Transportation Research Part E 188 (2024) 103638

Contents lists available at ScienceDirect

Transportation Research Part E

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

Platform-based stable truck matching problem with trailer-swapping mode

Wenxiang Peng [a,b], Xiangsheng Chen [a,b], Zhaojie Xue [a,b,c,*], Yubin Liao [a,b], Jintao You [d]

[a] College of Civil and Transportation Engineering, Shenzhen University, Shenzhen 518060, China
[b] Key Laboratory of Coastal Urban Resilient Infrastructures (Shenzhen University), Ministry of Education, Shenzhen 518060, China
[c] State Key Laboratory of Subtropical Building and Urban Science, Shenzhen 518060, China
[d] Shenzhen Research Institute of Big Data, Shenzhen 518172, China

ARTICLE INFO

Keywords:
Collaborative transportation
Platform
Trailer-swapping mode
Iterative preference-list-trim heuristic algorithm
Integer programming
Preference list

ABSTRACT

This study investigates the platform-based stable truck-matching problem with trailer-swapping mode (STMP-TSM). The TSM is a novel collaborative transportation approach in which trucks participate in trailer swapping, significantly decreasing the rate of empty trucks and reducing transportation costs. In the STMP-TSM, a platform delivers a trailer-swapping scheme that satisfies all participating trucks. Correspondingly, an integer linear programming model is developed to maximize the total truck utility of the STMP-TSM. A specific preference list based on a chain-data structure is meticulously constructed to obtain a stable matching scheme. The preference list enables more generalized truck matching. In addition, a series of acceleration strategies is proposed to expedite the generation of a preference list while effectively reducing its length. An iterative preference-list-trim heuristic algorithm is designed, which strategically trims chains in the preference list to solve the STMP-TSM efficiently. As a benchmark, an integer linear programming model is developed based on the preference list to solve the truck-matching problem using the TSM. Finally, a series of numerical experiments are conducted to evaluate the performance of the proposed algorithm, assess the practicality of the TSM, and analyze the influences of the key parameters.

1. Introduction

The trucking industry is an indispensable component of the global supply chain, but faces formidable challenges such as inefficiency and resource wastage. Traditional trucking modes, in which trucks operate independently with small and dispersed service scales, are characterized by frequent instances of empty-truck travel, which increases fuel waste and operational costs. Addressing these issues requires the exploration of innovative solutions (Islam and Olsen, 2014).

Collaborative transportation modes and central platforms have emerged to address the efficiency and resource-utilization challenges prevalent in the trucking industry (Basso et al., 2019; Deng et al., 2023). Collaborative transportation modes efficiently integrate capacity resources, potentially minimizing the distances traveled by empty trucks. Concurrently, the central platform, a pivotal

* Corresponding author at: Room A423, Zhigong Building, Shenzhen University, Shenzhen 518060, China.
E-mail addresses: pengwenxiang@126.com (W. Peng), xschen@szu.edu.cn (X. Chen), zjxue@szu.edu.cn (Z. Xue), 2110474158@email.szu.edu.cn (Y. Liao), youjintao@sribd.cn (J. You).

https://doi.org/10.1016/j.tre.2024.103638
Received 6 February 2024; Received in revised form 20 June 2024; Accepted 21 June 2024
Available online 26 June 2024

component of collaborative transportation, leverages advanced information technology and intelligent scheduling systems to optimize transportation processes (Badraoui et al., 2023). This not only enhances the ability of transportation companies to respond to market demands, but also contributes to an overall reduction in operational costs. Consequently, these innovative collaborative transportation modes and central platforms present innovative solutions for the trucking industry, providing new possibilities for achieving more efficient and sustainable transportation systems (Pan et al., 2019).

The trailer-swapping mode (TSM) is a novel and efficient collaborative transportation method. Diverging from traditional origin-to-destination transportation, this mode facilitates trailer swapping at specific locations after a portion of the transit, with each truck assuming responsibility for the transported trailer after swapping. Upon delivering the goods to the destination, the truck returns to its origin. In contrast to the traditional mode, in which trucks are typically empty on the return journey, the TSM enables trucks to interconnect during transportation, swap trailers, and promote resource sharing. This innovative method significantly reduces the empty-truck rate and driver workload. Moreover, this collaborative transportation concept not only mitigates resource wastage from empty travel, but also contributes to a reduction in overall transportation costs (Akpan, 2023; Fan et al., 2020). The TSM is further illustrated using Fig. 1 as a demonstration. The transportation network is assumed as a regular hexagon with a side length of a, and the trailer-swapping node is located at its center. In the traditional mode, each truck travels a laden mileage of $2a$ and an empty mileage of $2a$. In the TSM, three trucks are participating in swapping at the trailer-swapping node. Truck 1 leads the trailer of truck 2; truck 2 leads the trailer of truck 3; and truck 3 leads the trailer of truck 1. Each truck travels a laden mileage of $2a$ and an empty mileage of a. At this point, the empty mileage of all trucks is reduced by 50 %.

The central platform plays a crucial role in facilitating the successful operation of the TSM, as it collects truck-transportation information and computes essential scheduling and delivery schemes (Mei et al., 2021; Miller et al., 2020). As illustrated in Fig. 1, the TSM relies on the robust support of the central platform. Within a central platform that fosters information sharing, such as Uber Freight, Convoy, and Loadsmart in the current logistics industry, the participating trucks gain insight into the transportation information of other trucks (Sun and Yin, 2021). The platform shares the information only after collecting data from all trucks, preventing any truck from falsifying personal information to increase its own utility based on information submitted by other trucks. Therefore, the information provided by participating trucks is assumed to be reliable. In this setting, all trucks participate in trailer-swapping schemes that maximize their individual utility by rejecting schemes with lower returns. Given that the system optimum solution may not always be applicable in collaborative transportation modes with shared information, some trucks may refuse to accept the proposed solutions, leading to the spontaneous formation of small alliances (Chen et al., 2023). To eliminate these occurrences, stable matching solutions must be explored. Under stable matching solutions, all participating trucks accept the proposal, and the formation of small alliances is prevented. Transparency fostered by information sharing and a stable matching mechanism builds trust between truckers and the platform, fostering greater engagement and loyalty. This guarantees stable revenue for the platform through commissions. In this context, this study proposes the platform-based stable truck-matching problem with TSM (STMP-TSM).

The STMP-TSM aims to maximize the total utility of trucks while considering their stability constraints. To solve the STMP-TSM, three key concerns must be addressed:

(1) Obtaining all trailer-swapping groups beneficial to trucks.

To address this, we construct a truck preference list, effectively exploring all the trailer-swapping groups that are beneficial to trucks.

(2) Determining whether a trailer-swapping scheme is a stable matching.

By introducing relevant concepts, such as blocking groups, we can effectively determine whether a scheme is a stable matching.

(3) Finding a utility-optimized stable matching scheme from numerous trailer-swapping groups.

Finally, by developing a mathematical model and designing a solution algorithm, we successfully obtain a specific stable matching scheme.

By addressing these three concerns, we gain insights that facilitate the practical application of the TSM.

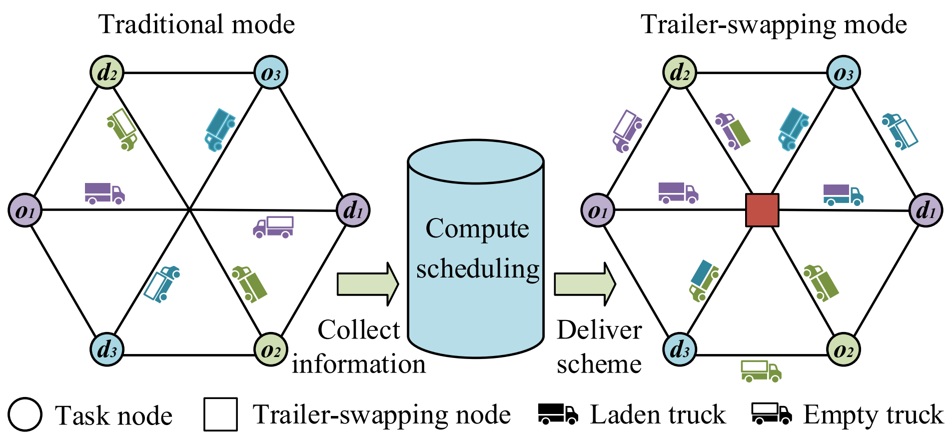

Fig. 1. Interactive process of platform-based TSM.

1.1. Literature review

The TSM employs vehicle separation technology to aid trucks in cost reduction and efficiency enhancement. Currently, research on transportation problems related to vehicle separation technology mainly focuses on the truck and trailer routing problem (TTRP) and the swap-body vehicle routing problem (SBVRP). However, TTRP (da Cruz and Salles da Cunha, 2023; Villegas et al., 2013) and SBVRP (Todosijević et al., 2017; Toffolo et al., 2018) mainly focus on optimizing truck routes, without addressing collaborative transportation involving multiple vehicles. In contrast, the TSM achieves optimization of truck transport tasks through spatiotemporal collaboration among multiple vehicles. This study primarily focuses on collaborative transportation and stable matching among trucks. Consequently, relevant literature on these two topics is reviewed in this study.

Collaborative transportation is commonly categorized as vertical and horizontal classifications. In vertical collaborative transportation, partners serve distinct segments of the transportation network, whereas in horizontal collaborative transportation, partners cater to the same or overlapping segments of the transport network (Cleophas et al., 2019; Yang et al., 2023). The TSM investigated in this study falls within the domain of horizontal collaborative transportation.

In horizontal collaborative transportation, multiple partners participate in collaborative efforts by pooling resources to achieve effectiveness. Consequently, each partner is inclined to choose collaborators who maximize their individual advantages. Thus, the central platform has the challenge of selecting matching schemes that ensure universal acceptance among the partners (Atef Yekta et al., 2023). This challenge resembles the classical stable matching problem initially proposed in Gale and Shapley (1962). Stable matching problems can be primarily classified into two types: stable marriage problems, applicable when partners possess different attributes, and stable roommate problems, which are relevant when partners belong to the same category. The STMP-TSM examined in this study serves as an extension of the stable roommate problem applied to the trucking context. We subsequently review the literature relevant to stability in the context of horizontal collaborative transportation.

The study of collaborative transportation stability involves two primary perspectives. The first considers the reallocation of costs or utility among partners, whereas the second focuses on the preferences among partners. In the former, the system optimum solution is initially determined, and based on a game theory framework, the stability of the allocation results is then considered. Stability denotes the absence of scenarios in which participants can attain superior outcomes by altering their strategies (Guajardo et al., 2018). From the latter perspective, within the context of the stable matching problem, stability refers to a set of matches in which there are no individuals willing to depart from their current matching and form better ones (Fenoaltea et al., 2021; Danilov, 2023).

Guajardo and Rönnqvist (2016) presented a thorough review of cost allocation methods for collaborative transportation, summarizing relevant research up to 2015. Recently, an adaptive large-neighborhood search algorithm was introduced to address collaboration among carriers dealing with large-scale pickup and delivery problems with time windows (Farvaresh and Shahmansouri, 2020). After determining the near-optimum coalitions, this study allocates outcomes among partners and ensures fairness and stability in cost allocation by employing the Shapley value solution concept. Similarly, numerous studies employed the Shapley value solution concept for cost or utility allocation (Ben Jouida et al., 2021; Bouchery et al., 2022; Giudici et al., 2021; Wang, 2023; Zheng et al., 2019). van Zon et al. (2021) explored the cost allocation problem within a joint network vehicle routing game, proposing a row generation algorithm to ascertain a core allocation or validate the non-existence of such an allocation. Similarly, Chen et al. (2023) employed a row generation algorithm to address cost allocation models in the context of cooperative autonomous truck platooning. Lai et al. (2022) addressed the issue of collaborative less-than-truckload transportation within a shipper consortium overseen by third-party logistics providers. The study formulated the cost allocation problem for shippers as a cooperative game and introduced an efficient computable cost allocation rule derived from a linearized dual model. Sun and Yin (2019) and Sun and Yin (2021) investigated the truck behavioral stability of trucks in decentralized and centralized platooning systems, and proposed a fair cost allocation mechanism based on the marginal contributions of platoon trucks. Although cost allocation among partners can ensure the overall efficiency of the system, these a posteriori gain-sharing mechanisms rarely guarantee that all partners perceive fair treatment (Soriano et al., 2023).

In contrast, when examining collaborative transportation from the perspective of stable matching, all partners choose collaborators based on their preferences, ensuring that all partners perceive fairness, albeit at the expense of a marginal reduction in efficiency (Park et al., 2023). Park et al. (2023) investigated the determinants of successful matching on online freight exchange platforms to facilitate connections between truckers and shippers. This study emphasized that platforms that allow significant price adjustments, longer lead times for loading, and customer revisions of shipment information enhance the likelihood of successful matching. Peng et al. (2016) introduced a model for stable vessel-cargo matching in the dry bulk shipping market, integrating a price game mechanism to simulate the bidding behavior of disadvantaged participants during the matching process for preferred objects. Li et al. (2020) proposed a mixed integer linear programming approach for optimizing a novel multiple-delivery-point service on freight O2O platforms. This study jointly considered matching and pricing strategies for optimum delivery routes to multiple retailers. Both Peng et al. (2016) and Li et al. (2020) considered stable matching between partners from a price game perspective. Stable matching among partners in collaborative transportation from the perspective of preference lists is gradually gaining research attention. Zhang et al. (2020) addressed the autonomous vehicle co-ownership problem among partners based on the preferences of partners and developed an integer linear programming model to solve the problem. Barua et al. (2023) first collected partner information to construct a preference list and then employed a two-stage algorithm based on this list to effectively address the maximum stable truck platooning participation problem.

However, Barua et al. (2023) focused on pairwise matching using an algorithm specifically designed for such matching. Moreover, their problem did not permit trucks to form platoons by adjusting route plans. In contrast, Zhang et al. (2020) addressed a more generalized stable matching problem, accommodating stable matching for any number of partners. Nevertheless, they opted for a greedy heuristic strategy to expedite the model-solving process using trim groups, where the final solution may not guarantee stable matching. The present study addresses these gaps.

1.2. Objectives and contributions

This study investigates the operational decisions of a central platform to achieve stable matching among participating partners. We achieved this objective through the development of a mathematical model and algorithm design, validated by numerical experiments, to assess the performance of the algorithm and feasibility of the mode. The contributions of this study are as follows:

(1) We construct a specific preference list based on the information provided by the partners. This allows partners to select trailer-swapping nodes based on their preferences. In addition, this preference list supports matching among any number of partners.
(2) Based on the characteristics of the problem, we derive relevant properties, enabling the trimming of the preference list without compromising the optimum stable matching. This enhances the efficiency of data preprocessing and algorithmic solutions.
(3) Solving the STMP-TSM using a mathematical model is extremely challenging because of the overwhelmingly large number of feasible groups. To address this issue, we design a heuristic algorithm that consistently identifies a satisfactory stable matching solution within an acceptable time.

The subsequent sections of this paper are structured as follows: Section 2 provides a definition of the STMP-TSM, and Section 3 introduces a mathematical model for addressing the STMP-TSM. Section 4 delineates the steps and procedures for constructing and trimming the preference list. Section 5 elucidates the characteristics and solving process of the iterative preference-list-trim heuristic algorithm. Section 6 introduces an integer programming model designed to resolve the truck matching problem with the TSM (TMP-TSM) based on the preference list. Section 7 presents the numerical experiments and analysis, and Section 8 concludes the study by summarizing the research findings and discussing potential future research directions.

2. Problem description

We use $G = (N, E)$ to represent the transport network of the STMP-TSM, where N denotes the set of nodes and E represents the set of edges. Each edge $(i,j) \in E$, $i,j \in N$, corresponds to a road segment in the network, and each node $i \in N$ represents an intersection in the road network. Let T be the set of trucks involved, assumed to be homogeneous and capable of trailer swapping. Each truck $k \in T$ has a trip characterized by an origin node $o_k \in N$ and a destination node $d_k \in N$, with $o_k \neq d_k$. Assuming all trucks travel at free-flow speed in the network, the time required to traverse an edge $(i,j) \in E$ is denoted by $t_{i,j}$. When $(i,j) \notin E$, $i,j \in N$, $t_{i,j}$ represents the shortest time from i to j. Kindly note that all notations used in this paper can be referred to in the Appendix.

In the traditional mode, each truck transports goods individually. Upon departing from the origin node, the trucks follow the shortest route to their destination, unload upon arrival, and usually return empty to the origin node. At this point, the transportation task is completed. To distinguish the transportation costs for laden and empty trips, we use c_1/c_2 to represent the unit travel-time cost for laden/empty trucks, with the condition $c_1 > c_2$. The benchmark cost for truck k to complete the transportation task individually is f_k. w_k represents the estimated departure time for truck k from its origin node. In the traditional mode, the arrival time of goods at the destination is denoted by g_k, where g_k is the benchmark time for truck k.

If trucks participate in a TSM facilitated by a central platform, they can collaboratively accomplish transportation tasks. For example, the platform delivers a scheme for trucks k and l to execute trailer swapping at node i. Trucks k and l will follow the shortest route to node i sequentially, adhering to their originally scheduled departure times. Upon arrival, a trailer-swapping operation with a duration of θ takes place. After swapping, truck k leads the trailer of truck l to node d_l, unloads it, and returns empty to node o_k. Simultaneously, the trailer of truck k is led by truck l to node d_k, and after unloading, it returns empty to node o_l.

When two trucks are involved in trailer swapping, they must arrive at the swapping node before a trailer-swapping operation can be performed. However, if multiple trucks participate in trailer swapping at a specific node, all trucks do not necessarily need to arrive before the trailer swaps. For example, trucks k, l, and q participate in trailer swapping at a node and the platform delivers a scheme for truck k to lead the trailer of truck l, truck l to lead the trailer of truck q, and truck q to lead the trailer of truck k. In this case, truck k does not need to wait for truck q to arrive at the swapping node; it can depart from i to d_l after receiving the trailer of truck l. Similarly, trucks l and q do not need to wait for trucks k and l to arrive, respectively. However, because of detours and waiting at the swapping node, the arrival time of the goods at the destination is later than the benchmark time, leading to delayed delivery. This is disadvantageous for trucks, particularly when transporting time-sensitive goods. To address this, we introduce a unit-time penalty cost p_k to account for the delayed delivery of the goods of truck k. Additionally, from a practical standpoint, we set a maximum number of trucks α for a trailer-swapping group to avoid excessive waiting and operational load at the swapping node.

We express the utility obtained by a truck in a trailer-swapping scheme to indicate whether the scheme is favorable for the truck. Utility refers to the cost savings that a truck can achieve by participating in trailer swapping. This was calculated by subtracting the actual transportation cost and delayed-delivery penalty cost from the benchmark cost of the truck. The actual transportation cost represents the travel expenses incurred by a truck participating in the TSM for completing transportation tasks and returning to its origin, while the delayed-delivery penalty cost refers to the penalties faced by the trailer initially led by the truck for delayed delivery. We regard the utility of a truck transporting goods individually as zero. Theoretically, the larger the utility a truck obtains in a trailer-swapping scheme, the more it prefers the scheme. All trucks wish to maximize their utility by participating in the most advantageous trailer-swapping scheme. In the studied STMP-TSM, our goal is to determine a trailer-swapping scheme under stable matching conditions, where no truck rejects the swapping scheme proposed by the platform. If the platform proposes a swapping scheme that maximizes the utility of all trucks but overlooks the stability constraints, trucks may prefer to participate in swapping with trucks that increase their utility. For simplicity, we term this solution the system optimum solution, which is also the solution for the TMP-TSM. In some cases, the optimum solution of the system may also be a stable matching. In Fig. 1, trucks participating in the TSM obtain equal utility, and they cannot achieve higher utility by forming smaller alliances. Therefore, the system optimum solution is also a stable matching solution.

Using Fig. 2 as an example, we further illustrate the STMP-TSM. The transportation network in Fig. 2 is consistent with that in Fig. 1. However, unlike Fig. 1, in Fig. 2, the distribution of truck task nodes varies. In the system optimum solution, truck 1 leads the trailer of truck 2; truck 2 leads the trailer of truck 3; and truck 3 leads the trailer of truck 1. Truck 3 derives no utility from participating in this swapping and rejects the proposed swapping scheme from the platform because its laden and empty mileage remain equal in both the traditional mode and the TSM. In this case, the swapping scheme is an unstable matching. In the stable matching scheme, only trucks 1 and 2 participate in swapping at the trailer-swapping node, whereas truck 3 cannot find suitable swapping partners and complete its transportation task individually. Although the swapping scheme may not be the system optimum, the trucks do not reject the proposal. A rigorous definition of stable matching is introduced in Section 3.

3. Problem modeling

We first present the relevant notations and definitions for the STMP-TSM and then provide a mathematical model for the STMP-TSM.

3.1. Notations and definitions

Definition 1. (Swapping Group). Any set of trucks with a length greater than zero containing specific trucks, swapping orders, and swapping nodes is referred to as a swapping group. For example, the group $\{a,b,c\}^i$ represents trucks a, b, and c swapping at node i, where truck a leads the trailer of b, truck b leads the trailer of c, and truck c leads the trailer of a. Additionally, group $\{a\}^{o_i}$ denotes truck a transporting goods individually. Notably, the swapping node and order also serve as identifiers for the group. For example, $\{a,b,c\}^i$, $\{a,b,c\}^j$, and $\{b,a,c\}^i$ are distinct swapping groups.

Here, we use $|\lambda|$ to represent the length of group λ, and μ^λ to represent the total utility of group λ, where $\mu^\lambda = \sum_{k\in\lambda}\mu_k^\lambda$, and μ_k^λ denotes the utility that truck k obtains by participating in group λ. We use Φ to represent the set of all groups, where $\lambda \in \Phi$. Furthermore, Φ_k represents all groups that include truck k, where $\Phi_k \subseteq \Phi$. If there exists a truck $k \in \lambda, \eta$, and $\mu_k^\lambda > \mu_k^\eta$, then we consider that truck k prefers group η less than group λ. We use Φ_k^λ to represent the set of groups that truck k prefers less than group λ.

Definition 2. (*Matching Scheme*). Choose multiple groups from Φ, with each truck participating in only one group. This set of groups is referred to as the matching scheme.

Definition 3. (*Blocking Group*). Assume that some trucks are unwilling to accept the current matching scheme $\mathscr{M}$, and these trucks prefer to spontaneously form a new group λ, and $\lambda \notin \mathscr{M}$. Then, we refer to group λ as a blocking group for the current matching scheme $\mathscr{M}$.

For example, consider groups $\eta = \{a,b\}^i$ and $\psi = \{c,d\}^j$ in $\mathscr{M}$ and another group $\lambda = \{b,c\}^i$ with $\mu_b^\lambda > \mu_b^\eta$ and $\mu_c^\lambda > \mu_c^\psi$. In this case, trucks b and c are unwilling to accept the current matching scheme $\mathscr{M}$ and prefer to spontaneously form a new group λ. Therefore,

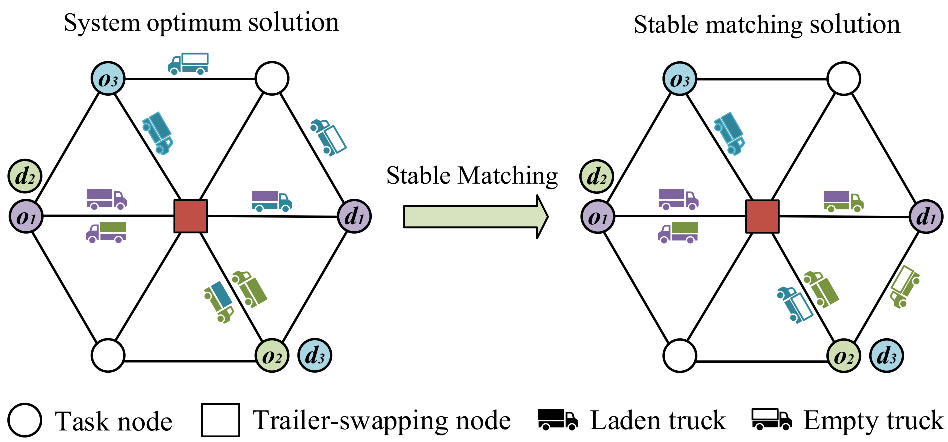

Fig. 2. Illustration of the STMP-TSM.

group λ is a blocking group for the matching scheme $\mathscr{M}$.

Definition 4. (*Stable Matching Scheme*). For a matching scheme $\mathscr{M}^*$, if there are no blocking groups, we refer to $\mathscr{M}^*$ as a stable matching scheme.

In a stable matching solution, not every truck needs to be matched with others. For example, in a solution $\{a,b\}^i$ and $\{c\}^{o_c}$, where trucks a and b refuse to form alliances with truck c, this solution qualifies as a stable matching that meets the definition of no blocking groups.

3.2. Mathematical model

The model includes the following decision variable:
X_λ is a binary variable. $X_\lambda = 1$, if group λ is chosen in the matching scheme; $X_\lambda = 0$, otherwise, $\forall \lambda \in \Phi$.
Using the above notation, the following stable matching model [SMM] is given:

$$\max \sum_{\lambda\in\Phi}\mu^\lambda X_\lambda \tag{1}$$

$$\sum_{\lambda\in\Phi_k} X_\lambda = 1, \ \ \forall k \in T \tag{2}$$

$$X_\lambda + \frac{2}{2|\lambda|-1}\sum_{l\in\lambda}\sum_{\eta\in\Phi_l^\lambda} X_\eta \leq 1, \ \ \forall \lambda \in \Phi \tag{3}$$

$$X_\lambda \in \{0,1\}, \ \ \forall \lambda \in \Phi \tag{4}$$

Objective function (1) aims to maximize the total utility of all the trucks. Constraint (2) ensures that each truck is assigned to a single group in the matching scheme. Constraint (3) ensures the absence of blocking groups for the final matching scheme. Constraint (4) defines the type of decision variable.

Proposition 1. Constraint (3) ensures the absence of blocking groups for the final matching scheme.

Proof. A group can either be chosen ($X_\lambda = 1$) or not chosen ($X_\lambda = 0$).

(1) When $X_\lambda = 1$, $\sum_{l\in\lambda}\sum_{\eta\in\Phi_l^\lambda} X_\eta$ must be 0. This is because for any $\eta \in \Phi_l^\lambda$, where $l \in \lambda$, groups η and λ share at least one common truck l. Constraint (2) ensures that groups containing the same trucks can only have one in the matching scheme. Therefore, constraint (3) is satisfied, ensuring that no blocking groups exist for the final matching scheme.

(2) When $X_\lambda = 0$, without loss of generality, suppose the blocking group of the current matching scheme is λ.

1) If $\sum_{l\in\lambda}\sum_{\eta\in\Phi_l^\lambda} X_\eta = |\lambda|$, meaning that all trucks forming λ are allocated to groups with utilities lower than μ_l^λ, then these trucks would prefer to leave the current group and form a new group λ. In this case, $\frac{2}{2|\lambda|-1}\sum_{l\in\lambda}\sum_{\eta\in\Phi_l^\lambda} X_\eta > 1$ holds, indicating that the current matching scheme has a blocking group λ. Therefore, we consider $\frac{2}{2|\lambda|-1}\sum_{l\in\lambda}\sum_{\eta\in\Phi_l^\lambda} X_\eta \leq 1$ to ensure the absence of a blocking group λ. At this point, constraint (3) is satisfied.

2) If $\sum_{l\in\lambda}\sum_{\eta\in\Phi_l^\lambda} X_\eta < |\lambda|$, indicating that some trucks in λ are assigned to groups with utilities higher than μ_l^λ, then these trucks are unwilling to leave the current group to form the new group λ. In this case, blocking group λ is absent. At this point, constraint (3) is satisfied. □

Despite the conciseness of the model [SMM], it requires significant time to generate all feasible groups (ensuring positive utility for each truck in the group) by treating them as known parameters. As the number of trucks increases, the number of groups in Φ will significantly rise, leading to a sharp increase in the solving time for model [SMM]. For example, when both the number of trucks and nodes is 10, the total number of groups to be enumerated and evaluated is $|N|\sum_{1\leq j\leq|T|} A_{|T|}^j = 10\sum_{1\leq j\leq 10}\frac{10!}{(10-j)!}$. Thus, relying solely on computational power may not effectively address this issue.

In this context, we explore a more efficient approach for generating all possible groups. In the classic stable roommate matching problem, a preference list enables quick assessment of whether any two participants can form a feasible matching group (Liu et al., 2015; Cseh et al., 2019). Therefore, we adopted the approach of constructing a preference list to generate all the feasible groups. However, the traditional preference list only considers whether two participants can match. The STMP-TSM that we are investigating considers a more generalized truck matching, the traditional preference list structure is not applicable. To address this issue, we introduce a chain-data structure within the preference list to facilitate matching among any number of trucks. A detailed explanation of the preference list and chain is provided in Section 4.

4. Truck preference list

The platform initially collects all relevant information about the participating trucks, including their origin nodes, destination nodes, departure times, and penalty costs for delayed delivery per unit time. Subsequently, these data undergo preliminary processing to generate a preliminary preference list (Section 4.1) and further processing to obtain the final truck preference list (Section 4.2).

4.1. Data extraction

In this section, we initially introduce the chain-data structure, followed by the presentation of notations and formulas utilized for calculating the utility of the chain. Subsequently, we present data preprocessing techniques aimed at expediting the generation of the preliminary preference list.

4.1.1. Chain definition

Truck preferences for trailer-swapping groups are determined by their utility. The utility a truck gains from participating in a group is solely related to the saved transportation cost and penalty cost for the delayed delivery of the corresponding trailer. Correspondingly, the utility that a truck gains is associated with which trailer it leads and which truck leads its trailer. For example, in the group $\{\cdots,q,k,l,\cdots\}^i$, the utility of truck k is associated with truck l, whose trailer is led by truck k, and truck q, who leads the trailer of truck k. Information about which trailer a truck leads and which truck leads its trailer at a trailer-swapping node is recorded and organized into a data structure called a chain.

Definition 5. (*Chain*). For any $q,k,l\in T,i\in N$, we refer to $(q,k,l)^i$ as a chain for truck k. $(q,k,l)^i$ indicates that trucks q, k and l swapping at node i, where the trailer of truck k is led by truck q, and truck k leads the trailer of truck l. Specifically, when $q = l$, trucks k and l swap trailers at node i.

For example, each of trucks a, b, c and d have its own chain $(d,a,b)^i$, $(a,b,c)^i$, $(b,c,d)^i$ and $(c,d,a)^i$, respectively. At this point, trucks a, b, c and d swapping at node i, where truck a leads the trailer of b, truck b leads the trailer of c, truck c leads the trailer of d, and truck d leads the trailer of a. Given this information, we can obtain a feasible trailer-swapping group $\{a,b,c,d\}^i$ by combining chains $(d,a,b)^i$, $(a,b,c)^i$, $(b,c,d)^i$ and $(c,d,a)^i$.

4.1.2. Utility of chain

The specific calculations of the benchmark cost f_k and benchmark time g_k are given by equations (5) and (6), respectively:

$$f_k = c_1 t_{o_k,d_k} + c_2 t_{d_k,o_k} \tag{5}$$

$$g_k = w_k + t_{o_k,d_k} \tag{6}$$

The specific calculation formulas for parameters δ^i_{kl}, σ^i_{kl}, and σ^i_{kl} are given by equations (7), (8), and (9), respectively:

$$\delta^i_{kl} = f_k - (c_1 t_{o_k,i} + c_1 t_{i,d_l} + c_2 t_{d_l,o_k}) \tag{7}$$

$$\sigma^i_{kl} = max\{w_k + t_{o_k,i}, w_l + t_{o_l,i}\} + \theta + t_{i,d_l} \tag{8}$$

$$\varepsilon^i_{kl} = p_l(\sigma^i_{kl} - g_l) \tag{9}$$

δ^i_{kl} represents the saved transportation costs for truck k when it leads the trailer of truck l, with the corresponding trailer-swapping node being i, where $k,l\in T,k\neq l,i\in N$. σ^i_{kl} represents the time at which truck k leads the trailer of truck l to reach the destination node d_l, with the corresponding trailer-swapping node being i, where $k,l\in T,k\neq l,i\in N$. ε^i_{kl} represents the penalty cost for truck l when the trailer of truck l is led by truck k, with the corresponding trailer-swapping node being i, where $k,l\in T,k\neq l,i\in N$.

After obtaining the aforementioned information, we can assess whether a given chain is profitable for the trucks. For truck k, the utility obtained by choosing chain $(q,k,l)^i$ are given by $\delta^i_{kl} - \varepsilon^i_{qk}$. When $\delta^i_{kl} - \varepsilon^i_{qk} > 0$ holds, we call $(q,k,l)^i$ an acceptable chain for truck k. Unless otherwise specified, the chains mentioned in the following context refer to acceptable chains for trucks.

4.1.3. Expedite method

To expedite data preprocessing, we further deduce **Propositions 2** and **3**, in which a potential trailer-swapping node for a truck refers to a node where the truck may participate in trailer swapping. When a node is a non-potential trailer-swapping node for a truck, the truck does not participate in trailer-swapping at that node. By excluding non-potential trailer-swapping nodes for trucks, data preprocessing can be accelerated.

Proposition 2. For any truck $k\in T$, its potential trailer-swapping node i must satisfy $f_k \geq c_1 t_{o_k,i} + c_2 t_{i,o_k} + p_k\theta$.

Proof. If node i is a potential trailer-swapping node for truck k, then truck k must have at least one chain and it corresponds to node i. Without loss of generality, assume $(q,k,l)^i$ is a feasible chain for truck k, then $\delta^i_{kl} - \varepsilon^i_{qk} > 0$ must hold.

Because $c_1 > c_2$, it follows that $\delta^i_{kl} = f_k - c_1 t_{o_k,i} - c_1 t_{i,d_l} - c_2 t_{d_l,o_k} \leq f_k - c_1 t_{o_k,i} - c_2 t_{i,d_l} - c_2 t_{d_l,o_k}$. Considering that the shortest distance from i to o_k is t_{i,o_k}, then $t_{i,o_k} \leq t_{i,d_l} + t_{d_l,o_k}$ holds. Thus, $\delta^i_{kl} \leq f_k - c_1 t_{o_k,i} - c_2 t_{i,o_k}$.

Moreover, as $\sigma^i_{qk} = max\{w_q + t_{o_q,i}, w_k + t_{o_k,i}\} + \theta + t_{i,d_k} \geq w_k + t_{o_k,i} + \theta + t_{i,d_k}$, and $t_{o_k,i} + t_{i,d_k} \geq t_{o_k,d_k}$, it follows that $\sigma^i_{qk} \geq w_k + t_{o_k,d_k} + \theta = \theta + g_k$. Therefore, $\varepsilon^i_{qk} = p_k\left(\sigma^i_{qk} - g_k\right) \geq p_k\theta$.

In this case, the upper bound of δ^i_{kl} is $f_k - c_1 t_{o_k,i} - c_2 t_{i,o_k}$, and the lower bound of ε^i_{qk} is $p_k\theta$. Therefore, when

$f_k - c_1 t_{o_k,i} - c_2 t_{i,o_k} - p_k\theta > 0$, $(q,k,l)^i$ may be a chain for truck k, indicating that node i is a potential trailer-swapping node for truck k. □

The lower bound of ε^i_{qk} as $p_k\theta$ indicates that for any truck participating in trailer swapping, the corresponding trailer experiences a delayed delivery. This is because to swap trailers with other trucks, the truck must detour to the trailer-swapping node, possibly wait for the corresponding truck to arrive, conduct the trailer swapping, and only then will its trailer be led to the destination.

Proposition 3. For truck k to participate in trailer swapping at potential trailer-swapping node i, the existence of $\delta^i_{kl} > 0$ is a prerequisite, and node i must also be a potential trailer-swapping node for truck l.

Proof. According to Definition 5, the chain $(q,k,l)^i$ for truck k is required to satisfy $\delta^i_{kl} - \varepsilon^i_{qk} > 0$, indicating the necessity of $\delta^i_{kl} > 0$. Moreover, if node i is not a potential trailer-swapping node for truck l, truck l cannot participate in trailer swapping with any other truck at node i. Therefore, the condition that node i is a potential trailer-swapping node for truck l is imperative. □

Utilizing **Propositions 2** and **3**, we have determined the set of potential trucks at each node. The set of potential trucks at a node refers to the trucks that may participate in trailer swapping at that specific node. Furthermore, truck chains can be generated by traversing a set of potential trucks at each node. Let $Q^i(i \in N)$ denote the set of potential trucks at node i. The time complexities of **Propositions 2** and **3** are $O(|T||N|)$ and $O\left(|T|^2|N|\right)$, respectively. The time complexity of computing all chains for the trucks is $O\left(\sum_{i\in N}|Q^i|^3\right)$. Because $|Q^i| \le |T|(i \in N)$, it follows that $O\left(\sum_{i\in N}|Q^i|^3\right) \le O\left(|T|^3|N|\right)$.

4.2. Chain deletion

Sorting all the chains for each truck in descending order of utility yields the truck preference list. The order of chains in the preference list aligns with their respective preference values. Without loss of generality, chains with equal utility are assigned equal preference values.

As shown in Table 1, five trucks and two swapping nodes are included. For example, if truck a chooses the chain $(b,a,c)^i$ with a preference value of 1, the current trailer-swapping group is $\{b,a,c\}^i$. Then, truck c needs to choose $(a,c,x)^i$ ($x \in T$ and $x \neq a$). At this point, chains $(a,c,b)^i$ and $(a,c,d)^i$ for truck c are valid. The trailer-swapping group is further expanded to $\{b,a,c,b\}^i$ or $\{b,a,c,d\}^i$. Because $\{b,a,c,b\}^i$ has overlapping ends, the trailers of trucks a, c and b are led by b, a and c, respectively, satisfying the precondition for a feasible trailer-swapping group. If truck b has the chain $(c,b,a)^i$, the trailer-swapping group $\{b,a,c\}^i$ is a feasible trailer-swapping group. Further verification is conducted for $\{b,a,c,d\}^i$. Truck d does not have the chain $(c,d,x)^i$ for any truck $x \in T$ and $x \neq a,c$. At this point, the trailer of truck b are not led by other trucks, which does not satisfy the precondition for a feasible trailer-swapping group. This implies that $\{b,a,c,d\}^i$ is not a feasible trailer-swapping group.

Through this exhaustive search approach, we can determine all feasible trailer-swapping groups. The simple deduction above shows that not all truck chains are feasible. We further preprocess and delete the invalid chains for trucks based on **Propositions 4** and **5**. The invalid chain $(q,k,l)^i$ for truck k is acceptable, but it results in zero or negative utility for either truck q or l.

Proposition 4. For any truck $k \in T$, a necessary condition for its chain $(q,k,l)^i$ to be feasible is that trucks q and l each have chains $(x,q,k)^i$ and $(k,l,y)^i$, respectively, where $x,y \in T$.

Proof. Cost saving for truck q when leading the trailer of truck k, denoted as δ^i_{qk}, is achieved when truck q willingly leads the trailer of truck k and hands over its own trailer to truck x (where x can be any truck, $x \in T$). The corresponding penalty cost for delayed delivery incurred by truck q in this scenario is ε^i_{xq}. The condition for feasibility is $\delta^i_{qk} - \varepsilon^i_{xq} > 0$, indicating that truck q gains a positive utility and is willing to lead the trailer of truck k and hand over its own trailer to truck x. According to Definition 1, when $\delta^i_{qk} - \varepsilon^i_{xq} > 0$, $(x,q,k)^i$ is a chain for truck q. Therefore, a necessary condition for truck q to be willing to lead the trailer of truck k is the existence of chain $(x,q,k)^i$.

Similarly, for truck l to willingly let truck k lead its own trailer, chain $(k,l,y)^i$ (where y can be any truck, $y \in T$) must exist.

In the special case in which $q = l$, trucks q and k mutually swap trailers. In this case, for truck q to willingly lead the trailer of truck k, chain $(k,q,k)^i$ must exist, indicating that truck q is willing to lead the trailer of truck k and hand over its own trailer to truck k.

In summary, a feasible chain $(q,k,l)^i$ exists if and only if trucks q and l each have chains $(x,q,k)^i$ and $(k,l,y)^i$, respectively, and when $q = l$, chain $(k,q,k)^i$ exists. □

Table 1
An illustrative example of a preliminary preference list.

Truck	1	2	3	4
a	$(b,a,c)^i$	$(b,a,b)^i$	$(c,a,b)^i$	$(d,a,d)^j$
b	$(a,b,a)^i$	$(c,b,a)^i$	$(c,b,c)^i$	$(d,b,d)^j$
c	$(a,c,b)^i$	$(d,c,d)^j$	$(a,c,d)^i$	$(b,c,b)^i$
d	$(c,d,b)^j$	$(b,d,b)^j$	$(c,d,c)^j$	$(a,d,a)^j$
e	$(a,e,a)^i$	$(b,e,b)^i$	$(c,e,c)^j$	$(d,e,d)^j$

By applying **Proposition 4**, we can efficiently delete truck chains. Table 2 presents the results of deleting the truck preference list from Table 1. Not only can the length of the preference list be further reduced, but truck e is also removed from the list owing to a lack of feasible chains.

Proposition 5. For any truck $k \in T$, when it chooses a chain with a preference value of n, and the corresponding trailer-swapping group includes other trucks that have chosen chains with a preference value of 1, then, for truck k, chains with preference values greater than n are all invalid chains.

Proof. In a trailer-swapping group, where, apart from truck k, all other trucks have chosen chains with a preference value of 1, this implies that these trucks have already identified their best chains and are willing to accept the trailer-swapping scheme. For truck k, having chosen a chain with a preference value of n, as long as truck k is willing to accept the proposed trailer-swapping scheme, no truck rejects the scheme. Compared with chains with preference values greater than n, trucks are more inclined to choose chains with a preference value of n. In this scenario, for truck k, chains with preference values greater than n are invalid and cannot be chosen by truck k. □

By applying **Proposition 5**, we can further delete chains for trucks. Table 3 presents the results of deleting the truck preference list from Table 2. In the trailer-swapping group $\{b, a, c\}^i$, trucks a and c have chosen chains with a preference value of 1, whereas truck b has chosen the chain $(c, b, a)^i$ with a preference value of 2. Therefore, chains with preference values greater than 2, such as $(c, b, c)^i$ and $(d, b, d)^j$, are deleted from truck b. Following the deletion of $(c, b, c)^i$ and $(d, b, d)^j$ from truck b, chains $(b, c, b)^i$ and $(b, d, b)^j$ are also deleted from trucks c and d, respectively.

From Table 3, we obtain the following feasible trailer-swapping groups: $\{b, a, c\}^i$, $\{a, b\}^i$, $\{a, d\}^j$, and $\{c, d\}^j$. When we choose the trailer-swapping group $\{b, a, c\}^i$ as the final trailer-swapping scheme, even though truck b does not choose a chain with a preference value of 1, such as $(a, b, a)^i$, the scheme is still stable. This is because, in this case, no trucks are willing to form a new trailer-swapping group with truck b. Similarly, when we choose trailer-swapping groups $\{a, b\}^i$ and $\{c, d\}^j$ as the final trailer-swapping scheme, the scheme remains stable.

Barua et al. (2023) focused on maximizing the number of trucks participating in the platooning-based collaborative transportation mode; however, our goal is to maximize the total utility of all trucks while considering stable matching constraints. From the two schemes mentioned above, we choose the trailer-swapping scheme that maximizes the total truck utility as the final trailer-swapping scheme.

Notably, stable matching may not always exist. Table 4 presents an improvement based on Table 3, and the example in Table 4 demonstrates the absence of stable matching. Assuming that the proposed final trailer-swapping scheme is $\{b, a, c\}^i$, in this case, truck b would reject this scheme and spontaneously participate in trailer swapping with truck d at node j. This is because, by swapping with truck d, truck b achieves a higher utility, and truck d, functioning as an individual transporter, is willing to swap with truck b. Similarly, other trailer-swapping schemes remain unstable.

In cases with no stable matching, a feasible method entails the platform subsidizing a portion of the utility to the trucks, thereby ensuring stability in the proposed trailer-swapping scheme. Further investigation of this aspect is reserved for future research. Our focus is on the effective identification of stable matching that maximizes utility and meets the requirements of both the platform and partners.

By utilizing the final preference list, we efficiently extracted all feasible swapping groups, thereby reducing the preprocessing time compared to the enumerated approach. However, because of the substantial number of feasible groups, relying on the [SMM] model may not solve the STMP-TSM within a reasonable time. Therefore, in Section 5, we design a heuristic algorithm based on the preference list to solve the STMP-TSM effectively, ensuring the attainment of a stable solution. As a benchmark, Section 6 presents an integer linear programming model for solving the TMP-TSM.

5. Iterative preference-list-trim heuristic algorithm

This section introduces the iterative preference-list-trim heuristic (IPTH) algorithm for addressing the STMP-TSM. In Section 5.1, the solution strategy of the IPTH is outlined, and Section 5.2 delves into the solving procedure of the IPTH.

5.1. Solving strategy

Herein, we draw inspiration from the solution approach of the column generation algorithm, aiming to expedite the solution

Table 2
An illustrative example of performing chain deletion with Proposition 4.

Truck	1	2	3	4
a	$(b, a, c)^i$	$(b, a, b)^i$	$(d, a, d)^j$	
b	$(a, b, a)^i$	$(c, b, a)^i$	$(c, b, c)^i$	$(d, b, d)^j$
c	$(a, c, b)^i$	$(d, c, d)^j$	$(b, c, b)^i$	
d	$(c, d, c)^j$	$(b, d, b)^j$	$(a, d, a)^j$	

Table 3
An illustrative example of performing chain deletion with Proposition 5.

Truck	1	2	3
a	$(b,a,c)^i$	$(b,a,b)^i$	$(d,a,d)^j$
b	$(a,b,a)^i$	$(c,b,a)^i$	
c	$(a,c,b)^i$	$(d,c,d)^j$	
d	$(c,d,c)^j$	$(a,d,a)^j$	

Table 4
An illustrative example of the absence of stable matching.

Truck	1	2	3
a	$(b,a,c)^i$	$(d,a,d)^j$	
b	$(d,b,d)^j$	$(c,b,a)^i$	
c	$(a,c,b)^i$	$(d,c,d)^j$	
d	$(c,d,c)^j$	$(a,d,a)^j$	$(b,d,b)^j$

process of the STMP-TSM by accessing only a small portion of the chains in the preference list. To validate the feasibility of this approach, we derive **Propositions 6**, **7**, and **8**.

For simplicity, the preference value of a truck in the matching scheme refers to that of the corresponding chain in the truck preference list. For any given matching scheme $\mathscr{M}$, using the preference values of each truck in $\mathscr{M}$ as a benchmark, chains in the preference list with preference values exceeding this benchmark are considered feeble chains for trucks in $\mathscr{M}$. Additionally, when we refer to a matching scheme $\mathscr{M}$ as stable on a preference list, we mean that the groups formed by the chains in the preference list are not blocking groups for $\mathscr{M}$.

Proposition 6. For a matching scheme $\mathscr{M}$, all groups formed by feeble chains are not blocking groups for $\mathscr{M}$.

Proof. For any truck, choosing a feeble chain implies obtaining a lower utility than that in matching scheme $\mathscr{M}$. Therefore, the truck is unwilling to reject $\mathscr{M}$ and choose to participate in a group formed by feeble chains. Hence, the groups formed by the feeble chains are not blocking groups for $\mathscr{M}$. □

Proposition 7. When the preference value of a truck in matching scheme $\mathscr{M}$ is greater than that in $\mathscr{M}'$, the feeble chains for that truck in $\mathscr{M}$ are also feeble chains in $\mathscr{M}'$.

Proof. Lower preference values indicate higher utility for trucks. When the preference value of a truck in matching scheme $\mathscr{M}$ is greater than in $\mathscr{M}'$, this implies that the set of feeble chains for that truck in $\mathscr{M}$ is a proper subset of the set in $\mathscr{M}'$. Therefore, the feeble chains for that truck in $\mathscr{M}$ are also feeble chains in $\mathscr{M}'$. □

Proposition 8. In the stable matching scheme $\mathscr{M}'$ obtained based on $\mathscr{H}^{\mathscr{M}}$ (the preference list obtained after trimming these feeble chains from the current preference list $\mathscr{H}$ based on the given matching scheme $\mathscr{M}$), the matching among the trucks participating in the TSM is stable on $\mathscr{H}$.

Proof. Assume that the matching scheme among the trucks participating in the TSM is denoted as $\mathscr{M}''$ ($\mathscr{M}'' \subseteq \mathscr{M}'$). Because this is the stable matching scheme $\mathscr{M}'$ based on $\mathscr{H}^{\mathscr{M}}$, the preference values of trucks participating in the TSM in $\mathscr{M}''$ are lower than those in $\mathscr{M}$. According to Proposition 7, the feeble chains for the trucks in $\mathscr{M}$ are also feeble chains in $\mathscr{M}''$. According to Proposition 6, groups formed by these feeble chains are not blocking groups for $\mathscr{M}''$. Because $\mathscr{M}'$ is stable on $\mathscr{H}^{\mathscr{M}}$, no blocking groups exist for $\mathscr{M}''$ in $\mathscr{H}^{\mathscr{M}}$. Therefore, the matching scheme $\mathscr{M}''$ does not have blocking groups in $\mathscr{H}$, thereby indicating a stable matching. □

Concerns may arise regarding stable matching for trucks in $\mathscr{M}'$, excluding those in $\mathscr{M}''$, that deliver goods individually. In reality, the partners of these trucks in $\mathscr{M}$ may have chosen other partners in $\mathscr{M}'$. Upon removing the feeble chains for these trucks in $\mathscr{M}$, they deliver the goods individually because they have no feeble chains. Nevertheless, these trucks can choose feasible chains to participate in the TSM. Therefore, refilling the feeble chains for these trucks into the preference list and achieving stable matching based on the preference list of these trucks is essential.

Table 5
An illustrative example for Proposition 8.

Truck	1	2
a	$(b,a,c)^i$	$(c,a,c)^i$
b	$(d,b,d)^j$	$(c,b,a)^i$
c	$(a,c,b)^i$	$(a,c,a)^i$
d	$(b,d,b)^j$	

Table 5 presents an example illustrating this scenario. When we consider $\mathscr{M} = \{b, a, c\}^i$, the feeble chains for trucks a and c, corresponding to $(c, a, c)^i$ and $(a, c, a)^i$, must be removed from $\mathscr{H}$ to obtain $\mathscr{H}^{\mathscr{M}}$. The stable matching scheme $\mathscr{M}'$ based on $\mathscr{H}^{\mathscr{M}}$ includes transporting goods individually for trucks a and c, as well as participating in the TSM for trucks b and d. In practice, trucks a and c choose feeble chains to obtain benefits. We can observe that, in this case, $\mathscr{M}'' = \{b, d\}^j$ is a stable matching scheme for trucks b and d, and the trailer-swapping scheme $\{a, c\}^i$ for trucks a and c is also a stable matching scheme.

Based on **Propositions 6**, **7**, and **8**, the following solution strategy can be outlined. Initially, a matching scheme $\mathscr{M}$ is identified, and the preference list $\mathscr{H}$ is trimmed based on $\mathscr{M}$ to obtain $\mathscr{H}^{\mathscr{M}}$. Subsequently, the stable matching scheme $\mathscr{M}'$ is determined based on $\mathscr{H}^{\mathscr{M}}$. In $\mathscr{M}'$, the matching among the trucks participating in the TSM is stable. Stable matching is continually sought among trucks transporting goods individually, resulting in the final stable matching scheme.

Determining a matching scheme $\mathscr{M}$ that is close to the optimum stable matching is an important task. This is because when the preference values of trucks in $\mathscr{M}$ are small, a significant portion of the feeble chains can be trimmed, thereby enhancing the solution efficiency. Constraint (2) in model [SMM] ensures that each truck is in only one group in the final matching scheme. Moreover, we can utilize this feature to identify matching scheme $\mathscr{M}$. Additionally, we can gradually increase the maximum group length to reduce the number of groups accessed during each iteration, progressively approaching a stable solution. For simplicity, we define $\mathscr{M}^*_n$, representing the optimum stable matching solution obtained when the maximum group length is set to n.

First, we solve for $\mathscr{M}^*_2$ based on preference list $\mathscr{H}$. Then, we utilize the preference values of trucks in $\mathscr{M}^*_2$ to trim the chains in preference list $\mathscr{H}$, and solve for $\mathscr{M}^*_3$ based on the trimmed preference list $\mathscr{H}^{\mathscr{M}^*_2}$. Subsequently, we employ the preference values of the trucks in $\mathscr{M}^*_3$ to trim the chains in preference list $\mathscr{H}$, and solve for $\mathscr{M}^*_4$ based on the trimmed preference list $\mathscr{H}^{\mathscr{M}^*_3}$. This process is repeated until a predetermined stopping condition is reached. Because each iteration is based on $\mathscr{H}$, the trucks transporting goods individually in $\mathscr{M}^*_3$ are stable matches within $\mathscr{M}^*_4$, and so forth. For trucks transporting goods individually in $\mathscr{M}^*_\alpha$, an additional stable matching verification is required. This iterative solving approach enables the discovery of a satisfactory and stable matching scheme by accessing only a subset of groups.

5.2. Solving procedure

Algorithm 1 outlines the procedure for addressing STMP-TSM. It takes as input the original preference list $\mathscr{H}$ and maximum group length α. The corresponding output is the final stable matching scheme. The algorithm is initiated by setting the current maximum group length to 2 (line 1). In the first step of each iteration, the preference values of the trucks in the stable match $\mathscr{M}^*_{i-1}$ are utilized to trim the chains in $\mathscr{H}$, resulting in a modified preference list $\mathscr{H}^{\mathscr{M}^*_{i-1}}$ (line 3). In addition, $\mathscr{M}^*_1$ is established by assuming that no trucks participate in swapping, implying an infinite preference value for all trucks. Subsequently, feasible swapping groups are extracted from the preference list $\mathscr{H}^{\mathscr{M}^*_{i-1}}$, ensuring that the group length does not exceed the current maximum group length (line 4). The algorithm then utilizes the [SMM] model to solve for the current optimum stable matching based on the obtained group set Φ (line 5). Subsequently, the current maximum group length is incrementally increased by 1. The algorithm iterates until the current maximum group length surpasses α, at which point it terminates (line 6). Trucks participating in the TSM in $\mathscr{M}^*_\alpha$ are removed from $\mathscr{H}$ to obtain a new preference list. Subsequently, all feasible groups are extracted, and a stable matching solution is determined using model [SMM] (line 8). Finally, we merge the stable matching schemes from the two parts to obtain a final stable matching scheme.

Algorithm 1 IPTH algorithm solving procedure

Input: The truck preference list $\mathscr{H}$, maximum group length α
Output: Final stable matching scheme
1: Set $i \leftarrow 2$;
2: **while** $i \leq \alpha$ **do**
3: Trim chains in $\mathscr{H}$ according to $\mathscr{M}^*_{i-1}$ to obtain $\mathscr{H}^{\mathscr{M}^*_{i-1}}$;
4: Extract all possible groups Φ in $\mathscr{H}^{\mathscr{M}^*_{i-1}}$ with current maximum group length i;
5: Utilize model [SMM] to solve the STMP-TSM with Φ and obtain $\mathscr{M}^*_i$;
6: Set $i \leftarrow i + 1$;
7: end while
8: Perform stable matching for trucks transporting goods individually in $\mathscr{M}^*_\alpha$;
9: Return final stable matching scheme

Using the IPTH algorithm does not guarantee the discovery of the optimum stable matching, but ensures stable matching. This is because during the solving process, we focus solely on stable matching among trucks, neglecting the comparison of optimality between stable solutions. However, obtaining a stable solution is difficult. The further development of algorithms for solving the optimum stable matching problem is left for future research.

6. System optimum model

As a benchmark, we present an integer linear programming model to solve the TMP-TSM. The model is constructed based on the preference list. Below, we provide the relevant notation for the model.

Sets:

R_k^i represents the set of chains for truck k at node i. For $r = (r_1, k, r_2)^i \in R_k^i$, r_1/r_2 denotes the first/third element in the chain. $(k \in T, i \in N)$

F_{kl}^i represents the subset of R_k^i and $F_{kl}^i = \{r | r \in R_k^i, r_1 = l\}$. $(k, l \in T, k \neq l, i \in N)$

S_{kl}^i represents the subset of R_k^i and $S_{kl}^i = \{r | r \in R_k^i, r_2 = l\}$. $(k, l \in T, k \neq l, i \in N)$

Parameters:

φ_{kr} represents the utility obtained by truck k when choosing chain r. $\left(k \in T, r \in R_k^i, i \in N\right)$

Decision variable:

Y_{kr} 1, if truck k chooses chain r; 0, otherwise. $\left(k \in T, r \in R_k^i, i \in N\right)$

Using the above notation, the following system optimum model [SOM] is obtained:

$$\max \sum_{k \in T} \sum_{i \in N} \sum_{r \in R_k^i} \varphi_{kr} Y_{kr} \tag{10}$$

$$\sum_{i \in N} \sum_{r \in R_k^i} Y_{kr} \leq 1, \quad \forall k \in T \tag{11}$$

$$2Y_{kr} \leq \sum_{m \in S_{r_1 k}^i} Y_{r_1 m} + \sum_{m \in F_{r_2 k}^i} Y_{r_2 m}, \quad \forall k \in T, r \in R_k^i, i \in N \tag{12}$$

$$Y_{kr} \in \{0, 1\}, \quad \forall k \in T, r \in R_k^i, i \in N \tag{13}$$

The objective function (10) is similar to (1) and aims to maximize the total utility of all trucks. Constraint (11) ensures that at most, one chain is chosen for each truck, similar to constraint (2). Constraint (12) ensures the feasibility of the groups extracted from the preference list. When $Y_{kr} = 1$, the first element r_1 in chain r must lead the trailer of truck k at node i, i.e., $\sum_{m \in S_{r_1 k}^i} Y_{r_1 m} = 1$. Similarly, the second element r_2 in chain r must hand over its trailer to be led by truck k at node i, i.e., $\sum_{m \in F_{r_2 k}^i} Y_{r_2 m} = 1$. Constraint (13) defines the type of decision variable.

By removing constraint (3) from model [SMM], the resulting model can also solve the TMP-TSM. However, model [SMM] requires extracting all groups from the preference list and further processing them as input parameters before starting the solution process. The [SOM] model we constructed omits the step of extracting swapping groups and can directly solve the TMP-TSM based on the preference list.

7. Computational experiments

In this section, the performance of the proposed IPTH algorithm is evaluated through numerical experiments. Furthermore, we analyze the results of the TSM and perform a sensitivity analysis of the relevant parameters. All program codes are developed in Python 3.11, and both models are solved using the Gurobi optimization solver. The computations are executed on a PC equipped with a 12th Gen Intel(R) Core(TM) i7-12800HX 2.00 GHz processor and 64.0 GB RAM.

Table 6
The computational result for 10 instances.

$\|T\|$	IPTH				[SMM]				[SOM]		Gap^1
	$BestV^1$	$SolT^1$	$PreT^1$	$GroN^1$	$BestV^2$	$SolT^2$	$PreT^2$	$GroN^2$	$BestV^3$	$SolT^3$	
10	15,779	<1	<1	31	15,779	<1	<1	38	15,779	<1	0.0
20	25,278	<1	<1	331	25,278	<1	<1	684	26,592	<1	4.9
30	37,909	<1	<1	877	37,909	3	1	3384	41,688	6	9.0
40	62,723	2	3	3523	62,777	417	77	21,840	67,765	33	7.4
50	87,187	9	9	6284	–	7200	1400	126,823	101,947	273	14.5
60	108,891	48	29	17,981	–	7200	4441	193,231	126,869	408	14.2
70	127,431	25	41	21,085	–	7200	7200	–	136,436	434	6.6
80	135,581	76	107	30,270	–	7200	7200	–	147,625	7200	8.2
90	167,730	330	431	57,432	–	7200	7200	–	189,812	1958	11.6
100	185,909	608	453	96,217	–	7200	7200	–	212,390	7200	12.5
Average											8.9

$Gap^1 = \left(BestV^3 - BestV^1\right)/BestV^3 (\%)$.

7.1. Experiment setup

All the experiments are conducted based on a simplified highway network in China comprising 33 nodes representing major cities. The truck travel distances between two nodes are calculated using Google Maps. The uniform truck travel speed is set to 80 km/h. The departure times of the trucks are generated from a uniform distribution U[6:00,18:00]. In line with actual situations, the unit travel cost for a laden truck c_1 is set at 200 ¥/h, whereas that for empty trucks c_2 is set at 180 ¥/h. Considering the diverse types of goods transported by trucks, the penalty cost for delayed delivery is uniformly distributed within U[100,200] ¥/h. The duration required for trucks to perform a trailer swap operation θ is set at 12 min. Additionally, the maximum group length α is set at 6. Using these settings, we generate 10 instances, with the total number of trucks gradually increasing from 10 to 100, as listed in Table 6.

7.2. Performance of IPTH

We assess the performance of the IPTH algorithm by analyzing its effectiveness, convergence, and stability. To evaluate its effectiveness, we compare the results of the IPTH algorithm with those of models [SMM] and [SOM]. Furthermore, we examine the convergence process of the IPTH algorithm in detail. Finally, to verify the stability, we generate instances of the same scale 10 times consecutively, analyzing the difference in results between the IPTH algorithm and model [SMM].

7.2.1. Effectiveness of IPTH

We initially generate 10 instances, with the total number of trucks ($|T|$) gradually increasing from 10 to 100, where $|T|$ serves as the identifier for each instance. For the results obtained from the IPTH algorithm, we record the total truck utility, solution time, preprocessing time (including the time to extract all swapping groups from the preference list), and total number of groups in Φ, denoted as $BestV^1$, $SolT^1$, $PreT^1$, and $GroN^1$, respectively. Similarly, we record the results of the model [SMM], including the total truck utility ($BestV^2$), solution time ($SolT^2$), preprocessing time ($PreT^2$), and total number of groups ($GroN^2$). Additionally, we record the results of the model [SOM] encompassing the system optimum solution ($BestV^3$) and solution time ($SolT^3$, including the time to generate the preference list). The results are presented in Table 6, where Gap^1 denotes the gap between $BestV^1$ and $BestV^3$.

The results indicate that the IPTH algorithm consistently identifies satisfactorily stable solutions within a reasonable time. The solution time, preprocessing time, and number of groups increase with the scale of the instances. For the largest instance, the solution and preprocessing times are 608 and 503 s, respectively. However, using model [SMM], only the first four instances can be solved using the [SMM]. Instances with $|T| = 50$ and $|T| = 60$ can only complete data preprocessing within 2 h, failing to obtain truck-swapping schemes in the subsequent 2 h. For the last four instances, model [SMM] cannot provide useful information within a reasonable time. This is because, with an increase in the total number of trucks, the number of feasible groups increases considerably. Preprocessing not only requires extracting the group set Φ but also involves further processing to obtain sets Φ_k and Φ_k^λ ($k \in T, \lambda \in \Phi$), where the associated computational time and storage space are substantial. For the instance $|T| = 40$, the model [SMM] involves feasible groups reaching 193,231, with a preprocessing time of 4441 s. In contrast, by trimming the preference list, the IPTH algorithm accesses only a portion of the groups. For the instance $|T| = 100$, the IPTH algorithm involves only 17,981 groups, with a preprocessing time of only 29 s. Further comparison of the total truck utility reveals that for the first three instances, the IPTH algorithm identifies the optimum stable matching solutions. For the instance $|T| = 40$, although the IPTH algorithm does not find the optimum stable matching solution, it obtains a solution close to the optimum. Therefore, the IPTH algorithm effectively addresses the STMP-TSM.

The input data for the model [SOM], which has undergone preprocessing, can converge to the optimum solution within a reasonable time. Specifically, only instances $|T| = 80$ and $|T| = 100$ obtained a near-optimum solution within 2 h, whereas the remaining instances obtain the optimum solution. Further comparison between the stable and system optimum solutions reveals these differences. Observing the values of Gap^1 shows that the optimum stable matching solution for instance $|T| = 10$ coincides with the system optimum solution. The values of Gap^1 for all instances are below 15 %, with an average value of 8.9 %. Considering that a stable matching results in a marginal reduction in the utility of the central platform, all trucks accept proposals from the platform, thereby allowing stable utility dividends to the platform.

We further illustrate the detailed differences between the stable matching solution obtained using the IPTH algorithm and the optimum stable matching solution obtained using the model [SMM]. We consider instance $|T| = 40$ as an example, and the results are listed in Table 7. Table 7 presents the trailer-swapping schemes obtained using both methods, and the preference values of each truck in the swapping scheme. Except for trucks 12, 13, 21, and 24, the trailer-swapping schemes for the remaining trucks are identical under both solution methods. In the IPTH algorithm, trucks 21 and 24 swap trailers at node 22, and trucks 12 and 13 also swap trailers at node 22. However, in model [SMM], four trucks form a swapping group and swap trailers at node 22. Trucks 21, 24, and 13 have higher

Table 7
The detailed results for instance $|T| = 40$.

$\|T\|$	Trailer-swapping scheme	Preference values
IPTH	$\{18,33\}^6$; $\{9,8,11\}^8$; $\{7,28\}^{14}$; $\{1,40\}^{15}$; $\{4,23\}^{16}$; $\{6,27\}^{16}$; $\{25,29\}^{17}$; $\{30,17,37\}^{18}$; $\{22,32\}^{19}$; $\{21,24\}^{\mathbf{22}}$; $\{12,13\}^{\mathbf{22}}$	(1,42); (16,41,6); (1,31); (1,21); (20,4); (46,30); (28,12); (11,15,25); (45,40); **(69,3)**; **(3,72)**
[SMM]	$\{18,33\}^6$; $\{9,8,11\}^8$; $\{7,28\}^{14}$; $\{1,40\}^{15}$; $\{4,23\}^{16}$; $\{6,27\}^{16}$; $\{25,29\}^{17}$; $\{30,17,37\}^{18}$; $\{22,32\}^{19}$; $\{21,12,13,24\}^{\mathbf{22}}$	(1,42); (16,41,6); (1,31); (1,21); (20,4); (46,30); (28,12); (11,15,25); (45,40); **(59,9,52,2)**

preference values in group $\{21,12,13,24\}^{22}$, whereas truck 12 has a higher preference value in group $\{12,13\}^{22}$. Therefore, group $\{21,12,13,24\}^{22}$ is not a blocking group for the trailer-swapping scheme output by the IPTH algorithm, indicating that the trailer-swapping scheme obtained by the IPTH algorithm is stable.

7.2.2. Convergence of IPTH

The IPTH algorithm explores stable solutions by progressively increasing the maximum group length. Here, we present detailed data as the maximum group length varies from 2 to 6, as shown in Table 8. The table records the total truck utility for each iteration. The total truck utility gradually increases and tends to stabilize as the group length increases for the first eight instances. However, for the two largest instances, the total truck utility does not consistently increase with the maximum group length. Instead, they fluctuate. This is because, with a larger maximum group length, more trucks can form larger groups. When trucks with lower utility choose chains with higher preference values, the total utility may decrease. In fact, considering only a group length of 2, stable matching cannot be achieved in most cases. Except for instance $|T| = 30$, all instances involve trucks forming larger swapping groups. Therefore, when considering collaborative transportation modes, a generalized approach to stable truck matching is more applicable.

7.2.3. Stability of IPTH

The IPTH algorithm, which is characterized by its deterministic nature, consistently yields identical outcomes across successive runs. To assess its stability, we consecutively generated instances of the same scale ten times; the results are presented in Table 9. For instances $|T| \geq 60$, the model [SMM] fails to find solutions; therefore, we did not include relevant data in Table 9. Gap^2 denotes the gap between $BestV^1$ and $BestV^2$, where $BestV^1$ and $BestV^2$ denote the total truck utility obtained by the IPTH algorithm and model [SMM], respectively. CPU^1 denotes the average computation time of the IPTH algorithm for solving 10 instances of the same scale, including preprocessing and solving time. CPU^2 denotes the average computation time of model [SMM] for successfully solved instances.

For the instance $|T| = 10$, the IPTH algorithm consistently identifies the optimum stable matching solution. For the instance $|T| = 20$, an approximately stable matching solution is identified in three of the ten runs. For instances $|T| = 30$ and $|T| = 40$, the IPTH algorithm does not consistently identify the optimum stable matching solution. However, the average Gap^2 is -1.98 % and -1.54 %, respectively, indicating a very small deviation. For instances $|T| = 50$, model [SMM] only able to successfully solve 4 instances, with CPU^2 significantly exceeding CPU^1, and the average Gap^2 being -3.85 %, still within a relatively small deviation. This underscores the effectiveness and stability of the IPTH algorithm.

7.3. Results analysis

This section further analyzes the features of the TSM, including the impact of optimization objectives on mode operations, as well as specific scenarios related to the delayed-delivery and empty rates in the TSM.

7.3.1. Optimization objectives

Our previous optimization objective was to maximize the total truck utility (MTU). However, maximizing the number (MTN) of trucks participating in the TSM is also an important direction for in-depth exploration. In our construct solving approach, switching between these two optimization objectives is straightforward. Table 10 presents the results of the IPTH algorithm, recording the total truck utility and number of participating trucks for both objectives as comparative results. $BestV^1$ and Num^1 represent the total truck utility and number of participating trucks under MTU, respectively. $BestV^4$ and Num^2 represent the total truck utility and number of participating trucks under MTN, respectively. Gap^3 represents the gap between $BestV^1$ and $BestV^4$, and Gap^4 represents the gap between Num^1 and Num^2.

Except for instance $|T| = 10$, in all other instances, the total truck utility under MTU is always greater than that under MTN. Similarly, except for the first three instances, the number of participating trucks under MTN is always greater than that under MTU. Clearly, different optimization objectives produce differences in the obtained results. The average Gap^3 is 10.19 %, indicating a 10.19 % reduction in the revenue of the central platform under MTN. This reduction leads to an increase in the number of participating trucks by 4.40 %. Therefore, the specific choice of the optimization objective may need to be based on the positioning of the central platform. In practice, during the initial promotion of the central platform, adopting the MTN objective may attract more trucks to participate in the TSM, thereby enhancing the visibility of the platform at the expense of sacrificing revenue.

Table 8
Convergence of IPTH.

α	$\|T\|$									
	10	20	30	40	50	60	70	80	90	100
2	15,174	24,326	37,909	59,246	74,446	105,264	119,934	130,331	149,076	169,508
3	15,779	24,326	37,909	62,723	78,398	107,921	122,655	133,314	166,249	184,125
4	15,779	25,278	37,909	62,723	84,289	108,891	127,431	133,885	168,766	187,267
5	15,779	25,278	37,909	62,723	87,187	108,891	127,431	134,621	169,740	182,605
6	15,779	25,278	37,909	62,723	87,187	108,891	127,431	135,581	167,730	185,909

Table 9
Stability of IPTH.

$\|T\|$	*Gap*^{2}										Average	*CPU*^{1}	*CPU*^{2}
10	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	<1	<1
20	0.00	0.00	0.00	0.00	0.00	−2.04	−0.64	0.00	0.00	−1.08	−0.39	<1	<1
30	0.00	−3.30	−1.05	−0.19	−5.25	0.00	−3.00	0.00	−1.12	−5.93	−1.98	2	58
40	−0.09	−0.89	−3.22	−0.85	−0.61	−3.71	−2.75	−1.12	−1.85	−0.28	−1.54	2	242
50	–	–	–	–	−0.60	–	–	−7.44	−5.98	−1.37	−3.85	22	1546

$Gap^2 = (BestV^1 - BestV^2)/BestV^2(\%)$.

Table 10
Comparison between MTU and MTN.

$\|T\|$	MTU		MTN		*Gap*^{3}	*Gap*^{4}
	BestV^1	*Num*^{1}	*BestV*^{4}	*Num*^{2}		
10	15,779	5	15,779	5	0.00	0.00
20	25,278	8	23,745	8	6.46	0.00
30	37,909	14	34,881	14	8.68	0.00
40	62,723	24	55,670	26	12.67	−8.33
50	87,187	33	77,132	37	13.04	−12.12
60	108,891	36	89,275	37	21.97	−2.78
70	127,431	50	116,127	53	9.73	−6.00
80	135,581	50	132,003	53	2.71	−6.00
90	167,730	64	151,087	65	11.02	−1.56
100	185,909	69	160,745	74	15.65	−7.25
Average					10.19	−4.40

7.3.2. Delayed-delivery rate

In the traditional mode, trucks transport goods to their destination via the shortest route. However, trucks participating in the TSM may need to detour to a trailer-swapping node and await the arrival of other trucks for the trailer-swapping operation. Consequently, trucks participating in the TSM results in delayed delivery. Fig. 3 illustrates the delayed-delivery rates of the participating trucks. The delayed-delivery rate is the ratio of the time at which goods are delayed to the transport time from the origin node to the destination node for trucks. Presented in the form of a violin plot, Fig. 3 also shows the distribution of the delayed-delivery rate from the 25th to 75th percentile and the mean delayed-delivery rate for all trucks in each instance. Some trucks are willing to sacrifice significant timeliness to reduce transportation costs. For example, trucks in the instance $|T| = 70$ have a delayed-delivery rate of 200 %. However, the majority of trucks choose to participate in the TSM with a smaller sacrifice in timeliness, and the data distribution from the 25th to 75th percentile is concentrated at a lower level. Further observation of the connecting lines of the averages reveals that the average delayed-delivery rate does not gradually increase with the instance scale and that the connecting lines appear almost horizontal. Therefore, the average delayed-delivery rate for all trucks participating in the TSM is approximately 20 %.

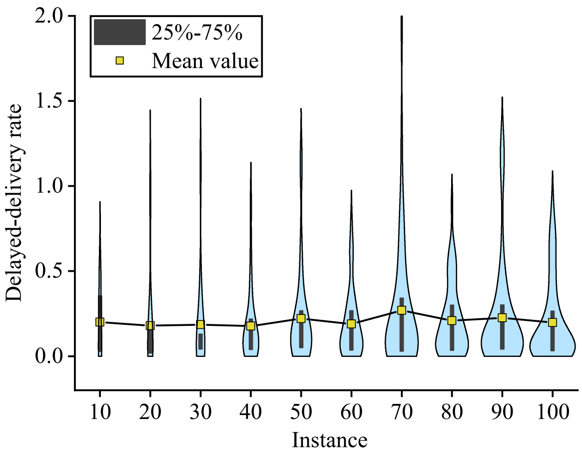

Fig. 3. Delayed-delivery rate of trucks.

7.3.3. Empty rate

The participation of trucks in the TSM can significantly reduce the empty rate. The empty rate of a truck is the ratio of the empty travel distance to the total travel distance. Therefore, the empty rate for trucks transporting goods individually is 50 %. Fig. 4 depicts the distribution of the empty rate under the TSM. Fig. 4, presented in the form of a violin plot akin to Fig. 3, provides a detailed distribution of the empty rate from the 25th to 75th percentile and the mean empty rate for all trucks in each instance.

Observations indicate that in each instance, some trucks transport goods individually, corresponding to an empty rate of 50 %. Additionally, some trucks exhibit a 0 % empty rate under the TSM, where the destination of the trailers led by these trucks coincides with their origin node. Excluding these scenarios, the remaining trucks that participated in the TSM exhibit an empty rate distributed between 0 % and 50 %. Further observation of the connecting lines of the average empty rate indicates an overall decreasing trend with an increase in the instance scale. This trend indicates that an increased number of trucks participating in the TSM correlates with a more favorable outcome in improving the empty rate. In summary, the TSM demonstrates the ability to reduce the empty rate by approximately 25 %. When only trucks participating in the TSM are considered (excluding trucks transporting goods individually), the advantages of reducing the empty rate are even more pronounced.

7.4. Sensitivity analysis

Numerical experiments were conducted under fixed parameter settings. This section discusses the impact of various critical parameters on the TSM performance. Three instances, $|T| = 20$, $|T| = 50$, and $|T| = 80$, are selected for the experiments. In each parameter analysis, the remaining parameters are maintained as a constant, and only the parameter under scrutiny is modified.

7.4.1. Duration of the swapping operation

In our previous setup, we designated the duration of the swapping operation θ as 12 min. Now, we vary θ from 0 to 60 min to analyze its impact on the delayed-delivery rate. Fig. 5 illustrates the impact of θ on the delayed-delivery rate. The delayed-delivery rate gradually increases with increasing values of θ, exhibiting a linear growth trend. This is because changes in θ do not affect the preference values for various chains in the preference list, but only impact the utility associated with each chain. Consequently, an increase in θ results in a corresponding increase in the delayed-delivery times for all trucks, leading to a gradual increase in the delayed-delivery rate. In practical applications of the TSM, judiciously reducing the duration of the swapping operation can effectively decrease the delayed delivery rate while simultaneously enhancing the total truck utility and platform revenue.

7.4.2. The ratio of c_2 toc_1

In our previous numerical experiments, we fixed the unit travel-time cost for an empty truck c_2 at 180 and that for a laden truck c_1 at 200, resulting in a ratio c_2/c_1 of 0.9. In this analysis, we systematically vary c_2/c_1 from 0.5 to 1.0 to assess its impact on the applicability of the TSM.

Fig. 6 illustrates the effects of c_2/c_1 on the total truck utility. Given that the total truck utility for the three instances is not at the same level, a numerical transformation is applied for a consistent representation in the figure. Using the total truck utility at $c_2/c_1 = 0.9$ as the benchmark value, the gap between the total truck utility under other c_2/c_1 values and the benchmark value is calculated, denoted as *Gap* as the vertical axis in the figure. Therefore, when $c_2/c_1 = 0.9$, the *Gap* value is zero. The observations reveal that as c_2/c_1 increases, *Gap* shows a linear growth trend, indicating that as c_2 approaches c_1, the total truck utility increases. This is because, as c_2 decreases, the benchmark cost for trucks transporting goods individually decreases, resulting in reduced utility derived from participating in the TSM.

We further explore the effects of c_2/c_1 on the empty rate, as shown in Fig. 7. The figure illustrates the impact of c_2/c_1 values on the

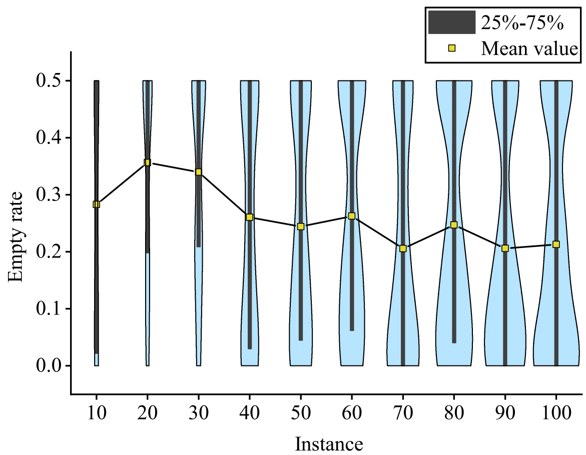

Fig. 4. Empty rate of trucks.

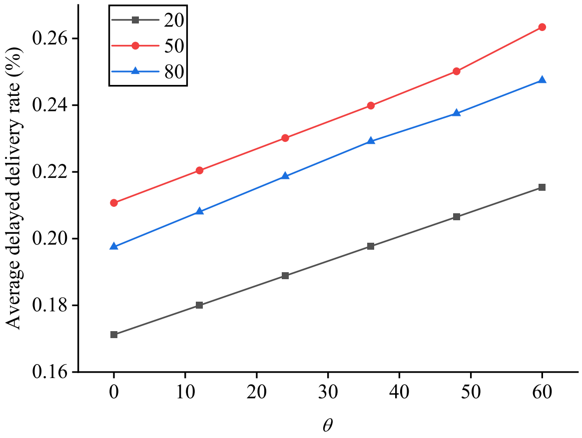

Fig. 5. Effects of θ on the delayed-delivery rate.

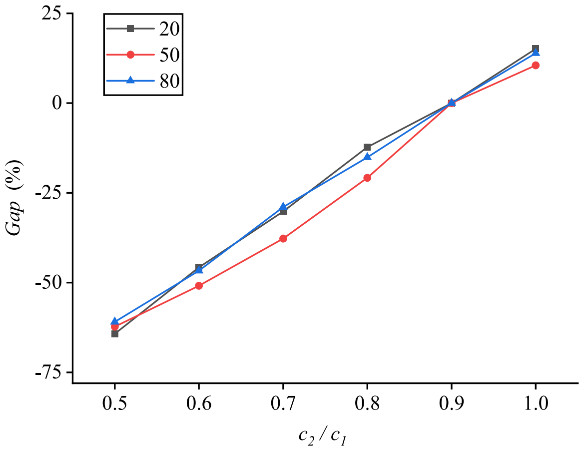

Fig. 6. Effects of c_2/c_1 on the total truck utility.

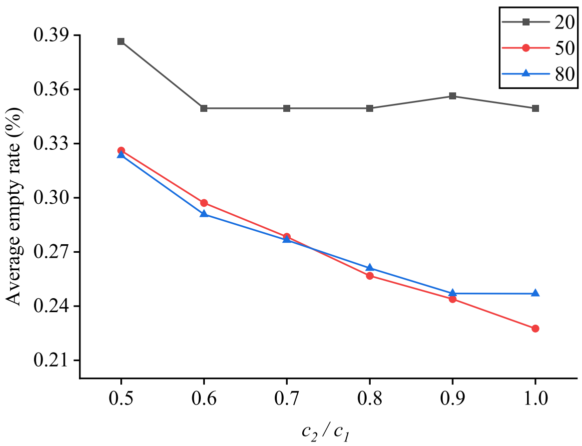

Fig. 7. Effects of c_2/c_1 on the empty rate.

empty rate of the trucks. The observation reveals that, except for the instance $|T| = 20$ where the empty rate remains relatively constant, in the other two instances, as c_2/c_1 gradually increases, the empty rate consistently decreases. As c_2/c_1 increases, the benchmark cost for the trucks also increases. Trucks participating in the TSM gain more utility and attract more trucks to participate in this mode. As the number of participating trucks increases, the empty rate decreases further.

8. Conclusions

In the context of a platform-based TSM, this study addresses the stable truck-matching problem among partners. All trucks first upload the relevant information to the platform, which, after extensive computations, delivers a stable matching scheme that satisfies all participating trucks. Trucks gain utility by executing a platform-delivered trailer-swapping scheme, and the platform secures stable revenue by charging commissions. Specifically, this study focuses on platform computation and elucidates the methods employed to achieve a stable matching scheme. Initially, a preliminary preprocessing step is performed by collecting truck information. During this preprocessing phase, the construction time of the preference list is effectively reduced by utilizing the derived acceleration strategies. Subsequently, the initially constructed preference list is further processed to delete the invalid chains, thereby reducing the length of the preference list. Upon obtaining the final preference list, the model [SMM] is employed to determine the optimum stable matching solution. However, for larger-scale instances, completing the preprocessing within a reasonable time is impractical, and the model [SMM] fails to converge within an acceptable time. To overcome this challenge, the IPTH algorithm is designed to efficiently solve the STMP-TSM. The IPTH algorithm gradually approaches a stable matching solution by continuously relaxing the constraint on the maximum group length. Moreover, by leveraging the matching scheme from the previous iteration, the algorithm trims chains in the preference list during the current iteration, thereby significantly reducing the number of feasible groups. This results in a satisfactory and stable matching solution within a shorter time. A series of numerical experiments is conducted to validate the effectiveness, convergence, and stability of the IPTH algorithm. The TSM is subjected to characteristic analyses considering the optimization objectives, delay-delivery rate, and empty rate. Sensitivity analyses of the key parameters are performed, leading to several recommendations and managerial insights.

This study explores the method to determine a satisfactorily stable matching scheme. When stable matching is not achievable, the question of how platforms can achieve stability through subsidies should be considered for further consideration. Currently, our developed IPTH algorithm is limited to solving instances involving up to 100 trucks, and it encounters challenges in effectively handling larger-scale instances. The investigation of effective algorithms for achieving optimal stable matching solutions and addressing larger-scale instances warrants further exploration. These aspects will be addressed in future studies.

CRediT authorship contribution statement

Wenxiang Peng: Writing – review & editing, Writing – original draft, Validation, Software, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Xiangsheng Chen:** Supervision, Resources, Project administration, Methodology, Funding acquisition, Conceptualization. **Zhaojie Xue:** Writing – review & editing, Writing – original draft, Supervision, Methodology, Funding acquisition, Conceptualization. **Yubin Liao:** Visualization, Software, Investigation, Data curation. **Jintao You:** Resources, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgements

This work was supported by the National Natural Science Foundation of China (grant number 52090084), the Strategic Research and Consulting Project of Chinese Academy of Engineering (grant number 2022-JB-02), the Internal Project Fund from Shenzhen Research Institute of Big Data (grant number J00220230013), the Guangdong Provincial Natural Science Foundation (GNSF) (grant number 2023A1515110228), and the Longgang District Special Funds for Science and Technology Innovation (grant number LGKCSDPT2023002).

Appendix

Notation	Definition
$G = (N,E)$	Graph with node set N and edge set E
(i,j)	Indices for edges
i,j	Indices for nodes
k,l,q	Indices for trucks
o_k/d_k	Index for origin/destination node of truck k
λ	Swapping group

(continued on next page)

(*continued*)

Notation	Definition
$\mathscr{M}$	Matching scheme
$\mathscr{M}^*$	Stable matching scheme
$\mathscr{H}$	Final preference list
Sets:	
N	Set of nodes
E	Set of edges
T	Set of trucks
Φ	Set of groups
Φ_k	Set of all groups that include truck k
R_{ki}	Set of chains for truck k at node i
F_{ki}^l	Subset of R_{ki}, where truck l is the first element in each chain
S_{ki}^l	Subset of R_{ki}, where truck l is the second element in each chain
Parameters:	
c_1/c_2	The unit travel-time cost for laden/empty trucks
f_k	The benchmark cost for truck k completing the transportation task individually
w_k	The estimated departure time of truck k from its origin node
g_k	The benchmark time for truck k completing the transportation task individually
θ	The duration of the trailer-swapping operation
p_k	The unit time penalty cost for trailers of truck k with delayed delivery
μ^λ	The total utility of group λ
μ_k^λ	The utility that truck k obtains by participating with group λ
δ_{kl}^i	The saved transportation costs for truck k when it leads the trailer of truck l, with the corresponding trailer-swapping node being i
σ_{kl}^i	The time at which truck k leads the trailer of truck l to reach the destination node d_l, with the corresponding trailer-swapping node being i
ε_{kl}^i	The penalty cost for truck l when the trailer of truck l is led by truck k, with the corresponding trailer-swapping node being i
φ_{kr}	The utility obtained by truck k when it chooses chain r
Decision variables:	
X_λ	Binary variable indicating whether group λ is chosen in the matching scheme.
Y_{kr}	Binary variable indicating whether truck k chooses chain r from R_{ki}

References

Akpan, I.J., 2023. Thirty years of International Transactions in Operational Research: past, present, and future direction. Int. Trans. Oper. Res. 30, 2709–2728. https://doi.org/10.1111/itor.13321.

Atef Yekta, H., Bergman, D., Day, R., 2023. Balancing stability and efficiency in team formation as a generalized roommate problem. Nav. Res. Logist. 70, 72–88. https://doi.org/10.1002/nav.22084.

Badraoui, I., van der Lans, I.A.M.C., Boulaksil, Y., van der Vorst, J.G.A.J., 2023. Horizontal logistics collaboration success factors: expectations versus reality. Benchmarking 31 (1), 29–52. https://doi.org/10.1108/BIJ-04-2022-0274.

Barua, L., Zou, B., Choobchian, P., 2023. Maximizing truck platooning participation with preferences. Transp Res E 179. https://doi.org/10.1016/j.tre.2023.103297.

Basso, F., D'Amours, S., Rönnqvist, M., Weintraub, A., 2019. A survey on obstacles and difficulties of practical implementation of horizontal collaboration in logistics. Int. Trans. Oper. Res. 26, 775–793. https://doi.org/10.1111/itor.12577.

Ben Jouida, S., Guajardo, M., Klibi, W., Krichen, S., 2021. Profit maximizing coalitions with shared capacities in distribution networks. Eur J Oper Res 288, 480–495. https://doi.org/10.1016/j.ejor.2020.06.005.

Bouchery, Y., Hezarkhani, B., Stauffer, G., 2022. Coalition formation and cost sharing for truck platooning. Transp Res B 165, 15–34. https://doi.org/10.1016/j.trb.2022.08.007.

Chen, S., Wang, H., Meng, Q., 2023. Cost allocation of cooperative autonomous truck platooning: Efficiency and stability analysis. Transp Res B 173, 119–141. https://doi.org/10.1016/j.trb.2023.04.008.

Cleophas, C., Cottrill, C., Ehmke, J.F., Tierney, K., 2019. Collaborative urban transportation: Recent advances in theory and practice. Eur J Oper Res 273, 801–816. https://doi.org/10.1016/j.ejor.2018.04.037.

Cseh, Á., Irving, R.W., Manlove, D.F., 2019. The Stable Roommates Problem with Short Lists. Theory Comput Syst 63, 128–149. https://doi.org/10.1007/s00224-017-9810-9.

da Cruz, H.F.A., Salles da Cunha, A., 2023. The Profitable Single Truck and Trailer Routing Problem with Time Windows: Formulation, valid inequalities and branch-and-cut algorithms. Comput Ind Eng 180. https://doi.org/10.1016/j.cie.2023.109238.

Danilov, V.I., 2023. Review of the theory of stable matchings and contract systems. Comput. Math. Math. Phys. 63, 466–490.

Deng, S., Zhou, D., Wu, G., Wang, L., You, G., 2023. Evolutionary game analysis of three parties in logistics platforms and freight transportation companies' behavioral strategies for horizontal collaboration considering vehicle capacity utilization. Complex and Intelligent Systems 9, 1617–1637. https://doi.org/10.1007/s40747-022-00873-9.

Fan, T., Pan, Q., Pan, F., Zhou, W., Chen, J., 2020. Intelligent logistics integration of internal and external transportation with separation mode. Transp Res E 133. https://doi.org/10.1016/j.tre.2019.10.011.

Farvaresh, H., Shahmansouri, S., 2020. A coalition structure algorithm for large-scale collaborative pickup and delivery problem. Comput Ind Eng 149. https://doi.org/10.1016/j.cie.2020.106737.

Fenoaltea, E.M., Baybusinov, I.B., Zhao, J., Zhou, L., Zhang, Y.C., 2021. The Stable Marriage Problem: An interdisciplinary review from the physicist's perspective. Phys Rep 917, 1–79. https://doi.org/10.1016/j.physrep.2021.03.001.

Gale, D., Shapley, L.S., 1962. College admissions and the stability of marriage. Am. Math. Mon. 69, 9–15.

Giudici, A., Lu, T., Thielen, C., Zuidwijk, R., 2021. An Analysis of the Stability of Hinterland Container Transport Cooperation. Transp Sci 55, 1170–1186. https://doi.org/10.1287/trsc.2021.1050.

Guajardo, M., Rönnqvist, M., 2016. A review on cost allocation methods in collaborative transportation. Int. Trans. Oper. Res. 23, 371–392. https://doi.org/10.1111/itor.12205.

Guajardo, M., Rönnqvist, M., Flisberg, P., Frisk, M., 2018. Collaborative transportation with overlapping coalitions. Eur J Oper Res 271, 238–249. https://doi.org/10.1016/j.ejor.2018.05.001.

Islam, S., Olsen, T., 2014. Truck-sharing challenges for hinterland trucking companies: A case of the empty container truck trips problem. Bus. Process. Manag. J. 20, 290–334. https://doi.org/10.1108/BPMJ-03-2013-0042.

Lai, M., Cai, X., Hall, N.G., 2022. Cost Allocation for Less-Than-Truckload Collaboration via Shipper Consortium. Transp Sci 56, 585–611. https://doi.org/10.1287/trsc.2021.1066.

Li, J., Zheng, Y., Dai, B., Yu, J., 2020. Implications of matching and pricing strategies for multiple-delivery-points service in a freight O2O platform. Transp Res E 136. https://doi.org/10.1016/j.tre.2020.101871.

Liu, W., Wu, C., Feng, B., Liu, J., 2015. Conditional preference in recommender systems. Expert Syst Appl 42, 774–788. https://doi.org/10.1016/j.eswa.2014.08.044.

Mei, Q., Li, J., Ursavas, E., Zhu, S.X., Luo, X., 2021. Freight transportation planning in platform service supply chain considering carbon emissions. Int J Prod Econ 240. https://doi.org/10.1016/j.ijpe.2021.108241.

Miller, J., Nie, Y.M., Liu, X., 2020. Hyperpath Truck Routing in an Online Freight Exchange Platform. Transp Sci 54, 1676–1696. https://doi.org/10.1287/trsc.2020.0989.

Pan, S., Trentesaux, D., Ballot, E., Huang, G.Q., 2019. Horizontal collaborative transport: survey of solutions and practical implementation issues. Int J Prod Res. https://doi.org/10.1080/00207543.2019.1574040.

Park, A., Chen, R., Cho, S., Zhao, Y., 2023. The determinants of online matching platforms for freight services. Transp Res E 179. https://doi.org/10.1016/j.tre.2023.103284.

Peng, Z., Shan, W., Guan, F., Yu, B., 2016. Stable vessel-cargo matching in dry bulk shipping market with price game mechanism. Transp Res E 95, 76–94. https://doi.org/10.1016/j.tre.2016.08.007.

Soriano, A., Gansterer, M., Hartl, R.F., 2023. The multi-depot vehicle routing problem with profit fairness. Int J Prod Econ 255. https://doi.org/10.1016/j.ijpe.2022.108669.

Sun, X., Yin, Y., 2019. Behaviorally stable vehicle platooning for energy savings. Transp Res C 99, 37–52. https://doi.org/10.1016/j.trc.2018.12.017.

Sun, X., Yin, Y., 2021. Decentralized game-theoretical approaches for behaviorally-stable and efficient vehicle platooning. Transp Res B 153, 45–69. https://doi.org/10.1016/j.trb.2021.08.012.

Todosijević, R., Hanafi, S., Urošević, D., Jarboui, B., Gendron, B., 2017. A general variable neighborhood search for the swap-body vehicle routing problem. Comput Oper Res 78, 468–479. https://doi.org/10.1016/j.cor.2016.01.016.

Toffolo, T.A.M., Christiaens, J., Van Malderen, S., Wauters, T., Vanden Berghe, G., 2018. Stochastic local search with learning automaton for the swap-body vehicle routing problem. Comput Oper Res 89, 68–81. https://doi.org/10.1016/j.cor.2017.08.002.

van Zon, M., Spliet, R., van den Heuvel, W., 2021. The joint network vehicle routing game. Transp Sci 55, 179–195. https://doi.org/10.1287/TRSC.2020.1008.

Villegas, J.G., Prins, C., Prodhon, C., Medaglia, A.L., Velasco, N., 2013. A matheuristic for the truck and trailer routing problem. Eur J Oper Res 230, 231–244. https://doi.org/10.1016/j.ejor.2013.04.026.

Wang, Y., 2023. A collaborative approach based on Shapley value for carriers in the supply chain distribution. Heliyon 9. https://doi.org/10.1016/j.heliyon.2023.e17967.

Yang, S., Zhang, J., Zhou, S., 2023. The cost transportation game for collaboration among transportation companies. Annals of Operations Research. https://doi.org/10.1007/s10479-023-05466-4.

Zhang, A., Kang, J.E., Kwon, C., 2020. Generalized stable user matching for autonomous vehicle co-ownership programs. Serv. Sci. 12, 61–79. https://doi.org/10.1287/serv.2020.0257.

Zheng, X.X., Liu, Z., Li, K.W., Huang, J., Chen, J., 2019. Cooperative game approaches to coordinating a three-echelon closed-loop supply chain with fairness concerns. Int. J. Prod. Econ. 212, 92–110. https://doi.org/10.1016/j.ijpe.2019.01.011.