tc
1	5	5	17.19	767.19	17	17	103.51	2,653.51	42	42	158.76	6,458.76	67	67	294.88	10,344.88
2	3	5	20.73	570.73	9	15	109.65	1,759.65	17	33	154.07	3,504.07	27	59	278.59	5,928.59
3	2	5	20.88	470.88	7	15	112.23	1,562.23	14	33	167.72	3,217.72	24	55	326.55	5,476.55
4	2	5	21.78	471.78	6	15	135.41	1,485.41	14	31	196.12	3,146.12	24	52	377.21	5,377.21
5	2	5	27.72	477.72	6	15	144.12	1,494.12	12	32	204.49	3,004.49	23	52	458.12	5,358.12
6	2	5	10.42	460.42	6	15	165.36	1,515.36	12	31	220.02	2,970.02	22	50	480.36	5,180.36

Table 10. Detailed solution results for the fuel and travel cost under different maximum platoon sizes

L	5+5			15+15			30+30			50+50			Average
	Fc^1	Fc^2	Fc^2 / Fc	Fc^1	Fc^2	Fc^2 / Fc	Fc^1	Fc^2	Fc^2 / Fc	Fc^1	Fc^2	Fc^2 / Fc	Fc^2 / Fc
1	17.19	0.00	0.00	101.14	2.37	0.02	140.52	18.24	0.11	263.66	31.22	0.10	0.06
2	17.08	3.65	0.18	98.23	11.42	0.10	122.18	31.89	0.21	210.18	68.41	0.25	0.19
3	16.00	4.88	0.23	94.26	17.97	0.16	133.33	34.39	0.21	242.2	84.35	0.26	0.22
4	16.65	5.13	0.24	108.61	26.80	0.20	153.65	42.47	0.22	270.98	106.23	0.28	0.24
5	21.14	6.58	0.24	117.06	27.06	0.19	150.44	54.05	0.26	333.98	124.14	0.27	0.24
6	23.63	7.10	0.23	138.56	32.80	0.20	160.62	59.40	0.27	345.16	135.20	0.28	0.25

Table 11. Effect of the value of (v,c) on the system performance

(v,c)	5+5				15+15				30+30				50+50			
	Dr	Tr	Fc	Tc	Dr	Tr	Fc	Tc	Dr	Tr	Fc	Tc	Dr	Tr	Fc	Tc
(35,0.5)	2	5	10.42	460.42	6	15	165.36	1515.36	12	31	220.02	2970.02	22	50	480.36	5180.36
(30,0.2)	2	5	8.04	458.04	6	15	116.96	1466.96	12	30	212.65	2912.65	22	50	435.04	5235.04
(65,1.5)	2	5	19.56	469.56	6	15	170.05	1520.05	12	31	305.68	3055.68	20	54	564.37	5264.37
(80,10)	2	5	29.78	479.78	6	15	186.44	1536.44	11	32	433.82	3133.82	20	56	626.49	5426.49
$(0, \infty)$	2	6	12.03	512.03	8	17	123.20	1773.20	18	34	258.94	3758.94	24	60	574.79	5974.79

Impact of the types of available transport modes

Another important factor that is believed to largely affect the system performance is the type of available alternative transport modes that can be used by drivers. Therefore, we will analyze the solution results to the proposed LCDP under four different available transport modes featured by different combinations of two important property values, including average travel speed v (measured by km/h) and unit travel cost c, by setting $(v,c) \in \{(35,0.5),(30,0.2),(65,1.5),(80,10)\}$, which will represent the motorcycle, bus, shared electric vehicles, and taxi, respectively. Kindly note that we try to include a special case by considering $(0,\infty)$ to represent that no alternative transport modes can be used by drivers, which can be seen as the operation mode considered in You et al. (2020). The outcomes under different values of (v,c) are tabulated in Table 11.

As shown in Table 11, we observe that the use of alternative transport modes can reduce the number of drivers and trucks needed to complete the same number of the customers' drayage requests. Notably, the number of engaged drivers and the number of deployed trucks can be reduced by 30.59% and 12.23% on average, respectively, when compared with the results of the model under $(0,\infty)$. This result aligns with our expectation that fewer drivers and trucks will be needed for the LCDP under the IPOM since the use of alternative transport modes enables drivers to conduct service tasks for other customers without waiting at the current customer sites after dropping off their trucks, resulting in higher operational efficiency and better utilization of resources. It is also worthwhile to note that such labor/truck saving benefits vary little with different types of available transport modes as the variations of the values of the two indicators, i.e., Dr and Tr, under different values of (v,c) are marginal for most instances. This result indicates that any types of transport modes available for the drivers to move among customers besides trucks would play a part in resource saving and cost reduction. Furthermore, we find that the use of alternative transport modes can significantly reduce the total operational cost by 19.29% on average when compared with that with $(0,\infty)$, which quantitatively validates the significant benefits of the use of alternative transport modes to effectively reduce the total operational cost for the LCDP.

6. Conclusions

This study investigates a more general local container drayage problem under an improved platooning operation mode, in which the trucks from different platoons can platoon

together for fuel savings in the transportation process of containers, the drivers can move by alternative transport modes besides trucks, and the sharing of empty trucks among customers is allowed. To determine the optimal numbers, routes, and schedules of the drivers and trucks that minimize the total operational cost, we developed a MILP model that can specifically incorporate the operational characteristics of the proposed platooning mode. Due to the structural complexity of the proposed model, a tailored heuristic construction method incorporating simulated annealing was proposed to obtain good-quality solutions for the underlying problem. Extensive numerical experiments were conducted to verify the effectiveness and efficiency of the proposed model and solution method. We also demonstrated the advantages of the improved platooning mode and explored the impacts of some major influential factors on the system performance.

Future research work can be undertaken in several aspects. First, more efficient heuristic algorithms are highly anticipated to realize the (real-time) planning for the large-scale LCDP. Second, the proposed model can be extended from various aspects considering stochastic parameters such as the request arrivals and travel speeds and more practical constraints like limited resources and customers' service time windows. Moreover, this study assumes that each driver will undertake at most one route. It would be a very interesting idea to allow drivers to perform multiple trips between terminal and customer locations. Last but not least, more efficient drayage operation modes for the LCDP can be developed by incorporating other emerging technologies, such as electric trucks, in which the recharging strategies should be considered in the routing and scheduling process of the trucks in the future.

References

APTA, 2021. Public Transportation Facts. https://www.apta.com/news-publications/public-transportation-facts/.

Benantar, A., Abourraja, M. N., Boukachour, J., Boudebous, D., Duvallet, C., 2020. On the integration of container availability constraints into daily drayage operations arising in France: Modelling and optimization. Transportation Research Part E: Logistics and Transportation Review, 140, 101969.

Bernardo, M., Pannek, J., 2018. Robust solution approach for the dynamic and stochastic vehicle routing problem. Journal of Advanced Transportation 34, 78-89.

Bhoopalam, A. K., Agatz, N., Zuidwijk, R., 2018. Planning of truck platoons: A literature review and directions for future research. Transportation research part B: methodological, 107, 212-228.

Braekers, K., An Caris, Janssens, K., 2014. Bi-objective optimization of drayage operations in the service area of intermodal terminals. Transportation Research Part E: Logistics and Transportation Review 65, 50-69.

Bruglieri, M., Mancini, S., Peruzzini, R., Pisacane, O., 2021. The multi-period multi-trip container drayage problem with release and due dates. Computers & Operations Research, 125, 105102.

Calvert, S. C., Schakel, W. J., van Arem, B., 2019. Evaluation and modelling of the traffic flow effects of truck platooning. Transportation research part C: emerging technologies, 105, 1-22.

Chen, R., Chen, S., Cui, H., Meng, Q., 2021. The container drayage problem for heterogeneous trucks with multiple loads: A revisit. Transportation Research Part E: Logistics and Transportation Review, 147, 102241.

Chen, R., Meng, Q., Jia, P., 2022. Container port drayage operations and management: Past and future. Transportation Research Part E: Logistics and Transportation Review 159: 102633.

Chen, S., Wang, H., Meng, Q., 2021. Autonomous truck scheduling for container transshipment between two seaport terminals considering platooning and speed optimization. Transportation Research Part B: Methodological 154: 289-315.

Cui, H., Chen, S., Chen, R., Meng, Q., 2022. A two-stage hybrid heuristic solution for the container drayage problem with trailer reposition. European Journal of Operational Research, 299(2), 468-482.

Davila, A., Aramburu, E., Freixas, A., 2013. Making the best out of aerodynamics: Platoons. SAE Technical Paper.

DHL, 2021. Ocean freight market update. https://www.dhl.com/content/dam/dhl/global/dhl-global-forwarding/documents/pdf/glo-dgf-ocean-market-update.pdf.

Engholm, A., Kristoffersson, I., Pernestål, A., 2021. Impacts of Large-Scale Driverless Truck Adoption on the Freight Transport System. Transportation Research Part A: Policy and Practice 154, 227-254.

Escudero, A., Cuberos-Gallardo, M., Muñuzuri, J., 2019. Using simulated annealing to solve the daily drayage problem with hard time windows. New Global Perspectives on Industrial Engineering and Management, 83-90.

Escudero, A., Muñuzuri, J., Guadix, J., Arango, C., 2013. Dynamic approach to solve the daily drayage problem with transit time uncertainty. Computers in industry 64(2), 165-175.

Ghosal, A., Sagong, S. U., Halder, S., Sahabandu, K., Conti, M., Poovendran, R., Bushnell, L., 2021. Truck platoon security: State-of-the-art and road ahead. Computer Networks, 185, 107658.

Imai, A., Nishimura, E., Current, J., 2007. A Lagrangian relaxation-based heuristic for the vehicle routing with full container load. European journal of operational research 176(1), 87-105.

Kuo, Y., 2010. Using simulated annealing to minimize fuel consumption for the time-dependent vehicle routing problem. Computers & Industrial Engineering 59(1), 157-165.

Lammert, M.P., Duran, A., Diez, J., Burton, K., Nicholson, A., 2014. Effect of platooning on fuel consumption of class 8 vehicles over a range of speeds, following distances, and mass. SAE International Journal of Commercial Vehicles 7(2), 626-639.

Larsen, R., Rich, J., & Rasmussen, T. K., 2019. Hub-based truck platooning: Potentials and profitability. Transportation Research Part E: Logistics and Transportation Review, 127, 249-264.

Liang, H., Zhou, S., Liu, X., Zheng, F., Hong, X., Zhou, X., Zhao, L., 2021. A dynamic resource allocation model based on SMDP and DRL algorithm for truck platoon in vehicle network. IEEE Internet of Things Journal, 9(12), 10295-10305.

Lu, X.Y., Shladover, S.E., 2014. Automated truck platoon control and field test. Road Vehicle Automation. Springer, 247-261.

Macharis, C., Bontekoning, Y.M., 2004. Opportunities for OR in intermodal freight transport research: A review. European Journal of Operational Research 153(2), 400-416.

Mahdinia, I., Arvin, R., Khattak, A.J., Ghiasi, A., 2020. Safety, energy, and emissions impacts of adaptive cruise control and cooperative adaptive cruise control. Transportation Research Record, 2674(6), 253–267.

Moghaddam, M., Pearce, R. H., Mokhtar, H., Prato, C. G., 2020. A generalised model for container drayage operations with heterogeneous fleet, multi-container sizes and two modes of operation. Transportation Research Part E: Logistics and Transportation Review, 139, 101973.

Mousavi, S., Vahdani, B., 2017. A robust approach to multiple vehicle location-routing problems with time windows for optimization of cross-docking under uncertainty. Journal of Intelligent & Fuzzy Systems, 32(1): 49-62.

Oudani, M., 2021. A Simulated Annealing Algorithm for Intermodal Transportation on Incomplete Networks. Applied Sciences 11(10), 4467.

SCHENKER, 2019. Platooning in the logistics industry: Researchers see great potential in real operations after tests. https://www.dbschenker.com/lu-en/about/press/corporate-news/platooning-in-the-logistics-industry--researchers-see-great-potential-in-real-operations-after-tests-594872.

She, R., Ouyang, Y., 2022. Generalized link cost function and network design for dedicated truck platoon lanes to improve energy, pavement sustainability and traffic efficiency. Transportation Research Part C: Emerging Technologies, 140, 103667.

Sivanandham S., Gajanand M., 2020. Platooning for sustainable freight transportation: an adoptable practice in the near future? Transport Reviews, 40(5): 581-606.

Song, Y., Zhang, J., Liang, Z., Ye, C., 2017. An exact algorithm for the container drayage problem under a separation mode. Transportation Research Part E: Logistics and Transportation Review 106, 231-254.

Wang, C., Mu, D., Zhao, F., 2015. A parallel simulated annealing method for the vehicle routing problem with simultaneous pickup–delivery and time windows. Computers & Industrial Engineering, 83: 111-122.

Wang, D., Zhang, R., 2019. Double-trailer drop-and-pull container drayage problem, 2019 Chinese Control And Decision Conference (CCDC). IEEE, pp. 4320-4325.

Wei, L., Zhang, Z., Zhang, D., Leung, S.C., 2018. A simulated annealing algorithm for the capacitated vehicle routing problem with two-dimensional loading constraints. European Journal of Operational Research 265(3), 843-859.

Xiao, Y., Zhao, Q., Kaku, I., Xu, Y., 2012. Development of a fuel consumption optimization model for the capacitated vehicle routing problem. Computers & Operations Research 39(7), 1419-1431.

Xue, Z., Lin, H., You, J., 2021. Local container drayage problem with truck platooning mode. Transportation Research Part E: Logistics and Transportation Review 147, 102211.

Xue, Z., Zhang, C., Lin, W.-H., Miao, L., Yang, P., 2014. A tabu search heuristic for the local container drayage problem under a new operation mode. Transportation Research Part E: Logistics and Transportation Review 62, 136-150.

You, J., Miao, L., Zhang, C., Xue, Z., 2020. A generic model for the local container drayage problem using the emerging truck platooning operation mode. Transportation Research Part B: Methodological 133, 181-209.

You, J, Wang, Y., Xue, Z. J., 2022. An exact algorithm for the multi-trip container drayage problem with truck platooning. https://papers.ssrn.com/sol3/papers.cfm?abstract_id=4006049

Zhang, R., Huang, C., Wang, J., 2020. A novel mathematical model and a large neighborhood search algorithm for container drayage operations with multi-resource constraints. Computers & Industrial Engineering, 139, 106143.

Zhang, R., Yun, W.Y., Moon, I.K., 2011. Modeling and optimization of a container drayage problem with resource constraints. International Journal of Production Economics 133(1), 351-359.

Zhang, R., Lu, J.-C., Wang, D., 2014. Container drayage problem with flexible orders and its near real-time solution strategies. Transportation Research Part E: Logistics and Transportation Review 61, 235-251.

Zhang, R., Wang, D., Wang, J., 2020. Multi-Trailer Drop-and-Pull Container Drayage Problem. IEEE Transactions on Intelligent Transportation Systems.

Zhang, R., Zhao, H., Moon, I., 2018. Range-based truck-state transition modeling method for foldable container drayage services. Transportation Research Part E: Logistics and Transportation Review 118, 225-239.

Appendix: Notations

Indices and sets

$\mathbf{N}$	Set of nodes
$\mathbf{A}$	Set of arcs
$\mathbf{G=(N,A)}$	Graph with node set $\mathbf{N}$ and arc set $\mathbf{A}$
$\mathbf{C}$	Set of tasks
$\mathbf{V}$	Set of drivers
$\mathbf{K}$	Set of trucks
i, j, g	Indices for nodes
(i, j)	Index for arc
v	Index for driver
k	Index for truck

Known parameters

t_{ij}	Time required to traverse arc (i, j) by truck
t'_{ij}	Time required to traverse arc (i, j) by alternative transport modes
c_{ij}	The incurred fuel consumption cost to traverse arc (i, j) by trucks
c'_{ij}	The incurred travel cost to traverse arc (i, j) by alternative transport modes
p_i	The packing/unpacking time required by task node $i \in \mathbf{C}_1$
L	Maximum platoon size
T	Maximum planning horizon
η	Fuel reduction rate by platooning
λ_1	Daily fixed cost to employ a driver
λ_2	Daily fixed cost to deploy a truck

Decision variables

x_{ij}^{v}	Binary variable indicating whether driver $v \in \mathbf{V}$ serves nodes $i \in \mathbf{N}$ and $j \in \mathbf{N}$ consecutively in a route
y_{ij}^{k}	Binary variable indicating whether truck $k \in \mathbf{K}$ traverses arc (i, j)
α_{ij}^{od}	Binary variable indicating whether arc (i, j) is used to connect the (o, d) pair
β_{ij}	Binary variable indicating whether there is a truck that traverses arc (i, j)
s_i	The service beginning time of task node $i \in \mathbf{C}$