ARTICLE IN PRESS

European Journal of Operational Research xxx (xxxx) xxx

Contents lists available at ScienceDirect

European Journal of Operational Research

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

Production, Manufacturing, Transportation and Logistics

A two-step large neighborhood search for a collaborative two-tier city logistics system

Johannes Gückel [a,*], Teodor Gabriel Crainic [b], Pirmin Fontaine [a]

[a] Catholic University of Eichstätt-Ingolstadt, Ingolstadt School of Management & Mathematical Institute of Machine Learning and Data Science, Auf der Schanz 49, 85049 Ingolstadt, Germany

[b] Interuniversity Research Centre on Enterprise Networks, Logistics and Transportation (CIRRELT) and Department of Management and Technology, University of Quebec in Montréal, Canada

ARTICLE INFO

Keywords:
Logistics
City logistics
Large neighborhood search
Horizontal collaboration
Cost allocation

ABSTRACT

The rapid transport of freight is an essential feature of modern societies and an enabling factor for economic trade and growth. Nevertheless, the negative impact of freight transportation in urban areas poses challenges for Logistics Service Providers (LSPs) as well as for municipalities. In this context, a centrally coordinated Two-Tier City Logistics System (2T-CLS), in which LSPs voluntarily agree to collaborate with each other, has the potential to reduce both economic and environmental impact costs. In order to plan such a system, it is important not only to make efficient use of the resources provided but also to have a mechanism that allocates the costs incurred to the individual LSPs. We introduce a mixed-integer linear program (MILP) formulation for the tactical planning of a 2T-CLS involving multiple LSPs that share their resources and customer demands. This MILP comprises a service network design formulation on the first tier and a vehicle routing problem formulation on the second tier, which are connected with each other. To address larger instances, we introduce an Integrative Two-Step Large Neighborhood Search with adaptive components that integrates first and second-tier decisions. In order not only to minimize the costs incurred but also to distribute them fairly, we investigate different problem-specific proportional methods, as well as more advanced game theoretical methods. Numerical experiments show that collaboration leads to average cost savings of 26.91%, which primarily stems from first-tier collaboration.

1. Introduction

The transportation of goods and people is an indispensable part of modern societies and economies. At the same time, transportation is a major disruptive factor, especially for urban life, due to congestion, noise, emissions, space consumption, and other negative external effects. According to a United Nations (2018) forecast, urbanization will increase from 55% in 2018 to 68% in 2050. Simultaneously, the increasing degree of digitalization is leading to a trend towards more online orders, which means that e-commerce volume will rise as well Lone et al. (2021). These two developments result in an increasing number of goods and people that need to be transported within cities. Making cities sustainable requires urban logistics concepts that not only meet the increasing demand for goods and services efficiently but also prioritize environmental responsibility as a key aspect of sustainability. While the need for efficient and sustainable transport concepts for inner-city transportation is more important than ever, Logistics Service Providers (LSPs) struggle with low load factors, empty trips, long dwell times at loading and unloading points, and a large number of deliveries to individual customers (Cepolina & Farina, 2015). The goal of city logistics is to efficiently manage the transportation of goods while minimizing the negative consequences of transportation (Savelsbergh & van Woensel, 2016). Introduced by Crainic et al. (2004), 2T-CLSs feature the preliminary delivery of freight to CDCs located on the outskirts of urban areas. From these CDCs, large urban vehicles deliver freight to satellites — small facilities located in the inner city. Proceeding from the satellites, the last-mile deliveries to customer locations are completed using vehicles, hereinafter referred to as city freighters, which are smaller, environmentally friendly, and cost-efficient. This two-tier structure leads to challenging optimization problems because of the appearance of NP-hard problems on both tiers that require coordination and synchronization.

Efficiently managing 2T-CLSs can be achieved either by a single LSP operating throughout a city, as is the case in Monaco, or through collaboration — i.e., resource and/or demand sharing among multiple

* Corresponding author.
E-mail address: jgueckel@ku.de (J. Gückel).

https://doi.org/10.1016/j.ejor.2025.08.016
Received 23 December 2024; Accepted 8 August 2025
Available online 3 October 2025

LSPs (Fontaine et al., 2023). While the first option may be difficult to implement in many cities due to political barriers, collaboration often faces resistance from LSPs because they wish to retain direct contact with customers. This raises the question of whether collaboration on both tiers is necessary and what benefits it might offer. Although collaboration in vehicle routing in single-tier environments has been well-studied in the literature, collaboration within 2T-CLSs remains relatively unexplored.

Previous studies on the collaboration of LSPs have focused on single-tier environments, particularly in vehicle routing or similar settings, and consistently reported significant cost savings. However, these studies do not consider the interdependencies between tiers, where collaboration on one tier can influence operations and costs on the other.

This paper specifically examines collaboration among LSPs in the tactical planning of 2T-CLSs, allowing us to assess how collaboration on each tier contributes to overall cost savings. By doing so, we can analyze the interaction effects between tiers and evaluate the mutual impact of collaboration on both tiers.

In a survey, Cruijssen, Cools et al. (2007) found that LSPs have a strong belief in the efficiency gains that collaboration can offer. Further, Cruijssen, Dullaert et al. (2007) surveyed a considerable number of LSPs in Belgium, finding that although collaboration presents clear advantages, developing a justifiable cost-sharing plan poses a significant obstacle to LSPs' joint ventures.

To maintain the stability of collaboration among partners, the question of how to allocate the resulting total system cost to the participating LSPs in a way that is perceived as fair is essential. Such a mechanism provides each partner with a strong incentive to participate. In 2T-CLSs, cost structures are shaped by interdependent decisions on both tiers, such as service selection, routing efficiency, demand pooling, and temporal alignment of operations. Models that focus on one tier of the system while approximating the other can capture only part of the potential cost savings and do not allow for a comprehensive evaluation of their allocation. Furthermore, current studies do not provide insights into the impact of different cost allocation methods in city logistics systems, even though these systems are typically regarded as inherently collaborative. Our study is the first to analyze these aspects in a fully modeled two-tier setting, providing new insights into both the impact of collaboration on monetary and environmental outcomes and the design of fair allocation mechanisms in collaborative city logistics.

While there is increasing interest in collaborative two-tier city logistics, there is still insufficient insight into how resources and customer demands should be shared among LSPs to achieve efficient and equitable outcomes in tactical planning. Existing cooperative studies focus on vehicle routing problems rather than service network design models, and do not evaluate the effects of different cost allocation methods, even though service network design is commonly used for tactical planning in city logistics. Furthermore, models and solution approaches for collaborative two-tier city logistics remain limited, especially when it comes to providing insights into how resources should be deployed and how costs arising from the tactical plan should be allocated to individual LSPs. This paper fills this research gaps by making three key contributions.

1. We introduce an innovative Mixed-Integer Linear Program (MILP) formulation for tactical planning of a 2T-CLS with collaborating LSPs that combines and connects both first-tier multi-modal service network design with second-tier arc-based vehicle routing. This formulation includes the collaboration of LSPs on both tiers individually and regulates the collaboration intensity through constraints that limit the demand sharing among these LSPs.
2. We develop a problem-specific two-step large neighborhood search with adaptive components that leverages the two-tier problem structure through problem-specific operators as well as through thresholds and a solution memory. This method is designed to efficiently solve larger instances, characterized by increased demands, services, and network complexity with additional City Distribution Centers (CDCs) and satellites.
3. Through an extensive numerical study, we show the performance of our metaheuristic and, as the first to do so in this context, provide managerial insights into the environmental and monetary impacts of collaboration in 2T-CLSs. We also evaluate the effects of different proportional and game-theoretical cost allocation methods, particularly with respect to LSPs of different sizes, which is a question that naturally arises in collaborative planning. These insights are highly relevant for both LSPs and municipalities pursuing collaborative city logistics systems.

The paper is organized as follows: After the literature review in Section 2, we detail the problem setting in Section 3. We present the MILP for resource planning in Section 4. Section 5 details our metaheuristic. Section 6 introduces problem-specific proportional methods and shows game theoretical cost allocation methods. In Section 7, we conduct an extensive numerical study regarding the performance of the metaheuristic as well as cost benefits and cost allocations. The paper concludes with a summary and future research directions in Section 8.

2. Literature review

The literature review is divided into four sections. Section 2.1 presents literature related to 2T-CLSs. Section 2.2 focuses on collaboration in transportation, while Section 2.3 reviews the literature on cost allocation methods. Finally, Section 2.4 outlines the research gaps.

2.1. Two-tier city logistics

In their review on two-echelon vehicle routing problems, Sluijk et al. (2023) highlight that there are numerous publications with various model formulations and objective functions. They specifically identify city logistics as a key application area, where the two-tiered structure helps achieve demand consolidation and environmental benefits. They showed that there are essentially two ways of formulating such problems. An arc-based vehicle routing formulation (Jepsen et al., 2013) or a route-based formulation (Baldacci et al., 2013) in which there are sets for feasible routes. Our study aligns with the second formulation grounded in the specific literature on tactical planning of 2T-CLSs, a field primarily characterized by the use of service network design models, which are based on the pre-generation of services that represent, for example, a tram service traveling from a CDC to a sequence of satellites at specific times. Consideration of 2T-CLSs in the scientific community was introduced by the work of Crainic et al. (2004), which addressed the problem of the optimal location of satellites. A case study with data from the city of Rome clearly showed that a 2T-CLS can greatly reduce the distance traveled by large trucks within the city and, in return, smaller, more environmentally friendly vehicles can be used for the final delivery to the customers. The general framework for service network design models in the tactical planning problem in a 2T-CLS was introduced by Crainic et al. (2009). They took into account the time dimension of the demand and the associated need to synchronize and schedule the vehicles of the first- and second tier. In Crainic and Sgalambro (2014), a service network design formulation was proposed for the tactical planning problem in a 2T-CLS, including a discussion on algorithmic solution perspectives. Crainic et al. (2016) explicitly considered demand uncertainty in tactical planning. They proposed a two-stage stochastic programming formulation and presented different strategies for adjusting the plan to the observed demand. Crainic et al. (2021) proposed a scheduled service network design formulation as a modeling framework for tactical planning of a 2T-CLS and adapted it to the specific problem characteristics. Scherr et al. (2019) introduced a service network design formulation for the tactical planning of parcel delivery using mixed autonomous fleets, highlighting cost savings and coordination strategies based on infrastructure and demand. Fontaine et al. (2021) expanded the existing literature by introducing a more realistic problem setting, incorporating both inbound and outbound demand, a multimodal network with diverse vehicle types, and satellite capacity constraints tailored to demand volume and vehicle modes. They presented a scheduled service network design formulation and developed an efficient Benders decomposition algorithm to solve it. In their investigation, they demonstrated that a multi-modal fleet on the first tier of a two-tier system can reduce costs and improve utilization. Considering the decisions about the number and location of facilities on both tiers, Winkenbach et al. (2016) presented a MILP for the two-echelon capacitated location routing problem. Schmidt et al. (2022) investigated the integration of public transport service providers in a two-tier system to deliver freight from the outskirts to satellite depots. Further, Fontaine et al. (2023) recently analyzed when single-tier or two-tier systems benefit cities, finding that higher customer density and greater distances between depots and distribution areas favor two-tier strategies. They also showed that consolidation among LSPs reduces emissions. Related to the literature on 2T-CLSs, Perboli et al. (2021) addressed the joint problem of operating satellites and providing services to customers.

Overall, there are no models that combine service network design on the first tier with arc-based vehicle routing on the second tier and allow LSPs to collaborate with a certain intensity on both tiers. The only investigation explicitly considering different LSPs collaborating in a 2T-CLS was conducted by Crainic et al. (2020). They assumed that demand could be satisfied by any participant of the coalition of LSPs. As they pointed out, the collaboration leads to significant efficiency improvements in terms of monetary costs and environmental footprint. However, they did not consider the allocation of costs to the LSPs and only considered collaboration on the first tier of the system.

2.2. Collaboration in transportation

While collaboration in the maritime and airline industries has been used and explored for some time, it is still a growing area of research in urban freight management, taking on different forms. Vertical collaboration involves organizations at different levels of the supply chain, such as suppliers, retailers, and LSPs. On the other hand, horizontal collaboration refers to collaboration among organizations operating at the same level of the supply chain. Additionally, diagonal collaboration incorporates both horizontal and vertical collaboration (see Rusich et al. (2017) for a framework of collaborative logistics). In this work, we focus on horizontal collaboration. According to EU Commission (2001), horizontal collaboration refers to collaboration through an agreement or concerted practice between companies operating at the same level in the market that are potentially competing with each other.

The literature on horizontal collaboration largely consists of studies on the classification of its forms and empirical studies on its potential benefits and barriers. Cruijssen, Dullaert et al. (2007) highlighted that horizontal collaboration can identify and exploit win-win situations between companies at the same supply chain level. This collaboration can take various forms. As Pérez-Bernabeu et al. (2015) noted, collaborating companies—whether competing or unrelated suppliers, manufacturers, retailers, receivers, or LSPs—can share information, facilities, or resources to reduce costs and/or improve service.

In the context of transportation logistics, horizontal collaboration has gained increased attention in recent years as an efficient instrument for the sustainable and efficient design of freight transport (Pan et al., 2019). For example, Agarwal and Ergun (2010) proposed a mechanism for directing liner shipping carriers to prioritize the collective interests of the alliance while optimizing their own profitability. Similarly, Nataraj et al. (2019) investigated the joint opening of a CDC and concluded that significant overall cost savings can be achieved with higher collaboration intensity. Regarding satellite depots, Bruni et al. (2024) explored the advantages of vertical and horizontal collaboration, identifying substantial cost reductions in both cases. For a comprehensive review of problems, approaches, and further research areas related to horizontal collaboration among LSPs, we refer to Pan et al. (2019).

Recent years have also seen numerous publications presenting the potential benefits of collaborative planning in single-tier settings, particularly regarding total profit improvements (e.g., Montoya-Torres et al., 2016). As highlighted by Gansterer and Hartl (2018) in their survey on collaborative vehicle routing, horizontal collaboration can lead to cost savings ranging from 20% to 30%. However, findings from single-tier settings cannot fully reflect the effects of collaboration across upstream or downstream tiers, nor determine which proportion of costs can be extracted from collaboration in each tier.

When constructing stable collaborations, ensuring the fair treatment of coalition members is crucial. Beyond the cost allocation methods considered in this study, some research has introduced constraints to address the workload of LSPs (e.g., Gansterer et al., 2018; Mancini et al., 2021). Additionally, auction models have been employed to manage demand exchange, where a shared pool allows participants to submit their demands, which are then traded among them through auctions (e.g., Elting et al., 2025; Gansterer et al., 2020; Karels et al., 2020).

Finally, despite the growing body of literature in this area, Parisa et al. (2019) pointed out that the implementation of collaborative approaches in practice remains challenging due to the lack of a viable business model. This highlights the urgent need for a fair cost allocation mechanism to distribute the benefits of collaboration among participating LSPs.

2.3. Cost allocation methods

Cost allocation in collaborative games has already been studied in the literature for a variety of use cases. For a detailed review on cost allocation mechanisms in collaborative transportation, we refer to Guajardo and Rönnqvist (2016) in which over 40 different allocation mechanisms were identified in the reviewed literature. The mechanisms most commonly used in practice are based on proportional methods, where each participant is allocated a cost share in relation to a predefined reference value. In the simplest case, each participant bears an equal share of the costs. Other classic criteria are, for example, the share of demand or the stand-alone costs (Guajardo & Rönnqvist, 2016).

Widely used methods based on cooperative game theory are the Shapley Value (Shapley, 1953), the core allocation (Gillies, 1959), and the nucleolus (Schmeidler, 1969). In addition, cost allocation mechanisms were introduced in the context of specific applications, such as the Equal Profit Method (EPM) developed by Frisk et al. (2010) in the context of collaborative forest transportation. Further cost allocation mechanisms based on the distinction between separable and non-separable costs are introduced by Tijs and Driessen (1986).

Building on these approaches, recent studies have examined cost allocation methods in specific collaborative scenarios. For instance, Vanovermeire et al. (2014) compared different cost allocation methods in collaborative transport, especially with partners who have different characteristics. They showed that the choice of the appropriate cost allocation method depends heavily on the characteristics of the coalition. An example of cost allocation in horizontal carrier collaboration can be found in Verdonck et al. (2016), where they investigated the cooperative carrier facility location problem, comparing three different cost allocation mechanisms. Kimms and Kozeletskyi (2016b) determined the Shapley Value for the cooperative traveling salesman problem. Further, Kimms and Kozeletskyi (2016a) proposed an a priori core cost allocation for a horizontal cooperating traveling salesman, which provided expected costs for the coalition participants.

Nevertheless, neither proportional nor more advanced game-theoretical cost allocation methods have been investigated in the context of 2T-CLSs, highlighting a significant gap in the literature. Despite city logistics systems often being regarded as inherently collaborative, no concrete insights into cost allocation mechanisms have been developed for this domain, even though the consequences of different allocation mechanisms are highly relevant for the success of future collaborative city logistics projects.

2.4. Research gap

Based on our literature analysis, three key research gaps emerge. First, collaboration in 2T-CLSs remains largely unexplored, and little is known about the cost implications of (partial) collaboration at one or both tiers or how collaboration at one tier impacts cost and operations on the other tier, as single-tier collaborative models cannot reflect this. Second, the literature lacks efficient heuristic solution methods for multi-modal service network design at the first tier connected with vehicle routing at the second tier. Third, while cost allocation mechanisms have been studied in cooperative game theory, no studies provide insights into the potential collaborative savings in fully modeled 2T-CLSs and their implications for different cost allocation methods, including both proportional and game-theoretical approaches.

3. Problem setting

We describe the general functioning of 2T-CLSs in Section 3.1. Section 3.2 describes the tactical planning in 2T-CLSs. Section 3.3 outlines the collaborative aspects of the problem.

3.1. 2T-CLS

For the illustration of 2T-CLSs, we use the terms introduced by Crainic et al. (2009). A 2T-CLS consists of two levels of physical infrastructure, the CDCs and the satellite platforms.

CDCs represent the first tier of facilities and are typically located on the outskirts of a city. Freight is first delivered from its origin location to the CDCs. For this, we assume costs for the delivery of each customer demand to each CDC. At the CDCs, the loads are sorted and consolidated for further distribution to satellites via urban vehicle services. A service starts at a certain time period at a CDC, visits one or more satellites, and then returns to the same CDC. Multimodality is represented by different transport modes for the first tier, including different types of urban vehicles. As Fontaine et al. (2021), we differentiate between line-based modes (e.g., tram) that can only drive along predefined lines and free-roaming modes (e.g., truck). Satellites represent the second-tier facilities and are usually located near the city center. Here, the load is again consolidated before the final last-mile delivery to the demand location takes place, using city freighters. These city freighters start at a satellite, visit several demand locations and return back to the satellite. A major challenge in these systems is the synchronization between the first-tier operations and the second-tier operations.

3.2. Tactical planning in a 2T-CLS

According to Crainic (2000) and Crainic and Kim (2007), tactical planning aims to develop a transportation plan to facilitate efficient operations and resource utilization while satisfying the demand for transportation within the quality criteria publicized or agreed upon with the respective customers. The tactical plan is designed for a short time horizon, called schedule length, which corresponds to the length of regularity for parameter setting. It must be set up sometime before based on forecasted parameters. Further, we assume a known deterministic demand, each demand associated with a specific volume. Daily demand fluctuations and uncertainties are carried out on the operational level and are therefore out of the scope of our paper.

The tactical plan of a 2T-CLS determines which services to operate, how to allocate demands to those services and satellites, and addresses second-tier routing. The goal is to satisfy the regular demand most efficiently in terms of a cost-efficient and environmentally friendly use of resources, and at the same time to meet demand conditions such as release and due dates. Strategic decisions, such as the location of satellites, are assumed to be given and not addressed in tactical planning. A key challenge within these two-tier systems is to ensure that the tactical plan enables the synchronization of operations between the first and second tiers at the operational level. This synchronization involves precise coordination in terms of location and timing. Concretely, the satellite to which the urban vehicle delivers a demand determines the starting position for that demand on the second tier. Further, in terms of timing, the delivery to a satellite must precede the departure from that satellite of each demand.

3.3. Collaboration in a 2T-CLS

We assume that multiple LSPs operate within the city using the same CDCs and satellites. These LSPs form a coalition and collaborate on a voluntary basis. Each LSP has its own set of demands, each with a certain demand volume, which is brought into the coalition when they join. On the other side, each LSP has its own resources for transportation. These resources are first-tier services, capacities at satellites that they are allowed to use, as well as vehicle fleets. We assume that these LSPs agree to collaborate with each other through a central coordinator responsible for deciding which resources to use in order to fulfill all demands. The central coordinator can either be a third-party provider or a coordinating entity that is jointly operated by the LSPs involved. In either case, each LSP must be willing to share its information about demands and transportation resources.

We assume that the participating LSPs can offer transportation capacities on both the first tier and second tier. Thus, each LSP could satisfy its own demand. Therefore, collaboration can take place in two possible ways. First, by sharing their customer demands, so that each LSP can also have their demands delivered by services of other LSPs and vice versa. Second, by utilizing the same resources such as vehicle fleets, satellite depots, and CDCs. This enables freight consolidation of the participating LSPs by using common transport resources for first- and second-tier delivery. Fig. 1 represents such a 2T-CLSs in which two LSPs collaborate with each other.

Within collaborations, it is important not only that the total costs resulting from the tactical planning of the provided resources are low but also that each individual LSP benefits from the collaboration and perceives the distribution of costs as fair. Therefore, a cost allocation method is required that takes the results of tactical planning as input and allocates the total system costs to the LSPs involved in order to enable fair and stable collaboration.

4. General approach and model

This section introduces a formulation that integrates first-tier multimodal service network design with second-tier vehicle routing, linking the two through connection constraints that align decisions by time periods and handover locations (satellites). We are the first to combine service network design and vehicle routing in this way, considering (partial) collaboration across both tiers with specific collaboration intensity constraints that limit the volume of shared demands on both tiers individually. First-tier vehicles handle larger volumes where collaboration mainly improves utilization, while on the second tier, cost savings largely result from more efficient route planning, which requires explicit vehicle routing modeling. Section 4.1 introduces the general modeling outline and notation. Building upon this, Section 4.2 presents our mathematical formulation for resource planning in a collaborative 2T-CLS.

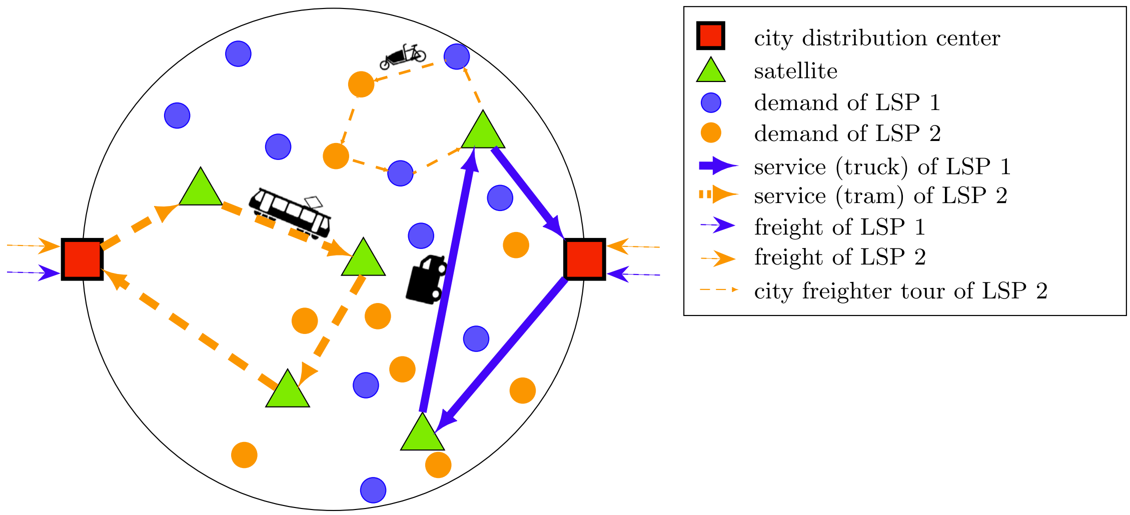

Fig. 1. 2T-CLS with collaborating LSPs.

4.1. General outline and notation

At the core of the problem is a set of LSPs $\mathcal{N}$. Each LSP $n \in \mathcal{N}$ has a subset of customer demands $\mathcal{D}(n)$ out of the set of all demands $\mathcal{D}$. Each demand $d \in \mathcal{D}$ is characterized by a volume v_d, the origin, the destination, the release date that specifies when the demand is available at the CDCs and a due date b_d that specifies when the demand must be delivered to its final location.

Consistent with Crainic et al. (2004), we model the first tier as a service network design. The time dimension is represented by a schedule length that is divided in the periods $1, \ldots, |\mathcal{P}|$. All considered time-related parameters are assumed to be integer multiples of the period length. Let $\mathcal{E}$ be the set of CDCs and $\mathcal{S}$ the set of satellites. For the first-tier deliveries, a set of services $\mathcal{R}$ is available, characterized by transportation mode m_r, vehicle type t_r and cost c_r, representing not only monetary but also environmental impact cost. Each service $r \in \mathcal{R}$ starts at a CDC $e_r \in \mathcal{E}$ and visits an ordered sequence of satellites. Transportation modes are represented by the set $\mathcal{M}$. Each mode of transportation is associated with different vehicle types (e.g., the mode "truck" can consist of the types "small truck" and "big truck"), where $\mathcal{T}$ represents the set of vehicle types. Each type t has a specific capacity u_t and a specific fleet size h_{etn} at CDC e provided by LSP n.

We consider three different capacity constraints on satellites. Operations at satellite $s \in \mathcal{S}$ are limited by a maximum number of urban vehicles a_{spn} LSP n can transfer in period p. The same type of constraint is additionally considered by $\bar{a}_{spmn}$ for each mode m to account for different capacity constraints per mode. In addition, we consider an upper limit for the total volume of freight g_{spn} LSP n can accommodate at satellite s in period p. The costs incurred to deliver demand d from its origin to a CDC e are also externally given and denoted as f_{de}. Note that these costs could further include reallocation costs of demands to CDCs of other LSPs if CDCs are not operated by all LSPs together.

To account for the different LSPs, each LSP n is able to provide a subset of services $\mathcal{R}(n) \subseteq \mathcal{R}$. Thereby $\mathcal{R}(n) \cap \mathcal{R}(\hat{n}) = \emptyset \quad \forall n, \hat{n} \in \mathcal{N} : \hat{n} \neq n$ holds. Further, each service has a certain service time w_r for unloading operations at each satellite, as well as an arrival time τ_{rs} at satellite s, respectively. $\mathcal{R}(p,s)$ and $\mathcal{R}(p,s,m)$ define the subsets of $\mathcal{R}$ that include all services (of mode m) that operate at satellite s during period p, while $\mathcal{R}(p,e,t)$ represents the subset that includes services of vehicle type t, starting at CDC e and operating in period p. We consider the subset $\mathcal{R}(d,s)$ that represents the services that fulfill the release date for demand d at the CDC the service starts from, and further include satellite s in their route.

We include the second-tier routing in our model rather than relying on approximations. This enables us to model collaboration effects on both tiers individually and to evaluate their respective cost impacts in a differentiated manner. To achieve this, we integrate a Vehicle Routing Problem with Release and Due Dates (VRPRDD) (Shelbourne et al., 2017) on each satellite, where the release dates of each demand are determined by the arrival period of the service that the demand is assigned to at the corresponding satellite resulting from the first-tier, plus a service time w_r. For the second-tier routing, we construct a directed graph $\mathcal{G} = (\mathcal{V}, \mathcal{A})$ where $\mathcal{V} = \mathcal{D} \cup \mathcal{S} \cup \hat{\mathcal{S}}$ are the nodes consisting of the set of demand locations $\mathcal{D}$, satellite nodes $\mathcal{S}$ that serve as the starting points of the city freighters and a duplication of the satellite nodes $\hat{\mathcal{S}}$ that serve as the ending points of the city freighters. $\mathcal{K}$ represents the set of city freighters. Each city freighter k has a satellite node $s_k^+ \in \mathcal{S}$ at which it starts and a satellite node $s_k^- \in \hat{\mathcal{S}}$ at which it ends. These two satellite nodes are identical, i.e., they refer to the same physical location, as each city freighter starts and ends at the same satellite. The subset $\mathcal{K}(s)$ defines the city freighters starting from satellite s. The set $\mathcal{A}$ represents the arcs connecting the nodes. The subsets $\mathcal{A}(k)$ represent the set of arcs that city freighter k can travel, as each city freighter is only allowed to start and end at the satellite it is assigned to. We specify the allowable arcs for city freighter k to end in node i as $\delta_k^-(i)$, and the arcs starting from node i as $\delta_k^+(i)$. The costs that arise if arc $(i,j) \in \mathcal{A}$ is used by a city freighter are represented by $\hat{c}_{ij}$ and the time distance of each arc $(i,j) \in \mathcal{A}$ is represented by $\bar{t}_{ij}$. We assume service times l of city freighters at each node. Each city freighter has a capacity of q. Again, we account for different LSPs through subsets of city freighter $\mathcal{K}(n)$ provided by LSP n.

Additionally, we define two parameters, α_1 and α_2. These parameters specify for each LSP the lower bound of demand volume that must be assigned to own services on the first tier (α_1) and to own city freighters on the second tier (α_2), respectively. For example, $\alpha_1 = 0.2$ and $\alpha_2 = 0.3$ imply that at least 20% of an LSP's demand volume must be assigned to its own services and at least 30% of an LSP's demand volume must be fulfilled by its own city freighters. Therefore, these two parameters limit the demand sharing between the LSPs. A full table with the notation is provided in Appendix A.

4.2. Mathematical problem formulation

In this section, we introduce the MILP for our problem setting in which a central coordinator minimizes the total system costs. After presenting the decision variables and the objective function, we start with the first-tier service network design constraints. Then, we describe the second-tier vehicle routing constraints. On both tiers, we introduce constraints that limit the sharing of demands according to the parameters α_1 and α_2. Further, we introduce linking constraints, which connect the first-tier with the second-tier and enable synchronization on the operational level.

We consider the following decision variables: the binary variable y_r takes the value one if service $r \in \mathcal{R}$ is selected, the binary variable x_{dsr} takes the value one if demand $d \in \mathcal{D}$ is assigned to satellite $s \in \mathcal{S}$ and service $r \in \mathcal{R}$ and the binary variable z_{ijk} takes the value one if second-tier city freighter k uses arc $(i,j) \in \mathcal{A}(k)$. Finally, p_{ik} is specifying the arrival time of city freighter k at vertex $i \in \mathcal{D} \cup \{s_k^+, s_k^-\}$.

Grounded on this, we formulate the following MILP:

Objective function

$$min \quad \sum_{r\in\mathcal{R}} c_r \cdot y_r + \sum_{d\in\mathcal{D}}\sum_{s\in\mathcal{S}}\sum_{r\in\mathcal{R}} f_{de_r} \cdot x_{dsr} + \sum_{k\in\mathcal{K}}\sum_{(i,j)\in\mathcal{A}(k)} \hat{c}_{ij} \cdot z_{ijk} \tag{1}$$

The objective function (1) minimizes the total cost incurred. These costs are made up of three components: 1. the costs related to the operation of services, 2. the costs for assigning demands to services and thus simultaneously to the CDCs from which the services start, and 3. the second-tier costs associated with routing the demands from the satellites to their final location. This objective function thus differs from other service network design models in the field of 2T-CLSs, in which the costs of the second tier are either linearly approximated or neglected at all (e.g., Fontaine et al., 2021; Scherr et al., 2019). Such linear approximations typically rely on the Euclidean distance between customer locations and satellites, offering only a rough estimate of actual routing costs. Moreover, they fail to capture the operational efficiencies enabled by cooperation, such as higher vehicle utilization, improved demand consolidation, and better temporal coordination. By explicitly modeling second-tier routing, our approach accounts for these effects and enables a more accurate assessment of collaboration benefits on each tier.

First-tier service network design constraints

$$\sum_{s\in\mathcal{S}}\sum_{r\in\mathcal{R}} x_{dsr} = 1 \qquad \forall d \in \mathcal{D} \tag{2}$$

$$\sum_{s\in\mathcal{S}}\sum_{r\in\mathcal{R}\setminus\mathcal{R}(d,s)} x_{dsr} = 0 \qquad \forall d \in \mathcal{D} \tag{3}$$

$$\sum_{d\in\mathcal{D}}\sum_{s\in\mathcal{S}} v_d \cdot x_{dsr} \le u_{t_r} \cdot y_r \qquad \forall r \in \mathcal{R} \tag{4}$$

$$\sum_{r\in\mathcal{R}(p,e,t)} y_r \le \sum_{n\in\mathcal{N}} h_{etn} \qquad \forall e \in \mathcal{E}, t \in \mathcal{T}, p \in \mathcal{P} \tag{5}$$

$$\sum_{d\in\mathcal{D}}\sum_{r\in\mathcal{R}(p,s)} v_d \cdot x_{dsr} \le \sum_{n\in\mathcal{N}} g_{spn} \qquad \forall s \in \mathcal{S}, p \in \mathcal{P} \tag{6}$$

$$\sum_{r\in\mathcal{R}(p,s)} y_r \le \sum_{n\in\mathcal{N}} a_{spn} \qquad \forall s \in \mathcal{S}, p \in \mathcal{P} \tag{7}$$

$$\sum_{r\in\mathcal{R}(p,s,m)} y_r \le \sum_{n\in\mathcal{N}} \bar{a}_{spmn} \qquad \forall s \in \mathcal{S}, p \in \mathcal{P}, m \in \mathcal{M} \tag{8}$$

$$\alpha_1 \cdot \sum_{d\in\mathcal{D}(n)} v_d \le \sum_{d\in\mathcal{D}(n)}\sum_{s\in\mathcal{S}}\sum_{r\in\mathcal{R}(n)} x_{dsr} \cdot v_d \qquad \forall n \in \mathcal{N} \tag{9}$$

$$x_{dsr} \in \{0,1\} \qquad \forall d \in \mathcal{D}, s \in \mathcal{S}, r \in \mathcal{R} \tag{10}$$

$$y_r \in \{0,1\} \qquad \forall r \in \mathcal{R} \tag{11}$$

Constraints (2) and (3) guarantee that each demand is assigned to a single service for which the release date at the CDC is not violated. The capacity limit for each urban vehicle type of each service is ensured by Constraints (4). Constraints (5) limit the maximum number of used urban vehicle types t at CDC e for each period p. Constraints (6) limit the amount of freight operated in period p at satellite s. The number of urban vehicles operating in period p at satellite s is limited by Constraints (7) while Constraints (8) explicitly limit the number of urban vehicles of mode m. To limit the demand sharing, Constraints (9) enforce that each LSP satisfies at least a fraction α_1 of its own demand volume by its own services.

Second-tier vehicle routing constraints

For the resulting VRPRDD on the second tier, we use the previously defined graph $\mathcal{G}$.

$$\sum_{k\in\mathcal{K}}\sum_{j\in\delta_k^+(d)} z_{djk} = 1 \qquad \forall d \in \mathcal{D} \tag{12}$$

$$\sum_{j\in\delta_k^+(s_k^+)} z_{s_k^+ jk} = 1 \qquad \forall k \in \mathcal{K} \tag{13}$$

$$\sum_{j\in\delta_k^-(d)} z_{jdk} - \sum_{j\in\delta_k^+(d)} z_{djk} = 0 \qquad \forall d \in \mathcal{D}, k \in \mathcal{K} \tag{14}$$

$$\sum_{i\in\delta_k^-(s_k^-)} z_{i s_k^- k} = 1 \qquad \forall k \in \mathcal{K} \tag{15}$$

$$\sum_{d\in\mathcal{D}}\sum_{j\in\delta_k^+(d)} v_d \cdot z_{djk} \le q \qquad \forall k \in \mathcal{K} \tag{16}$$

$$p_{ik} + \bar{t}_{ij} + l - p_{jk} \le M_1 \cdot (1 - z_{ijk}) \qquad \forall (i,j) \in \mathcal{A}(k), k \in \mathcal{K} \tag{17}$$

$$p_{dk} \le b_d \qquad \forall k \in \mathcal{K}, d \in \mathcal{D} \tag{18}$$

$$\alpha_2 \cdot \sum_{d\in\mathcal{D}(n)} v_d \le \sum_{d\in\mathcal{D}(n)}\sum_{k\in\mathcal{K}(n)}\sum_{j\in\delta_k^+(d)} z_{djk} \cdot v_d \qquad \forall n \in \mathcal{N} \tag{19}$$

$$z_{ijk} \in \{0,1\} \qquad (i,j) \in \mathcal{A}(k), \forall k \in \mathcal{K} \tag{20}$$

Constraints (12) ensure that each demand location is visited by exactly one city freighter. Constraints (13) to (15) guarantee that each city freighter starts and ends at the respective satellite the city freighter belongs to while ensuring the correct flow. Constraints (16) limit the load of each city freighter to the maximum capacity. Constraints (17) ensure the correct time-flow of each city freighter route, while Constraints (18) guarantee that each demand location is delivered before its due date. Similar to Constraints (9), Constraints (19) limit the demand sharing on the second tier.

Connection constraints

We connect the two tiers with the following constraints:

$$\sum_{r\in\mathcal{R}} x_{dsr} = \sum_{k\in\mathcal{K}(s)}\sum_{j\in\delta_k^+(d)} z_{djk} \qquad \forall d \in \mathcal{D}, s \in \mathcal{S} \tag{21}$$

$$\sum_{r\in\mathcal{R}}\sum_{s\in\mathcal{S}} \tau_{rs} \cdot x_{dsr} + w_r \le p_{s_k^+ k} + M_2 \cdot \Big(1 - \sum_{j\in\delta_k^+(d)} z_{djk}\Big) \quad \forall d \in \mathcal{D}, k \in \mathcal{K} \tag{22}$$

Constraints (21) state that each demand must be assigned to a city freighter that departs from the same satellite the demand is released to by the chosen first-tier service. Constraints (22) ensure that for each demand, the second-tier departure period at the satellite must be greater or equal to the first-tier arrival period plus service time. The incorporation of these constraints forms the basis for the synchronization of the two tiers at the operational level.

5. Solution approach

As our problem formulation combines two NP-hard problems, we propose a metaheuristic to address large instances. Therefore, we develop an Integrative Two-Step Large Neighborhood Search (I2S-LNS) with adaptive components that integrates decisions on both tiers. This design makes it possible to utilize the existing two-tier problem structure efficiently. We first describe the solution representation and search space in Section 5.1. Afterward, we illustrate the general framework of the metaheuristic in Section 5.2. In Section 5.3, we show how we construct an initial feasible solution. Sections 5.4 and 5.5 describe the two steps of the procedure.

5.1. Solution representation and search space

A solution $\mathcal{S}$ is represented by a set of selected services $\mathcal{R}_s$ (referred to as the service design), the assignment of demands to services and satellites $\mathcal{X}$, and the second-tier routing $\mathcal{Y}$ capturing the sequence of nodes including information on starting and departure times at each visited node. During the entire procedure, the search space is limited to the space of feasible solutions.

5.2. General outline of the I2S-LNS

The general procedure of the I2S-LNS is depicted in Algorithm 1. We start by constructing an initial feasible solution (Section 5.3). Then, our heuristic consists of iteratively applying a two-step procedure, where the First-Step Large Neighborhood Search (1S-LNS) applies operators on the services, followed by the Second-Step Adaptive Large Neighborhood Search (2S-ALNS), which applies operators on the demands. Further, we introduce a solution memory to intensify the search in promising search regions. This two-step design makes it possible to efficiently utilize problem specifics of both service network design on the first tier and vehicle routing on the second tier and to generate globally good solutions.

Precisely in the 1S-LNS, we destroy and repair our solution by applying operators on the service design $\mathcal{R}_s$ and adapting $\mathcal{X}$ and $\mathcal{Y}$ to ensure feasibility. Since it is unlikely to find new best solutions with a service design that has high service operating costs, denoted as $c^r(\mathcal{R}_s)$, we employ a threshold $(1+\theta_1)\cdot c^r(\mathcal{R}^*)$ where $c^r(\mathcal{R}^*)$ represents the cheapest service operating cost that are found so far during the search process. If the service operating cost of the current solution exceeds this threshold, we repeat Step 1. If this happens ϕ times in a row, we randomly select one of the stored solutions in our solution memory Γ and go back to this solution. This solution memory is constantly updated during the search process and includes the $|\Gamma|$ best solutions with distinct service design $\mathcal{R}_s$. It plays a crucial role in relocating the search process back to the regions that have proven to lead to high-quality solutions and thus helps to intensify the search in those regions. In the 2S-ALNS, we destroy and repair our solution by applying operators on the demands. Consequently, the demand assignment $\mathcal{X}$ and routing $\mathcal{Y}$ changes while keeping $\mathcal{R}_s$ fixed. In both steps, the best solution $\mathcal{S}^*$ is always updated when a new best solution is found. We stop after a minimum number of iterations without improving the objective value of the best found solution $f(\mathcal{S}^*)$ or after a time limit is reached.

Algorithm 1: I2S-LNS

$\mathcal{S}_c \leftarrow$ Generate a feasible initial solution; // Section 5.3
while *Stopping criterion not met* **do**
  $it \leftarrow 0$;
  repeat
    if $it > \phi$ **then**
      $\mathcal{S}_c \leftarrow$ drawRandomElement(Γ); // use solution memory
      break;
    $\mathcal{S}_c \leftarrow$ 1S-LNS($\mathcal{S}_c$); // Step 1: Section 5.4
    $it \leftarrow it + 1$;
  until $c^r(\mathcal{R}_s) < (1+\theta_1)\cdot c^r(\mathcal{R}^*)$;
  $\mathcal{S}_c \leftarrow$ 2S-ALNS($\mathcal{S}_c$); // Step 2: Section 5.5

Compared to standard large neighborhood searches from the literature, our approach integrates two large neighborhood searches within a single procedure. One large neighborhood search with operators related to the services and another adaptive large neighborhood search with operators related to the demands. These two are connected and interact with each other through an intelligent embedding of thresholds on both steps and a solution memory. Within these procedures, we have operators that exploit both service network design and vehicle routing problem specificities.

In contrast, most existing LNS methods, such as those in Breunig et al. (2016), Grangier et al. (2016), and Voigt et al. (2022), use a single-step procedure that applies all operators globally, even when targeting different aspects like satellite selection, routing, or location decisions. Hemmelmayr et al. (2012) distinguish between small- and large-impact operators in a two-echelon setting, but their approach still uses a unified LNS structure and applies large-impact changes only periodically after a certain number of iterations without improvement. So far, no existing method combines two coordinated LNS components that separately address service network design and vehicle routing, making our approach particularly well suited to the structure of two-tier city logistics systems.

5.3. Construction heuristic

We sequentially generate a feasible initial solution for the first tier and then for the second tier and ensure connection.

First tier: On the first tier, the solution consists of $\mathcal{R}_s$ and $\mathcal{X}$. In a first step, a service that complies with all the capacity restrictions (see Constraints (5) to (8)) is randomly selected. Afterward, demands that are not assigned to a service yet are randomly drawn one after the other and assigned to the selected service and assigned to the closest possible satellite the service visits, if constraints allow. We repeat this procedure until all demands are assigned. Assignments to satellites and services are also only permitted if it is ensured that there is still enough time for at least a commuting tour of a city freighter from the satellite to the final demand location to ensure feasibility on the second tier. This procedure is repeated η_1 times to generate η_1 different starting solutions. The solution with the lowest costs is taken as the initial solution for the first tier.

Second tier: Resulting from the first-tier assignments to satellites, we have a VRPRDD on each satellite. For each satellite, we greedily insert each demand in its cheapest possible position until all demands are served. This generates the initial solution for the routing $\mathcal{Y}$.

5.4. 1S-LNS

We start by destroying the current solution $\mathcal{S}_c$ by applying removal operators on the service design $\mathcal{R}_s$. As the solution gets infeasible when removing services because previously assigned demands are getting unassigned, we try to reassign the unassigned demands to other services in the existing service design in their cheapest possible position (in terms of demand assignment to services and satellites as well as routing). If not all demands can be reassigned into the existing service design, we insert new services using the insertion operators. We repeat this until all demands are assigned to a service, resulting in a fully repaired solution. Algorithm 2 depicts the complete logic of the First step LNS.

Algorithm 2: 1S-LNS

$\mathcal{S}_c \leftarrow removalOperator(\mathcal{S}_c)$;
$\mathcal{D}_u \leftarrow$ Demands that have been assigned to removed services;
for $d \in \mathcal{D}_u$ **do**
  if $bestInsertion(d, \mathcal{S}_c)$ **then**
    $\mathcal{D}_u \leftarrow \mathcal{D}_u \setminus \{d\}$;
while $\mathcal{D}_u \neq \emptyset$ **do**
  $\mathcal{S}_c \leftarrow insertionOperator(\mathcal{S}_c)$;
  for $d \in \mathcal{D}_u$ **do**
    if $bestInsertion(d, \mathcal{S}_c)$ **then**
      $\mathcal{D}_u \leftarrow \mathcal{D}_u \setminus \{d\}$;
$\mathcal{S}^* \leftarrow update(\mathcal{S}^*, \mathcal{S}_c)$; $\Gamma \leftarrow update(\Gamma, \mathcal{S}_c)$; $\mathcal{R}^* \leftarrow update(\mathcal{R}^*, \mathcal{S}_c)$;

For removing services, we use the following operators:

- **Random removal**: We randomly select between 1 and $\lambda_{\mathcal{R}}$ services and remove them.
- **Worst removal**: For each currently active service, we determine the ratio between the demand volume that is currently assigned

to the service and the service operating cost. Between 1 and $\lambda_{\mathcal{R}}$ services with the worst ratio are selected and closed.

To insert new services, we first determine all services that can still be inserted without violating any capacity constraints. Further, we determine if the service must be provided by a specific LSP to not violate the demand sharing Constraints (9). Then, we apply the following operators to this set of services.

- **Random insertion**: We randomly select a service and add it to the set of selected services $\mathcal{R}_s$.
- **Best-fit insertion**: We select the service with the best ratio of demand volume of the unassigned demands for which the service fulfills the release and the due date, and service operating cost c_r. In a preprocessing step before the heuristic starts, we determine for each demand and satellite combination the subset of services, that fulfill the release and due dates. Through this preprocessing step, we can easily evaluate this operator.

For continuous diversification purposes, we multiply the sorting criteria for the worst removal and the best-fit insertion operators by a random number in the interval [0.8, 1.2]. The selection of operators is based on the probabilities μ_r and μ_i, which determine the probability of utilizing the random operator for removal and insertion, respectively. The calibration of these probabilities is detailed in Section 7.2.1.

5.5. 2S-ALNS

In this step, we iteratively destroy and repair the solution of Step 1 by applying operators on the demands. Throughout this step, the service design $\mathcal{R}_s$ is fixed. Only the demand assignment $\mathcal{X}$ and routing $\mathcal{Y}$ are modified. In each iteration, we first choose a demand removal operator to destroy our current solution. After that, we repair our solution by applying an insertion operator. We only update the current solution if the objective value is not increased by more than a factor of $(1+\theta_w)$. We again take advantage of a threshold to prevent wasting computation time in unpromising search regions. Specifically, we break the ALNS if the objective value of the current solution $f(\mathcal{S}_c)$ exceeds a dynamic threshold that starts in the first iteration with $(1+\theta_2)\cdot f(\mathcal{S}^*)$ and decreases linearly with the number of iterations until it reaches $f(\mathcal{S}^*)$. We employ this logic since strong improvements in the current solution can be expected, especially during early iterations. We set a maximum number of iterations η_2 as stopping criteria. Algorithm 3 depicts the complete logic of the second step ALNS.

Algorithm 3: 2S-ALNS

$it \leftarrow 0$;
while $it < \eta_2$ **do**
  $\mathcal{S}'_c \leftarrow Repair(Destroy(\mathcal{S}_c))$ through applying operators on $\mathcal{D}$;
  $it \leftarrow it + 1$;
  if $f(\mathcal{S}'_c) < (1+\theta_w)\cdot f(\mathcal{S}_c)$ **then**
    $\mathcal{S}_c \leftarrow \mathcal{S}'_c$; // update current solution
  $\mathcal{S}^* \leftarrow update(\mathcal{S}^*, \mathcal{S}_c)$; $\Gamma \leftarrow update(\Gamma, \mathcal{S}_c)$;
  if $f(\mathcal{S}_c) > (1+\theta_2\cdot(1-it/\eta_2))\cdot f(\mathcal{S}^*)$ **then**
    break; // break if threshold is exceeded

We consider the following problem-specific operators to remove demands from the solution:

- **Random removal**: We randomly draw between 1 and λ_D demands.
- **CDC regret removal**: For each demand d, we determine whether the assignment to another CDC e would lead to lower costs f_{de}. We then sort the demands according to the potential cost savings and take between 1 and λ_D demands with the highest potential cost savings when assigned to another CDC.
- **Satellite regret removal**: For each demand, we determine the deviation between the distance to the satellite the demand is assigned to and the distance to the closest other satellite. We sort the demands increasing according to this deviation and select between 1 and λ_D demands with the highest deviation. The idea behind this operator is that demands that are not assigned to their closest satellite are more likely to lead to cost savings.
- **Random route removal**: A city freighter route is randomly selected and all the demands that are on this city freighter are removed.
- **Worst route removal**: For each city freighter route, we determine the ratio between the routing costs and the demand volume that is fulfilled by this city freighter. All demands of the city freighter route with the worst ratio are removed.

For each of the operators that are not random, we multiply the sorting criteria by a random number in the interval [0.8, 1.2] for better continuous diversification.

We consider the following insertion operator

- **Best insert**: We insert the demand where the costs, in terms of service assignment cost as well as second-tier routing insertion cost, are lowest.
- **2-regret insert**: For each demand, we determine the cheapest insertion cost for assigning the demand to each service and satellite combination. The regret value is then defined by the difference between the cheapest and the second cheapest service/satellite combination. Then we sort the demands in decreasing order by their regret value and insert them one after the other in their cheapest possible position in terms of service assignment cost as well as second-tier routing cost.

If, for insertion purposes, a new city freighter must start, we assign the city freighter to the LSP of the respective demand.

Operator selection Operators are chosen based on a roulette wheel mechanism, as described by Pisinger and Ropke (2007), utilizing the reaction factor ρ. Each pairing of removal and insertion operators is assigned a reward σ, allocated as follows: σ_1 for worsening the current objective value, σ_2 for maintaining the current objective value, σ_3 for improving the current objective value. Unlike many studies, we do not assign a reward for finding a new best global solution because such outcomes are significantly influenced by the current service design $\mathcal{R}_s$, which is not impacted by these operators.

6. Cost allocation in a collaborative 2T-CLS

After minimizing the total system costs, the question now arises of how to allocate these costs to the participating LSPs. In this section, we present the different cost allocation methods that we analyze in our numerical study. As there are currently no insights into the consequences of different cost allocation methods in city logistics, we include some easy-to-communicate proportional methods (Section 6.1), as well as two very distinct game-theoretical cost allocation methods that represent different schools of thought: the Shapley value (Section 6.2) and the EPM. While the Shapley value is based on marginality, the EPM is based on equality.

Given that each method captures different aspects of cost allocation, it is unclear how they will behave or whether they will lead to vastly different results. This is especially relevant in the context of 2T-CLSs, where a fully collaborative model captures cost savings across both tiers through pooling of demand, coordinated service selection, and temporal alignment. In such systems, especially smaller LSPs may benefit even more in relative terms, which underlines the need to investigate suitable cost allocation methods.

6.1. Proportional methods

The methods most commonly used in practice because of their simplicity are proportional methods. Proportional methods distribute costs in proportion to a predefined reference value (Vanovermeire et al., 2014). Although these methods are easy to calculate and interpret, they only take one aspect into account and leave out other key elements that influence the cost of each LSP.

In our setting, the number of demands assigned to each LSP (demand numbers) and the associated total volume of these demands are intuitive, problem-specific metrics for cost allocation. However, these proportional methods do not account for the cost structure of an LSP, including neither the stand-alone cost nor the marginal costs, which reflect the additional costs incurred when collaborating with others. To address this limitation, we propose a further proportional method based on the stand-alone cost. Specifically, we consider the following cost allocation approaches:

- **Demand-based Allocation (DA)**: The cost allocation to LSP n is proportional to the number of demands.

$$DA_n = \frac{|\mathcal{D}(n)|}{|\mathcal{D}|} \tag{23}$$

- **Volume-based Allocation (VA):** The cost allocation is proportional to the total volume of demand. In contrast to considering only the number of demands, this approach accounts for the cumulative demand volume assigned to each LSP, reflecting differences in demand magnitude.

$$VA_n = \frac{\sum_{d \in \mathcal{D}(n)} v_d}{\sum_{d \in D} v_d} \tag{24}$$

- **Demand- and Volume-based Allocation (DVA):** The DVA is calculated as a weighted sum (weight ω) of the DA and VA where $0 \leq \omega \leq 1$.

$$DVA_n = \omega \cdot DA_n + (1-\omega) \cdot VA_n \tag{25}$$

- **Stand-Alone-based Allocation (SAA)**: The cost allocation is determined by the ratio of the stand-alone costs of each LSP $C(\{n\})$ when no collaboration occurs and the cumulated stand-alone cost of all LSPs. In contrast to the other proportional methods, however, it is necessary to calculate the costs that incur for each LSP when acting independently.

$$SAA_n = \frac{C(\{n\})}{\sum_{\hat{n} \in \mathcal{N}} C(\{\hat{n}\})} \tag{26}$$

Each of the four cost allocations is interpreted as a share of the total costs that LSP n must bear.

6.2. Shapley value

Shapley (1953) introduced the Shapley Value as a cost allocation method in cooperative game theory. The Shapley Value is calculated as the average marginal contribution that the participant makes to each possible subcoalition when added to the coalition according to the order. It provides a unique cost allocation solution that satisfies a lot of fairness axioms. For details about the formulation of these axioms, we refer to Shapley (1953). Analytically, the Shapley Value for a participant n is calculated as follows:

$$SV_n = \sum_{\mathcal{O} \subset \mathcal{N} : n \in \mathcal{O}} \frac{(|\mathcal{O}|-1)!(|\mathcal{N}|-|\mathcal{O}|)!}{|\mathcal{N}|!} \cdot [C(\mathcal{O}) - C(\mathcal{O} - \{n\})] \tag{27}$$

Thereby, the summation is over all subcoalitions $\mathcal{O}$ within the set of all participants $\mathcal{N}$ that contain participant n whereby $[C(\mathcal{O}) - C(\mathcal{O} - \{n\})]$ represents the participants marginal contribution to the subcoalition $\mathcal{O}$. The Shapley Value is therefore a marginal cost-based method that does not emphasize an equal distribution of costs but derives its fairness aspect from marginal costs. In the context of city logistics, the marginal cost perspective is particularly relevant because it highlights how smaller LSPs with high stand-alone costs can benefit disproportionately within a coalition. By sharing resources, these smaller LSPs achieve significant gains in utilization and efficiency, whereas larger LSPs, already operating with higher utilization and greater efficiency, see comparatively smaller marginal improvements. The Shapley Value thus captures not only individual cost burdens but also the unique collaborative benefits that may not align with proportional allocations, making it particularly well-suited for evaluating cost-sharing among LSPs with diverse operational scales.

6.3. Equal profit method

The EPM was developed by Frisk et al. (2010) and is based on the idea that the maximum difference of the pairwise relative savings should be minimized. Thus, the relative savings of the participants should be as similar as possible. With decision variable c_n representing the costs allocated to participant n and $C(\{n\})$ representing the stand-alone costs of participant n the EPM allocates the costs through the following linear model:

$$\min \quad f \tag{28}$$

$$\text{s.t.} \quad f \geq \frac{c_n}{C(\{n\})} - \frac{c_{\hat{n}}}{C(\{\hat{n}\})} \qquad \forall n, \hat{n} \in \mathcal{N} : n \neq \hat{n} \tag{29}$$

$$\sum_{n \in \mathcal{O}} c_n \leq C(\mathcal{O}) \qquad \forall \mathcal{O} \subset \mathcal{N} \tag{30}$$

$$\sum_{n \in \mathcal{N}} c_n = C(\mathcal{N}) \tag{31}$$

The objective function (28) minimizes the maximum pairwise difference in relative savings that is measured by Constraints (29). Constraints (30) and (31) ensure the rationality and the efficiency condition. The constraints ensure a core allocation if a cost allocation that lies in the core exists. In the case that the core is empty, we use the epsilon-core as proposed by Frisk et al. (2010) to keep the coalition stable.

7. Numerical study

Section 7.1 describes the numerical setup and the generation of instances. Then, we calibrate the parameters and evaluate the performance of the I2S-LNS in Section 7.2. Section 7.3 presents managerial insights regarding the impact of collaboration as well as about the cost allocation methods.

7.1. Numerical setup and instance generation

We implemented the MILP in Python 3.12 using Gurobi 11 as a solver. The I2S-LNS is implemented in C++. All experiments are carried out on an AMD Ryzen 9 5950X 16-Core Processor, 3.40 GHz with 128 GB RAM. We generate new specific instances for this problem. Similar to related papers in the field of 2T-CLSs, we generate the instances based on a real city (Crainic et al., 2004; Fontaine et al., 2021). In our case, we take a large German city as a basis. We locate CDCs in easily accessible places outside the city. For satellites, we mainly use tram stops or larger squares in the city center. We go one step further than other papers and sample demand destinations for all instances in relation to publicly available data on population density in different city districts. We use two different networks, hereinafter called N1 and N2. N1 has two CDCs and four satellites, N2 has three CDCs and six satellites. We consider a planning horizon of 36 periods, ten minutes each. We generate a set of services with different numbers of satellites starting and ending at a CDC. The first start period is chosen randomly between periods five and ten. Each drawn service is then replicated two times during the planning horizon, with eight periods in between.

We consider four different urban vehicle types: small tram, large tram, small truck, and large truck, with capacities of 500 for small vehicles and 750 for large vehicles. The cost of the services c_r depends on the vehicle type and the travel distance. We set the fixed cost for small vehicles to 15 and for large vehicles to 20. The variable costs are the travel distance multiplied by 1.2 for small trams, 1.5 for small trucks, 1.7 for large trams, and 2.0 for large trucks. The lower cost of trams is justified by their lower environmental impact cost. Further, the travel speed for trams is set to 25 km/h and for trucks to 20 km/h because of traffic. The capacity of city freighters on the second tier is 250, and the travel speed is 20 km/h. A service time of one period is assumed. In each instance, each LSP individually draws random services out of the generated services, with half of these services being operated by large urban vehicles. Demand volumes are uniformly distributed between 50 and 100. For each demand, the release date RD is randomly selected from the range 1 to 18. The due date DD is calculated using the formula $DD = RD + 12 + \text{randint}(0, 6)$ to ensure time feasibility. For each demand / CDC combination, we assume random assignment cost (f_{de}) ranging from one to five. For capacity constraints, we assume that each LSP is allowed to operate one vehicle at a time (a_{spn}) at each satellite, implying $a_{spnm} \leq 1$ for all modes, and use 300 units of the satellite capacity (g_{spn}). With this setting, we generate instances of different sizes with up to three LSPs, 100 demands, and 60 services for our numerical experiments in the following sections.

7.2. I2S-LNS performance

We start with calibrating the parameter setting of the I2S-LNS (Section 7.2.1). No benchmark instances in the literature exist integrating service network design on the first tier and vehicle routing on the second tier. Further, comparing to pure service network design problems or pure two-echelon vehicle routing problems does not consider essential parts of our problem setting. Therefore, we evaluate the performance of the I2S-LNS through three steps: First, by comparing it against Gurobi on small instances (Section 7.2.2), second, by analyzing its performance on larger instances (Section 7.2.3) and benchmarking against a single-step version of our heuristic with the same parameters and a solution memory, lastly, by measuring the performance impact of removing newly introduced components of the I2S-LNS (Section 7.2.4). Throughout all performance experiments, we assume two LSPs that are fully allowed to share their customers ($\alpha_1 = \alpha_2 = 0$). Demands and services are evenly split across these two LSPs.

7.2.1. Calibration

To calibrate the parameter setting, we conducted tests on medium-sized instances with 40 and 60 demands. Initial values for each parameter were determined based on preliminary tests conducted throughout the development phase, ensuring a realistic starting point. Subsequently, we systematically varied each parameter within a specified range while holding all others constant to identify the setting that yielded the best average performance. The finalized parameter settings that are used throughout our experiments are presented in Table 1.

7.2.2. Benchmark against Gurobi

We compare the performance of the I2S-LNS with the solutions obtained by solving the MILP using Gurobi. A time limit of one hour is imposed on Gurobi, whereas the I2S-LNS is set to terminate after 1.000 iterations (full iterations of Step 1 and Step 2) without improvement or once a time limit of ten minutes is reached. We execute the I2S-LNS five times for each instance. We conduct two experiments for this purpose. In the first experiment, we increase the number of demands ($|\mathcal{D}|$) while keeping the number of services constant ($|\mathcal{R}| = 24$). In the second experiment, the number of services ($|\mathcal{R}|$) is increased while the number of demands ($|\mathcal{D}| = 15$) is kept constant. To exclude the influence of the demands on the complexity, corresponding instances share identical demand sets. For example, the first instance with $|\mathcal{R}| = 12$ has the same demands as the first instance with $|\mathcal{R}| = 36$, and this pattern is maintained across all instances. For the first experiment, we use both networks. For the second experiment, we only use N2 to generate a wider range of services. The aggregated results for these experiments are shown in Tables 2 and 3 (detailed results for all benchmarks are presented in Appendix B). We report the best and the average objective values as well as the percentage deviation ($\sigma[\%]$) between the best and average. Further, we report the percentage difference between the objective value obtained by Gurobi and the average objective value of the I2S-LNS ($\Delta[\%]$) and the corresponding runtime.

Table 1
I2S-LNS parameter setting.

Parameter	Value	Description
η_1	100	Number of iterations of first-tier construction heuristic
η_2	$1.1 \cdot \lvert\mathcal{D}\rvert$	Number of iterations in Step 2
θ_1	0.25	Parameter for threshold in Step 1
θ_2	0.35	Parameter for threshold in Step 2
θ_W	0.01	Parameter for threshold for worsening the objective value
λ_D	6	Max. number of removed demands
λ_R	3	Max. number of removed services
μ_r	0.2	Probability for random removal operator
μ_i	0.4	Probability for random insertion operator
ϕ	5	Max. number of threshold violations in Step 1
$\lvert\Gamma\rvert$	5	Length of solution memory
ρ	0.1	ALNS reaction factor
σ_1, σ_2, σ_3	{1,3,5}	ALNS scores for updating weights

In the first experiment, we observe that Gurobi is only able to prove optimality for very small instances with $|\mathcal{D}| = 5$ and $|\mathcal{D}| = 10$. For larger instances, Gurobi exhibits large gaps of up to 33.91% on average for instances with $|\mathcal{D}| = 30$. In instances with $|\mathcal{D}| = 40$, Gurobi fails to obtain a feasible solution within the time limit. Our metaheuristic significantly outperforms Gurobi by finding exactly the same solutions for very small instances where Gurobi finds the optimal solution and better solutions for larger instances in less computation time. Notably, for larger instances, the performance of our metaheuristic remains very stable, with mean σ significantly below 1%.

In the second experiment, we observe a slight increase in the gaps shown by Gurobi as the number of services rises. Notably, the difference between the I2S-LNS and Gurobi also increases slightly as the number of services increases. In all instances, the I2S-LNS delivers equal or better solutions than Gurobi in a much shorter computation time. Our metaheuristic also shows remarkable stability, with σ equals 0 in most instances, reflecting minimal variation between the best and average results. The computation time is only slightly influenced by the increased number of services. This analysis highlights the number of demands as the key complexity driver in our problem setting.

7.2.3. Performance on larger instances

In Table 4, we assess the heuristics stability on larger instances, creating five instances each for three configurations: $\mathcal{D} = 50, \mathcal{R} = 36$; $\mathcal{D} = 75, \mathcal{R} = 48$; and $\mathcal{D} = 100, \mathcal{R} = 60$. In addition, we benchmark against a single-step version of our metaheuristic with the same operators and a solution memory embedded. This version of the heuristic is explained in pseudocode in Appendix C. We set a time limit of 30 min, which is the only stopping criterion for the single-step version whereas the two-step version has the stopping criterion described above. Since defining a comparable dynamic stopping criterion for the single-step version could introduce bias, we fixed its runtime at 1800 s, matching the maximum runtime of the I2S-LNS. Thus, the comparison is made with the single-step version always running for 1800 s, while the I2S-LNS stops earlier if its dynamic stopping criterion is met.

We observe that even for larger instances, our heuristic delivers very stable results with σ remaining below 1% in almost all of the instances. It also shows that the single-step version leads to significantly worse results with an average of 1.82% higher cost. This clearly shows that

Table 2
Aggregated comparison of Gurobi and the I2S-LNS with varying $|D|$ and constant $|R| = 24$.

Network	$\|D\|$	Gurobi			I2S-LNS				Δ[%]
		Costs	GAP [%]	Time [s]	Best costs	Avg. costs	σ [%]	Avg. time [s]	
N1	5	108.85	0.0	2	108.85	108.85	0.0	17	0.0
N1	10	169.63	1.17	1731	169.63	169.63	0.0	37	0.0
N1	15	234.33	24.95	3600	226.21	226.55	0.14	107	−3.01
N1	20	285.50	24.22	3600	275.30	275.81	0.19	180	−3.29
N2	30	412.80	33.91	3600	371.58	373.68	0.55	405	−9.32
N2	40	–	–	3600	486.65	490.13	0.65	544	–

Table 3
Aggregated comparison of Gurobi and the I2S-LNS with varying $|R|$ and constant $|D| = 15$.

$\|R\|$	Gurobi			I2S-LNS				Δ [%]
	Costs	GAP [%]	Time [s]	Best costs	Avg. costs	σ [%]	Avg. time [s]	
12	230.01	15.21	3600	226.62	226.62	0.00	93	−1.49
36	222.71	20.56	3600	215.45	215.46	0.00	104	−2.88
60	205.19	24.58	3600	198.87	198.91	0.02	109	−2.98

Table 4
Aggregated performance benchmark against a single-step version on larger instances.

$\|D\|$	$\|R\|$	I2S-LNS				Single-step version			Δ [%]
		Best costs	Avg. costs	σ [%]	Avg. time [s]	Best costs	Avg. costs	σ [%]	
50	36	587.16	590.48	0.57	1268	597.33	604.07	1.20	1.76
75	48	877.74	884.02	0.71	1778	893.89	905.20	1.02	1.75
100	60	1105.47	1113.75	0.75	1800	1126.56	1142.34	0.99	1.96

Table 5
Increase in objective value when taking out special components.

$\|D\|$	Threshold θ_1 and memory Γ	Threshold θ_2
50	2.66%	0.25%
75	2.65%	0.79%
100	3.17%	0.75%

our two-step version makes better use of the existing problem structure than standard single-step large neighborhood searches.

7.2.4. Impact of special components

In this section, we assess the performance impact of special components of the I2S-LNS, specifically the thresholds and the solution memory. Table 5 presents the average percentage increase in the objective value when removing the special components. Note that when leaving out threshold θ_1, we also do not take advantage of the solution memory Γ. We use the previously introduced larger instances with 50 to 100 demands.

As can be clearly seen, the inclusion of threshold θ_1 and, thus, the inclusion of the solution memory Γ dramatically boosts the performance of our heuristic among all instance sizes. Also, the inclusion of the threshold θ_2 significantly contributes to a better performance of our heuristic. These results show that the thresholds in this two-step procedure are enhancing the performance of the I2S-LNS.

7.3. Managerial insights

Firstly, we analyze the impact of varying the collaboration intensity through modifying the α−parameters. Subsequently, we compare and analyze the cost allocation methods that were previously introduced. For these investigations, we generate ten instances each with 48 demands distributed across three LSPs, each offering 21 services. We take three cases into account, with each case being a modification of the previous.

- In **Case 1**, demands are evenly split among the LSPs, meaning each LSP has 16 demands.
- In **Case 2**, we adjust Case 1 by changing the allocation of demands to LSPs; one LSP receives eight demands, another 16, and the last one 24 demands.
- In **Case 3**, we further modify Case 2 by also varying the demand volume. The demand volume for the LSP with 16 demands is multiplied by 0.75, and for the LSP with eight demands by 0.5, to assess the effects of altering both the number and the volume of demands.

Thus, we have in Case 1 equally sized LSPs, in Case 2 LSPs that differ by the number of demands, and in Case 3 LSPs that differ by the number of demands as well as by the demand volume. For all experiments LSP 1 refers to the smallest, LSP 2 to the medium and LSP 3 to the largest LSP.

7.3.1. Impact of varying the collaboration intensity

In this section, we examine the effects of the collaboration intensity constraints. Starting from a fully collaborative system in which both resources and demands are fully shared ($\alpha_1 = \alpha_2 = 0$), we gradually increase the α parameters to 1 (no demand sharing). First, we increase α_1 and α_2 both at the same time (Fig. 2). Then, we increase them individually while holding the other α parameter constant at 0 (Fig. 3).

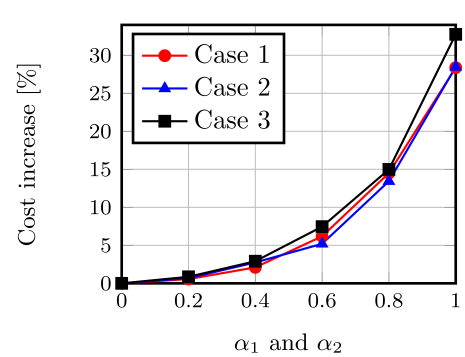

Fig. 2. Sensitivity analysis for increasing α_1 and α_2.

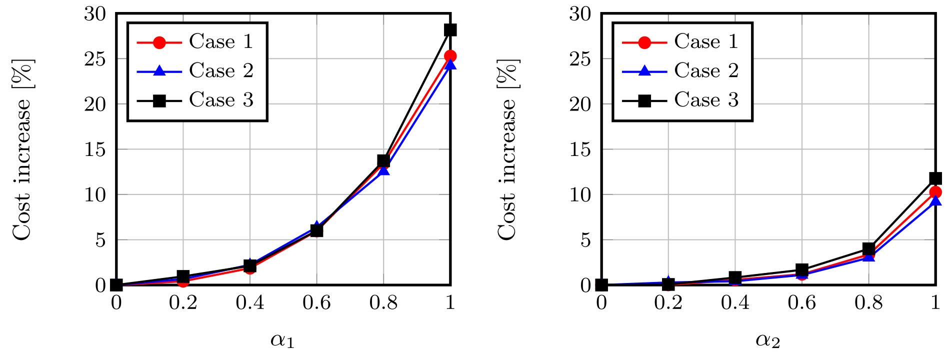

Fig. 3. Sensitivity analysis for increasing α_1 and α_2 individually.

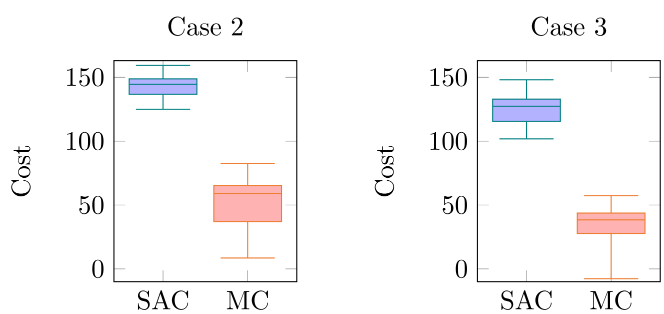

Fig. 4. Distribution of stand-alone cost and marginal cost for LSP 1.

We observe that increasing the α−parameters to 1 leads to significant increases in total system costs across all cases. On average, total cost increases by 29.25% when there is no demand sharing compared to total demand sharing. Additionally, we find that sharing even 20% of demands ($\alpha = 0.8$) can lead to a large portion of the potential cost savings achievable through full demand sharing. This is especially true for collaboration on the second tier, as nearly the whole increase in cost is happening for $\alpha_2 > 0.8$. This is attributed to the fact that LSPs use identical city freighters, eliminating structural cost differences among these LSPs. Further, we identify that sharing demands in the first tier leads to a higher cost impact than sharing demands in the second tier. This is due to the fact that first-tier collaboration impacts the assignment to CDCs, the utilization of the services, as well as the assignment to the satellites and thus the starting point for the second-tier routing. Moreover, the larger capacity of first-tier urban vehicles enhances the demand pooling potential across different LSPs. These effects result in a very interesting insight for LSPs, as a very large part of the potential cost savings is possible through collaboration without many of the customers even realizing this, as the final delivery to their homes is carried out by the LSPs they have engaged. This is demonstrated by the fact that almost the full amount of potential cost savings can be realized through full collaboration on the first tier and only the sharing of a small part of the demands on the second tier.

In addition to the cost increase compared to full collaboration, we further analyze the effects of demand sharing on several KPIs. These include the utilization of first-tier urban vehicles, the utilization of city freighters, the share of provided capacity by selected services operated by large vehicles (large trams and large trucks), and the cost increases at the individual tiers. The first-tier costs are calculated as $\sum_{r\in\mathcal{R}} c_r \cdot y_r + \sum_{d\in D}\sum_{s\in S}\sum_{r\in\mathcal{R}} f_{de_r} \cdot x_{dsr}$, while the second-tier costs are represented by $\sum_{k\in\mathcal{K}}\sum_{(i,j)\in\mathcal{A}(k)} \hat{c}_{i,j} \cdot z_{ijk}$. The results are presented in Table 6.

We observe the following effects: (1) Collaboration, in general, increases vehicle utilization and leads to a higher share of larger urban vehicles being employed. (2) First-tier costs are almost exclusively influenced by collaboration at the first tier, while second-tier costs are affected by collaboration at both tiers and fall particularly sharply when collaboration takes place at both the first and second tiers. (3) Full collaboration leads to disproportionately high cost savings on the second tier. (4) Collaboration restricted to the first tier improves vehicle utilization at this tier but reduces it slightly on the second tier. This effect is due to pooling demands of different LSPs into a single service. Consequently, diverse demands from these LSPs reach the satellites leading to reduced utilization on the second tier as each LSP is required to operate its own city freighters independently. Nevertheless, even though the utilization is lower, it still leads to cost savings on the second tier through more efficient route planning through shorter traveled distances.

These results clearly highlight the interaction effects between the two tiers, where, for example, collaboration on the first tier also impacts operations on the second tier, demonstrating that demand sharing not only generates monetary savings for the LSPs, but also significantly reduces the environmental impact on each tier.

7.3.2. Cost allocation

In this section, we examine the effects of the different cost allocation methods. In particular, we analyze the consequences for LSPs of different sizes when applying the distinct methods, and to investigate whether there are differing interests in the selection of the cost allocation method depending on LSPs size. This is the first study that compares and evaluates different cost allocation methods in the context of city logistics, providing important insights for future collaborative projects. For this analysis, we evaluate the characteristic function of each instance by running the heuristic for each possible sub-coalition of LSPs. We then apply the previously introduced cost allocation methods to the individual instances. We use $\omega = 0.5$ as the weight for the DVA.

Our analysis reveals that by fully sharing resources and demands, the coalition could achieve cost savings of on average 26.91% by comparing the total stand-alone costs of all LSPs to the costs of the entire coalition across all cases. To compare the different cost allocation methods, Table 7 shows the average percentage cost savings of each LSP compared to its stand-alone cost using the cost allocation methods. Thereby, LSP 1 stands for the smallest, LSP 2 for the medium, and LSP 3 for the largest LSP.

Based on the results, we identify the following effects for the three cases:

Table 6
KPIs for different collaboration scenarios.

KPI	Full coll.	First tier coll. only	Second tier coll. only	No coll.
Util. first tier vehicles [%]	89.07	89.05	76.13	77.90
Util. second tier vehicles [%]	82.42	69.39	79.07	72.77
Share of large vehicles [%]	60.14	61.14	47.09	43.61
Cost increase [%]	–	10.40	25.89	29.87
Cost increase first tier [%]	–	1.44	25.06	24.25
Cost increase second tier [%]	–	24.16	28.20	40.19

Table 7
Average cost savings [%] compared to stand-alone cost.

Case	LSP	SV	EPM	DA	VA	DVA	SAA
Case 1	1	26.82	26.32	24.59	25.87	25.23	26.32
	2	25.71	26.32	26.96	26.44	26.70	26.32
	3	26.54	26.32	26.98	26.34	26.66	26.32
Case 2	1	41.29	26.36	36.69	38.62	37.65	26.36
	2	25.24	26.36	24.36	24.07	24.21	26.36
	3	21.11	26.36	23.25	25.51	24.38	26.36
Case 3	1	47.31	28.08	33.59	61.77	47.68	28.08
	2	27.84	28.08	21.78	29.77	25.78	28.08
	3	21.41	28.08	29.51	14.82	22.17	28.08

- In Case 1, the average cost savings do not differ significantly neither between the LSPs nor between the individual cost allocation methods. This is not surprising as the LSPs do not differ significantly as they have exactly the same number of demands and services drawn from the same population. Differences at the level of the individual instances balance each other out on average.
- In Case 2, we find that LSP 1 is particularly favored by the use of the demand-based proportional methods and the Shapley Value, as the relative cost savings are significantly higher when using this method than when using the other methods. In contrast, it is better for the two larger LSPs to allocate costs using methods that generate similar relative cost savings. This is due to the fact that larger LSPs usually already have higher utilization of services because of their high number of demands. Demand-based proportional cost allocation methods do not take this effect into account, which leads to lower relative cost savings of larger LSPs compared to smaller LSPs when using these allocation methods.
- In Case 3, we observe similar effects. LSP 1 benefits in particular from the use of the Shapley Value as well as from the use of the VA, as this takes into account both the lower number of demands and the lower demand volume. LSP 3 again benefits, in particular, from the EPM and the SAA, but this time also from the DA, as this method does not take into account the higher demand volume compared to the other LSPs.

Notably, we identify that in every instance, the SAA and EPM deliver the same results. This is due to the fact, that allocating costs according to the SAA method already satisfies the additional constraints that are included in the EPM method.

Further, the experiments show that there are diverging interests of the LSPs with regard to the selection of a cost allocation method based on their size. Simple proportional methods may seem plausible and easy to calculate but can lead to very different relative cost savings among the LSPs. In particular, it would be difficult to agree on a fair reference value for the proportional methods, as this leads to immense differences in cost allocation. With respect to the two game theoretical allocation methods, it is particularly desirable for large LSPs to use the EPM or the SAA, as they lead to the same relative cost savings among the LSPs. For the smallest LSP, the allocation according to the Shapley Value is advantageous, as it takes into account the comparatively low marginal costs once the smallest LSP joins a coalition. These results show the diverging interests of the individual LSPs in relation to the selection of a fair allocation method.

Detailed analysis further reveals the following effects:

- In Case 3, the VA results in cost allocations that seem to be unfair for the two larger LSPs. Specifically, in three out of ten instances, the costs allocated to the two larger LSPs exceed the costs of the subcoalition of these two LSPs. In Case 2, we observe a similar effect in the allocation according to the DA. In several cases, this leads to an allocation where the two largest LSPs have almost no cost savings compared to forming a joint subcoalition. This suggests that simple proportional allocation methods can potentially lead to cost allocations that do not allow for stable coalitions, as some subcoalitions are more profitable than the coalition of all LSPs.
- Although the average cost allocations in Case 1 are quite similar, significant differences in individual instances exist, especially between the game theoretical methods and the proportional methods (except SAA). This discrepancy arises because the proportional methods do not account for variations in costs, which are influenced by the location of demands, the composition of the services, and the release and due dates for these demands.

Another important insight that we identify is that the smallest LSP (LSP 1) only incurs a very small marginal cost when added to the coalition of LSP 2 and LSP 3. Fig. 4 displays both the stand-alone costs (SAC) of LSP 1 when acting independently and the marginal costs (MC) of adding LSP 1 to the coalition throughout the ten instances for Case 2 and 3. As shown, the marginal costs are by far lower compared to the stand-alone costs. Remarkably, in one instance in Case 3, the marginal costs are even negative. This is attributed to the fact that LSP 1 contributes only a few low-volume demands to the coalition but offers as much capacity at the satellites and provides as many services as the other LSPs. These counteracting cost effects result in very low marginal costs. This clearly indicates that incorporating smaller LSPs, which also have transport capacities, can be highly beneficial for the coalition overall.

8. Conclusion and further research

This study presented a decision model for collaboration in 2T-CLSs, which integrates a service network design on the first tier and a vehicle routing problem on the second tier. Additionally, we proposed an efficient problem-specific metaheuristic that is based on a two-step procedure consisting of a large neighborhood search for the service design and an adaptive large neighborhood search for the demand assignment and routing. Furthermore, various cost allocation methods were applied for this problem setting.

Through an extensive numerical study, we demonstrate the performance of our solution approach. Further, we could show that the collaboration among LSPs can result in significant cost savings. Even when only a small percentage of the demands is shared among the coalition, a large part of the potential cost savings through collaboration can be achieved. Further, collaboration contributes to increased vehicle utilization on both tiers and thereby contributes towards a more sustainable city logistics system. What is particularly important for LSPs is that we were able to show that almost all potential cost savings through full collaboration can be realized through full collaboration on the first tier and the sharing of only a small share of the demands on the second tier. Regarding the cost allocation methods, our experiments showed diverging interests of the LSPs. While for larger LSPs, a cost allocation based on methods that lead to similar relative cost savings is advantageous, for smaller LSPs, the Shapley Value is advantageous because it pays particular attention to the very low marginal costs. Simple proportional methods based on demand characteristics such as the number or the volume of demand also lead to very different cost savings for LSPs depending on their demand characteristic. Within a coalition, it may be difficult to agree on a simple proportional method, as depending on the choice of the reference value, some LSPs will be advantaged and others disadvantaged.

For future research, it could be exciting to investigate other aspects of collaboration, such as risk sharing, which could lead to the development of a more resilient collaborative city logistics system. With regard to the strategic level, the collaborative selection and location of sites for satellites and CDCs could offer scope for further research.

CRediT authorship contribution statement

Johannes Gückel: Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Methodology, Investigation, Formal analysis, Data curation. **Teodor Gabriel Crainic:** Writing – review & editing, Validation, Supervision, Resources, Project administration, Methodology, Funding acquisition, Formal analysis, Conceptualization. **Pirmin Fontaine:** Writing – review & editing, Validation, Supervision, Resources, Project administration, Methodology, Investigation, Funding acquisition, Formal analysis, Conceptualization.

Acknowledgments

We thank the anonymous reviewers for their valuable recommendations, which have significantly improved our paper. The paper was

Table A.8
Sets, parameters and decision variables.

Sets and subsets	
$\mathcal{A}$	Set of arcs
$\mathcal{A}(k)$	Set of arcs city freighter k can use
$\mathcal{D}$	Set of demands
$\mathcal{D}(n)$	Set of demands assigned to LSP n
$\delta_k^-(i)$	Set of arcs for city freighter k that start in node i
$\delta_k^-(i)$	Set of arcs for city freighter k that end in node i
$\mathcal{E}$	Set of CDCs
$\mathcal{K}$	Set of city freighters
$\mathcal{K}(n)$	Set of city freighters of LSP n
$\mathcal{K}(s)$	Set of city freighters available at satellite s
$\mathcal{M}$	Set of modes of transportation
$\mathcal{N}$	Set of all LSPs
P	Set of periods
$\mathcal{R}$	Set of services
$\mathcal{R}(d,s)$	Set of services fulfilling time windows for demand d and satellite s
$\mathcal{R}(n)$	Set of services that LSP n operates
$\mathcal{R}(p,e,t)$	Set of services of type t, starting from CDC e and operating during period p
$\mathcal{R}(p,s)$	Set of services operating at satellite s during period p
$\mathcal{R}(p,s,m)$	Set of services of mode m operating at satellite s during period p
$\mathcal{S}$	Set of satellite platforms
$\mathcal{T}$	Set of urban vehicle types
$\mathcal{T}(m)$	Set of urban vehicle types for mode m
Parameters	
a_{spn}	Maximum number of urban vehicles that LSP n is allowed to accommodate at satellite s in period p
$\bar{a}_{spmn}$	Maximum number of urban vehicles of mode m that LSP n is allowed to accommodate at satellite s in period p
b_d	Latest possible delivery time period for demand d
c_r	Operating cost of service r
$\hat{c}_{ij}$	Cost of city freighter using arc (i,j)
e_r	CDC of urban vehicle service r
f_{de}	Assignment cost of demand d to CDC e
g_{spn}	Total volume of freight that LSP n is allowed to accommodate at satellite s in period p
h_{etn}	Fleet size of urban vehicles at CDC e of type t owned by LSP n
l	Service time of city freighters at each demand location
M_1	Big M 1 for Constraints (17)
M_2	Big M 2 for Constraints (22)
m_r	Transportation mode of service r
q	Capacity of city freighters
ρ	Handling time for each demand at each satellite
s_d	Service time at demand location d.
s_k^+	Satellite at which city freighter k starts
s_k^-	Satellite at which city freighter k ends
$\bar{t}_{ij}$	Second-tier travel time between node i and node j
t_r	Urban vehicle type of service r
τ_{rs}	Arrival time of service r at satellite s
u_t	Urban vehicle capacity of type t
v_d	Volume of demand d
w_r	Service time of service r at each satellite
Decision variables	
p_{ik}	Variable specifying the start of service time at vertex i serviced by city freighter k
x_{dsr}	Taking the value one if demand d is assigned to satellite s and service r, zero otherwise
y_r	Taking the value one if service r is selected, zero otherwise
z_{ijk}	Taking the value one if city freighter k uses arc $(i,j) \in \mathcal{A}$, zero otherwise

Table B.9
Comparison of Gurobi and the I2S-LNS with varying $|D|$ and constant $|R| = 24$.

Instance[a]	Costs	GAP [%]	Time [s]	Best costs	Avg. costs	σ [%]	Avg. time [s]	Δ[%]
N1-D5-1	94.55	0.00	1.25	94.55	94.55	0.00	10.24	0.00
N1-D5-2	77.15	0.00	0.50	77.15	77.15	0.00	17.19	0.00
N1-D5-3	121.71	0.00	4.97	121.71	121.71	0.00	19.56	0.00
N1-D5-4	123.86	0.00	2.09	123.86	123.86	0.00	17.95	0.00
N1-D5-5	126.97	0.00	1.41	126.97	126.97	0.00	18.95	0.00
N1-D10-1	173.41	0.00	209.70	173.41	173.41	0.00	48.73	0.00
N1-D10-2	177.29	4.92	3600	177.29	177.29	0.00	40.54	0.00
N1-D10-3	164.92	0.00	1045.30	164.92	164.92	0.00	31.87	0.00
N1-D10-4	166.96	0.93	3600	166.96	166.96	0.00	29.93	0.00
N1-D10-5	165.58	0.00	197.92	165.58	165.58	0.00	35.57	0.00
N1-D15-1	233.22	19.39	3600	228.43	228.96	0.23	102.20	−1.83
N1-D15-2	219.92	16.49	3600	211.45	211.67	0.10	111.50	−3.30
N1-D15-3	241.12	23.00	3600	228.08	228.08	0.00	144.20	−5.41
N1-D15-4	203.75	24.95	3600	203.75	203.75	0.00	94.96	0.00
N1-D15-5	273.66	40.92	3600	259.32	260.30	0.38	82.57	−4.52
N1-D20-1	307.00	32.73	3600	299.35	300.43	0.36	187.74	−2.15
N1-D20-2	268.18	16.76	3600	264.77	264.84	0.03	210.58	−1.25
N1-D20-3	306.51	24.92	3600	276.44	276.44	0.00	218.21	−9.80
N1-D20-4	258.46	28.23	3600	248.62	250.02	0.56	158.62	−3.26
N1-D20-5	287.34	18.48	3600	287.34	287.34	0.00	125.33	0.00
N2-D30-1	396.12	41.11	3600	342.06	343.05	0.29	273.63	−13.41
N2-D30-2	388.48	30.47	3600	382.67	386.09	0.89	486.02	−0.61
N2-D30-3	429.20	31.15	3600	386.41	390.57	1.08	297.51	−8.98
N2-D30-4	406.94	28.96	3600	370.28	371.59	0.35	500.25	−8.69
N2-D30-5	443.28	37.86	3600	376.48	377.10	0.16	467.68	−14.93
N2-D40-1	–	–	3600	487.97	491.03	0.75	600.00	–
N2-D40-2	–	–	3600	484.37	485.43	0.22	592.30	–
N2-D40-3	–	–	3600	492.01	499.20	1.46	573.82	–
N2-D40-4	–	–	3600	488.69	489.23	0.11	513.37	–
N2-D40-5	–	–	3600	480.22	485.74	0.73	442.63	–

[a] N1-D5-1 refers to the first out of five instances with network N1 and $|D| = 5$.

written as part of a project funded by the German Research Foundation. We would like to express our gratitude to the German Research Foundation (project number: ROCOCO FO1468/1-1) for the financial support of our research project. Additionally, our gratitude goes to the Bavarian Research Alliance for their support of the research stay in Montreal, which contributed significantly to the development of this paper. While working on this paper, the second author was Adjunct Professor, Department of Computer Science and Operations Research, Université de Montréal, Canada. We gratefully acknowledge the financial support provided by the Canadian Natural Sciences and Engineering Research Council (NSERC) through its Discovery grant program, as well as by the Fonds de recherche du Québec through their Teams and CIRRELT infrastructure grants.

Appendix A. MILP notation

See Table A.8.

Appendix B. Performance benchmark

See Tables B.9–B.11.

Appendix C. Single-step version

The single-step version of the heuristic is outlined in pseudocode in Algorithm 4. First, a continuous random number p between 0 and 1 is drawn in each iteration. If this random number falls below a threshold θ_3, a destroy-and-repair operation is performed using the operators $\mathcal{O}_1$. These operators correspond to the destroy-and-repair operators described in Section 5.4. Otherwise, a destroy-and-repair operation is carried out using the demand-based operators $\mathcal{O}_2$ from Section 5.5.

When applying the operators $\mathcal{O}_1$, we allow solutions with an objective value worse than the current solution. However, when using the operators $\mathcal{O}_2$, the current solution is only updated if the objective value falls within a predefined threshold, as is also the case in the second step described in Section 5.5. Additionally, the solution memory is incorporated by reverting to one of the stored solutions in memory if it_{max} iterations pass without an update to the global best solution $\mathcal{S}^*$. The functionality of the solution memory remains otherwise identical to that in the two-step version. All other parameters of the heuristic also remain unchanged. The probabilities for selecting operators are analogous to those in Sections 5.4 and 5.5. The parameters θ_3 and it_{max} were tuned in the same manner as described in Section 7.2.1 and are set to $\theta_3 = 0.015$ and $it_{max} = 1500$. Please note that we tune θ_3 as a fixed parameter, as adaptivity would strongly disadvantage service operators and would not be fair, since the modification of the service design often makes the solutions much worse in the short term, but can be advantageous in the long term. For a fair comparison, we allow the algorithm to run for the entire runtime without imposing an early stopping criterion.

Algorithm 4: Single-step version

$\mathcal{S}_c, \mathcal{S}^* \leftarrow$ Generate a feasible initial solution; Section 5.3
$it \leftarrow 0$;
while *Time limit not reached* **do**
  $p \leftarrow \text{randomNumber}(0, 1)$;
  if $p < \theta_3$ **then**
    $\mathcal{S}_c \leftarrow \text{DestroyandRepair}(\mathcal{S}_c, \mathcal{O}_1)$; // operators on the services
  else
    $\mathcal{S}_c' \leftarrow \text{DestroyandRepair}(\mathcal{S}_c, \mathcal{O}_2)$; // operators on the demands
    if $f(\mathcal{S}_c') < (1 + \theta_w) \cdot f(\mathcal{S}_c)$ **then**
      $\mathcal{S}_c \leftarrow \mathcal{S}_c'$; // update current solution if threshold is met
  if $f(\mathcal{S}_c) > f(\mathcal{S}^*)$ **then** $it \leftarrow it + 1$;
  else $it \leftarrow 0$; $\mathcal{S}^* \leftarrow \mathcal{S}_c$;
  if $it > it_{max}$ **then**
    $\mathcal{S}_c \leftarrow \text{drawRandomElement}(\Gamma)$; $it \leftarrow 0$; // use solution memory
  $\Gamma \leftarrow \text{update}(\Gamma, \mathcal{S}_c)$;

Table B.10
Comparison of Gurobi and the I2S-LNS with varying $|R|$ and constant $|D| = 15$.

Instance[a]	Gurobi			I2S-LNS				Δ [%]
	Costs	GAP [%]	Time [s]	Best costs	Avg. costs	σ [%]	Avg. time [s]	
R12-1	268.74	23.39	3600	268.74	268.74	0.00	107.00	0.00
R12-2	213.72	10.20	3600	209.17	209.17	0.00	63.50	−2.13
R12-3	234.14	12.53	3600	224.77	224.77	0.00	85.12	−4.00
R12-4	227.22	16.53	3600	224.20	224.20	0.00	106.96	−1.33
R12-5	206.22	13.40	3600	206.22	206.22	0.00	101.56	0.00
R36-1	203.11	14.76	3600	202.34	202.35	0.00	80.92	−0.37
R36-2	205.21	13.57	3600	205.21	205.21	0.00	95.83	0.00
R36-3	258.75	26.47	3600	236.47	236.48	0.00	165.57	−8.61
R36-4	201.09	13.88	3600	201.09	201.09	0.00	77.94	0.00
R36-5	245.40	34.10	3600	232.16	232.16	0.00	99.09	−5.40
R60-1	206.49	22.48	3600	197.38	197.50	0.06	96.84	−4.35
R60-2	198.17	21.86	3600	191.14	191.14	0.00	110.29	−3.55
R60-3	200.71	17.36	3600	193.26	193.36	0.05	89.83	−3.66
R60-4	179.82	31.15	3600	179.82	179.82	0.00	103.94	−0.00
R60-5	240.74	30.07	3600	232.75	232.75	0.00	145.33	−3.32

[a] R12-1 refers to the first out of five instances with $|\mathcal{R}| = 12$.

Table B.11
Performance benchmark against a single-step version on larger instances.

$\|D\|$	$\|R\|$	I2S-LNS				Single-step version			Δ [%]
		Best costs	Avg. costs	σ [%]	Avg. time [s]	Best costs	Avg. costs	σ [%]	
50	36	552.32	555.85	0.64	1283.70	564.14	568.94	0.85	2.14
50	36	559.20	563.13	0.70	1258.43	569.61	578.61	1.58	1.86
50	36	613.62	615.01	0.23	1434.71	623.31	625.70	0.38	1.58
50	36	626.99	629.19	0.35	1230.52	629.64	644.14	2.30	0.42
50	36	583.67	589.24	0.95	1134.03	599.95	606.95	0.87	2.79
75	48	895.75	904.07	0.93	1800	911.71	919.82	0.89	1.78
75	48	844.22	848.36	0.49	1800	856.59	875.23	2.18	1.47
75	48	873.83	879.79	0.68	1800	894.13	907.89	0.63	2.32
75	48	894.47	898.29	0.43	1740.36	900.97	911.16	0.39	0.73
75	48	880.42	889.61	1.04	1751.60	902.06	913.90	1.31	2.46
100	60	1105.81	1111.72	0.53	1800	1120.08	1141.26	1.89	1.29
100	60	1110.46	1120.61	0.91	1800	1136.92	1144.45	0.66	2.38
100	60	1140.63	1147.24	0.58	1800	1146.18	1154.73	0.75	0.49
100	60	1092.63	1099.54	0.63	1800	1135.46	1138.17	0.24	3.92
100	60	1077.81	1089.62	1.10	1800	1096.16	1113.07	1.54	1.70

References

Agarwal, R., & Ergun, Ö. (2010). Network design and allocation mechanisms for carrier alliances in liner shipping. *Operations Research*, *58*(6), 1726–1742. http://dx.doi.org/10.1287/opre.1100.0848.

Baldacci, R., Mingozzi, A., Roberti, R., & Calvo, R. W. (2013). An exact algorithm for the two-echelon capacitated vehicle routing problem. *Operations Research*, *61*(2), 298–314. http://dx.doi.org/10.1287/opre.1120.1153.

Breunig, U., Schmid, V., Hartl, R. F., & Vidal, T. (2016). A large neighbourhood based heuristic for two-echelon routing problems. *Computers & Operations Research*, *76*, 208–225. http://dx.doi.org/10.1016/j.cor.2016.06.014.

Bruni, M. E., Khodaparasti, S., & Perboli, G. (2024). A bi-level approach for last-mile delivery with multiple satellites. *Transportation Research Part C: Emerging Technologies*, *160*, Article 104495. http://dx.doi.org/10.1016/j.trc.2024.104495.

Cepolina, E. M., & Farina, A. (2015). A new urban freight distribution scheme and an optimization methodology for reducing its overall cost. *European Transport Research Review*, *7*(1), 1–14. http://dx.doi.org/10.1007/s12544-014-0149-x.

Crainic, T. G. (2000). Service network design in freight transportation. *European Journal of Operational Research*, *122*(2), 272–288. http://dx.doi.org/10.1016/S0377-2217(99)00233-7.

Crainic, T. G., Errico, F., Rei, W., & Ricciardi, N. (2016). Modeling demand uncertainty in two-tier city logistics tactical planning. *Transportation Science*, *50*(2), 559–578. http://dx.doi.org/10.1287/trsc.2015.0606.

Crainic, T. G., Gendreau, M., & Jemai, L. (2020). Planning hyperconnected, urban logistics systems. *Transportation Research Procedia*, *47*, 35–42. http://dx.doi.org/10.1016/j.trpro.2020.03.070.

Crainic, T. G., & Kim, K. H. (2007). Intermodal transportation. *Handbooks in Operations Research and Management Science*, *14*, 467–537. http://dx.doi.org/10.1016/S0927-0507(06)14008-6.

Crainic, T. G., Perboli, G., & Ricciardi, N. (2021). City logistics. In *Network design with applications to transportation and logistics* (pp. 505–534). Springer, http://dx.doi.org/10.1007/978-3-030-64018-7_16.

Crainic, T. G., Ricciardi, N., & Storchi, G. (2004). Advanced freight transportation systems for congested urban areas. *Transportation Research Part C: Emerging Technologies*, *12*(2), 119–137. http://dx.doi.org/10.1016/j.trc.2004.07.002.

Crainic, T. G., Ricciardi, N., & Storchi, G. (2009). Models for evaluating and planning city logistics systems. *Transportation Science*, *43*(4), 432–454. http://dx.doi.org/10.1287/trsc.1090.0279.

Crainic, T. G., & Sgalambro, A. (2014). Service network design models for two-tier city logistics. *Optimization Letters*, *8*(4), 1375–1387. http://dx.doi.org/10.1007/s11590-013-0662-1.

Cruijssen, F., Cools, M., & Dullaert, W. (2007). Horizontal cooperation in logistics: Opportunities and impediments. *Transportation Research Part E: Logistics and Transportation Review*, *43*(2), 129–142. http://dx.doi.org/10.1016/j.tre.2005.09.007.

Cruijssen, F., Dullaert, W., & Fleuren, H. (2007). Horizontal cooperation in transport and logistics: A literature review. *Transportation Journal*, *46*(3), 22–39. http://dx.doi.org/10.2307/20713677.

Elting, S., Ehmke, J. F., & Gansterer, M. (2025). Preference learning for efficient bundle selection in horizontal transport collaborations. *European Journal of Operational Research*.

EU Commission (2001). Guidelines on the applicability of article 81 of the EC treaty to horizontal cooperation agreements. *Official Journal of the European Communities*, *100*, 2–30.

Fontaine, P., Crainic, T. G., Jabali, O., & Rei, W. (2021). Scheduled service network design with resource management for two-tier multimodal city logistics. *European Journal of Operational Research*, *294*(2), 558–570. http://dx.doi.org/10.1016/j.ejor.2021.02.009.

Fontaine, P., Minner, S., & Schiffer, M. (2023). Smart and sustainable city logistics: Design, consolidation, and regulation. *European Journal of Operational Research*, *307*(3), 1071–1084. http://dx.doi.org/10.1016/j.ejor.2022.09.022.

Frisk, M., Göthe-Lundgren, M., Jörnsten, K., & Rönnqvist, M. (2010). Cost allocation in collaborative forest transportation. *European Journal of Operational Research*, *205*(2), 448–458. http://dx.doi.org/10.1016/j.ejor.2010.01.015.

Gansterer, M., & Hartl, R. F. (2018). Collaborative vehicle routing: A survey. *European Journal of Operational Research*, *268*(1), 1–12. http://dx.doi.org/10.1016/j.ejor.2017.10.023.

Gansterer, M., Hartl, R. F., & Salzmann, P. E. (2018). Exact solutions for the collaborative pickup and delivery problem. *Central European Journal of Operations Research*, *26*, 357–371. http://dx.doi.org/10.1007/s10100-017-0503-x.

Gansterer, M., Hartl, R. F., & Savelsbergh, M. (2020). The value of information in auction-based carrier collaborations. *International Journal of Production Economics*, *221*, Article 107485. http://dx.doi.org/10.1016/j.ijpe.2019.09.006.

Gillies, D. (1959). Solutions to general non-zero-sum games. *vol. 4*, In *Contributions to the theory of games* (pp. 47–85). Princeton University Press, http://dx.doi.org/10.1515/9781400882168-005, (40).

Grangier, P., Gendreau, M., Lehuédé, F., & Rousseau, L.-M. (2016). An adaptive large neighborhood search for the two-echelon multiple-trip vehicle routing problem with satellite synchronization. *European Journal of Operational Research*, *254*(1), 80–91. http://dx.doi.org/10.1016/j.ejor.2016.03.040.

Guajardo, M., & Rönnqvist, M. (2016). A review on cost allocation methods in collaborative transportation. *International Transactions in Operational Research*, *23*(3), 371–392. http://dx.doi.org/10.1111/itor.12205.

Hemmelmayr, V. C., Cordeau, J.-F., & Crainic, T. G. (2012). An adaptive large neighborhood search heuristic for two-echelon vehicle routing problems arising in city logistics. *Computers & Operations Research*, *39*(12), 3215–3228. http://dx.doi.org/10.1016/j.cor.2012.04.007.

Jepsen, M., Spoorendonk, S., & Ropke, S. (2013). A branch-and-cut algorithm for the symmetric two-echelon capacitated vehicle routing problem. *Transportation Science*, *47*(1), 23–37. http://dx.doi.org/10.1287/trsc.1110.0399.

Karels, V. C., Veelenturf, L. P., & Van Woensel, T. (2020). An auction for collaborative vehicle routing: Models and algorithms. *EURO Journal on Transportation and Logistics*, *9*(2), Article 100009. http://dx.doi.org/10.1016/j.ejtl.2020.100009.

Kimms, A., & Kozeletskyi, I. (2016a). Core-based cost allocation in the cooperative traveling salesman problem. *European Journal of Operational Research*, *248*(3), 910–916. http://dx.doi.org/10.1016/j.ejor.2015.08.002.

Kimms, A., & Kozeletskyi, I. (2016b). Shapley value-based cost allocation in the cooperative traveling salesman problem under rolling horizon planning. *EURO Journal on Transportation and Logistics*, *5*(4), 371–392. http://dx.doi.org/10.1007/s13676-015-0087-3.

Lone, S., Harboul, N., & Weltevreden, J. (2021). European E-commerce report.

Mancini, S., Gansterer, M., & Hartl, R. F. (2021). The collaborative consistent vehicle routing problem with workload balance. *European Journal of Operational Research*, *293*(3), 955–965. http://dx.doi.org/10.1016/j.ejor.2020.12.064.

Montoya-Torres, J. R., Muñoz-Villamizar, A., & Vega-Mejía, C. A. (2016). On the impact of collaborative strategies for goods delivery in city logistics. *Production Planning and Control*, *27*(6), 443–455. http://dx.doi.org/10.1080/09537287.2016.1147092.

Nataraj, S., Ferone, D., Quintero-Araujo, C., Juan, A., & Festa, P. (2019). Consolidation centers in city logistics: A cooperative approach based on the location routing problem. *International Journal of Industrial Engineering Computations*, *10*(3), 393–404. http://dx.doi.org/10.5267/j.ijiec.2019.1.001.

Pan, S., Trentesaux, D., Ballot, E., & Huang, G. Q. (2019). Horizontal collaborative transport: Survey of solutions and practical implementation issues. *International Journal of Production Research*, *57*(15-16), 5340–5361. http://dx.doi.org/10.1080/00207543.2019.1574040.

Parisa, D. N., Samuel, K. E., & Espinouse, M.-L. (2019). Systematic literature review on city logistics: Overview, classification and analysis. *International Journal of Production Research*, *57*(3), 865–887. http://dx.doi.org/10.1080/00207543.2018.1489153.

Perboli, G., Brotcorne, L., Bruni, M. E., & Rosano, M. (2021). A new model for last-mile delivery and satellite depots management: The impact of the on-demand economy. *Transportation Research Part E: Logistics and Transportation Review*, *145*, Article 102184. http://dx.doi.org/10.1016/j.tre.2020.102184.

Pérez-Bernabeu, E., Juan, A. A., Faulin, J., & Barrios, B. B. (2015). Horizontal cooperation in road transportation: A case illustrating savings in distances and greenhouse gas emissions. *International Transactions in Operational Research*, *22*(3), 585–606. http://dx.doi.org/10.1111/itor.12130.

Pisinger, D., & Ropke, S. (2007). A general heuristic for vehicle routing problems. *Computers & Operations Research*, *34*(8), 2403–2435. http://dx.doi.org/10.1016/j.cor.2005.09.012.

Rusich, A., et al. (2017). *Collaborative logistics networks* (Ph.D. thesis), Università degli Studi di Trieste.

Savelsbergh, M., & van Woensel, T. (2016). City logistics: Challenges and opportunities: 50th anniversary invited article. *Transportation Science*, *50*(2), 579–590. http://dx.doi.org/10.1287/trsc.2016.0675.

Scherr, Y. O., Saavedra, B. A. N., Hewitt, M., & Mattfeld, D. C. (2019). Service network design with mixed autonomous fleets. *Transportation Research Part E: Logistics and Transportation Review*, *124*, 40–55. http://dx.doi.org/10.1016/j.tre.2019.02.001.

Schmeidler, D. (1969). The nucleolus of a characteristic function game. *SIAM Journal on Applied Mathematics*, *17*(6), 1163–1170. http://dx.doi.org/10.1137/0117107.

Schmidt, J., Tilk, C., & Irnich, S. (2022). Using public transport in a 2-echelon last-mile delivery network. *European Journal of Operational Research*, *317*(3), 827–840. http://dx.doi.org/10.1016/j.ejor.2022.10.041.

Shapley, L. S. (1953). A value for n-person games. In H. W. Kuhn, & A. W. Tucker (Eds.), *Contributions to the theory of games II* (pp. 307–317). Princeton: Princeton University Press, http://dx.doi.org/10.7249/P0295.

Shelbourne, B. C., Battarra, M., & Potts, C. N. (2017). The vehicle routing problem with release and due dates. *INFORMS Journal on Computing*, *29*(4), 705–723. http://dx.doi.org/10.1287/ijoc.2017.0756.

Sluijk, N., Florio, A. M., Kinable, J., Dellaert, N., & Van Woensel, T. (2023). Two-echelon vehicle routing problems: A literature review. *European Journal of Operational Research*, *304*(3), 865–886. http://dx.doi.org/10.1016/j.ejor.2022.02.022.

Tijs, S. H., & Driessen, T. S. (1986). Game theory and cost allocation problems. *Management Science*, *32*(8), 1015–1028. http://dx.doi.org/10.1287/mnsc.32.8.1015.

United Nations (2018). *World urbanization prospects*. United Nations, Dept. of Economic and Social Affairs, Population Division, URL: https://population.un.org/wup/publications/.

Vanovermeire, C., Vercruysse, D., & Sörensen, K. (2014). Analysis of different cost allocation methods in a collaborative transport setting. *International Journal of Engineering Management and Economics*, *4*(2), 132–150. http://dx.doi.org/10.1504/IJEME.2014.066576.

Verdonck, L., Beullens, P., Caris, A., Ramaekers, K., & Janssens, G. K. (2016). Analysis of collaborative savings and cost allocation techniques for the cooperative carrier facility location problem. *Journal of the Operational Research Society*, *67*(6), 853–871. http://dx.doi.org/10.1057/jors.2015.106.

Voigt, S., Frank, M., Fontaine, P., & Kuhn, H. (2022). Hybrid adaptive large neighborhood search for vehicle routing problems with depot location decisions. *Computers & Operations Research*, *146*, Article 105856. http://dx.doi.org/10.1016/j.cor.2022.105856.

Winkenbach, M., Kleindorfer, P. R., & Spinler, S. (2016). Enabling urban logistics services at la poste through multi-echelon location-routing. *Transportation Science*, *50*(2), 520–540. http://dx.doi.org/10.1287/trsc.2015.0624.