Transportation Research Part E 214 (2026) 105025

Contents lists available at ScienceDirect

Transportation Research Part E

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

Solving a rich vehicle routing problem with a heterogeneous fleet and trailers in medium-sized transport companies

Robin Aninger, Stefan Voigt *

Catholic University of Eichstätt-Ingolstadt, Ingolstadt School of Management, Auf der Schanz 49, 85049, Ingolstadt, Germany

ARTICLE INFO

Keywords:
Rich vehicle routing problem
Heterogeneous fleet
Trailers
Matheuristic

ABSTRACT

Efficient route planning is a critical challenge for transport companies operating heterogeneous vehicle fleets. Manual planning often fails to leverage cost-saving consolidation opportunities, leading to suboptimal capacity utilization and increased operational costs. This paper addresses a real-world vehicle routing problem involving pickup and delivery tasks, heterogeneous vehicles and trailers, time windows, service equipment requirements, and labor constraints. We develop a matheuristic method that combines heuristic tour construction with exact optimization using a set partitioning model leveraging the manageable number of stops per tour.

The proposed matheuristic is validated using real operational data from a medium-sized logistics provider based in Germany. Our experiments demonstrate that the matheuristic achieves average cost savings of 16.7% compared to historical plans, with solution times well under three minutes for instances up to 55 nodes. A comparison with a pure optimization model shows that while exact methods can yield marginally better solutions (on average 2–3% improvement), they are computationally impractical for real-time use. The matheuristic strikes a practical balance between solution quality and runtime, enabling automated, scalable, and cost-efficient routing for medium-sized transport companies with a heterogeneous fleet.

1. Introduction

Efficient routing of transport vehicles is critical to the logistics industry, particularly given the substantial freight volumes handled by road transport. In Germany alone, approximately 2.9 billion tonnes of goods were transported by truck in 2023, with the vast majority being domestic freight (Federal Statistical Office of Germany (Destatis), 2025). While large logistics providers often operate centralized planning systems and proprietary optimization software, many small and medium-sized transport companies continue to rely on manual route planning processes, which are often constrained by time pressures, lack of optimization expertise, and limited access to advanced decision-support systems (Caceres-Cruz et al., 2014). Manual planning in such contexts frequently results in suboptimal utilization of vehicle capacities, missed consolidation opportunities, increased operational costs, and emissions. Furthermore, human planners often struggle to account for the combinatorial complexity introduced by diverse vehicle capabilities, time window restrictions, and heterogeneous customer demands. While academic research on vehicle routing problems (VRPs) has yielded a wide array of sophisticated algorithms, these methods often prioritize benchmark performance at the expense of practical applicability, overlooking constraints such as heterogeneous fleets, varying service requirements, and tight computation time budgets (Drexl, 2012).

This paper addresses the need for practically viable and computationally efficient route optimization methods tailored to the operational realities of medium-sized transport companies managing a heterogeneous fleet. We consider a real-world variant of the

* Corresponding authors.
E-mail addresses: robin.aninger@aninger.de (R. Aninger), stefan.voigt@ku.de (S. Voigt).

https://doi.org/10.1016/j.tre.2026.105025
Received 30 October 2025; Received in revised form 24 April 2026; Accepted 11 June 2026
Available online 22 June 2026

VRP that involves transporting individual orders, each defined by a pickup and delivery location. Some customers are subject to strict temporal constraints in the form of hard time windows, and each tour is additionally constrained by a maximum allowable duration. Additional complexity arises from the need to assign appropriate combinations of trucks and trailers, respecting constraints such as payload capacity, loading equipment (e.g., tail lifts), and driver working time regulations. Our objective is to develop a matheuristic solution approach that effectively integrates mathematical optimization with heuristic techniques to minimize total operational costs, including distance-based costs and labor expenses, while ensuring efficient utilization of available equipment and producing high-quality, feasible tour plans. The selected case company is representative of the class of small- to medium-sized operators and provides a realistic setting in which solution quality, robustness, and short runtimes are equally critical. Consequently, the proposed matheuristic is designed not to target large-scale benchmark instances (for large-scale instances, see for example Arnold et al., 2019), but to deliver practically viable and implementable solutions for medium-sized transport operations.

The key contributions of this study are as follows:

1. We introduce a rich, application-driven VRP variant that reflects the operational constraints of a medium-sized transport provider operating a heterogeneous vehicle fleet.
2. We formulate a mathematical model that incorporates core logistical elements such as time windows, vehicle-trailer compatibility, capacity limits, and service durations.
3. We propose a matheuristic method that combines an integer programming model for optimal tour selection with a heuristic procedure for generating candidate tours.
4. We conduct extensive numerical experiments using real operational data, evaluating the solution quality, computational efficiency, and robustness of our approach.

The remainder of this paper is structured as follows. Section 2 provides a detailed problem definition. Section 3 reviews the relevant literature on related rich VRP variants. Section 4 presents our mathematical model. Section 5 outlines the matheuristic solution approach. Section 6 reports on numerical experiments and their outcomes. Finally, Section 7 concludes the article and discusses directions for future research.

2. Problem description

This section formalizes the transport planning problem addressed in this study, inspired by real-world operational challenges encountered by medium-sized logistics providers. The goal is to efficiently assign transport jobs to a heterogeneous fleet of vehicles and trailers, aiming to minimize total operational costs while respecting various practical constraints.

2.1. Transport jobs and operational structure

Each transport job involves shipping from a pickup (loading) location to a delivery (unloading) location. Jobs have associated attributes that define their operational requirements:

- Pickup and delivery nodes with geographic coordinates,
- Time windows specifying earliest and latest allowed service start times for both loading and unloading operations,
- Service durations at each node,
- Load characteristics including weight and required pallet space,
- Optional service equipment constraints, such as the need for tail lifts at certain locations.

Transport jobs may be composed of multiple sub-jobs, typically from the same client, which must be served within the same route. These sub-jobs may include direct return loads, i.e., transport requests where the destination of one job becomes the origin of another, provided that all precedence and compatibility constraints are respected.

The time windows vary in precision from tightly defined (e.g., 10-minute slots) to more flexible spans that cover a full working day. All loading operations must precede their corresponding unloading operations and all service activities must respect specified time windows, with the required service time occurring in addition to the time window constraints.

2.2. Fleet and equipment constraints

The transport provider operates a heterogeneous fleet consisting of several types of trucks and trailers. Vehicles differ significantly in payload capacity, pallet space, operating costs, and available equipment (such as tail lifts). Trailers increase load capacity but are compatible only with specific vehicles. The proper assignment of vehicles and trailers to jobs depends on:

- **Compatibility:** Each trailer can be towed only by compatible vehicles;
- **Capacity:** The combined load of assigned jobs must not exceed the payload or volume limit of the vehicle-trailer combination at any time;
- **Equipment availability:** Some loading/unloading locations require tail lifts;
- **Driving time limits:** Each route must comply with applicable labor regulations and operational policies regarding maximum route durations.

The fleet includes various vehicle classes, summarized with representative operational costs in Table 1.

Each tour must start and end at a central depot and use exactly one vehicle, optionally paired with a compatible trailer.

Table 1
Fleet composition and representative costs.

Vehicle class	Cost per kilometer (€)	Cost per hour (€)
Small vans	0.40	13
Medium transporters	0.50	14
Light trucks	0.60–0.80	14.5–19
Trailers	0.12	–

2.3. Objectives and cost structure

The primary objective is to assign all jobs to feasible routes such that total operational costs are minimized. Costs consist of three main components:

- **Distance-based vehicle costs**, proportional to the distance traveled by each vehicle;
- **Distance-based trailer costs**, incurred when a vehicle tows a trailer and proportional to the distance traveled;
- **Time-based labor costs**, proportional to the total tour duration.

The distance-based cost components represent the average operational costs per kilometer provided by the case company. They are aggregated cost rates and include fuel consumption, maintenance, depreciation, insurance, and administrative overhead expenses. As the purpose of this study is to evaluate routing and tour selection decisions rather than detailed cost accounting, the distance-based cost components are treated as exogenous inputs and are not further decomposed into individual cost components. Time-based labor costs reflect driver wages and associated personnel-related expenses per unit of working time.

2.4. Complexity and need for automation

Given the multiple constraints and rich structure of the problem, including heterogeneous resources, sub-job precedence, time windows, and compatibility rules, manual planning is time-consuming and prone to suboptimality. Human dispatchers may overlook feasible consolidations or create schedules that violate minor constraints, leading to higher costs or operational delays. Even experienced planners face significant challenges in maintaining full feasibility and cost-effectiveness as the size of the problem increases. Automated tour planning enables more consistent and efficient use of resources. According to Toth and Vigo (2002), algorithmic methods can reduce total routing costs significantly compared to manual dispatching. In addition, efficient vehicle utilization contributes not only to economic performance, but also to environmental goals, by reducing total vehicle kilometers and associated CO_2 emissions (McKinnon and Edwards, 2010). To support practical adoption, the solution approach must produce feasible, interpretable, and near-optimal routing plans within minutes, a requirement addressed by the matheuristic optimization approach developed in this study.

3. Literature

Tour planning problems are among the most fundamental and practically relevant research domains in operations research. Due to their central role in logistics and distribution systems, they have generated a large body of literature over the past decades (Laporte, 2009). The classical *capacitated vehicle routing problem* (CVRP), introduced by Dantzig and Ramser (1959), describes the task of serving multiple customers from a central depot using a fleet of identical vehicles, each constrained by capacity, with the goal of minimizing the total travel distance. This classical model provides the foundation for a vast array of problem extensions. In practice, transport operations are rarely as simple as assumed in the CVRP. Real-world constraints such as customer time windows, multiple depots, varying vehicle types, and dynamic demand necessitate richer models. This has led to the development of the *rich vehicle routing problem* (rich VRP) family, which generalizes the VRP to account for a broader and more realistic set of operational requirements (Drexl, 2012). Section 3.1 examines the notion of richness in VRPs, Section 3.2 reviews related articles particularly those involving heterogeneous fleets and trailers, Section 3.3 briefly reviews solution approaches, and lastly Section 3.4 summarizes the research gap.

3.1. Defining rich vehicle routing

Early VRP research focused on stylized laboratory problems: homogeneous vehicles departing from a single depot, serving a set of customers with deterministic demands and simple capacity constraints. Over time, two broad strands emerged. On one hand, theoretical contributions explored algorithmic and combinatorial properties of benchmark instances under controlled conditions; on the other hand, case-oriented papers progressively introduced constraints motivated by specific industries (Lahyani et al., 2015b).

Pioneering articles by Sörensen et al. (2008) and Drexl (2012) critically compare these theoretical VRP models against the complexity encountered in operational environments. Both studies demonstrate that academic formulations often neglect features that practitioners face daily such as multiple depots, driver-regulation constraints, equipment requirements (e.g., tail lifts, cooling units), and dynamic customer requests. For example, Sörensen et al. (2008) highlight that real-life systems require rapid optimization, integration with warehouse and inventory decisions, and compatibility checks between vehicle equipment and customer sites. Drexl

(2012) further emphasizes that routing software must accommodate customer-driven complexities; if not, solutions derived in laboratory settings risk being inapplicable or suboptimal when deployed. Both authors call for richer models and algorithms that bridge the gap between academic VRPs and industrial practice.

Despite the introduction of *rich* VRP variants, there is no universally accepted definition of when a problem qualifies as a *rich* VRP. Early works often equated richness with the mere addition of one or two features (e.g., time windows or pickup-and-delivery constraints). Lahyani et al. (2015b) observe that many papers label their variant as rich simply because it integrates multiple but loosely related constraints, thus obscuring the distinction between a moderately extended VRP and a truly comprehensive, real-world model. Caceres-Cruz et al. (2014) arrive at a similar conclusion: Rich VRP has become a catch-all term for any model that incorporates more than the classical assumptions, although real-world distribution systems typically encompass a high number of interacting features.

To address this ambiguity, Lahyani et al. (2015b) propose a precise definition. Their approach begins with an extensive literature review to identify recurring strategic/tactical aspects (e.g., multi-period planning, multiple depots, stochastic demand) and operational/physical restrictions (e.g., heterogeneous fleets, time windows, driver regulations). They assert that a VRP must include at least four strategic/tactical aspects and six operational/physical restrictions to qualify as rich. If a model focuses primarily on operational characteristics, it must encompass at least nine such constraints to meet their threshold. However, we follow the more comprehensive (but more fuzzy) definition of Caceres-Cruz et al. (2014). These authors emphasize that rich VRP is not a single problem but a class of problems. Drawing on multiple sources, they note that richness arises from integrating multiple optimization criteria, constraints, and preferences to mirror actual logistics operations.

Prominent enrichments include:

- Time windows for deliveries and pickups;
- Pickup-and-delivery relationships;
- Heterogeneous vehicle fleets;
- Equipment-based restrictions (e.g., tail lifts, refrigeration units);
- Compatibility constraints between customers and vehicles;
- Environmental or regulatory restrictions (e.g., low-emission zones);
- Driver working-hour regulations and mandatory breaks.

Thus, rather than defining richness by any single feature, they frame rich VRP as an umbrella term for models that combine multiple, often interdependent, real-world constraints. Numerous studies illustrate how combining heterogeneous fleets, customer requirements, and operational constraints gives rise to rich VRP variants that bridge academic models and real-world applications. For a review on rich routing problems in supply chain management we refer the interested reader to Schmid et al. (2013) and for a review on less constrained heterogeneous VRPs to Baldacci et al. (2008). The following section reviews studies most closely related to the problem setting considered in this work.

3.2. Related studies

Rich VRPs with heterogeneous fleets. A first closely related stream of literature focuses on rich vehicle routing problems involving heterogeneous fleets. Sicilia et al. (2016) examine urban distribution where narrow streets, congestion, and tight delivery schedules necessitate a heterogeneous fleet of small and medium-sized vehicles, strict pickup-and-delivery windows, equipment compatibility (e.g., refrigerated versus dry goods), limits on the number of orders per vehicle, and open routes that need not return to the depot. To solve this problem, they propose a hybrid metaheuristic that integrates variable neighbourhood search and tabu search. Similarly, Lahyani et al. (2015a) tackle olive-oil collection in Tunisia through a multi-product, multi-period, multi-compartment model: trucks feature dedicated compartments for each oil grade (with mandatory cleaning between tours), capacity heterogeneity, and dynamic demand forecasts across several collection periods. Their mixed-integer programming formulation, solved by branch-and-cut, highlights how compartmentalization, cleaning activities, and seasonal demand patterns can be captured within a rich, multi-period framework that closely aligns with actual operational costs and service requirements. Marques et al. (2020) extend the richness paradigm to the forestry sector by integrating inbound logistics (raw material deliveries from suppliers to a mill) and outbound logistics (finished product shipments from the mill to customers) under a common heterogeneous fleet. Their model incorporates backhauling, allowing vehicles to pick up raw materials on return trips. They propose a matheuristic that iteratively fixes binary variables. A recent contribution by Silva et al. (2025) addresses a rich VRP with heterogeneous fleet and time windows, arising from a technology-based logistics provider. This study formulates an asymmetric mixed (close–open) rich vehicle routing problem incorporating capacity, heterogeneous vehicle types, and operational implementation constraints, and evaluates its performance on both benchmark and real company instances. De Armas et al. (2015) focus on a Spanish distribution company with a fixed heterogeneous fleet, soft and multiple time windows per customer, varying customer priorities, and vehicle–customer compatibility constraints. Their objectives include minimizing total distance while balancing route durations to ensure equitable workloads. Exact methods proved impractical for instances exceeding 100 customers and 20 vehicles, so they develop a general variable neighborhood search for the static setting and a dynamic approach that re-optimizes routes within seconds upon arrival of new requests (de Armas and Melián-Batista, 2015). Dabia et al. (2019) consider a VRP with private fleet and common carrier by allowing a private heterogeneous fleet to serve customers unless it is capacity-constrained or outsourcing to a common carrier subject to quantity-discounted costs proves more economical. Their mixed-integer formulation integrates binary decisions for private-fleet assignments and outsourcing quantities, piecewise-linear cost functions for discounted outsourcing, and customer time windows. A branch-and-price algorithm efficiently solves moderate-sized instances. Ceselli et al. (2009) develop a column-generation framework for a daily planning problem faced by a distribution-software company that must route a heterogeneous fleet from multiple depots to customers subject to numerous constraints: time windows, incompatibilities between goods, vehicles, and customers; maximum route length and duration, driver-hour regulations, optional outsourcing of low-volume customers to express couriers, split deliveries, open routes, and a complex cost structure dependent on load, distance, and number of stops. Recent research has further expanded the scope of rich VRPs. Alcaraz et al. (2019) formulate a rich VRP integrating long-haul transport, driver-hours regulation, heterogeneous fleets, incompatibility among goods, and last-mile outsourcing decisions. They develop tailored heuristics and metaheuristics to balance transport cost, regulatory compliance, and outsourcing efficiency. Kramer et al. (2019) address pharmaceutical distribution through a multi-depot, heterogeneous-fleet model with flexible time windows, periodic demand, and vehicle–customer incompatibilities, solved via a multi-start iterated local search. Yang et al. (2020) propose a cooperative rich VRP for rural last-mile logistics, combining cooperative game theory with branch-price-and-cut optimization to allocate cost savings among competing logistics providers. More recently, Yang et al. (2025) introduce a neighborhood-based matheuristic for the vehicle routing problem with delivery options, addressing alternative delivery locations, shared drop-points, and service-level synchronization in last-mile logistics.

Routing with trailers. A second relevant stream concerns vehicle routing problems with trailers, commonly referred to as the truck-and-trailer routing problem (TTRP). Early heuristic approaches include Scheuerer (2006), who propose a tabu search for the TTRP, and Lin et al. (2011), who extend the problem to include time windows. Subsequent contributions have explored both heuristic and exact solution methods. For example, Villegas et al. (2013) develop a set-partitioning-based matheuristic for the TTRP, while Derigs et al. (2013) provide a comprehensive treatment of truck-and-trailer routing variants. On the exact side, Rothenbächer et al. (2018) propose a branch-price-and-cut algorithm for the TTRP with time windows, and Bartolini and Schneider (2020) introduce a two-commodity flow formulation for capacitated variants. Accorsi and Vigo (2020) work on the single truck-and-trailer routing problem, while da Cruz and Salles da Cunha (2023) focus on a profitable variant with time windows.

3.3. Solution approaches

There is a growing movement toward unified or framework-based solution approaches that can accommodate multiple rich VRP formulations under a single engine. Early work by Hasle and Kloster (2007) exemplifies the unified-heuristic philosophy. The authors employ a single, extensible rich VRP model and apply a metaheuristic framework to solve MDVRP, PVRP, VRPB, and their combinations without altering the algorithm's high-level structure. Subsequent efforts formalized this idea: Ropke and Pisinger (2006) proposed an adaptive large neighborhood search (ALNS) that solves the rich PDPTW problem; by plugging in specific removal and insertion operators, their ALNS can be deployed for any subvariant of rich PDPTW without rewriting core logic. Derigs and Vogel (2013) introduced a generic metaheuristic framework capable of addressing several rich VRPs within one algorithmic schema.

On the exact-method side, Errami et al. (2024) release VRPSolverEasy, a Python interface to the VRPSolver branch-cut-and-price framework that abstracts away low-level MIP modeling. Users simply define depots, customers, links, and vehicle types, VRPSolverEasy then handles heterogeneous fleets, time windows, multi-compartment vehicles, and virtually any combination of popular VRP constraints (Errami et al., 2024). This generic approach attains optimality for smaller instances (up to 100 customers) and provides benchmark solutions for larger cases.

In contrast to unified solver frameworks, matheuristics are specifically designed for a given problem variant yet retain sufficient flexibility to incorporate new constraints with relatively minor modifications (see Archetti and Speranza, 2014, for an extensive review on matheuristics for routing problems). Following the definition of Boschetti et al. (2009), a matheuristic is a heuristic algorithm made by the interoperation of metaheuristics and mathematical programming techniques. By interweaving exact methods (e.g., integer programming subroutines) with metaheuristic search components, matheuristics exploit the strengths of both paradigms: they use mathematical programming to guide large-scale neighborhood explorations, while metaheuristic strategies provide robustness in navigating complex, high-dimensional solution spaces. Doerner and Schmid (2010) identify three principal hybridization strategies that combine exact algorithms with metaheuristics:

- Set-covering-/set-partitioning based matheuristics, where a pool of promising routes (generated by a heuristic) is collected and then selected via a mathematical formulation;
- Local branching approaches, which embed integer cuts or branching constraints within a metaheuristic to intensify search around incumbent solutions; and
- Decomposition techniques, where the original problem is partitioned into smaller subproblems (e.g., by geographic region or customer clusters) that are solved precisely or heuristically and then recombined.

3.4. Positioning of the present study

Positioning of the problem setting. Despite the breadth of literature, there remain open challenges in aligning academic methods with real-world operational constraints. While time windows and vehicle heterogeneity are frequently studied, the integration of specialized equipment requirements (e.g., tail lifts), trailer attachment rules, or driver shift modeling is less common. Moreover, many studies focus on abstract benchmark instances (e.g., Solomon or Golden datasets), which may not reflect the complexity of actual dispatching environments. Against this background, the problem addressed in this paper occupies a distinct niche. In contrast to classical TTRP formulations, which typically emphasize trailer detachment, parking, or transshipment decisions, we model trailers as optional capacity extensions within a heterogeneous fleet without intermediate decoupling operations. At the same time, our model extends beyond much of the heterogeneous-fleet VRP literature by integrating multiple interacting features, including paired pickup-and-delivery requests, grouped sub-jobs that must be served on the same route, tail-lift compatibility constraints, explicit vehicle–trailer compatibility, and maximum tour duration limits. To the best of our knowledge, the combination of these elements within a medium-sized, real-world routing context remains underrepresented in the literature.

Positioning of the problem scale. In the heterogeneous-fleet stream (without a rich set of constraints), early contributions such as Liu and Shen (1999) and Repoussis and Tarantilis (2010) heuristically address fleet size and mix problems with time windows with up to 100 customers, while more application-driven studies such as Belfiore and Yoshida Yoshizaki (2009) tackle an instance with 519 supermarkets. More recently, research on very large-scale VRPs with homogeneous fleets has focused on decomposition, clustering, and learning-based frameworks capable of handling hundreds to tens of thousands of customers, e.g., Arnold et al. (2019) and Kerscher and Minner (2025). These approaches typically prioritize scalability over the inclusion of detailed operational constraints. In contrast, the present study focuses on medium-sized instances (up to 55 nodes in our experiments) where computational tractability is less restrictive, but the integration of multiple interacting real-world constraints becomes central. Our objective is therefore not to compete with large-scale VRP methods, but to address a richer and more application-driven routing problem that reflects the operational realities of medium-sized transport providers.

Set-partitioning matheuristics and methodological positioning. In our work, we focus on a set-partitioning matheuristic: we first generate a diverse route pool via a tailored brute force heuristic and then solve a restricted set-partitioning model to assemble these routes into a globally feasible solution. This choice reflects both the empirical effectiveness of set-partitioning formulations in rich VRP literature and their relative ease of extension when new constraints are introduced. Recent empirical evidence further supports the effectiveness of set-partitioning and set-covering based matheuristics for VRPs. Arenas-Vasco et al. (2025) conduct a large-scale meta-analysis of more than 200 VRP studies employing route-pool-based matheuristics and statistically quantify the contribution of set-partitioning components to solution quality. Their results show that formulations based on set partitioning consistently outperform pure metaheuristics in terms of optimality gap, particularly for rich VRPs with heterogeneous fleets, time windows, and multiple interacting constraints. In most existing contributions, however, the quality of the final solution depends critically on sophisticated route construction procedures, typically based on adaptive large neighborhood search, variable neighborhood search, or other advanced metaheuristics.

In contrast, the present study deliberately adopts a different design choice for the route generation phase. Instead of relying on complex metaheuristic search, we employ a structured brute-force enumeration of all feasible permutations within small node groups. This approach is enabled by the medium-sized problem setting and the empirically limited number of stops per tour, and allows us to exhaustively evaluate all valid sequences within each group while fully enforcing feasibility constraints. As a result, the generated route pool is not biased by heuristic search trajectories and guarantees that the best feasible tour for each combination (and vehicle class) is retained.

This distinction directly addresses the trade-off between solution completeness and computational scalability emphasized in the literature. While advanced heuristics are indispensable for large-scale instances, our results indicate that for medium-sized, highly constrained problems, exhaustive enumeration can be both computationally tractable and methodologically advantageous. It ensures high-quality candidate tours without requiring extensive parameter tuning or complex search operators, thereby providing a transparent, explainable and robust alternative within the set-partitioning matheuristic framework.

4. Model formulation

This section provides a complete mathematical formulation of the vehicle routing problem under consideration. It includes a detailed description of the sets, parameters, and decision variables, followed by the objective function and constraints. The model accounts for vehicle and trailer assignments, load propagation, service time windows, precedence relations, and cost components.

In the following, we model the problem as a mixed-integer programming model and use the notation given in Table 2. All parameters are indexed by their associated sets, while decision variables additionally use superscripts to indicate the corresponding vehicle and, where applicable, the assigned trailer. Node 0 denotes the depot. Subscripts indicate the associated index (e.g., Q_k^{weight} for the maximum weight capacity of vehicle k, Q_l^{pallet} for the maximum pallet capacity of trailer l, or e_i for the earliest arrival time at node i). Given the large number of constraints, we introduce them in logically related groups for clarity. The complete model is presented in Appendix A.

Objective. The objective function minimizes the total operational cost, combining travel-related expenses (vehicle and trailer distance) with time-based labor costs incurred by each vehicle.

$$\min C = \sum_{k\in K} c_k^{\text{dist}} D^k + \sum_{k\in K}\sum_{l\in L} c_l^{\text{dist}} D^{k,l} + \sum_{k\in K} c_k^{\text{time}} T^k \tag{1}$$

Constraints. The following constraints ensure that all customer requests are fulfilled through feasible and operationally consistent routing plans. They enforce exact customer service, route continuity, depot start and end, valid trailer assignments, capacity compliance, service time adherence, precedence for pickups and deliveries, grouped order handling, equipment compatibility (e.g., tail lift), and define valid variable domains.

Customer service assignment. Each customer node must be visited exactly once:

$$\sum_{k\in K} y_i^k = 1 \qquad \forall i \in N \setminus \{0\} \tag{2}$$

Table 2
Model notation.

Sets	
N	Set of all nodes (including depot 0)
N^+	Set of pickup nodes
N^-	Set of delivery nodes
K	Set of vehicles
L	Set of trailers
$\delta^+(i)$	Set of outgoing arcs from node $i \in N$
$\delta^-(i)$	Set of incoming arcs to node $i \in N$
Parameters	
c_k^{dist}	Distance-based cost rate for vehicle k [€/km]
c_l^{dist}	Distance-based cost rate for trailer l [€/km]
c_k^{time}	Time-based labor cost for vehicle k [€/min]
$d_{i,j}$	Distance between nodes i and j
Q_k^{weight}	Max weight capacity of vehicle k [kg]
Q_k^{pallet}	Max pallet capacity of vehicle k [pallets]
Q_l^{weight}	Max weight capacity of trailer l [kg]
Q_l^{pallet}	Max pallet capacity of trailer l [pallets]
q_i^{weight}	Weight at node i
q_i^{pallet}	Pallets at node i
e_i	Earliest arrival time at node i
l_i	Latest arrival time at node i
s_i	Service duration at node i
τ_k	Speed factor (travel time per distance) for vehicle k [min/km]
$T^{\max}$	Maximum allowed tour duration
a_k	1 if vehicle k is trailer-compatible, 0 otherwise
b_k	1 if vehicle k has tail lift, 0 otherwise
b_i	1 if node i requires tail lift, 0 otherwise
j_i	Job ID (links pickup and delivery for job i)
g_i	Order group ID (bundles sub-jobs)
M	A large constant
Decision and auxiliary variables	
$x_{i,j}^k$	1 if vehicle k travels directly from node i to j, 0 otherwise
$z^{k,l}$	1 if trailer l is assigned to vehicle k, 0 otherwise
y_i^k	1 if node i is visited by vehicle k, 0 otherwise
D^k	Total distance traveled by vehicle k
$D^{k,l}$	Total distance driven by vehicle k while towing trailer l
t_i^k	Arrival time of vehicle k at node i
T^k	Total tour duration for vehicle k
u_i^k	Cumulative weight on vehicle k (including assigned trailer, if any) at node i
v_i^k	Cumulative pallet load on vehicle k (including assigned trailer, if any) at node i

Routing flow conservation. Each visited node must be entered and exited exactly once by the assigned vehicle:

$$\sum_{(i,j)\in\delta^+(i)} x_{i,j}^k = y_i^k \qquad \forall i \in N, k \in K \tag{3}$$

$$\sum_{(j,i)\in\delta^-(i)} x_{j,i}^k = y_i^k \qquad \forall i \in N, k \in K \tag{4}$$

Route start and end. Each vehicle starts and ends its route at the depot (node 0):

$$\sum_{j\in N\setminus\{0\}} x_{0,j}^k \le 1 \qquad \forall k \in K \tag{5}$$

$$\sum_{i\in N\setminus\{0\}} x_{i,0}^k \le 1 \qquad \forall k \in K \tag{6}$$

Trailer assignment. Each vehicle may tow at most one trailer, and trailers are exclusive:

$$\sum_{l\in L} z^{k,l} \le a_k \qquad \forall k \in K \tag{7}$$

$$\sum_{k\in K} z^{k,l} \le 1 \qquad \forall l \in L \tag{8}$$

Distance tracking. Total route distances are computed and assigned to vehicles and trailers:

$$D^k = \sum_{i\in N}\sum_{j\in N} d_{i,j} x_{i,j}^k \qquad \forall k \in K \tag{9}$$

$$D^{k,l} \le D^k \quad \forall k \in K, l \in L \tag{10}$$
$$D^{k,l} \le M z^{k,l} \quad \forall k \in K, l \in L \tag{11}$$
$$D^{k,l} \ge D^k - M(1 - z^{k,l}) \quad \forall k \in K, l \in L \tag{12}$$

Load and capacity propagation. To explicitly capture the dynamics of loading and unloading along a route, cumulative load variables are introduced for both weight and pallet space. These variables track the vehicle's weight and pallet utilization after servicing each node and ensure that capacity constraints are respected at every point along the tour, rather than only in aggregate. Weight and pallet loads accumulate along routes and must remain within combined vehicle and trailer capacity:

$$u_0^k = 0 \quad \forall k \in K \tag{13}$$
$$v_0^k = 0 \quad \forall k \in K \tag{14}$$
$$u_j^k \ge u_i^k + q_j^{\text{weight}} x_{i,j}^k - M(1 - x_{i,j}^k) \quad \forall i \in N, j \in N^+, k \in K \tag{15}$$
$$u_j^k \ge u_i^k - q_j^{\text{weight}} x_{i,j}^k - M(1 - x_{i,j}^k) \quad \forall i \in N, j \in N^-, k \in K \tag{16}$$
$$v_j^k \ge v_i^k + q_j^{\text{pallet}} x_{i,j}^k - M(1 - x_{i,j}^k) \quad \forall i \in N, j \in N^+, k \in K \tag{17}$$
$$v_j^k \ge v_i^k - q_j^{\text{pallet}} x_{i,j}^k - M(1 - x_{i,j}^k) \quad \forall i \in N, j \in N^-, k \in K \tag{18}$$
$$u_i^k \le Q_k^{\text{weight}} + \sum_{l \in L} Q_l^{\text{weight}} z^{k,l} \quad \forall i \in N, k \in K \tag{19}$$
$$v_i^k \le Q_k^{\text{pallet}} + \sum_{l \in L} Q_l^{\text{pallet}} z^{k,l} \quad \forall i \in N, k \in K \tag{20}$$

At pickup nodes, the cumulative weight and pallet loads increase according to the corresponding demand, while at delivery nodes they decrease accordingly. By enforcing capacity limits for both dimensions at each node, the formulation prevents intermediate overloads even if the total tour demand would be feasible in aggregate. If a trailer is assigned, its weight and pallet capacities are added to those of the vehicle, allowing larger intermediate loads to be handled consistently.

Grouped delivery. Sub-jobs with the same group ID must be served by the same vehicle:

$$y_i^k = y_j^k \quad \forall i, j \in N \setminus \{0\} \text{ with } g_i = g_j \tag{21}$$

Precedence and timing. Pickups must precede deliveries and tours must respect timing:

$$t_j^k \ge t_i^k + s_i + d_{i,j}\tau_k - M(1 - x_{i,j}^k) \quad \forall i \in N, j \in N \setminus \{0\}, k \in K \tag{22}$$
$$t_j^k \ge t_i^k + s_i + d_{i,j}\tau_k \quad \forall i \in N^+, j \in N^- \text{ with } j_i = j_j \tag{23}$$

Service time windows. Arrival at each node must occur within its predefined time window:

$$t_i^k \ge e_i - M \cdot (1 - y_i^k) \quad \forall i \in N \setminus \{0\}, k \in K \tag{24}$$
$$t_i^k \le l_i + M \cdot (1 - y_i^k) \quad \forall i \in N \setminus \{0\}, k \in K \tag{25}$$

Tour duration. Each tour must comply with the maximum duration policy:

$$T^k \ge t_i^k + s_i + d_{i,0}\tau_k - t_0^k \quad \forall i \in N \setminus \{0\}, k \in K \tag{26}$$
$$T^k \le T^{\max} \quad \forall k \in K \tag{27}$$

Service equipment. Only vehicles with tail lifts may visit nodes that require them:

$$y_i^k b_i \le b_k \quad \forall i \in N \setminus \{0\}, k \in K \tag{28}$$

Variable domains. All variables must adhere to their respective binary or continuous domains:

$$x_{i,j}^k \in \{0,1\} \quad \forall i \in N, j \in N, k \in K \tag{29}$$
$$z^{k,l} \in \{0,1\} \quad \forall k \in K, l \in L \tag{30}$$
$$y_i^k \in \{0,1\} \quad \forall i \in N, k \in K \tag{31}$$
$$u_i^k, v_i^k, t_i^k \in \mathbb{R}_0^+ \quad \forall i \in N, k \in K \tag{32}$$
$$D^k, T^k \in \mathbb{R}_0^+ \quad \forall k \in K \tag{33}$$
$$D^{k,l} \in \mathbb{R}_0^+ \quad \forall k \in K, l \in L \tag{34}$$

5. Matheuristic solution approach

This section presents the matheuristic solution approach developed to solve the vehicle routing problem introduced in this study. The approach combines a heuristic tour construction phase with a subsequent mathematical optimization step that selects an optimal subset of tours. This hybrid design leverages the ability of heuristics to quickly produce diverse and feasible routes and the strength of optimization models to identify cost-optimal decisions under complex constraints. This section is structured into two parts: The mathematical tour selection model in Section 5.1 and the heuristic for generating feasible tours in Section 5.2. Appendix B provides a consolidated overview of the mathematical formulation of the tour selection phase used in the solution approach.

5.1. Mathematical formulation of the tour selection phase

The optimization phase relies on a set partitioning formulation that selects a subset of precomputed feasible tours. All relevant symbols and variables are summarized in Table 3. The notation follows the convention introduced in the main model formulation. Tour-specific attributes are indexed by $r \in R$, while vehicle- and trailer-related parameters are defined in Section 4.

Table 3
Notation for the set partitioning-based tour selection model.

Sets	
R	Set of precomputed feasible tours
N_r	Set of customer nodes included in tour $r \in R$
Parameters	
D_r	Total distance of tour r [km]
T_r	Total duration of tour r [min]
Q_r^{weight}	Total weight requirement of tour r [kg]
Q_r^{pallet}	Total pallet requirement of tour r [pallets]
τ_r	Average travel speed or speed index requirement of tour r
b_r	1 if tour r requires a tail lift, 0 otherwise
Decision variables	
x_r^k	1 if tour r is assigned to vehicle k, 0 otherwise
z_r^l	1 if trailer l is assigned to tour r, 0 otherwise

Note: The parameters D_r, T_r, Q_r^{weight}, Q_r^{pallet}, τ_r, and b_r are computed during the heuristic preprocessing phase and serve as input to the tour selection model.

Objective. The objective function minimizes the overall costs of selected tours by accounting for vehicle distance costs, trailer distance costs, and time-dependent labor expenses.

$$\min C = \sum_{r\in R}\sum_{k\in K} c_k^{\text{dist}} D_r x_r^k + \sum_{r\in R}\sum_{l\in L} c_l^{\text{dist}} D_r z_r^l + \sum_{r\in R}\sum_{k\in K} c_k^{\text{time}} T_r x_r^k \tag{35}$$

Constraints. The following constraints ensure that tour assignments are feasible, mutually exclusive, and operationally valid. They enforce full customer coverage, respect vehicle and trailer capacity limits, ensure equipment compatibility (e.g., tail lift), comply with timing restrictions, and define valid variable domains.

Vehicle and trailer assignment. Each vehicle and trailer may be assigned to at most one tour to ensure exclusivity and avoid resource conflicts:

$$\sum_{r\in R} x_r^k \le 1 \qquad \forall k \in K \tag{36}$$

$$\sum_{r\in R} z_r^l \le 1 \qquad \forall l \in L \tag{37}$$

Trailer compatibility. A trailer can only be assigned to a tour if the assigned vehicle is capable of towing it:

$$\sum_{l\in L} z_r^l \le \sum_{k\in K} a_k x_r^k \qquad \forall r \in R \tag{38}$$

Customer coverage. Each customer node must be visited exactly once by a selected tour, ensuring complete and non-redundant service:

$$\sum_{r\in R: i\in N_r}\sum_{k\in K} x_r^k = 1 \quad \forall i \in N \setminus \{0\} \tag{39}$$

Capacity constraints. The total weight and pallet demand of a tour must not exceed the combined capacities of the assigned vehicle and trailer:

$$Q_r^{\text{weight}} x_r^k \le Q_k^{\text{weight}} + \sum_{l\in L} Q_l^{\text{weight}} z_r^l \qquad \forall r \in R, k \in K \tag{40}$$

$$Q_r^{\text{pallet}} x_r^k \le Q_k^{\text{pallet}} + \sum_{l\in L} Q_l^{\text{pallet}} z_r^l \qquad \forall r \in R, k \in K \tag{41}$$

Capacity feasibility is enforced based on peak load requirements computed for each tour during the heuristic preprocessing phase. For every candidate tour, the exact sequence of pickup and delivery operations is simulated, and vehicle load is dynamically updated along the route. The parameters Q_r^{weight} and Q_r^{pallet} therefore represent the maximum cumulative weight and pallet load encountered at any point along tour r. A tour can only be assigned to a vehicle (and optional trailer) whose combined capacity is sufficient to cover these peak requirements. This guarantees that all intermediate loading and unloading operations remain feasible without requiring explicit load propagation in the tour selection model.

Timing constraints. Tours must be executable within the travel speed limitations of the assigned vehicle, ensuring feasibility of duration:

$$\sum_{k \in K} \tau_k x_r^k \leq \tau_r \qquad \forall r \in R \tag{42}$$

Timing feasibility is enforced through a preprocessing-based speed requirement. During tour generation, each candidate tour is evaluated with respect to travel times, service times, waiting times induced by time windows, and a maximum tour duration constraint. Based on this evaluation, the minimum vehicle speed factor required to execute the tour feasibly is determined and stored as parameter τ_r. In the tour selection model, a tour can only be assigned to a vehicle whose speed factor τ_k is at least as high as the required tour-specific value. This ensures that all timing constraints are satisfied without explicitly modeling arrival times in the set partitioning formulation.

Tail lift requirements. Tours that include nodes requiring a tail lift may only be assigned to vehicles equipped with this feature:

$$b_r x_r^k \leq b_k \qquad \forall r \in R, k \in K \tag{43}$$

Variable domains. All decision variables must adhere to their respective binary domains, indicating assignment decisions:

$$x_r^k \in \{0,1\} \qquad \forall r \in R, k \in K \tag{44}$$

$$z_r^l \in \{0,1\} \qquad \forall r \in R, l \in L \tag{45}$$

The formulation guarantees feasible assignments under all operational constraints, provided that the preprocessing heuristic generates a sufficiently rich and covering tour set, which depends critically on the group size parameter N (see next subsection).

5.2. Heuristic tour construction

Before the optimization model can be applied, a pool of feasible tours must be generated using a structured preprocessing algorithm. The pseudocode of the algorithm is provided in Appendix C. As input, the heuristic requires the full set of customer nodes with associated parameters (e.g., pickup and delivery pairing, load volume, time windows, and tail lift requirements), as well as vehicle and trailer attributes such as capacity, speed factors, and equipment availability. These parameters are used to generate and validate tour feasibility, and to compute the inputs required by the mathematical model for tour selection. This section describes the stepwise heuristic for constructing tours based on node grouping, combinatorial permutations, and constraint-based filtering.

It is important to note that, in the context of the real-world setting considered, the majority of feasible tours comprise only a limited number of stops. Consequently, the total number of nodes per tour remains constrained, which allows for the use of a brute-force permutation approach. This makes it possible to evaluate all valid sequences without excessive computational burden, thereby eliminating the need for advanced heuristic techniques.

Step 1: Node grouping. Customer nodes are grouped such that all locations associated with the same order, including its sub-tasks, are placed in a common group. This ensures that each group contains all required pickup and delivery points that must be served together within a single tour.

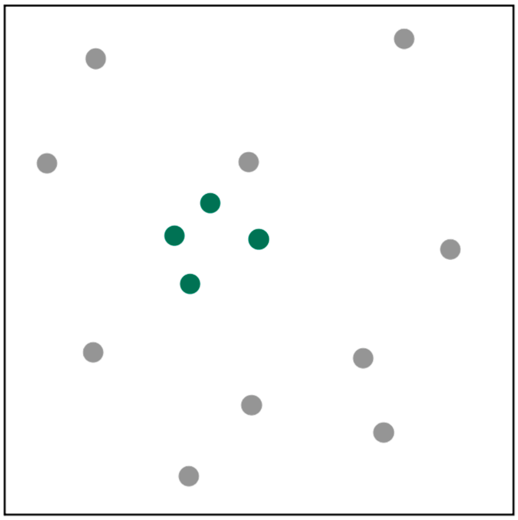


Fig. 1. Grouping of nodes of the same order.

Step 2: Group expansion. Each initial group is augmented with its nearest unassigned pickup-delivery pairs until a maximum group size of $2N$ nodes (i.e., N jobs) is reached. The selection ensures that pickup and delivery remain paired. Here, “nearest” is defined in terms of spatial proximity, measured by the distance between the geographical coordinates of unassigned pickup–delivery pairs and the nodes already contained in the group.

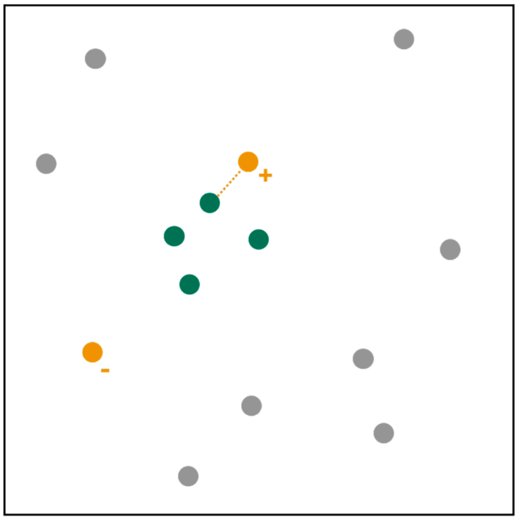


Fig. 2. Nearest-pair-based expansion of groups.

Step 3: Combination generation. All valid node combinations are generated for group sizes ranging from a problem-specific lower bound $n_{\min}$ up to the group size limit $2N$. The lower bound $n_{\min}$ is determined by the maximum number of suborders associated with any single parent order in the dataset, ensuring that node groups containing such orders can generate valid combinations that include all their dependent suborders. The upper bound of $2N$ corresponds to the maximum size of the previously expanded groups, ensuring that all relevant combinations are considered without exceeding the scope of the group. Each combination must include both nodes of any pickup–delivery pair and all dependent suborders.

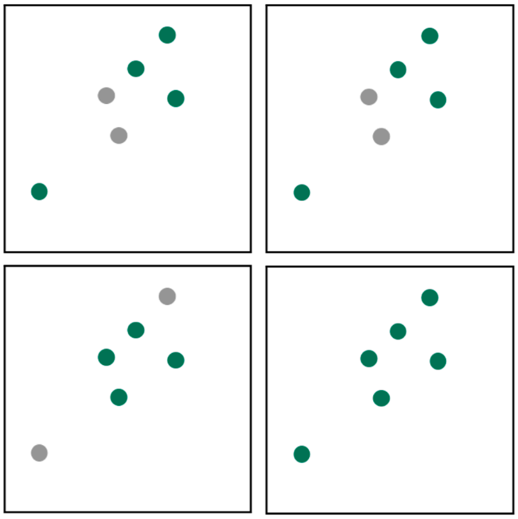


Fig. 3. Generation of valid combinations for group sizes.

Step 4: Permutation and tour building. Valid permutations are created for each combination, respecting pickup-before-delivery precedence. The depot node is added as start and end to form full tours.

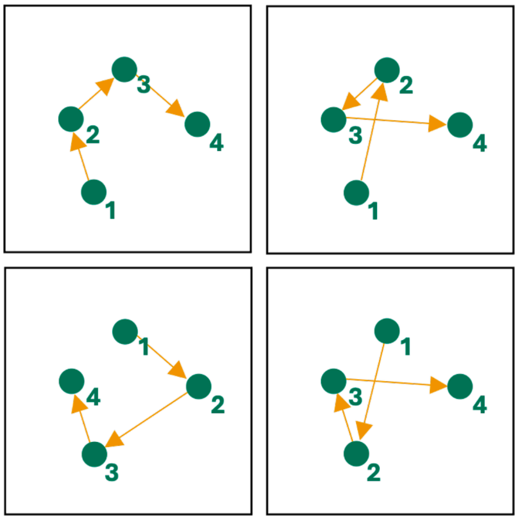


Fig. 4. Tour permutations with precedence constraints.

Step 5: Validity checking. Each tour is tested against vehicle class constraints including time windows, maximum route duration, payload, pallet space, and tail-lift availability. Tours violating any condition are discarded. Note that the cross represents the depot.

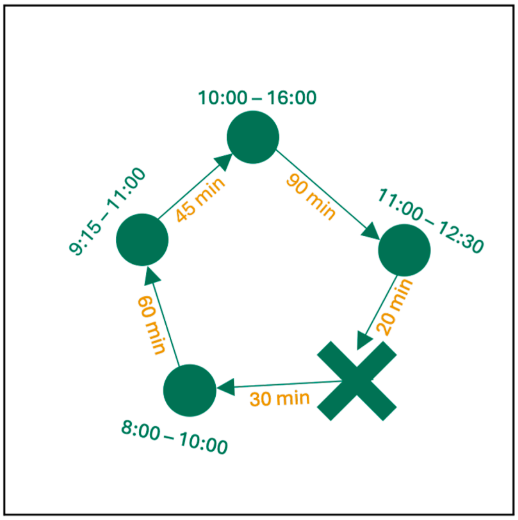


Fig. 5. Filtering of invalid tours.

Step 6: Best tour selection. For each vehicle class and node combination, the shortest valid tour is retained. This limits the final pool to at most one candidate tour per vehicle class and combination.

Tour attributes for optimization. Each accepted tour is annotated with parameters required for optimization: total distance D_r, duration T_r, max weight Q_r^{weight}, max pallets Q_r^{pallet}, speed factor τ_r, and a tail-lift flag b_r. To avoid redundant computation, tour attributes are computed once during the feasibility check and stored for use in the optimization model.

Design rationale of the enumeration-based route generation. In contrast to most set-partitioning matheuristics, where the quality of the final solution depends heavily on heuristic route construction procedures, the present approach adopts a structured enumeration strategy within bounded node groups. This design choice is motivated by the empirical observation that, in the considered application, feasible tours typically contain only a small number of stops. As a result, it becomes computationally tractable to evaluate all feasible permutations for each group while fully respecting pickup-and-delivery precedence and operational constraints.

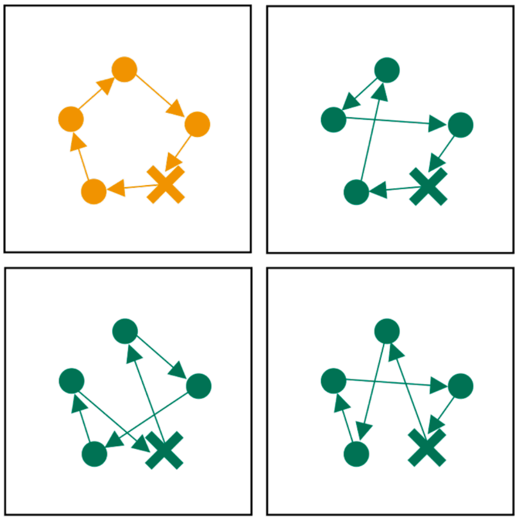


Fig. 6. Selection of the shortest valid tour.

Table 4
Instance properties used for the experiments.

Property	Instance I	Instance II	Instance III
Total orders	27	27	22
Suborders	11	13	7
Total nodes	55	55	45
Tail-lift nodes	8	10	4
Weight (mean ± std) [kg]	1045.2 ± 1368.3	949.0 ± 1232.7	599.6 ± 594.8
Pallets (mean ± std)	4.80 ± 5.81	5.11 ± 4.54	7.08 ± 6.34
Time window (mean ± std) [min]	402.2 ± 248.6	479.4 ± 366.2	356.6 ± 215.8
Distance (mean ± std) [km]	89.9 ± 76.5	81.3 ± 82.2	64.7 ± 60.9

A key advantage of this approach is that it guarantees the identification of the best feasible tour for each node combination and vehicle class, thereby eliminating the approximation bias inherent in heuristic route generation. Moreover, all constraints including capacity limits, time windows, grouped sub-orders, and equipment requirements, are enforced exactly during the generation phase. This avoids the need for penalty mechanisms, repair procedures, or feasibility restoration steps that are commonly required in neighborhood-based metaheuristics.

From a methodological perspective, this shifts the role of the matheuristic: instead of relying on stochastic search to discover high-quality routes, the approach constructs a high-quality and fully feasible route pool deterministically, and delegates the combinatorial decision-making entirely to the set-partitioning model. While this strategy would not scale to large instances with many stops per tour, it is particularly well suited for medium-sized, highly constrained routing problems, where solution quality depends more on accurately capturing operational constraints than on exploring large neighborhoods.

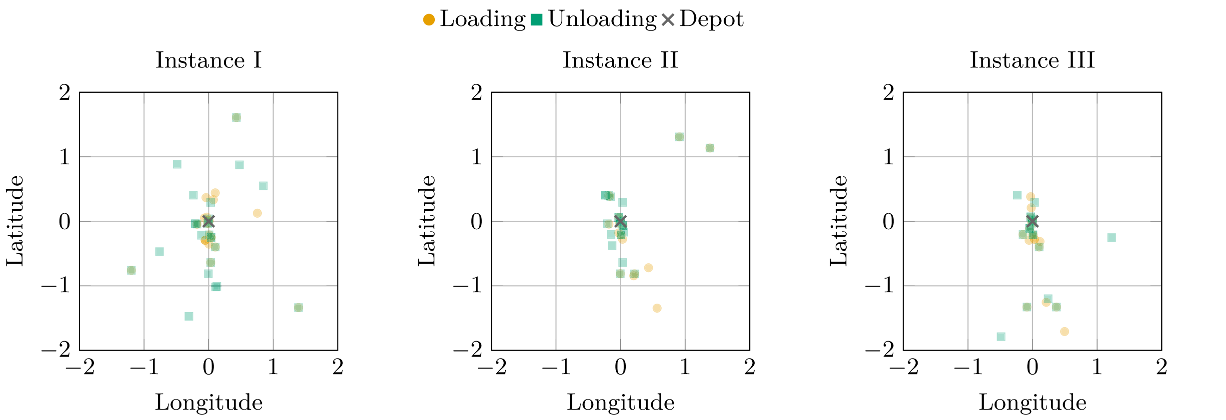
6. Numerical experiments

This section presents an extensive set of computational experiments designed to assess the performance and practical applicability of the proposed solution methods. The evaluation is based on real-world data from a mid-sized logistics provider. Three delivery days, referred to as Instance I, II, and III, are used to benchmark the methods. The experiments focus on two main questions: (1) how does the parameter N, governing group size in the heuristic, affect performance; and (2) how do the matheuristic and mathematical models compare in terms of cost, runtime, and solution quality, relative to each other and to the operational baseline.

All experiments were conducted on a Linux workstation equipped with an Intel Core i7-8700 processor (3.20 GHz) and 64 GB of RAM. The implementation was done in Python 3.12.3 using Gurobi 12.0.3 as the optimization solver.

6.1. Test instances and fleet overview

The three instances differ in the number of orders, node locations, and service requirements. For each order, pickup and delivery nodes are defined, with time windows and a fixed 10-minute service time, which is used as a representative planning assumption specified by the case company. Additionally, the maximum duration for any tour is limited to 9 hours, reflecting legal driving regulations and shift constraints. Table 4 summarizes the characteristics of the three instances.



(a) Instance I (b) Instance II (c) Instance III

Fig. 7. Node distributions across the three instances.

Table 5
Fleet composition and asset features.

Class	Units	Trailer-compatible	Tail-lift
Van	1	0	0
Transporter	4	1	0
Truck ≤ 3.5 t	12	3	7
Truck ≤ 7.5 t	8	1	5
Truck ≤ 16 t	4	2	2
Subtotal (vehicles)	**29**	**7**	**14**
Trailer	9	–	–

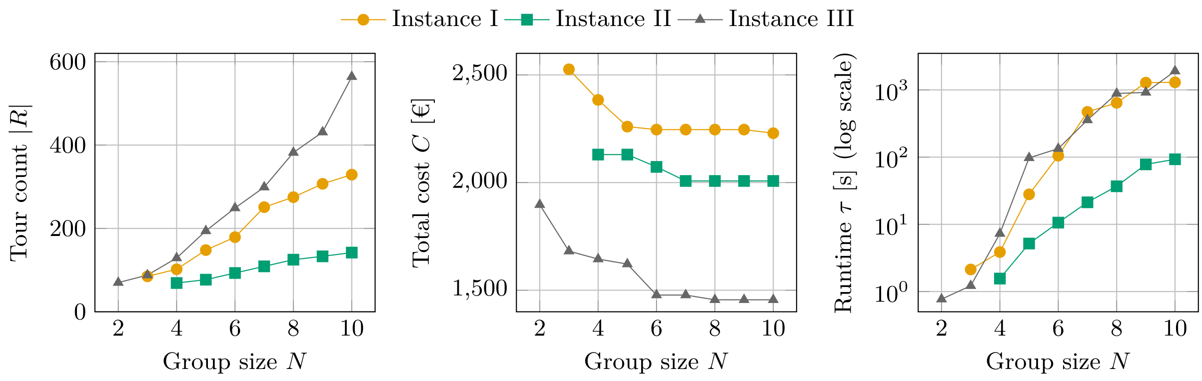
Table 6
Technical and cost specifications of vehicles and trailers.

Class	Payload [kg]	Pallet slots	Avg. speed [km/h]	Distance cost [€/km]	Labor cost [€/h]
Van	510–510	1–1	80	0.40	13
Transporter	1200–1300	3–4	75	0.50	14
Truck ≤ 3.5 t	680–1315	5–8	70	0.60	14.5
Truck ≤ 7.5 t	1905–3200	15–15	60	0.70	17
Truck ≤ 16 t	5550–7800	17–18	55	0.80	19
Trailer	2000–2664	10–12	–	0.12	–

The reported number of suborders reflects all delivery jobs that are dependent on a parent order and must be jointly served. Parent orders themselves are not included in the suborder count. Each suborder belongs to exactly one parent, but a parent may include multiple suborders. The total node count includes the depot as well as one pickup and one delivery location for each (sub-)order. Tail-lift nodes indicate any location requiring tail-lift equipment. Distance statistics are computed over all pairwise arcs within each instance. Distances between customer locations are computed using great-circle distances based on latitude and longitude, calculated via the Haversine formula (Sinnott, 1984). To approximate realistic road travel distances, these values are multiplied by a constant correction factor of 1.3. Country-level circuity estimates reported by Ballou et al. (2002) typically lie above 1.1 and are often around 1.2–1.5 for many industrialized countries, depending on network structure. The chosen factor lies within this empirically observed range.

Fig. 7 visualizes the spatial distribution of nodes in each instance, including loading locations, unloading locations, and the central depot. Some locations serve as both pickup and delivery points, often due to suborder structures. The spatial distribution of nodes varies substantially across instances, influencing the routing complexity.

The experiments are based on the real vehicle and trailer fleet of the case company. Fleet composition, trailer compatibility, and tail-lift availability were obtained from the company's fleet management and dispatching records. Vehicle payload capacities, pallet capacities, and technical characteristics correspond to the specifications of the deployed vehicles. Distance-based operating costs and labor costs were provided by the company and reflect internally used planning and accounting values. Details of the heterogeneous fleet, including vehicle classes and trailers, along with equipment compatibility (trailer and tail-lift support), capacity ranges, and operational costs, are shown in Tables 5 and 6.



(a) Generated tours (b) Total cost (c) Runtime

Fig. 8. Sensitivity to group size N: number of tours, cost, and runtime for three instances.

The average driving speeds are based on historical operational data provided by the case company. They reflect predominantly rural and inter-urban transport operations rather than dense urban traffic, which explains the comparatively high average values (50–80 km/h depending on vehicle class). The reported speeds are empirical averages and therefore already capture typical operational behavior, including trips with and without trailers. Only transporter and truck classes are trailer-compatible in the considered fleet, vans are not equipped for trailer usage and are therefore not affected. Potential speed reductions associated with trailer usage are thus implicitly reflected in the observed averages. To ensure consistency between the model and the company's actual operating conditions, vehicle-specific average speeds are applied unchanged, irrespective of trailer assignment. The impact of trailer usage is instead modeled explicitly through additional distance-based operating costs.

6.2. Sensitivity to group size

The heuristic's grouping parameter N governs the number of orders (i.e., pickup-delivery pairs) included in each candidate tour. As detailed in Section 5.2, increasing N expands the tour space and affects both the number of feasible combinations and the maximum stop count per tour. For computational tractability, combinations are capped at $N = 10$ (corresponding to 10 orders or 20 stops).

We analyze how the group size parameter N in the heuristic tour generation phase affects overall performance. Table D.9 reports the number of generated tours $|R|$, total cost C, and computation time τ for various values of N. As expected, increasing N enlarges the solution space, leading to potentially better solutions but longer runtimes. Minimum group sizes apply due to structural constraints in the data: Instance I requires $N \geq 3$, Instance II $N \geq 4$ and Instance III $N \geq 2$, as certain orders contain multiple suborders that must be grouped together to form feasible tours.

Fig. 8a, b, and c visualize the sensitivity of the heuristic solution quality and runtime to the group size parameter N. As N increases, the number of feasible tours grows significantly, enabling better cost optimization. However, the marginal improvement in cost diminishes beyond $N = 6$, while runtime increases rapidly.

We conclude that $N = 6$ offers a suitable balance between solution quality and computational effort, and use this value for all subsequent comparisons.

6.3. Solution quality comparison

To evaluate solution quality, we compare the proposed matheuristic against two distinct baselines:

- The **exact mathematical model**, solved using Gurobi with a runtime limit of 8 hours to serve as a strong lower-bound benchmark;
- The **company's historical operational plans**, representing the current status quo and manual planning efficiency.

The comparison covers all three benchmark instances (I–III), each evaluated at multiple instance sizes (i.e., node counts). Table 7 summarizes the results for each instance and solution method. Reported values for the number of selected tours $|R*|$, total cost C, and computation time τ represent the mean ± standard deviation over five independent runs.

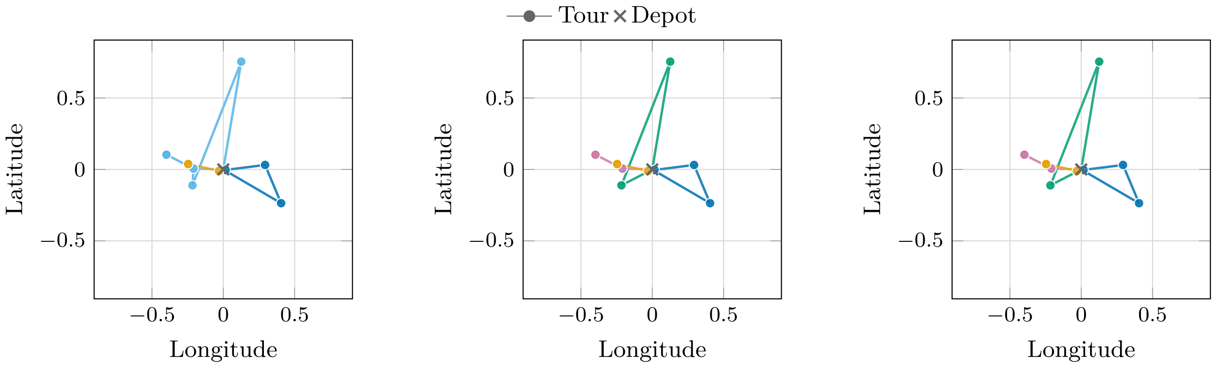
Fig. 9 shows the routing solutions for Instance I with $|i| = 15$ across the three planning approaches. While the mathematical model finds an optimal solution with three routes, the matheuristic uses four routes due to a different consolidation of orders. In this instance, the matheuristic and the operational solution exhibit the same route structures, but the matheuristic achieves lower total cost through a more effective assignment of vehicles and trailers. The difference to the optimal solution can be attributed to missing route combinations in the generated route pool, which prevents a more efficient consolidation.

Fig. 10 compares the total routing costs obtained by the mathematical model, the proposed matheuristic, and the company's historical (operational) plans across increasing instance sizes for all three test cases. For small instances, both optimization-based approaches yield similar solutions, with the mathematical model achieving slightly lower costs when it can be solved to optimality

Table 7
Comparison of solutions across all instances.

Instance	$\|i\|$	Mathematical model			Matheuristic			Operational	
		$\|R*\|$	C [€]	τ [s]	$\|R^*\|$	C [€]	τ [s]	$\|R^*\|$	C [€]
I	15	3 ± 0	364.86* ± 0	299.92 ± 32.47	4 ± 0	384.05 ± 0	4.25 ± 0.22	4	468.39
I	25	3 ± 0	831.90 ± 0	28,800 ± 0	4 ± 0	850.23 ± 0	14.78 ± 0.24	5	987.18
I	35	4.50 ± 0.71	1205.67 ± 18.65	28,800 ± 0	6 ± 0	1252.74 ± 0	16.53 ± 0.21	7	1447.28
I	45	6.50 ± 0.71	1701.36 ± 12.35	28,800 ± 0	8 ± 0	1735.63 ± 0	81.17 ± 0.92	9	1934.69
I	55	–	–	28,800 ± 0	9 ± 0	2246.01 ± 0	113.49 ± 0.49	10	2749.31
II	15	4 ± 0	563.13* ± 0	21.77 ± 1.70	4 ± 0	566.76 ± 0	4.82 ± 0.19	4	669.99
II	25	6.20 ± 0.45	1302.07 ± 2.12	28,800 ± 0	6 ± 0	1304.75 ± 0	6.02 ± 0.12	6	1458.15
II	35	6.25 ± 0.50	1632.82 ± 22.94	28,800 ± 0	7 ± 0	1674.02 ± 0	15.88 ± 0.29	8	1897.24
II	45	7.50 ± 0.71	1777.89 ± 32.16	28,800 ± 0	10 ± 0	1850.97 ± 0	19.28 ± 0.35	10	2084.26
II	55	–	–	28,800 ± 0	10 ± 0	2072.76 ± 0	12.07 ± 0.21	11	2332.67
III	15	2 ± 0	752.76* ± 0	48.14 ± 5.48	2 ± 0	752.89 ± 0	27.69 ± 0.40	5	946.54
III	25	5.20 ± 0.45	844.97* ± 0	24955.81 ± 3068.36	5 ± 0	849.92 ± 0	11.06 ± 0.22	6	1072.63
III	35	6 ± 0	1305.34 ± 9.29	28,800 ± 0	7 ± 0	1383.81 ± 0	54.61 ± 0.35	10	1872.11
III	45	6.25 ± 0.50	1450.51 ± 45.71	28,800 ± 0	8 ± 0	1477.65 ± 0	144.64 ± 3.27	12	1995.01

* in the cost column indicates that the solution is proven optimal by the mathematical model.



(a) Mathematical (b) Matheuristic (c) Operational

Fig. 9. Route comparison across the three solution approaches for instance I with 15 nodes.

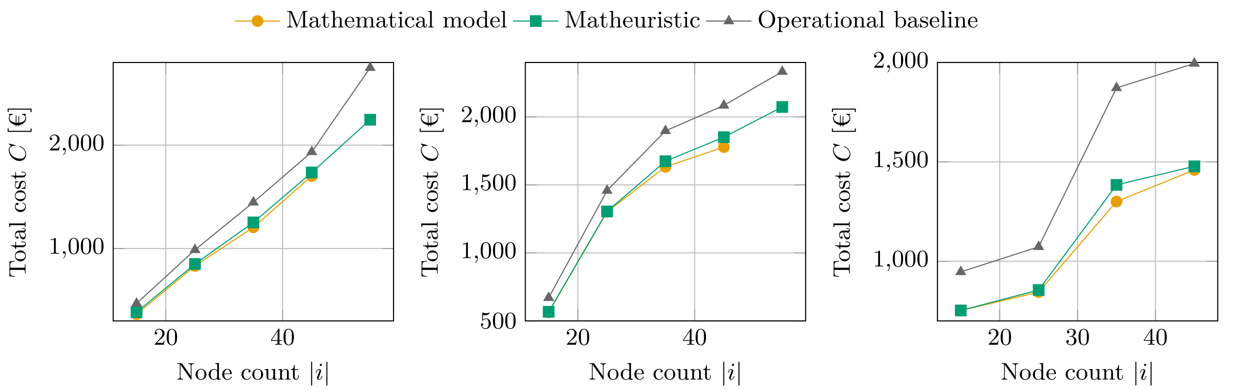
within the time limit. As the number of nodes increases, the exact model becomes computationally impractical; specifically, for instances with 55 nodes (Instance I and II), it failed to find any feasible solution within the 8-hour time limit. In contrast, the matheuristic remains scalable and continues to produce high-quality solutions. In all instances and sizes, both optimization-based approaches consistently outperform the operational baseline, demonstrating the practical effectiveness of the proposed method.

The following figures provide a summary comparison of the two solution methods across all instances. Fig. 11a shows the relative cost savings compared to the operational baseline, averaged across all tested instance sizes (15–55 nodes). While the mathematical model yields slightly higher savings (average 18.43%), it is limited by computational constraints. The matheuristic achieves nearly equivalent savings (16.70%) and performs consistently across all instance sizes. Fig. 11b highlights the runtime trend. The matheuristic remains computationally practical, requiring under 3 minutes even for instances that are considered large in the industry setting, whereas the mathematical model consistently reaches the 8-hour runtime limit on larger instances without proving optimality.

6.4. Cost structure

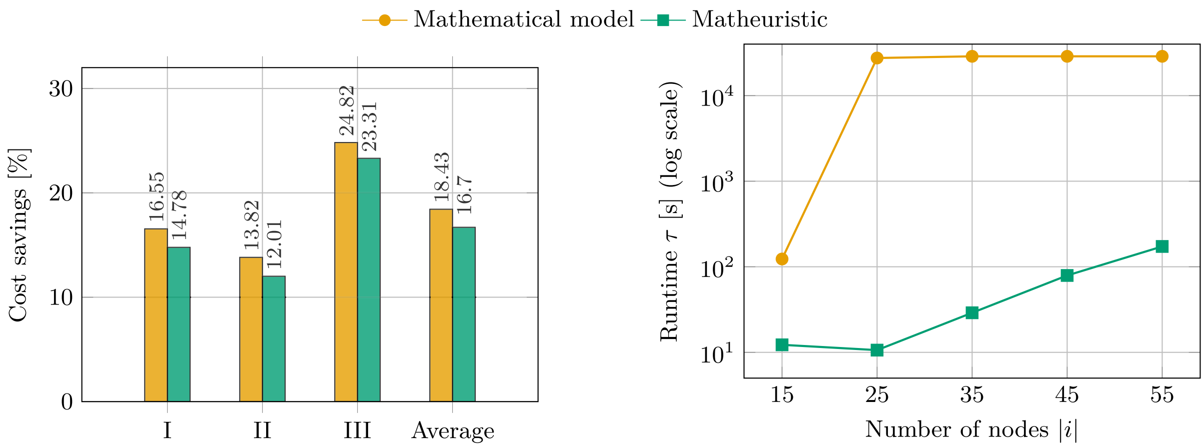
Fig. 12 shows the average cost structure across the three planning approaches. Vehicle-related costs constitute the largest component across all scenarios, primarily reflecting distance-proportional expenses such as fuel consumption, maintenance, and depreciation. Both optimization approaches (mathematical and matheuristic) exhibit substantially higher trailer usage (approximately four times that observed in the operational plans) indicating a more efficient utilization of available transport capacity. While we refrain from quantifying downstream effects such as emission savings in detail, these improvements suggest a general reduction in distance-related impacts. In contrast, the operational baseline incurs noticeably higher labor costs, which can be attributed to the execution of a larger number of shorter and less efficiently utilized tours, requiring more staff and vehicles.

These differences may stem from practical considerations beyond pure cost minimization. For instance, dispatchers might aim to balance workloads among drivers or avoid excessive tour durations, neither of which is explicitly modeled. This trade-off between



(a) Instance I (b) Instance II (c) Instance III

Fig. 10. Cost performance comparison across varying instance sizes.



(a) Relative cost savings (b) Runtime by instance size

Fig. 11. (a) Average cost savings vs. operational baseline (aggregated over all instance sizes). (b) Runtime trend for both methods.

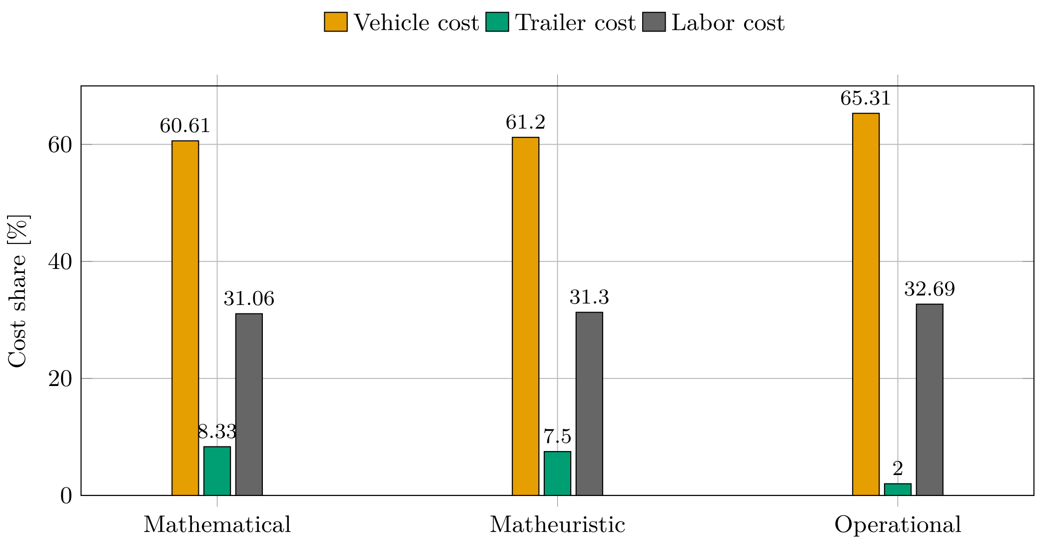


Fig. 12. Average cost breakdown by planning approach.

Table 8
Impact of trailer unavailability: Baseline (with trailers) vs. restricted fleet (no trailers).

	Instance I			Instance II			Instance III			Avg.
Metric	Base	Restr.	Δ	Base	Restr.	Δ	Base	Restr.	Δ	Δ
Total Cost [€]	2246.01	2443.22	+8.78%	2072.76	2398.33	+15.71%	1477.65	1873.21	+26.77%	+17.09%
Vehicle Cost [€]	1375.77	1609.65	+17.00%	1273.80	1594.43	+25.17%	874.36	1248.05	+42.74%	+28.30%
Time Cost [€]	718.96	833.57	+15.97%	664.08	803.89	+21.07%	481.94	625.17	+29.72%	+22.25%
Tours	9	9	0.00%	10	11	+10.00%	8	9	+12.50%	+7.50%
Runtime [s]	113.49	78.16	-31.13%	12.07	9.42	-21.96%	144.64	111.38	-23.00%	-25.36%

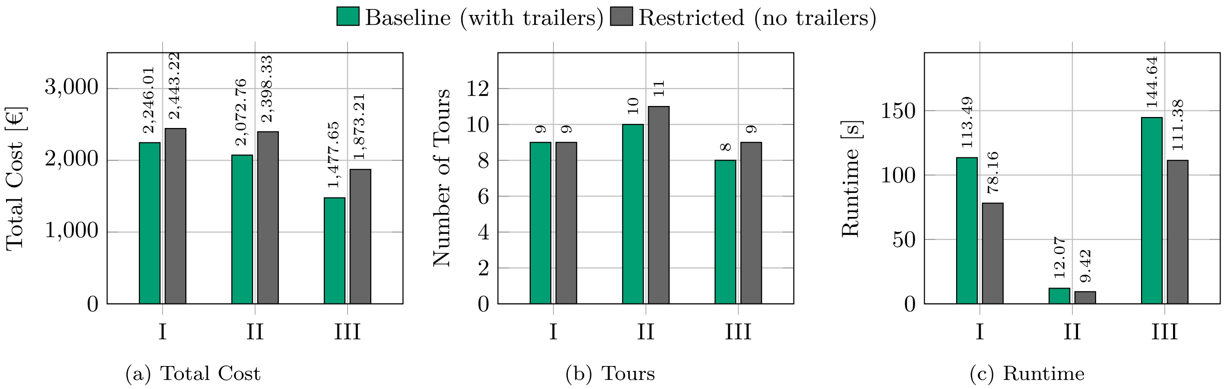


Fig. 13. Impact of trailer unavailability on (a) cost, (b) fleet size, and (c) computational performance across all instances.

equity and efficiency is well-documented in the vehicle routing literature. Matl et al. (2018) provide a comprehensive overview of equity-based criteria in routing models, highlighting how fairness constraints such as balancing total driving time or tour length across drivers can influence routing decisions and increase operational realism. In the present study, such considerations were deliberately excluded to isolate cost-based performance. However, future research may benefit from incorporating workload balancing objectives to better capture operational preferences and align model outputs with human planning behavior. Additionally, it is possible that recent shifts in cost structures, such as rising expenses for certain vehicle classes, have not yet fully influenced dispatcher behavior. This could make trailer-based combinations appear more favorable in optimization. While the models are cost-driven, including workload balance as an additional criterion could better reflect operational preferences.

6.5. *Impact of trailer availability*

To quantify the explicit economic benefit of the heterogeneous fleet strategy, specifically the utilization of trailers, we conducted a counterfactual experiment. In this scenario, we solved the same three benchmark instances (at full size) using the proposed matheuristic exclusively, but with the additional constraint that no trailers were available. This experiment effectively reduced the fleet to a mix of rigid trucks and vans, forcing the algorithm to serve all demands using only powered vehicles.

Table 8 and Fig. 13 summarize the quantitative results compared to the baseline configuration of the matheuristic (with trailers).

The results demonstrate the significant value of the trailer-compatible fleet. When trailers are removed, total costs increase by an average of 17.09% across the three instances. This cost penalty is driven primarily by two factors:

1. **Increased Vehicle & Labor Costs:** Without the ability to couple trailers, the algorithm must deploy more expensive powered vehicles or execute additional tours to transport the same cargo volume. In Instances II and III, the number of required tours increased by one. Consequently, vehicle-related costs rose by an average of 28.30%, and time-based labor costs increased by 22.25%.
2. **Loss of Consolidation Efficiency:** The trailer option allows a single driver to move up to twice the volume in a single leg. Removing this option forces a decomposition of heavy demands into separate trips, reducing the overall network efficiency.

The visualization in Fig. 13 clearly illustrates the trade-off: while eliminating trailers reduces computational runtime (c), it consistently drives up total costs (a) and the number of tours (b) across all instances. Interestingly, the computational runtime decreased by approximately 25% in the no-trailer scenario. This reduction is attributed to the smaller search space in the tour generation phase, as the combinatorial complexity of pairing trucks with compatible trailers is eliminated. However, this gain in speed comes at the expense of a severe loss in solution quality, confirming that the additional computational effort required to model the heterogeneous fleet is economically justified.

7. Conclusion

Summary. This study addressed a practical vehicle routing problem arising in a real-world transportation setting, characterized by a heterogeneous fleet and operational constraints. Two solution approaches were proposed: a pure mathematical optimization model and a matheuristic framework combining heuristic preprocessing with exact optimization. The main objective was to enable cost-efficient and feasible route planning while improving vehicle utilization in small to mid-sized transportation companies where manual planning remains prevalent. The proposed matheuristic method constructs feasible tours using a custom heuristic that respects operational constraints and vehicle characteristics. These tours serve as input to a set partitioning model, which selects the cost-optimal combination. The mathematical model alone provides marginally better solutions, on average 2–3% cheaper, but becomes computationally intractable for larger instances. In contrast, the matheuristic generates solutions within seconds to minutes even for instances with up to 55 customer nodes, making it highly suitable for practical applications. Numerical experiments confirm that both optimization-based methods significantly outperform the company's historical (operational) plans. The matheuristic yields an average cost saving of approximately 16.7%, while the exact model achieves around 18.4% when solvable within an 8-hour time limit. Both methods also show substantially higher trailer utilization compared to the operational baseline, suggesting improved transport capacity management. Furthermore, our counterfactual analysis demonstrates that restricting the fleet to non-trailer vehicles increases total costs by 17.1%, explicitly quantifying the economic value of fleet heterogeneity. From a managerial perspective, the matheuristic offers a scalable decision-support tool for automated tour planning. By intelligently combining orders into feasible tours and leveraging optimization to select among them, it reduces planning effort, and minimizes routing costs. Additional benefits include lower carbon emissions through reduced fleet mileage and fewer tours.

Directions for future work. There remains significant potential to refine the matheuristic. Future improvements could enhance the tour construction phase by incorporating maintenance histories, regulatory constraints such as driver rest times, and driver-tour compatibility based on past assignments. Integration of driver assignment and fairness metrics in the objective function could further align solutions with workforce satisfaction goals. In addition, future research could explicitly model speed differences associated with trailer usage, for example by distinguishing travel speeds for trucks with and without attached trailers, provided sufficiently detailed operational data are available. A promising direction would be a full-scale case study with live operational data, providing deeper insights into model applicability and further opportunities for real-world integration. This study was conducted using real-world data provided by the case company. However, larger-scale scenarios beyond 55 customer nodes could not be tested due to the limited data availability. Furthermore, the quality and granularity of historical order data may influence tour generation and model calibration. Future work should therefore consider extended datasets and stress-testing the method on synthetic large-scale instances and benchmark it against advanced metaheuristic approaches.

CRediT authorship contribution statement

Robin Aninger: Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Methodology, Investigation, Formal analysis, Data curation, Conceptualization; **Stefan Voigt:** Writing – review & editing, Writing – original draft, Validation, Supervision, Methodology, Formal analysis, Conceptualization.

Declaration of competing interest

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

Data availability

The data that has been used is confidential.

Appendix A. Model formulation

This appendix presents the complete formulation of the rich vehicle routing model. For a detailed overview of all sets, parameters, and decision variables, please refer to the notation summary in Table 2.

$$\min C = \sum_{k\in K} c_k^{\text{dist}} D^k + \sum_{k\in K}\sum_{l\in L} c_l^{\text{dist}} D^{k,l} + \sum_{k\in K} c_k^{\text{time}} T^k \tag{A.1}$$

s.t.

$$\sum_{k\in K} y_i^k = 1 \qquad \forall i \in N \setminus \{0\} \tag{A.2}$$

$$\sum_{(i,j)\in\delta^+(i)} x_{i,j}^k = y_i^k \qquad \forall i \in N, k \in K \tag{A.3}$$

$$\sum_{(j,i)\in\delta^-(i)} x_{j,i}^k = y_i^k \qquad \forall i \in N, k \in K \tag{A.4}$$

$$\sum_{j \in N \setminus \{0\}} x_{0,j}^k \leq 1 \qquad \forall k \in K \tag{A.5}$$

$$\sum_{i \in N \setminus \{0\}} x_{i,0}^k \leq 1 \qquad \forall k \in K \tag{A.6}$$

$$\sum_{l \in L} z^{k,l} \leq a_k \qquad \forall k \in K \tag{A.7}$$

$$\sum_{k \in K} z^{k,l} \leq 1 \qquad \forall l \in L \tag{A.8}$$

$$D^k = \sum_{i \in N} \sum_{j \in N} d_{i,j} x_{i,j}^k \qquad \forall k \in K \tag{A.9}$$

$$D^{k,l} \leq D^k \qquad \forall k \in K, l \in L \tag{A.10}$$

$$D^{k,l} \leq M z^{k,l} \qquad \forall k \in K, l \in L \tag{A.11}$$

$$D^{k,l} \geq D^k - M(1 - z^{k,l}) \qquad \forall k \in K, l \in L \tag{A.12}$$

$$u_0^k = 0 \qquad \forall k \in K \tag{A.13}$$

$$v_0^k = 0 \qquad \forall k \in K \tag{A.14}$$

$$u_j^k \geq u_i^k + q_j^{\text{weight}} x_{i,j}^k - M(1 - x_{i,j}^k) \qquad \forall i \in N, j \in N^+, k \in K \tag{A.15}$$

$$u_j^k \geq u_i^k - q_j^{\text{weight}} x_{i,j}^k - M(1 - x_{i,j}^k) \qquad \forall i \in N, j \in N^-, k \in K \tag{A.16}$$

$$v_j^k \geq v_i^k + q_j^{\text{pallet}} x_{i,j}^k - M(1 - x_{i,j}^k) \qquad \forall i \in N, j \in N^+, k \in K \tag{A.17}$$

$$v_j^k \geq v_i^k - q_j^{\text{pallet}} x_{i,j}^k - M(1 - x_{i,j}^k) \qquad \forall i \in N, j \in N^-, k \in K \tag{A.18}$$

$$u_i^k \leq Q_k^{\text{weight}} + \sum_{l \in L} Q_l^{\text{weight}} z^{k,l} \qquad \forall i \in N, k \in K \tag{A.19}$$

$$v_i^k \leq Q_k^{\text{pallet}} + \sum_{l \in L} Q_l^{\text{pallet}} z^{k,l} \qquad \forall i \in N, k \in K \tag{A.20}$$

$$y_i^k = y_j^k \qquad \forall i, j \in N \setminus \{0\} \text{ with } g_i = g_j \tag{A.21}$$

$$t_j^k \geq t_i^k + s_i + d_{i,j} \tau_k - M(1 - x_{i,j}^k) \qquad \forall i \in N, j \in N \setminus \{0\}, k \in K \tag{A.22}$$

$$t_j^k \geq t_i^k + s_i + d_{i,j} \tau_k \qquad \forall i \in N^+, j \in N^- \text{ with } j_i = j_j \tag{A.23}$$

$$t_i^k \geq e_i - M \cdot (1 - y_i^k) \qquad \forall i \in N \setminus \{0\}, k \in K \tag{A.24}$$

$$t_i^k \leq l_i + M \cdot (1 - y_i^k) \qquad \forall i \in N \setminus \{0\}, k \in K \tag{A.25}$$

$$T^k \geq t_i^k + s_i + d_{i,0} \tau_k - t_0^k \qquad \forall i \in N \setminus \{0\}, k \in K \tag{A.26}$$

$$T^k \leq T^{\max} \qquad \forall k \in K \tag{A.27}$$

$$y_i^k b_i \leq b_k \qquad \forall i \in N \setminus \{0\}, k \in K \tag{A.28}$$

$$x_{i,j}^k \in \{0, 1\} \qquad \forall i \in N, j \in N, k \in K \tag{A.29}$$

$$z^{k,l} \in \{0, 1\} \qquad \forall k \in K, l \in L \tag{A.30}$$

$$y_i^k \in \{0, 1\} \qquad \forall i \in N, k \in K \tag{A.31}$$

$$u_i^k, v_i^k, t_i^k \in \mathbb{R}_0^+ \qquad \forall i \in N, k \in K \tag{A.32}$$

$$D^k, T^k \in \mathbb{R}_0^+ \qquad \forall k \in K \tag{A.33}$$

$$D^{k,l} \in \mathbb{R}_0^+ \qquad \forall k \in K, l \in L \tag{A.34}$$

Appendix B. Mathematical formulation of the tour selection phase

This appendix provides the full formulation of the tour selection optimization model. For a detailed overview of all sets, parameters, and decision variables, please refer to the notation summary in Table 3.

$$\min C = \sum_{r \in R} \sum_{k \in K} c_k^{\text{dist}} D_r x_r^k + \sum_{r \in R} \sum_{l \in L} c_l^{\text{dist}} D_r z_r^l + \sum_{r \in R} \sum_{k \in K} c_k^{\text{time}} T_r x_r^k \tag{B.1}$$

s.t.

$$\sum_{r \in R} x_r^k \leq 1 \qquad \forall k \in K \tag{B.2}$$

$$\sum_{r \in R} z_r^l \leq 1 \qquad \forall l \in L \tag{B.3}$$

$$\sum_{l\in L} z_r^l \le \sum_{k\in K} a_k x_r^k \qquad \forall r \in R \tag{B.4}$$

$$\sum_{r\in R: i\in N_r} \sum_{k\in K} x_r^k = 1 \qquad \forall i \in N \setminus \{0\} \tag{B.5}$$

$$Q_r^{\text{weight}} x_r^k \le Q_k^{\text{weight}} + \sum_{l\in L} Q_l^{\text{weight}} z_r^l \qquad \forall r \in R, k \in K \tag{B.6}$$

$$Q_r^{\text{pallet}} x_r^k \le Q_k^{\text{pallet}} + \sum_{l\in L} Q_l^{\text{pallet}} z_r^l \qquad \forall r \in R, k \in K \tag{B.7}$$

$$\sum_{k\in K} \tau_k x_r^k \le \tau_r \qquad \forall r \in R \tag{B.8}$$

$$b_r x_r^k \le b_k \qquad \forall r \in R, k \in K \tag{B.9}$$

$$x_r^k \in \{0,1\} \qquad \forall r \in R, k \in K \tag{B.10}$$

$$z_r^l \in \{0,1\} \qquad \forall r \in R, l \in L \tag{B.11}$$

Appendix C. Heuristic tour construction

Algorithm 1: Heuristic tour construction procedure.

`// Step 1: Group nodes by shared order ID`
$\mathcal{G} \leftarrow \text{GroupNodes}(I, NO_i)$

`// Step 2: Expand each group with nearest node pairs`
for $g \in \mathcal{G}$ **do**
  while $|g| < 2N$ **do**
    $(i, j) \leftarrow \text{FindNearestPair}(g, I)$
    $g \leftarrow g \cup \{i, j\}$
  end
end
`// Step 3: Generate combinations for multiple sizes`
for $n \in \{n_{\min}, n_{\min}+2, \dots, 2N\}$ **do**
  $C \leftarrow \text{GenerateCombinations}(g, n)$

  `// Step 4: Generate valid permutations`
  for $c \in C$ **do**
    $\Pi \leftarrow \text{GeneratePermutations}(c)$
    $\mathcal{T} \leftarrow \text{BuildTours}(\Pi)$

    `// Step 5: Check feasibility per vehicle class`
    for $c \in KC$ **do**
      $\mathcal{T}_{\text{valid}} \leftarrow \text{CheckValidity}(\mathcal{T}, c)$

      `// Step 6: Select shortest valid tour`
      $t^* \leftarrow \arg\min_{t\in\mathcal{T}_{\text{valid}}} D_r$
      $\mathcal{T}_{\text{final}} \leftarrow \mathcal{T}_{\text{final}} \cup \{t^*\}$
    end
  end
end

Appendix D. Detailed numerical results

Table D.9
Sensitivity analysis of group size N.

N	Inst. I ($\|R\|$, C, τ)	Inst. II ($\|R\|$, C, τ)	Inst. III ($\|R\|$, C, τ)
2	–	–	70, €1897.70, 0.70s
3	79, €2526.41, 2.13s	–	92, €1654.14, 1.22s
4	104, €2384.26, 8.67s	69, €2130.13, 1.65s	142, €1631.85, 7.56s
5	148, €2260.24, 35.11s	77, €2130.13, 5.13s	205, €1608.67, 95.06s
6	179, €2246.45, 320.91s	93, €2072.90, 10.12s	249, €1478.76, 129.30s
7	251, €2246.45, 881.60s	109, €2007.64, 20.42s	299, €1478.76, 340.52s
8	268, €2246.45, 1016.64s	125, €2007.64, 36.41s	382, €1456.74, 861.88s
9	305, €2246.45, 3104.06s	133, €2007.64, 77.15s	431, €1456.74, 907.56s
10	329, €2229.51, 3230.63s	154, €2007.64, 137.37s	564, €1456.74, 1857.40s

References

Accorsi, L., Vigo, D., 2020. A hybrid metaheuristic for single truck and trailer routing problems. Transp. Sci. 54 (5), 1351–1371. https://doi.org/10.1287/trsc.2019.0943

Alcaraz, J.J., Caballero-Arnaldos, L., Vales-Alonso, J., 2019. Rich vehicle routing problem with last-mile outsourcing decisions. Transp. Res. E 129, 263–286. https://doi.org/10.1016/j.tre.2019.08.004

Archetti, C., Speranza, M.G., 2014. A survey on matheuristics for routing problems. EURO J. Comput. Optim. 2 (4), 223–246. https://doi.org/10.1007/s13675-014-0030-7

Arenas-Vasco, A., Alcázar, D., Villegas, J.G., 2025. A meta-analysis of set partitioning/set covering based matheuristics for vehicle routing problems. Oper. Res. Perspect. 15, 100357. https://doi.org/10.1016/j.orp.2025.100357

de Armas, J., Melián-Batista, B., 2015. Variable neighborhood search for a dynamic rich vehicle routing problem with time windows. Comput. Ind. Eng. 85, 120–131. https://doi.org/10.1016/j.cie.2015.03.006

de Armas, J., Melián-Batista, B., Moreno-Pérez, J.A., Brito, J., 2015. GVNS for a real-world rich vehicle routing problem with time windows. Eng. Appl. Artif. Intell. 42, 45–56. https://doi.org/10.1016/j.engappai.2015.03.009

Arnold, F., Gendreau, M., Sörensen, K., 2019. Efficiently solving very large-scale routing problems. Comput. Oper. Res. 107, 32–42. https://doi.org/10.1016/j.cor.2019.03.006

Baldacci, R., Battarra, M., Vigo, D., 2008. Routing a Heterogeneous Fleet of Vehicles. Springer US. p. 3–27. https://doi.org/10.1007/978-0-387-77778-8_1

Ballou, R.H., Rahardja, H., Sakai, N., 2002. Selected country circuity factors for road travel distance estimation. Transp. Res. A Policy Pract. 36 (9), 843–848. https://doi.org/10.1016/S0965-8564(01)00044-1

Bartolini, E., Schneider, M., 2020. A two-commodity flow formulation for the capacitated truck-and-trailer routing problem. Discrete Appl. Math. 275, 3–18. https://doi.org/10.1016/j.dam.2018.07.033

Belfiore, P., Yoshida Yoshizaki, H.T., 2009. Scatter search for a real-life heterogeneous fleet vehicle routing problem with time windows and split deliveries in brazil. Eur. J. Oper. Res. 199 (3), 750–758. https://doi.org/10.1016/j.ejor.2008.08.003

Boschetti, M.A., Maniezzo, V., Roffilli, M., Bolufé Röhler, A., 2009. Matheuristics: Optimization, Simulation and Control. Springer Berlin Heidelberg. p. 171–177. https://doi.org/10.1007/978-3-642-04918-7_13

Caceres-Cruz, J., Arias, P., Guimarans, D., Riera, D., Juan, A.A., 2014. Rich vehicle routing problem: survey. ACM Comput. Surv. 47 (2), 1–28. https://doi.org/10.1145/2666003

Ceselli, A., Righini, G., Salani, M., 2009. A column generation algorithm for a rich vehicle-routing problem. Transp. Sci. 43 (1), 56–69. https://doi.org/10.1287/trsc.1080.0256

da Cruz, H. F.A., Salles da Cunha, A., 2023. The profitable single truck and trailer routing problem with time windows: formulation, valid inequalities and branch-and-cut algorithms. Comput. Ind. Eng. 180, 109238. https://doi.org/10.1016/j.cie.2023.109238

Dabia, S., Lai, D., Vigo, D., 2019. An exact algorithm for a rich vehicle routing problem with private fleet and common carrier. Transp. Sci. 53 (4), 986–1000. https://doi.org/10.1287/trsc.2018.0852

Dantzig, G.B., Ramser, J.H., 1959. The truck dispatching problem. Manag. Sci. 6 (1), 80–91. https://doi.org/10.1287/mnsc.6.1.80

Derigs, U., Pullmann, M., Vogel, U., 2013. Truck and trailer routing–problems, heuristics and computational experience. Comput. Oper. Res. 40 (2), 536–546. https://doi.org/10.1016/j.cor.2012.08.007

Derigs, U., Vogel, U., 2013. Experience with a framework for developing heuristics for solving rich vehicle routing problems. J. Heuristics 20 (1), 75–106. https://doi.org/10.1007/s10732-013-9232-z

Doerner, K.F., Schmid, V., 2010. Survey: Matheuristics for Rich Vehicle Routing Problems. Springer Berlin Heidelberg. p. 206–221. https://doi.org/10.1007/978-3-642-16054-7_15

Drexl, M., 2012. Rich vehicle routing in theory and practice. Logist. Res. 5 (1–2), 47–63. https://doi.org/10.1007/s12159-012-0080-2

Errami, N., Queiroga, E., Sadykov, R., Uchoa, E., 2024. VRPSolverEasy: a python library for the exact solution of a rich vehicle routing problem. INFORMS J. Comput. 36 (4), 956–965. https://doi.org/10.1287/ijoc.2023.0103

Federal Statistical Office of Germany (Destatis), 2025. Goods transport.

Hasle, G., Kloster, O., 2007. Industrial Vehicle Routing. Springer Berlin Heidelberg. p. 397–435. https://doi.org/10.1007/978-3-540-68783-2_12

Kerscher, C., Minner, S., 2025. Decompose-route-improve framework for solving large-scale vehicle routing problems with time windows. Transp. Res. E 204, 104409. https://doi.org/10.1016/j.tre.2025.104409

Kramer, R., Cordeau, J.-F., Iori, M., 2019. Rich vehicle routing with auxiliary depots and anticipated deliveries: an application to pharmaceutical distribution. Transp. Res. E 129, 162–174. https://doi.org/10.1016/j.tre.2019.07.012

Lahyani, R., Coelho, L.C., Khemakhem, M., Laporte, G., Semet, F., 2015a. A multi-compartment vehicle routing problem arising in the collection of olive oil in tunisia. Omega 51, 1–10. https://doi.org/10.1016/j.omega.2014.08.007

Lahyani, R., Khemakhem, M., Semet, F., 2015b. Rich vehicle routing problems: from a taxonomy to a definition. Eur. J. Oper. Res. 241 (1), 1–14. https://doi.org/10.1016/j.ejor.2014.07.048

Laporte, G., 2009. Fifty years of vehicle routing. Transp. Sci. 43 (4), 408–416. https://doi.org/10.1287/trsc.1090.0301

Lin, S.-W., Yu, V.F., Lu, C.-C., 2011. A simulated annealing heuristic for the truck and trailer routing problem with time windows. Expert Syst. Appl. 38 (12), 15244–15252. https://doi.org/10.1016/j.eswa.2011.05.075

Liu, F.-H., Shen, S.-Y., 1999. The fleet size and mix vehicle routing problem with time windows. J. Oper. Res. Soc. 50 (7), 721–732. https://doi.org/10.1057/palgrave.jors.2600763

Marques, A., Soares, R., Santos, M.J., Amorim, P., 2020. Integrated planning of inbound and outbound logistics with a rich vehicle routing problem with backhauls. Omega 92, 102172. https://doi.org/10.1016/j.omega.2019.102172
Matl, P., Hartl, R.F., Vidal, T., 2018. Workload equity in vehicle routing problems: a survey and analysis. Transp. Sci. 52 (2), 239–260. https://doi.org/10.1287/trsc.2017.0744
McKinnon, A., Edwards, J., 2010. Opportunities for improving vehicle utilization. In: McKinnon, A., Browne, M., Whiteing, A., Cullinane, S. (Eds.), Green Logistics: Improving the Environmental Sustainability of Logistics, Kogan Page, London, p. 195–210.
Repoussis, P.P., Tarantilis, C.D., 2010. Solving the fleet size and mix vehicle routing problem with time windows via adaptive memory programming. Transp. Res. C Emerg. Technol. 18 (5), 695–712. https://doi.org/10.1016/j.trc.2009.08.004
Ropke, S., Pisinger, D., 2006. An adaptive large neighborhood search heuristic for the pickup and delivery problem with time windows. Transp. Sci. 40 (4), 455–472. https://doi.org/10.1287/trsc.1050.0135
Rothenbächer, A.-K., Drexl, M., Irnich, S., 2018. Branch-and-price-and-cut for the truck-and-trailer routing problem with time windows. Transp. Sci. 52 (5), 1174–1190. https://doi.org/10.1287/trsc.2017.0765
Scheuerer, S., 2006. A tabu search heuristic for the truck and trailer routing problem. Comput. Oper. Res. 33 (4), 894–909. https://doi.org/10.1016/j.cor.2004.08.002
Schmid, V., Doerner, K.F., Laporte, G., 2013. Rich routing problems arising in supply chain management. Eur. J. Oper. Res. 224 (3), 435–448. https://doi.org/10.1016/j.ejor.2012.08.014
Sicilia, J.A., Quemada, C., Royo, B., Escuín, D., 2016. An optimization algorithm for solving the rich vehicle routing problem based on variable neighborhood search and tabu search metaheuristics. J. Comput. Appl. Math. 291, 468–477. https://doi.org/10.1016/j.cam.2015.03.050
Silva, A. F.R., Pozos, A.T., Martinez, D.A., Escobar, J.W., 2025. An iterated local search based approach for a real rich vehicle routing problem with time windows. Comput. Ind. Eng. 208, 111422. https://doi.org/10.1016/j.cie.2025.111422
Sinnott, R.W., 1984. Virtues of the haversine. Sky Telesc. 68 (2), 158.
Sörensen, K., Sevaux, M., Schittekat, P., 2008. “Multiple Neighbourhood” Search in Commercial VRP Packages: Evolving Towards Self-Adaptive Methods. Springer Berlin Heidelberg. p. 239–253. https://doi.org/10.1007/978-3-540-79438-7_12
Toth, P., Vigo, D., 2002. 1. An Overview of Vehicle Routing Problems. Society for Industrial and Applied Mathematics. p. 1–26. https://doi.org/10.1137/1.9780898718515.ch1
Villegas, J.G., Prins, C., Prodhon, C., Medaglia, A.L., Velasco, N., 2013. A matheuristic for the truck and trailer routing problem. Eur. J. Oper. Res. 230 (2), 231–244. https://doi.org/10.1016/j.ejor.2013.04.026
Yang, F., Dai, Y., Ma, Z.-J., 2020. A cooperative rich vehicle routing problem in the last-mile logistics industry in rural areas. Transp. Res. E 141, 102024. https://doi.org/10.1016/j.tre.2020.102024
Yang, K., Zhang, Z., Zhao, J., Liang, Z., 2025. A neighborhood-based matheuristic for the vehicle routing problem with delivery options. Transp. Res. E 203, 104366. https://doi.org/10.1016/j.tre.2025.104366