ECONSTOR
Make Your Publications Visible.

A Service of

ZBW Leibniz-Informationszentrum Wirtschaft
Leibniz Information Centre for Economics

Lee, Chungjae; Dalmeijer, Kevin; Van Hentenryck, Pascal; Zhang, Peibo

Article

Optimizing Autonomous Transfer Hub Networks: Quantifying the potential impact of self-driving trucks

EURO Journal on Transportation and Logistics (EJTL)

Provided in Cooperation with:
Association of European Operational Research Societies (EURO), Fribourg

Suggested Citation: Lee, Chungjae; Dalmeijer, Kevin; Van Hentenryck, Pascal; Zhang, Peibo (2024) : Optimizing Autonomous Transfer Hub Networks: Quantifying the potential impact of self-driving trucks, EURO Journal on Transportation and Logistics (EJTL), ISSN 2192-4384, Elsevier, Amsterdam, Vol. 13, Iss. 1, pp. 1-15, https://doi.org/10.1016/j.ejtl.2024.100141

This Version is available at:
https://hdl.handle.net/10419/325212

Standard-Nutzungsbedingungen:

Die Dokumente auf EconStor dürfen zu eigenen wissenschaftlichen Zwecken und zum Privatgebrauch gespeichert und kopiert werden.

Sie dürfen die Dokumente nicht für öffentliche oder kommerzielle Zwecke vervielfältigen, öffentlich ausstellen, öffentlich zugänglich machen, vertreiben oder anderweitig nutzen.

Sofern die Verfasser die Dokumente unter Open-Content-Lizenzen (insbesondere CC-Lizenzen) zur Verfügung gestellt haben sollten, gelten abweichend von diesen Nutzungsbedingungen die in der dort genannten Lizenz gewährten Nutzungsrechte.

Terms of use:

Documents in EconStor may be saved and copied for your personal and scholarly purposes.

You are not to copy documents for public or commercial purposes, to exhibit the documents publicly, to make them publicly available on the internet, or to distribute or otherwise use the documents in public.

If the documents have been made available under an Open Content Licence (especially Creative Commons Licences), you may exercise further usage rights as specified in the indicated licence.


https://creativecommons.org/licenses/by-nc-nd/4.0/

WWW.ECONSTOR.EU

Mitglied der Leibniz-Gemeinschaft

EURO Journal on Transportation and Logistics 13 (2024) 100141

Contents lists available at ScienceDirect

EURO Journal on Transportation and Logistics

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

Optimizing Autonomous Transfer Hub Networks: Quantifying the potential impact of self-driving trucks

Chungjae Lee [a], Kevin Dalmeijer [a,*], Pascal Van Hentenryck [a], Peibo Zhang [b,a]

[a] H. Milton Stewart School of Industrial and Systems Engineering, Georgia Institute of Technology, United States of America
[b] Goizueta Business School, Emory University, United States of America

ARTICLE INFO

Keywords:
Autonomous Transfer Hub Networks
Autonomous trucking
Load planning
Mixed-integer linear programming
Case study

ABSTRACT

Autonomous trucks are expected to fundamentally transform the freight transportation industry. In particular, Autonomous Transfer Hub Networks (ATHNs), which combine autonomous trucks on middle miles with human-driven trucks on the first and last miles, are seen as the most likely deployment pathway for this technology. This paper presents a framework to optimize ATHN operations and evaluate the benefits of autonomous trucking. By exploiting the problem structure, this paper introduces a flow-based optimization model for this purpose that can be solved by blackbox solvers in a matter of hours. The resulting framework is easy to apply and enables the data-driven analysis of large-scale systems. The power of this approach is demonstrated on a system that spans all of the United States over a four-week horizon. The case study quantifies the potential impact of autonomous trucking and shows that ATHNs can have significant benefits over traditional transportation networks.

1. Introduction

Self-driving trucks are expected to fundamentally transform the freight transportation industry. Morgan Stanley estimates the potential savings from self-driving trucks at $168 billion annually for the United States alone (Greene, 2013). Additionally, autonomous transportation may improve on-road safety, and reduce emissions and traffic congestion (Short and Murray, 2016; Slowik and Sharpe, 2018).

SAE International defines different levels of driving automation, ranging from L0 to L5, corresponding to no-driving automation to full-driving automation (SAE International, 2018). The current focus is on L4 technology (high automation), which aims at delivering automated trucks that can drive without any human intervention in specific domains, e.g., on highways. The trucking industry is actively involved in making L4 vehicles a reality. Daimler Trucks, one of the leading heavy-duty truck manufacturers in North America, acquired a majority stake in self-driving truck developer Torc Robotics, which laid out a roadmap to launch autonomous trucks in 2027 (Transport Topics, 2023). Autonomous trucking company TuSimple has recently completed the first driverless tests on Chinese public roads (TechCrunch, 2023). In the US, Aurora Innovation teamed up with FedEx to haul freight between Fort Worth and El Paso, Texas, and the company reports that 60,000 miles have been completed without incidents (FedEx, 2022). These are just some of the companies involved in autonomous trucking, and others include Embark, Gatik, Kodiak, and Plus (FleetOwner, 2021; Forbes, 2021; FreightWaves, 2021).

A study by Viscelli describes different scenarios for the adoption of autonomous trucks by the industry (Viscelli, 2018). The most likely scenario, according to some of the major players, is the *transfer hub business model* (Viscelli, 2018; Berger, 2018; Shahandasht et al., 2019). Joanna Buttler, head of Daimler's global autonomous technology group, for example, stated that "We are staying laser focused on U.S. hub-to-hub, on-highway" (Transport Topics, 2023). An Autonomous Transfer Hub Network (ATHN) makes use of autonomous truck ports, or *transfer hubs*, to hand off trailers between human-driven trucks and driverless autonomous trucks. Autonomous trucks then carry out the transportation between the hubs, while regular trucks serve the first and last miles (see Fig. 1). Orders are split into a first-mile leg, an autonomous leg, and a last-mile leg, each of which served by a different vehicle. A human-driven truck picks up the freight at the customer location, and drops it off at a nearby transfer hub. A driverless self-driving truck moves the trailer to a transfer hub close to the destination, and another human-driven truck performs the last leg.

The ATHN applies automation where it counts: Monotonous highway driving is automated, while more complex local driving and customer contact is left to humans. Global consultancy firm Berger estimates operational cost savings between 22% and 40% in the transfer

* Correspondence to: 755 Ferst Dr NW, Atlanta, GA 30318, United States of America.
E-mail address: dalmeijer@gatech.edu (K. Dalmeijer).

https://doi.org/10.1016/j.ejtl.2024.100141
Received 12 February 2024; Received in revised form 15 July 2024; Accepted 5 August 2024
Available online 6 August 2024
2192-4376/© 2024 The Authors. Published by Elsevier B.V. on behalf of Association of European Operational Research Societies (EURO). This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/).

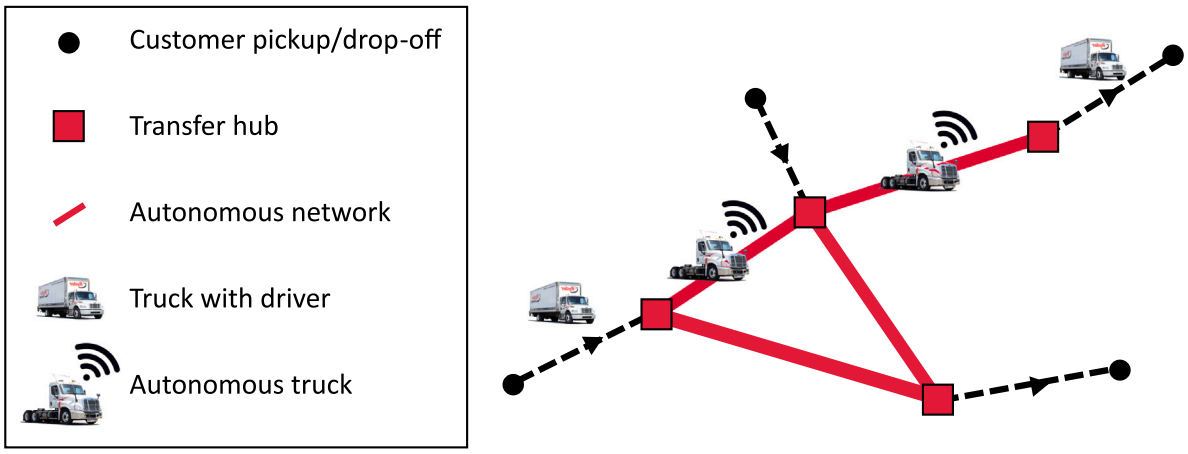


Fig. 1. Example of an autonomous transfer hub network.

hub model, based on cost estimates for three example trips (Berger, 2018). A recent white paper published by Ryder System, Inc. and the Socially Aware Mobility Lab studies whether these savings can be attained for actual operations and realistic orders in Full Truckload (FTL) shipping (Ryder System and Socially Aware Mobility Lab, 2021). It models ATHN operations as a scheduling problem and uses a Constraint Programming (CP) model to minimize empty miles and produce savings from 27% to 40% on a case study in the Southeast of the United States.

The current paper is an extension of the Ryder white paper that substantially improves, simplifies, and generalizes the methodology. It is the culmination of two years of research into the core computational difficulty of optimizing ATHN operations, reported in the Ryder white paper and in technical reports by the authors (Dalmeijer and Van Hentenryck, 2021; Lee et al., 2022). The CP model presented in Dalmeijer and Van Hentenryck (2021) produces solutions that outperform the current operations, but that do not provide a bound on optimality. Lee et al. (2022) introduces a Column Generation (CG) approach and a bespoke Network Flow (NF) model. It is shown that the CP solution can be more than 10% from optimal, and that the NF model can quickly produce solutions within 1% from optimality. These earlier findings motivate the flow-based optimization model in this paper that exploits the problem structure and is solved to optimality by blackbox solvers in a matter of hours. The resulting framework is easy to apply and enables the data-driven analysis of large-scale systems. It has also enabled a follow-up study on the role of hub capacities in ATHNs (Lee et al., 2023).

The power of the new methodology is demonstrated on an FTL system that spans all of the United States over a four-week horizon, expanding both the region and time horizon used in earlier reports. The case study quantifies the potential impact of self-driving trucks and shows that ATHNs yield significant benefit over traditional transportation networks. The main contributions of this work can be summarized as follows:

1. The paper provides a high-level framework to optimize ATHN operations.
2. The paper demonstrates that this enables the study of large-scale systems, requiring only a blackbox solver.
3. The paper uses realistic order data to quantify the potential impact of FTL autonomous trucking in the US on a national scale.

The remainder of this paper is organized as follows. Section 2 presents an overview of the literature. Section 3 provides the problem description and Section 4 discusses the methodology for optimizing ATHNs. This methodology is applied to a case study in the US that is introduced in Section 5. The baseline results and the analysis of the potential impact of autonomous trucking are presented in Section 6 and a detailed sensitivity analysis is provided by Section 7. Finally, Section 8 provides the conclusions.

2. Literature review

As autonomous technology advances, more papers are studying the effect of autonomous vehicles on transportation systems. Flämig (2016) provides an overview of the different ways that autonomous vehicles can be used both on public infrastructure and on private property (e.g., warehouses or company grounds). In the urban transportation setting, de Almeida Correia and van Arem (2016) studies the effect of autonomous vehicles on traffic delays and parking demand in a city. The authors use convex optimization to determine traffic assignments and a mixed integer nonlinear formulation to assign autonomous vehicles to households. A case study for the city of Delft, The Netherlands, demonstrates a positive impact on the road network.

In the freight transportation context, routing and scheduling problems with autonomous trucks have gained attention very recently. Chen et al. (2021) considers scheduling a platoon of autonomous trucks to reduce air resistance when traveling between two seaport terminals in Singapore. The authors present a mixed integer second-order-cone formulation that is solved with a column-generation based heuristic. In the area of service network design, Scherr et al. (2018) proposes a problem where a human-driven truck leads a platoon of autonomous vehicles in the first tier of city logistics. An arc-based mixed integer programming model on a time-space network is presented, but empirical observations show that only small problem instances are tractable. Scherr et al. (2020) extends this work by introducing a dynamic discretization discovery approach that outperforms a commercial solver, and also present a heuristic to quickly generate solutions.

In the Less-Than-Truckload (LTL) context, Al Hajj Hassan et al. (2022) studies the daily load planning problem under different levels of automation. The paper focuses on modifying a given base plan to deal with dynamic load requests and other aspects that are important during operations, including driver regulations where drivers are involved. The authors present a column-generation based heuristic to solve industry-based instances with up to 20 hubs and 1500 loads over a one-week horizon.

In terms of the problem structure, optimizing full truckload ATHN operations can be seen as a Pickup and Delivery Problem with Time Windows (PDPTW), where trucks pick up and drop off loads within the time windows prescribed by the customers. The book Toth and Vigo (2014) provides a survey of this vehicle routing problem and other variants. However, instead of routing, this paper will exploit the problem structure and take the perspective of scheduling a sequence of tasks (combined pickups and deliveries), which is closely related to the Vehicle Scheduling Problem with Time Windows (VSPTW, Desrosiers et al. (1995)). These problems are well studied, and several exact and heuristic solution methods exist. For example, Freling et al. (2001)

presents a solution method based on the primal–dual algorithm framework for VSPTW with a single depot. Ribeiro and Soumis (1994) proposes a column-generation approach for the VPSTW with multiple depots, and Hadjar et al. (2006) presents a branch-and-cut algorithm for the same problem. Steinzen et al. (2010) considers solving the time-extended variant of the VSPTW with multiple depots using a heuristic based on the branch-and-price framework. Campbell and Savelsbergh (2004) presents insertion heuristics for vehicle routing and scheduling problems.

The Vehicle Routing Problem with Full Truckloads (VRPFL, Arunapuram et al., 2003) is the specific variant that perhaps most structurally resembles the ATHN problem. Similar to the current paper, the VRPFL asks for minimum-cost truck routes to serve a set of loads that are specified by an origin, destination, and a pickup time window. Arunapuram et al. (2003) proposes a branch-and-price framework as the solution approach. The authors assume that each order consumes the full capacity of the truck, and the same assumption is made for optimizing ATHN operations, which reflects that autonomous trucks are expected to be mostly used for long-haul trips. A crucial technical difference between (Arunapuram et al., 2003) and the current paper is that autonomous trucks are assumed to be completely interchangeable. This will allow for a flow-based optimization model that is amenable to blackbox solving.

This paper introduces a high-level framework to optimize ATHN operations. The goal of this framework is to provide a practical way to study large-scale autonomous FTL systems and to quantify the potential impact of autonomous trucking. Previous works often rely on advanced optimization techniques such as cutting planes or column generation, or provide methods that do not scale to industry-sized problems. For example, the largest problem considered by Arunapuram et al. (2003) involves only 5 hubs and 160 loads. In contrast, this paper exploits the problem structure to provide a model that is blackbox solvable on a large scale (up to 200 hubs and 6000+ loads over a four-week horizon). Another benefit of the high-level framework is that it can be used to generate a base plan that forms the basis for the operational decisions, e.g., as studied by Al Hajj Hassan et al. (2022) for LTL trucking.

3. Problem description

This section introduces the problem of optimizing ATHN operations, while the solution methodology is presented in Section 4. Table 1 summarizes the nomenclature for the problem description. The goal is to serve a set of n full truckloads L at minimum cost with a combination of deliveries through the autonomous network and direct deliveries with regular trucks. Each load $l \in L$ is identified by an origin location $o(l)$, a destination location $d(l)$, and a planned departure time, or release time, $r(l)$. The autonomous network is based on a set of transfer hubs V_H. Every load $l \in L$ is associated with an origin hub $h_l^+ \in V_H$ near the origin $o(l)$ and a destination hub $h_l^- \in V_H$ near the destination $d(l)$.

Solution. A solution consists of three types of decisions that are made jointly. First, it is determined how each load $l \in L$ is served. It is assumed that there are exactly two options:

- **Autonomous:** The load follows the path $o(l) \to h_l^+ \to h_l^- \to d(l)$. The first and last legs are performed by a regular truck, while the connection between the hubs is served by an autonomous truck.
- **Direct:** The load follows the path $o(l) \to d(l) \to o(l)$. Both legs are served by a single regular truck that returns empty. Note that the case study will consider challenging orders that actually incur such an empty return in practice.

Second, the autonomous legs ($h_l^+ \to h_l^-$) of the loads that are served autonomously are combined into *routes* for at most $K \geq 0$ autonomous trucks. Note that these routes may include empty relocations from h_l^+ to $h_{l'}^-$ between loads l and l'. It is assumed that sufficient regular trucks are available to perform the traditional legs. The corresponding costs will be captured in the objective function, but the regular truck routes are not modeled explicitly. This is motivated by the fact that, in practice, the first- and last-mile problems are not very constrained. Third, it is decided at which time each load is picked up. It is assumed that every load $l \in L$ admits a flexibility of $\Delta \geq 0$ around the planned departure time $r(l)$, leading to a time window of $[r(l) - \Delta, r(l) + \Delta]$ for pickup. This time window is translated to h_l^+, h_l^-, and $d(l)$ according to the travel times to maintain this flexibility throughout. The travel times include time for loading and unloading the autonomous truck, which is assumed to be $S \geq 0$. A solution is feasible if each load is served autonomously or directly, all implied autonomous legs are covered by autonomous truck routes, and the autonomous truck routes are feasible with respect to time. Note that it is always feasible to replicate the current situation by serving all loads directly and not using any autonomous trucks.

Location graph. Before defining the objective, it is convenient to define a *location graph*. The location graph models all relevant locations and potential connections in the ATHN. Let the location graph be denoted by the directed graph $G = (V, A)$. Vertex set V contains a vertex for every hub location, and two vertices for every load $l \in L$ that correspond to the origin $o(l)$ and the destination $d(l)$, respectively. Arcs $a \in A$ are defined from every origin to the hubs (traditional first mile), between all the hubs (autonomous middle mile), from the hubs to every destination (traditional last mile), and between origin and destination directly (traditional direct delivery and empty return). Note that the arcs between the hubs form a complete graph. Every arc $a \in A$ is associated with a distance $c_a \geq 0$ and a travel time $\tau_a > 0$ obtained from OpenStreetMap (2021). For convenience, the cost and travel time from $i \in V$ to itself are defined as 0.

Objective. The objective is to serve all loads at minimum cost. While any non-negative arc-additive cost structure is supported, this paper will define cost as the total distance in traditional mileage equivalent. Autonomous trucks incur a cost of $(1-\alpha)c_a$ for every arc $a \in A$ on their routes, including arcs that represent empty relocations. The parameter $\alpha \in [0, 1]$ discounts the autonomous distance to correct for reduced labor cost. A direct trip for load $l \in L$ has a cost equal to its distance of $c_{o(l)d(l)} + c_{d(l)o(l)}$. Note that the discount does not apply to regular trucks. Finally, each first/last-mile arc $a \in A$ is assigned a cost of $\frac{1}{1-\beta}c_a$. The parameter $\beta \in [0, 1)$ represents the first/last-mile inefficiency, which assumes that a fraction β of the first/last-mile route mileage would be empty. The factor $\frac{1}{1-\beta}$ increases the cost of the first/last-mile arcs to compensate for the fact that these routes are not modeled explicitly. The total objective is the sum of the above components and can be interpreted as the total distance measured in equivalent traditional mileage.

4. Methodology

This section introduces the methodology that enables a large-scale data-driven study to quantify the impact of self-driving trucks. The nomenclature for this section is summarized by Table 2. Practical assumptions and preprocessing steps lead to a model that is easy to implement, can immediately be solved by blackbox solvers, and is highly extensible. The section ends by providing a practical guide to enable regional and temporal analysis in this framework, which requires only minor modifications to the input and the model.

Task graph. The optimization model considers the problem of optimizing ATHN operations from the perspective of scheduling *tasks* for autonomous trucks. Similar transformations are common in the arc-routing literature (e.g., see Black et al. (2013)). One task $t \in T$ is created for every load $l \in L$. If an autonomous truck performs a task, it means that the corresponding load is served through the autonomous network, and this truck serves the middle mile. If a task is not performed by any autonomous truck, this means that the corresponding

Table 1
Nomenclature problem description.

Symbol	Definition
Sets and graphs	
L	Set of loads, each load $l \in L$ consists of an origin $o(l)$, a destination $d(l)$ and a release time $r(l)$.
V_H	Set of autonomous transfer hub locations, $V_H \subseteq V$.
G	$= (V, A)$, location graph that models locations and connections in the ATHN.
V	Set of locations.
A	Set of location arcs, each arc $(i, j) \in A$ corresponds to travel from location $i \in V$ to location $j \in V$.
Parameters	
n	Number of loads, $n = \|L\|$.
h_l^+	Origin hub for load $l \in L$, $h_l^+ \in V_H$.
h_l^-	Destination hub for load $l \in L$, $h_l^- \in V_H$.
K	Maximum number of autonomous trucks.
Δ	Flexibility around the planned departure time (depart up to Δ earlier or later than planned).
S	Autonomous truck loading/unloading time.
c_a	Distance to travel location arc $a \in A$, $c_a \geq 0$.
τ_a	Time to travel location arc $a \in A$, $\tau_a > 0$.
α	Discount factor for autonomous mileage, $\alpha \in [0, 1]$
β	First/last-mile inefficiency, $\beta \in [0, 1)$

Table 2
Nomenclature methodology.

Symbol	Definition
Sets and graphs	
T	Set of tasks, each task $t \in T$ corresponds one-to-one to a load $l(t) \in L$, and represents serving this load on the ATHN with pickup time $p(t)$ at its origin hub.
$\bar{G}$	$= (\bar{V}, \bar{A})$, task graph that models the sequence of tasks.
$\bar{V}$	Set of vertices $\{0, \dots, n+1\}$ with source 0, sink $n+1$, and tasks $1, \dots, n$.
$\bar{A}$	Set of task arcs, each arc $(t, t') \in \bar{A}$ indicates that vertex $t \in \bar{V}$ is followed immediately by vertex $t' \in \bar{V}$.
Parameters	
$\bar{\tau}_a$	Duration of task arc $(t, t') \in \bar{A}$, $\tau_a > 0$, which consists of loading a truck for task t, driving between hubs, unloading, and relocating to the origin hub of task t'.
$\bar{C}_t$	Baseline cost for serving load $l(t)$ directly with a regular truck.
$\bar{c}_a$	Cost of task arc $a \in \bar{A}$, which is the difference between serving load $l(t)$ compared to the baseline.
$M_{tt'}$	$= p(t) - p(t') + 2\Delta + \bar{\tau}_{tt'}$, for $t, t' \in T$, sufficiently large big-M for Constraints (2e).
Variables	
x_t	Continuous variable that indicates the start time of task $t \in T$.
y_a	Binary variable that takes value one if $a \in \bar{A}$ is selected (i.e., the corresponding tasks are performed sequentially by the same vehicle), and zero otherwise.

load is served directly by a regular truck. Note that while performing tasks is optional, all loads are served in the end: performing a task only indicates that the task is served autonomously. Appropriate benefits will be assigned to performing tasks to match the cost structure in Section 3.

To capture this perspective, the location graph is transformed into a directed *task graph* $\bar{G} = (\bar{V}, \bar{A})$ in which the nodes are tasks and the arcs indicate the sequence of tasks performed by the same autonomous truck. The set $\bar{V}$ includes a source node 0 where each sequence starts, and a sink node $n+1$ where it ends. As the nodes now represent tasks instead of locations, the number of nodes in the task graph is typically larger than the number of nodes in the location graph. Arcs are defined from the source to the tasks, between the tasks (bi-directional), and from the tasks to the sink. Fig. 2 provides an illustrative example. The location graph shows four hubs and two loads l_1 and l_2 associated with tasks t_1 and t_2, respectively. Visiting node t_1 means that an autonomous truck loads at origin hub $h_{l_1}^+$, drives to destination hub $h_{l_1}^-$, and unloads there. It also implies that the first and last miles are performed by regular trucks (not pictured). After that, the autonomous truck may either perform another task $t_2 \in T$, which first requires a relocation from $h_{l_1}^-$ to $h_{l_2}^+$, or it may end its sequence. Loads for which the corresponding task is not covered are served by a direct trip with a regular truck (not pictured). This means that all operations in the ATHN are captured in the task graph by a set of paths from the source to the sink, where each path corresponds to an autonomous truck.

Routes and costs. Optimizing ATHN operations now amounts to choosing a set of feasible autonomous truck routes that minimize the total cost. A route is defined as a simple path in the task graph from source to sink, together with a starting time for every task. Arcs between tasks $t_1, t_2 \in T$ model the passage of time between picking up loads l_1 and l_2, respectively. That is, the duration is defined as $\bar{\tau}_{t_1 t_2} = S + \tau_{h_{l_1}^+ h_{l_1}^-} + S + \tau_{h_{l_1}^- h_{l_2}^+} > 0$, which sums the time for loading, performing the middle mile of load l_1, unloading, and relocating to the starting point of load l_2. Task t_1 must start in the correct time window, which is obtained by shifting the original time window of load l_1 by the time it takes to perform the first mile. This time window is given by $[p(t_1) - \Delta, p(t_1) + \Delta]$, where $p(t_1) = r(l_1) + t_{o(l_1)h_{l_1}^+}$.

Not covering task $t_1 \in T$ is associated with a constant baseline cost of $\bar{C}_{t_1}$ for performing a direct trip. This value is given by $\bar{C}_{t_1} = c_{o(l_1)d(l_1)} + c_{d(l_1)o(l_1)}$. If task t_1 *is* performed, the cost on the outgoing arc replaces the baseline cost with the appropriate costs for serving the load autonomously. More precisely, if task t_1 appears in a sequence followed by task $t_2 \in T$, the cost $\bar{c}_{t_1 t_2}$ of arc $(t_1, t_2) \in A$ is defined as follows:

$$\bar{c}_{t_1 t_2} = \underbrace{\frac{1}{1-\beta} c_{o(l_1)h_{l_1}^+}}_{\text{first mile}} + \underbrace{(1-\alpha) c_{h_{l_1}^+ h_{l_1}^-}}_{\text{middle mile}} + \underbrace{\frac{1}{1-\beta} c_{h_{l_1}^- d(l_1)}}_{\text{last mile}} + \underbrace{(1-\alpha) c_{h_{l_1}^- h_{l_2}^+}}_{\text{relocation}} - \underbrace{\bar{C}_{t_1}}_{\text{direct}}. \tag{1}$$

Along the same lines, source arcs $a \in \bar{A}$ have cost $\bar{c}_a = 0$ and sink arcs omit the relocation term. Note that $\bar{c}_a < 0$ when an autonomous delivery is preferred over a direct delivery, which encourages the task to be performed.

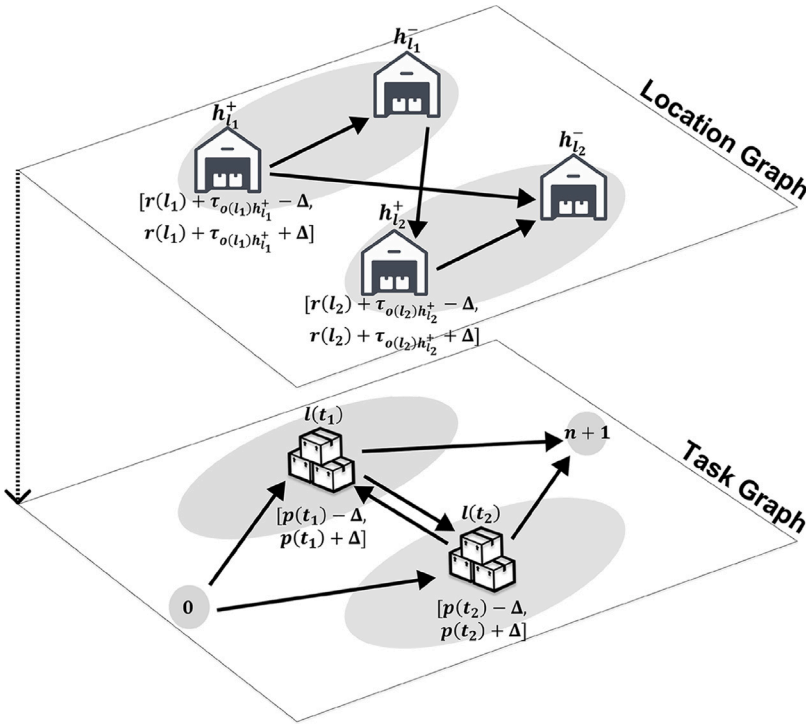


Fig. 2. Constructing the task graph from the location graph.

Optimization model. The optimization problem can now be stated as follows:

$$\min \quad \sum_{t\in T} \bar{C}_t + \sum_{a\in\bar{A}} \bar{c}_a y_a, \tag{2a}$$

$$\text{s.t.} \quad \sum_{a\in\delta_t^+} y_a \le 1 \qquad \forall t \in T, \tag{2b}$$

$$\sum_{a\in\delta_t^+} y_a = \sum_{a\in\delta_t^-} y_a \qquad \forall t \in T, \tag{2c}$$

$$\sum_{a\in\delta_0^+} y_a \le K, \tag{2d}$$

$$x_{t'} \ge x_t + \bar{\tau}_{tt'} - M_{tt'}(1-y_a), \qquad \forall t,t' \in T, (t,t') \in \bar{A} \tag{2e}$$

$$x_t \in [p(t)-\Delta, p(t)+\Delta] \qquad \forall t \in T, \tag{2f}$$

$$y_a \in \mathbb{B} \qquad \forall a \in \bar{A}. \tag{2g}$$

Let $x_t \in [p(t)-\Delta, p(t)+\Delta]$ be the start time of task $t \in T$. The variable $y_a \in \mathbb{B}$ is the flow on arc $a \in \bar{A}$, i.e., it takes value one if the tasks are performed sequentially by the same vehicle and zero otherwise. For convenience, let δ_v^+ and δ_v^- denote the out-arcs and in-arcs of $v \in \bar{V}$, respectively. Problem (2) then models the optimization of ATHN operations. Objective (2a) minimizes the system cost as discussed above. Constraints (2b) require that each task is performed at most once, and Constraints (2c) ensure flow conservation. The number of vehicles is limited by Constraint (2d). Constraints (2e) are Miller, Tucker, and Zemlin (1960) constraints that model the passage of time and eliminate cycles. It is straightforward to show that the constants

$$M_{tt'} = \bar{\tau}_{tt'} - \underbrace{(p(t')-\Delta)}_{\text{lowerbound on } x_{t'}} + \underbrace{(p(t)+\Delta)}_{\text{upperbound on } x_t} \tag{3}$$

are sufficiently large to make the constraint inactive when $y_a = 0$. Finally, Eqs. (2f)–(2g) define the variables. For a consistent analysis, the solution is postprocessed to shift the start times to as early in time as possible.

4.1. Acceleration techniques

The size of Problem (2) can be reduced significantly by recognizing that many arcs $(t,t') \in \bar{A}$ are either trivially time-feasible because task t' is planned much later than task t, or trivially time-infeasible because task t' is planned much earlier than task t. These observations are formalized in the following proposition.

Proposition 1 (*Preprocessing Rules*). *Let $a_t = p(t) - \Delta$ and $b_t = p(t) + \Delta$ be the earliest and latest possible start time of task $t \in T$, respectively. The following preprocessing rules are valid for arc $(t,t') \in \bar{A}$ between two tasks $t,t' \in T$.*

1. ***Arc is always time feasible:*** *$b_t + \bar{\tau}_{tt'} \le a_{t'} \Rightarrow$ remove Constraint (2e) for arc (t,t').*
2. ***Arc is never time feasible:*** *$a_t + \bar{\tau}_{tt'} > b_{t'} \Rightarrow$ remove arc (t,t') from $\bar{A}$.*

Proof. By definition, x_t is only allowed to take values in $x_t \in [a_t, b_t]$. The condition in the first rule implies $x_t + \bar{\tau}_{tt'} \le b_t + \bar{\tau}_{tt'} \le a_{t'} \le x_{t'} \Leftrightarrow x_{t'} \ge x_t + \bar{\tau}_{tt'}$ for all feasible values of x_t and $x_{t'}$. It follows that the time constraint is redundant and can be removed. Similarly, the condition in the second rule implies $x_t + \bar{\tau}_{tt'} \ge a_t + \bar{\tau}_{tt'} > b_{t'} \ge x'_t \Leftrightarrow x'_t < x_t + \bar{\tau}_{tt'}$. It follows that $y_{tt'} = 1$ would violate Constraint (2e). As such, $y_{tt'}$ may be set to zero, which is achieved by simply removing the arc. Hence, these preprocessing rules are valid. □

Both preprocessing rules eliminate time constraints, which can make them incredibly powerful. If all time Constraints (2e) are eliminated, then the x-variables (2f) are automatically satisfied, and the remainder of Problem (2) can be seen as a minimum-cost network flow problem. The only constraints that are not in standard form are Constraints (2b) and (2d), but they take the form of node capacities that can be handled through node splitting (Ahuja et al., 1993). It is well-known that the min-cost flow problem exhibits the integrality property, and can be solved in polynomial time by linear programming. As the flexibility Δ decreases, the preprocessing rules become more effective, and Problem (2) gets closer to a minimum-cost network flow problem.

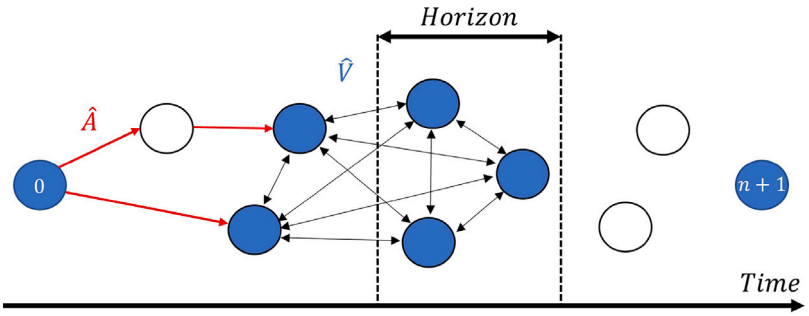


Fig. 3. Arc filtering for temporal decomposition ($\hat{A}$ in red and $\hat{V}$ in blue).

In fact, this situation is reached for the no-flexibility $\Delta = 0$ case, when every arc $(t, t') \in \bar{A}$ either satisfies Rule 1 or Rule 2 and all time constraints are eliminated. Informally, it is easier to optimize ATHN operations when there is less flexibility, to the point where it becomes provably easy without flexibility.

MIP start. In addition to preprocessing, this paper will try to improve the optimization process by providing the solver with an initial feasible solution. This solution is known as a *MIP start* and provides an upper bound that can assist the branch-and-bound process. Regardless of the flexibility Δ, a feasible solution can be calculated efficiently by solving the case when flexibility is set to zero. The calculated solution is then used as a starting point for the actual problem. To avoid the overhead of building an additional model, the solver is provided a partial MIP start of only $x_t = p(t)$ for all $t \in T$, which is sufficient to find the same solution.

The fact that $\Delta = 0$ is easy to solve and guarantees a valid upper bound is specifically because the trucks are autonomous. The main technical difference is that autonomous trucks are completely interchangable, while human-driven trucks need to be distinguished to ensure that drivers return to their specific starting point (Arunapuram et al., 2003) or that they do not exceed the maximum driving time (Gronalt et al., 2003). Network flow relaxations that aggregated drivers have been used to derive lower bounds (Gronalt et al., 2003), but it is not obvious how to transform the outcome into a feasible solution. For autonomous trucks these human factors do not apply, which enables the framework in this paper.

4.2. Regional and temporal decomposition

The framework in this paper is easily extended to perform regional and temporal decomposition. This requires only minor modifications to the input and to the model.

Regional decomposition. The goal of the regional decomposition is to compare the global optimization of ATHN operations to a situation in which each region (e.g., the South of the US) has dedicated autonomous trucks that only pick up loads that start in that region. In this case, trucks can serve loads within their region and loads that are moving out of the region. However, after leaving the region to drop off a load, the truck has to return before it can perform another task. The model will be modified to jointly optimize how trucks are assigned to regions and how to operate the ATHN under these restrictions. Performing an analysis in this setting helps answer questions about the scale at which autonomous trucks are effective and where they should be deployed.

Regional decomposition can easily be performed by filtering arcs from the task graph. First assign every task $t \in T$ to a region based on the location of the origin hub $h^+_{l(t)}$. Next, remove all arcs $(t, t') \in \bar{A}$ between tasks $t, t' \in T$ if the regions are different. It follows that when a flow reaches a task that starts from a specific region, there is no way to reach tasks that start from a different region, as intended. The amount of flow from the source to each region represents the amount of trucks that are assigned to that region. The truck assignments and operations are then jointly optimized by solving this filtered instance of Problem (2).

Temporal decomposition. The goal of the temporal decomposition is to plan ATHN operations on a rolling horizon (e.g., one week at a time), rather than for a full period at once (e.g., four weeks). Optimizing over a shorter horizon requires less information and is easier computationally. However, the model does not explicitly rebalance trucks at the end of the period. This means that optimizing myopically may leave the trucks ill-positioned for the next period. The temporal decomposition can be used to explore these trade-offs.

Implementing a rolling horizon for Problem (2) is relatively straightforward. A practical way to do so is by reusing the existing structures and modifying the model as little as possible. Fig. 3 provides an illustrative example. First, build the task graph for the full period. For a given horizon, identify the arcs $\hat{A}$ of the partial routes created previously (without sink arcs). Fix these arcs $a \in \hat{A}$ to $y_a = 1$ in the optimization model to stay consistent with the past. To plan for the current horizon, only the following nodes $\hat{V}$ are relevant: the source, the sink, the current route endpoints, and the tasks that start during the horizon. Now filter the task graph to only keep the arcs in $\hat{A}$ and the arcs in the subgraph induced by $\hat{V}$. This makes it impossible to plan outside of the horizon. The model is solved and the steps are repeated until the full period is planned.

5. Case study

To quantify the impact of autonomous trucking on a realistic transportation network, a case study is presented for the dedicated transportation business of Ryder System, Inc. (Ryder). Ryder is one of the largest transportation and logistics companies in North America, and provides fleet management, supply chain, and dedicated transportation services to over 50,000 customers.

Data. Ryder has provided a dataset that is representative for its dedicated transportation business in the US, reducing the scope to orders that are strong candidates for automation. The case study focuses on the *challenging orders* that currently consist of a single delivery followed by an empty return trip. These orders are highly inefficient and contribute significantly to the overall empty mileage, such that ATHN can potentially have a big impact. The challenging orders also allow for a clean comparison to the current situation: These are orders for which Ryder was unable to find a backhaul, and returning empty after delivery is how they would be served in practice. The challenging orders are converted into loads with an origin, destination, and planned departure time. The ATHN operations are optimized for loads that start during the first four weeks of October 2019. This corresponds to 6842 loads, with an average distance of 390 km (242 mi).

Network design. To design an effective network, it is important to select hubs that are 1. close to frequently used origins and destinations to minimize the first/last mile, and 2. easily accessible from the highway

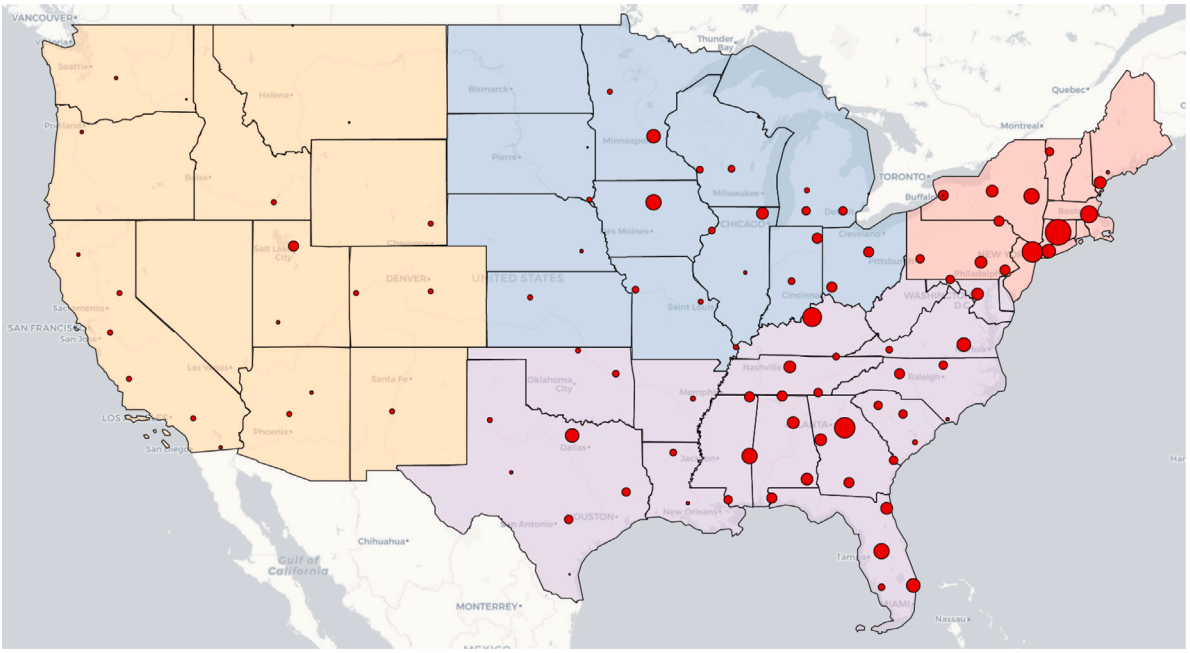


Fig. 4. ATHN design for 100 hubs (Circle area proportional to number of assigned loads).

to enable automation between the hubs. This is achieved by first using the K-means algorithm in Scikit-learn (Pedregosa et al., 2011) to cluster the origins and destinations into the desired number of hubs, based on data from October to December 2019. Next, the centroids are mapped onto the closest US truck stop obtained from the U.S. Department of Transportation (2019), according to haversine distance. The origin and destination hubs $h_l^+ \neq h_l^-$ for load $l \in L$ are chosen to minimize the value of $c_{o(l)h_l^+} + (1-\gamma)c_{h_l^+h_l^-} + c_{h_l^- d(l)}$, where $\gamma \in [0,1]$ is a discount factor for autonomous trucks. For $\gamma = 0$ this minimizes the total distance, for $\gamma = 1$ this minimizes the first/last-mile distance, and $\gamma \in (0,1)$ minimizes a combination of the two. By taking both the origin and the destination into account, the rule allows for assigning hubs in the right direction that are not necessarily the closest. Fig. 4 visualizes the 100-hub design for $\gamma = 40\%$, in which the area of each hub is proportional to the number of loads that are assigned to it. It can be seen that many loads are concentrated in the South (purple) and in the Northeast (red).

Experimental settings. Table 3 provides an overview of the baseline parameter values used in the case study. Various sensitivity analyses will be performed to observe how these parameters affect the system. The baseline includes two autonomous discount factors: $\alpha = 25\%$ and $\alpha = 40\%$. This results in a conservative estimate of the benefits of autonomous trucking, which is predicted to be 29% to 45% cheaper per mile (Engholm et al., 2020). All experiments use a consistent hub-assignment rule with autonomous discount factor $\gamma = 40\%$ to ensure that the results are comparable. A higher value of γ tends towards load paths that include more autonomous mileage. The baseline also uses multiple values of K to observe the impact of increasing availability of autonomous trucks, with the value $K = 100$ as the standard. These instances are solved sequentially by increasing the right-hand side of Constraint (2d) and reoptimizing. All steps in Section 4 are implemented in Python 3.9 and the ATHN operations are optimized with Gurobi 9.5.2. Gurobi is given three hours of solving time for each instance, unless stated otherwise. Each experiment is run on a Linux machine with dual Intel Xeon Gold 6226 CPUs on the PACE Phoenix cluster (PACE, 2017), using a single node with 24 cores and 192 GB of RAM. If memory is insufficient, the experiment is repeated on a machine with 384 GB of RAM. Note that if high-memory machines are not available, memory could also be traded for computing time by reducing the number of parallel threads.

Table 3
Baseline parameter values for the case study.

Parameter	Value
n	6842 loads
$\lvert V_H \rvert$	100 transfer hubs
K	$\in \{0, 50, \underline{100}, 150, 200, 250\}$ autonomous trucks
Δ	1 h pickup-time flexibility
S	30 min autonomous truck loading/unloading time
α	$\in \{25\%, 40\%\}$ discount for autonomous mileage
β	25% first/last-mile inefficiency
γ	40% discount for autonomous mileage during hub-assignment
Preprocessing	Applied
MIP start	Disabled
Solver time limit	3 h

6. Baseline results

This section discusses the baseline results for the case study. It first presents computational results to demonstrate that the presented framework can handle large-scale systems. Next, it analyzes the impact of autonomous trucking for the case study.

6.1. Computational results

Table 4 presents the computational results for the baseline instances. The instances differ by the discount for autonomous mileage α and the number of vehicles K. As described in the previous section, the instances for different K are run sequentially and reuse the same model, as would be done in practice to study the system. The 'LP Relaxation' columns present the time to solve the linear programming relaxation (before cuts) and the corresponding root gap. The 'Branch and Bound' columns summarize the full branch-and-bound process, reporting the number of nodes in the tree, the solution time (10,800 if the time limit of three hours is reached), and the final gap. The table omits the time

Table 4
Baseline computation statistics.

Parameters		LP relaxation		Branch and bound		
α	K	Seconds	Gap %	Nodes	Seconds	Gap (%)
25%	0	0	0	0	0	0
	50	218	19.50	1	10,800	0.03
	100	170	6.44	1	4,133	0
	150	183	1.81	1	3,261	0
	200	181	0.70	1	2,754	0
	250	177	0.40	1	2,850	0
40%	0	0	0	0	0	0
	50	268	25.10	1	10,800	0.18
	100	206	10.40	3382	6,430	0
	150	232	3.20	1	2,972	0
	200	207	1.00	1	3,149	0
	250	189	0.74	1	2,026	0

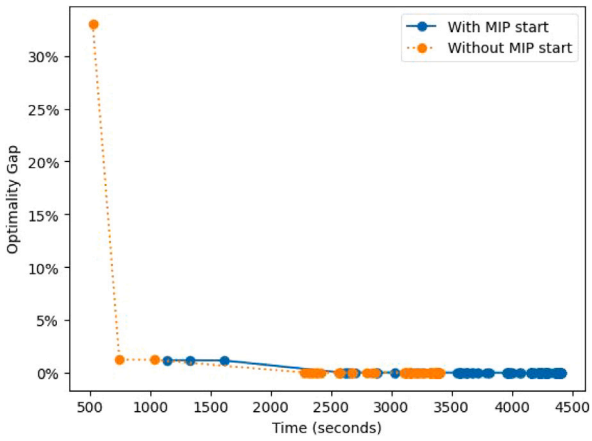


Fig. 5. Optimality gap over time for $\alpha = 25\%$ and 100 trucks.

for building the model, which was less than six minutes, and the time for presolve, which took less than four minutes in all cases.

Despite the fact that each model has over 22M binary variables and close to 300k constraints, Gurobi is able to find optimal solutions in most cases. For $K = 0$, the problem is trivial and is solved immediately in presolve. The cases with fewer vehicles are challenging to the solver, presumably because the tasks are packed more densely into the schedule, as will be discussed in Section 7.2. Only the $K = 50$ cases where not solved to optimality, and remain at 0.03% and 0.18% gap. Both for $\alpha = 25\%$ and $\alpha = 40\%$ the solver tends to keep adding cutting planes rather than branch. This strategy is successful to solve all other instances to optimality within the time limit. The only instance that stands out is $\alpha = 40\%$ and $K = 100$, which explores 3382 nodes in the branch-and-bound tree. For this instance, the log shows that a gap of 0.01% is found after 4845 s. When the gap is still at 0.01% at 5921 s, the solver decides to start branching to close the gap. This behavior can likely be explained by symmetry in the solution space, e.g., if two vehicles swap half of their tasks, the solution is likely to be of similar quality. The result is that good solutions are found quickly, but it takes a substantial number of cuts or branches to find the optimum. Overall, the computations for the baseline instances show that the proposed methodology can find optimal or close to optimal solutions in a short amount of time compared to the planning horizon.

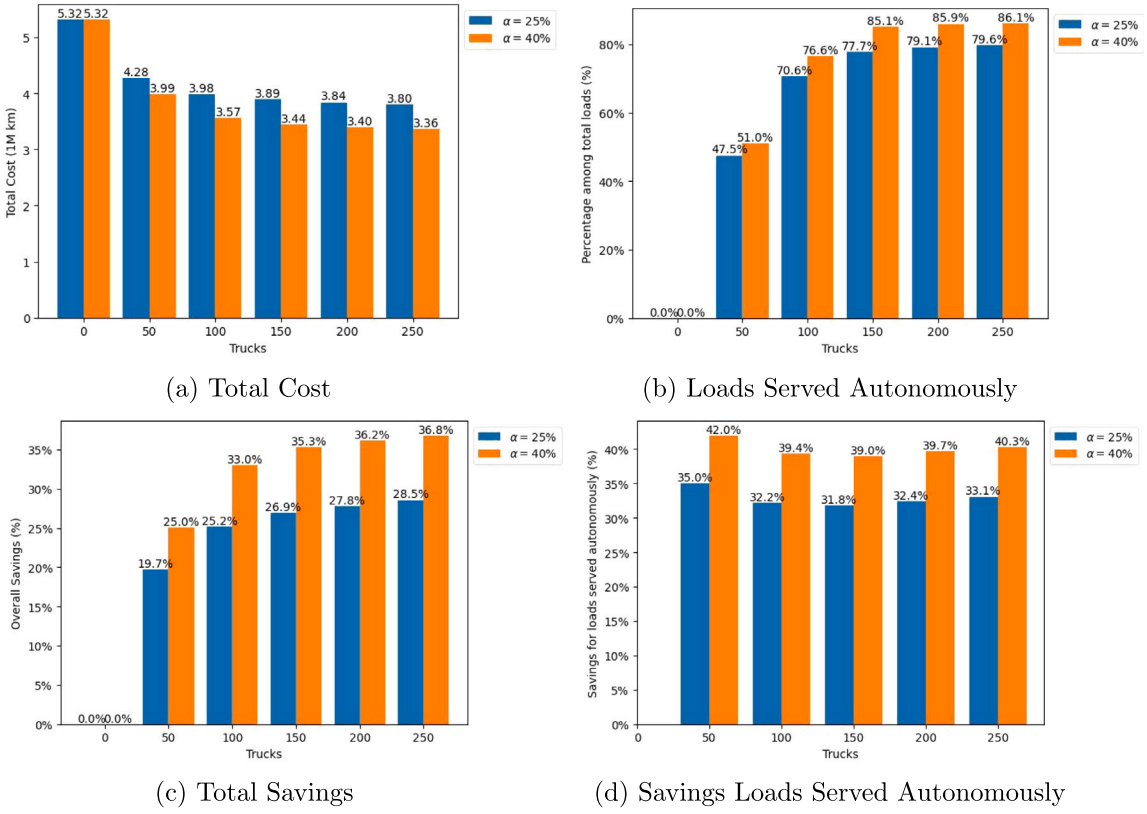
Using MIP starts. Enabling MIP start forces the solver to construct an initial feasible solution before starting the search (Section 4). Fig. 5 provides an example of the effect of enabling MIP start for the $\alpha = 25\%$ baseline with $K = 100$ trucks. Note that these tests are run independently without reusing the model for $K = 50$ as in the experiments above. Without any guidance, Gurobi takes 520 s to report the first optimality gap of 33%. At 741 s, a significantly better solution of 1.25% gap is found, and the problem is solved to optimality in under an hour. When MIP start is enabled, it takes more time for the search proper to start, and the first gap is reported at 1139 s. However, spending time to construct an initial feasible solution immediately leads to a gap of only 1.19% because of the improved upper bound. The full problem is solved in under one hour and 15 min.

Two observations are made for the case study. First, using a MIP start does not seem to improve solution time, but it does create a more predictable result. Especially if the instance cannot be solved to optimality, the MIP start is more likely to produce a reasonable solution before the time limit. Second, the small initial gap for the MIP start suggests that the initial solution is already of high quality. Recall that this zero-flexibility $\Delta = 0$ case can be seen as a min-cost flow problem, which gives practitioners the possibility to avoid commercial software and instead plan ATHN operations with highly-efficient open source solvers such as the LEMON graph library (Dezső et al., 2011). As most instances can be solved to optimality, MIP starts will only be enabled for the difficult large-flexibility instances in Section 7.1.

6.2. Impact of autonomous trucking

Fig. 6 presents the impact of autonomous trucking for the baseline, where autonomous mileage is discounted by either $\alpha = 25\%$ or $\alpha = 40\%$. It is clear that introducing autonomous trucks leads to substantial benefits. Fig. 6(a) shows that the first 50 trucks already lower the operational cost of the system (including first/last miles) by the equivalent of more than one million traditional kilometers. E.g., at \$1.25/km (≈\$2/mile) for traditional trucks, this corresponds to a value of about \$1.3M per four weeks or \$16.9M per year. The percentage savings for the overall system are provided in Fig. 6(c). These savings range from 20% for 50 trucks in the more expensive scenario to 37% for 250 trucks when autonomous trucking is less expensive. It is interesting to observe that adding vehicles clearly satisfies the law of diminishing returns. As more vehicles are added, more loads are served autonomously (Fig. 6(b)) and more savings are obtained (Fig. 6(c)), but the benefits level out at about 100 trucks for the Ryder case study. Note that this is a relatively small number of trucks compared to the 6842 loads, which reflects the fact that autonomous trucks can operate around the clock. Fig. 6(d) looks at the savings percentage only for loads that are served autonomously. It can be seen that, on average, loads that are served autonomously save between 31% and 42% in costs compared to traditional transportation.

Table 5 and Fig. 7 dive deeper into the results for $\alpha = 25\%$ and 100 trucks specifically. The table shows that the total distance driven in the ATHN (including empty miles) is 13.0% lower than for the current system. These savings are due to the flexibility of autonomous trucks that can operate throughout the night and never need to return home. This allows for only 29% empty miles on the autonomous middle mile, which is a substantial improvement over the rate of 50% in the current system. When labor cost reduction is taken into account, the savings increase to 25.2%. The truck schedule in Fig. 7 also shows that relocations are small compared to the work performed: Every row represents a single truck, where blue bars correspond to performing tasks, and the red bars correspond to driving empty. The schedule is relatively tight, except for the four 'gaps' during the weekends, in which not many loads are planned. This is an artifact of the data that results from planning around people, Section 7 will explore the value of increasing flexibility and allowing autonomous trucks to pick up loads on any day.



(a) Total Cost (b) Loads Served Autonomously

(c) Total Savings (d) Savings Loads Served Autonomously

Fig. 6. Baseline results.

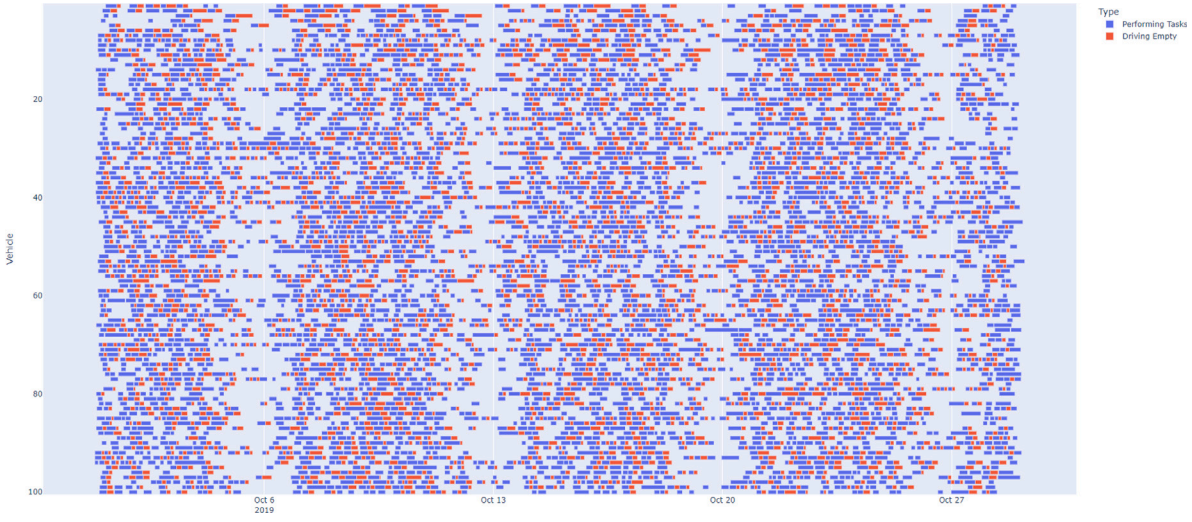
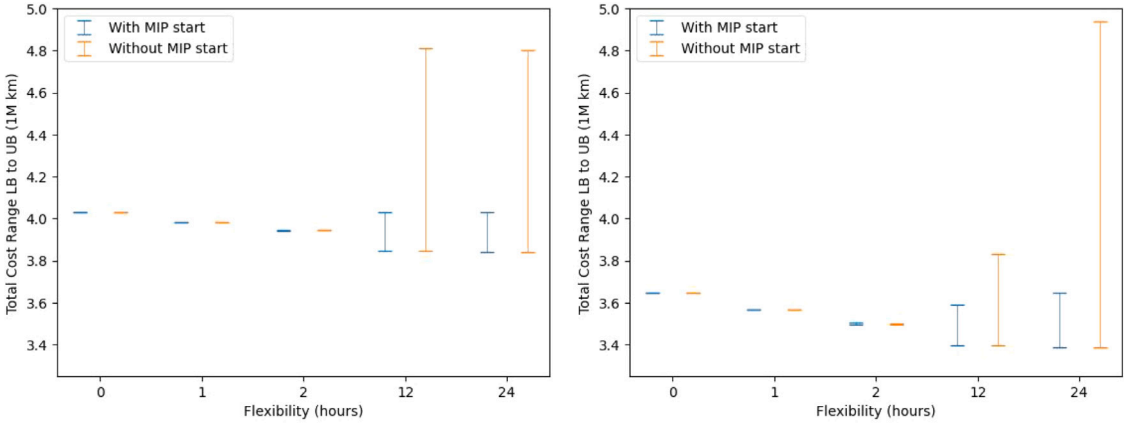


Fig. 7. Truck schedule for $\alpha = 25\%$ and 100 trucks.

Table 5
Statistics for $\alpha = 25\%$ and 100 trucks.

	Service	Loads	Segment	Total km	Empty	Cost factor	Cost
Current	Direct	6842	Full	5,323,357	50%	1	5,323,357
ATHN	Autonomous	4828	Middle	2,598,418	29%	0.75	1,948,814
			First/last	875,215	25%	1	875,215
	Direct	2014	Full	1,159,133	50%	1	1,159,133
	Total	6842		4,632,766			3,983,162
Savings ATHN compared to Current				690,591			1,340,195
				13.0%			25.2%



(a) Case $\alpha = 25\%$ and 100 trucks. (b) Case $\alpha = 40\%$ and 100 trucks.

Fig. 8. Upper and lower bounds for different time flexibilities Δ.

Table 6
Optimality gaps at time limit for large-flexibility instances.

Δ	$\alpha = 25\%$		$\alpha = 40\%$	
	Without MIP start	With MIP start	Without MIP start	With MIP start
2 h	0.04%	0.07%	0.17%	0.20%
12 h	20.05%	4.52%	11.38%	5.50%
24 h	14.80%	4.69%	29.77%	4.62%

7. Sensitivity analysis

The baseline results demonstrate significant benefits of autonomous trucking for the case study. This section provides an extensive sensitivity analysis of these results. It explores flexibility in pickup times, the trade-off between operating cost and autonomous truck utilization, the effect of regional and temporal decomposition, and various changes in input parameters.

7.1. Pickup-time flexibility

Even for a limited pickup-time flexibility of one hour, the baseline showed significant benefits of ATHN compared to traditional transportation. This section explores whether increasing the flexibility to up to 24 h further improves network performance. If there are major gains in efficiency, it may be worth negotiating new pickup times or new service-level agreements with the customers. From a computational standpoint, Section 6.1 explained that large-flexibility instances are more difficult to solve. For this reason, this section will rely heavily on MIP starts.

Figs. 8(a) and 8(b) present upper bounds (feasible solutions) and lower bounds for pickup-time flexibilities ranging from 0 to 24 h. In terms of computational performance, the figures show that the 0 h and 1 h instances can be solved to optimality, the 2 h instances have a small remaining gap, and the 12 h and 24 h instances prove difficult to solve. The exact optimality gaps are reported in Table 6. It can be seen that for the difficult instances, using a MIP start really helps to obtain better solutions, e.g., reducing the gap from 30% to 5% in the $\alpha = 40\%$ and $\Delta = 24$h case. Increasing the flexibility from 1 h to 2 h reduces the overall cost of the system by 0.9% for $\alpha = 25\%$ and 1.8% for $\alpha = 40\%$. With the current algorithm, increasing the flexibility to 12 h and 24 h does not lead to better solutions (upper bounds). However, the lower bounds show potential savings of up to 5.3% compared to the $\Delta = 1$h baseline. This makes it an interesting direction for future research to develop methods that are effective for larger flexibility. For $\alpha = 40\%$ and $\Delta = 24$h, the solver is able to improve over the MIP start solution. The resulting schedule is presented by Fig. 9. It shows that the increased flexibility creates a schedule that is more sparse and departs from the current weekly pattern shown by Fig. 7.

7.2. Number of autonomous trucks

This section explores the trade-off between operating cost and autonomous truck utilization in the ATHN. Fig. 10 presents the total cost of the ATHN operations from zero to 250 trucks in increments of 10, and all solutions are within 0.25% of optimality. The *inactivity* is defined as the percentage of time that autonomous trucks are waiting and not performing any tasks. The convex cost curves clearly show that the first autonomous trucks will be the most impactful and will almost never be inactive. This property makes it easier to run successful pilots and encourage the adoption of ATHN. Cost improvements start leveling out as more vehicles are added to the system. Inactivity goes up because more vehicles are available, but also because there is sufficient extra capacity to wait for the next load at the current location rather than to relocate.

Fig. 11 analyzes how long it would take to make back the purchasing costs of the autonomous trucks with the savings obtained by the ATHN. The plot presents results for autonomous trucks that cost \$150k to \$250k each, based on the \$200k estimated by Arizona Bank & Trust (2022). For the baseline scenario with $K = 100$ vehicles, the figure shows a payback period of less than 1.5 years, demonstrating a rapid return on investment for all parameters. Again it can be seen that the first vehicles are the most profitable, and a system with only 10 trucks would pay back the trucks within half a year.

The number of autonomous trucks in the system also affects the routes that are created. Figs. 12 and 13 present routes taken from the $K = 10$ and $K = 150$ vehicle solutions, respectively. Blue arrows represent loaded trucks, and red arrows correspond to driving empty. For the $K = 10$ case, nine of the ten routes are clustered in the East (Figs. 12(a) and 12(b)), and one route also serves loads in the West (Fig. 12(c)). It is clear that the optimization uses the limited number of trucks to serve the most dense region, as shown in Fig. 4. When the number of trucks increases, Fig. 13 shows that routes start to cover all regions of the US. The Ryder case study therefore suggests a roll-out strategy that starts in the East and expands from there.

7.3. Regional and temporal decomposition

In practice it may be necessary to plan ATHN operations for a specific region or for a specific time horizon. Examples are if states do not share order information, or if loads are only announced a week in

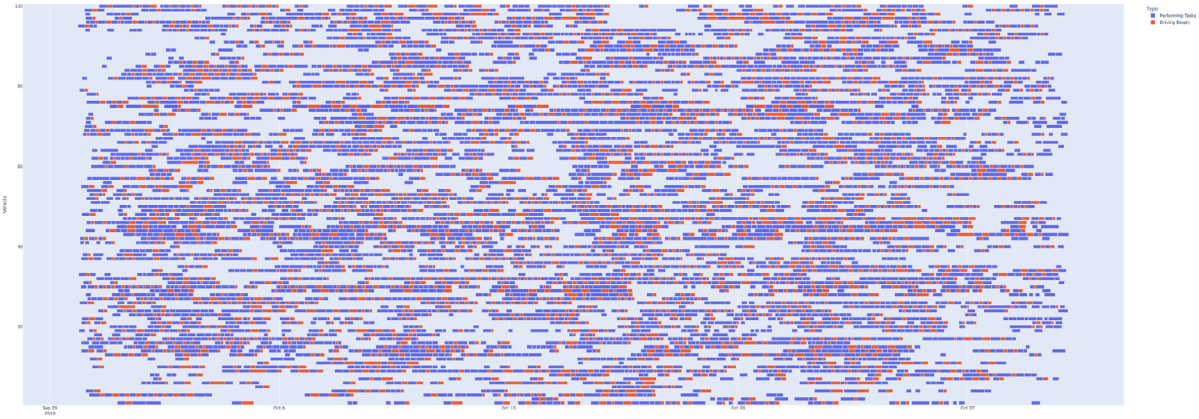


Fig. 9. Truck schedule for increased flexibility of $\Delta = 24$h ($\alpha = 40\%$, MIP start enabled).

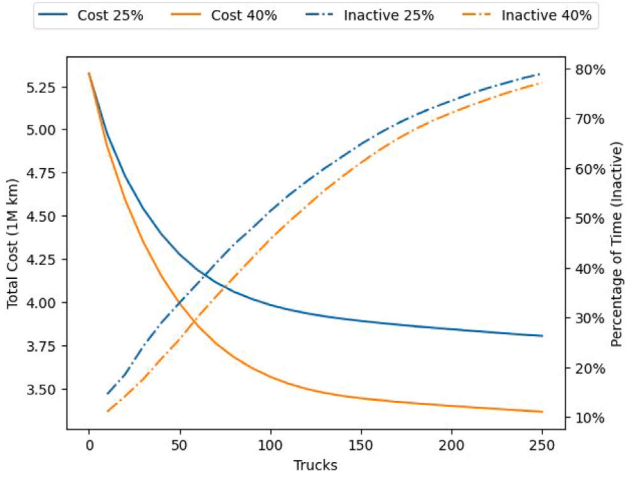


Fig. 10. Total cost and autonomous truck inactivity.

advance. Decomposing the problem is also a strategy to speed up the optimization.

Regional decomposition. The US is comprised of smaller areas where the ATHN framework can be applied separately according to Section 4.2. This study focuses on the impact of regional decomposition on the ATHN problem. Three regional decomposition schemes: 1. Region, 2. Division, and 3. State, adopted from the U.S. Census Bureau (2010), are used to solve the base cases with $K = 100$.

Fig. 14 presents the summary of the regional decomposition. As expected, a finer regional decomposition decreases the number of loads served by the ATHN and lowers the savings. But surprisingly, most of the benefits are still obtained when ATHN operations are planned on the region or division level. Planning at the state level is significantly more expensive, although ATHN still outperforms the current system by 13% to 21%. This indicates that a successful system is likely to require collaboration between states. Fig. 15 presents the allocation of autonomous trucks to regions or divisions. Recall that this is optimized by the model as a byproduct of the regional decomposition (Section 4.2). Both plots indicate that a majority of the trucks are allocated to the South and in the Northeast, in line with the data presented in Fig. 4. Fig. 16 presents an example route for the Northeast region for $\alpha = 25\%$. Note that vehicles are allowed to leave the region, but have to return empty before they can pick up the next load. This corresponds to a situation where regions have joint infrastructure but do not communicate about orders.

Temporal decomposition. Fig. 17 summarizes the results for decomposing the optimization problem into smaller time periods that are solved on a rolling horizon as described in Section 4.2. Surprisingly, Fig. 17(a) shows that the ATHN can be operated on a weekly basis at only a minimal increase in cost. This indicates that a week is sufficiently long to deal with the vehicle imbalances created during the previous period, without the need for explicit rebalancing. Fig. 17(b) shows that essentially the same number of loads are served whether the operations are planned on a monthly, biweekly, or weekly basis. This has the potential to greatly simplify planning, as loads do not have to be announced too far in advance. When the planning period is reduced to a single day, it can be seen that the costs do increase. More noticeably, the number of served loads drops significantly. This suggests that the optimization model is still able to perform the most profitable tasks, but is unable to exploit all the opportunities because the autonomous trucks are not in the right place at the beginning of the day.

In terms of solution time, there is also a significant advantage to planning on a weekly basis. Fig. 17(a) includes two line plots that indicate the *total* solving time over the full horizon. For the $\alpha = 25\%$ case, for example, it shows that planning biweekly instead of monthly only requires 35% of the solving time, while planning weekly instead of monthly only requires 12% of the solving time. It is concluded for the case study that planning weekly is a practical and fast alternative to planning monthly. Although the quality is worse, planning daily is extremely fast and may be useful to adjust schedules when unforeseen events happen.

7.4. Loading and unloading time

The baseline assumes that loading or unloading an autonomous truck takes 30 min. However, there remains uncertainty in the time this will take in practice. For example, operators may choose to perform more extensive inspections that increase the loading and unloading time. Fig. 18 presents results for the $K = 100$ case when the loading/unloading time S is varied from zero up to two hours. All instance were solved to optimality, except for the case $\alpha = 40\%$ and $S = 2$ hours which remained at a small gap of less than 0.01%. It can be seen that increasing the loading/unloading time reduces the amount of loads that are served autonomously. At the same time, the overall savings go down, but not by as much. This indicates that when loading/unloading time increases, the optimization model becomes more selective about which loads to serve autonomously and which loads to serve with

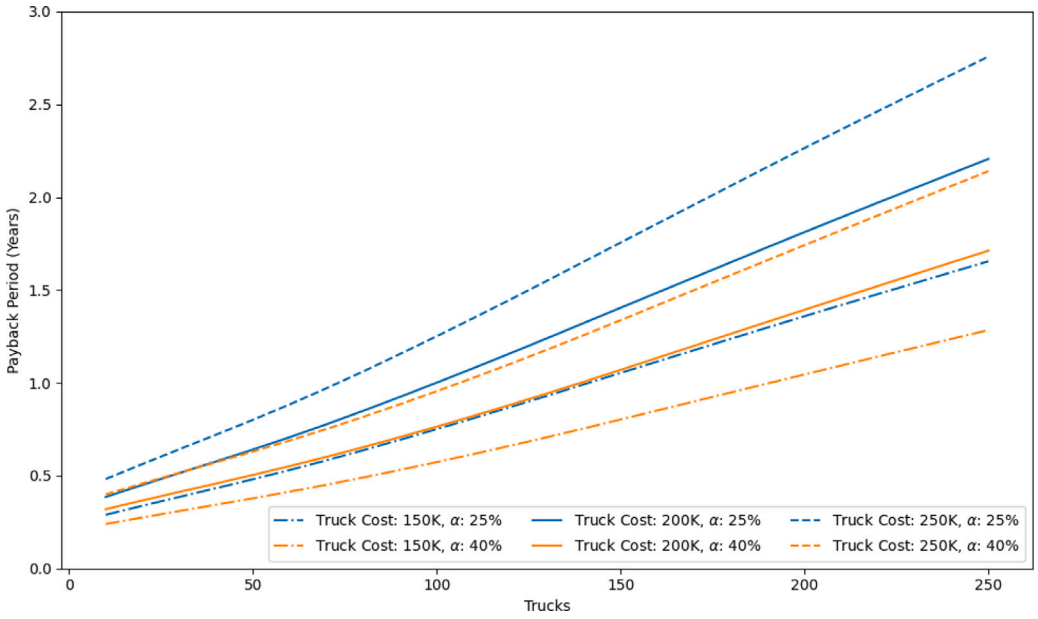
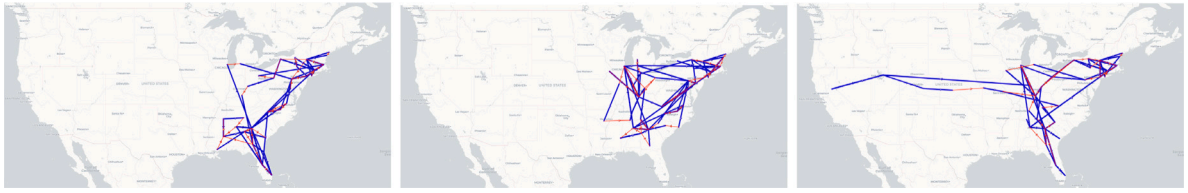
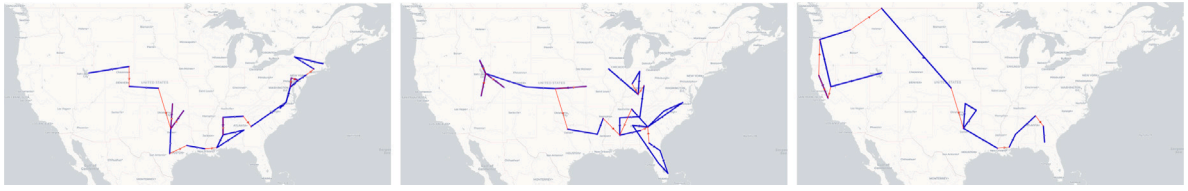


Fig. 11. Payback period for capital investments in autonomous trucks.



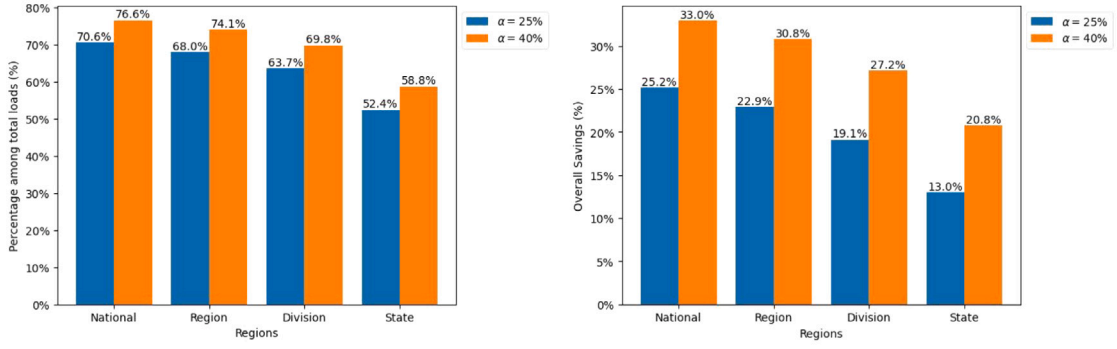
(a) East-focused Route 1 (b) East-focused Route 2 (c) West-focused Route

Fig. 12. Example routes for $K = 10$.



(a) Route 1 (b) Route 2 (c) Route 3

Fig. 13. Example routes for $K = 150$.



(a) Loads (b) Overall Savings

Fig. 14. Statistics for regional decomposition.

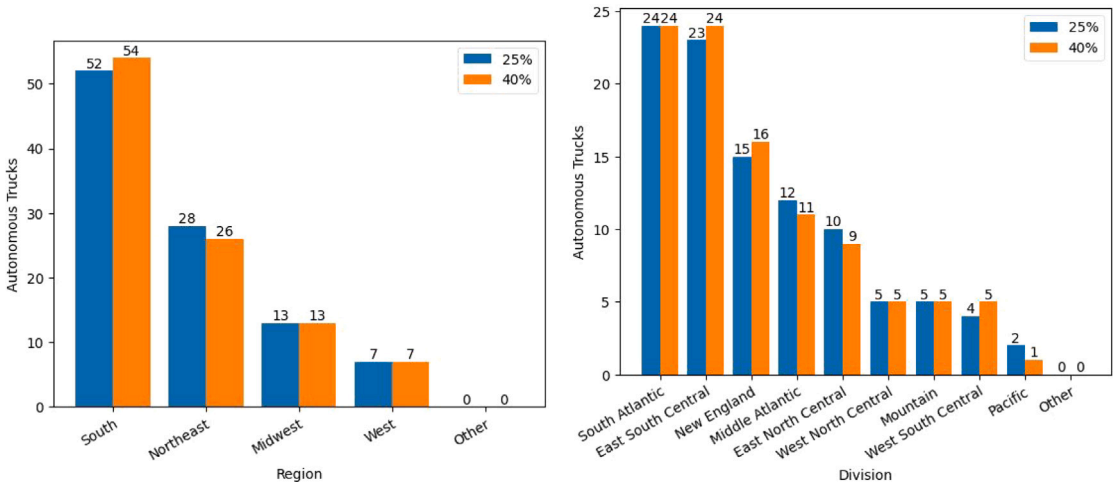


Fig. 15. Vehicle allocation per area.

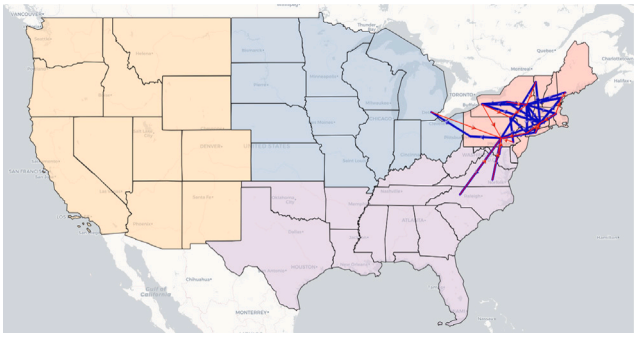


Fig. 16. Example route northeast region for $\alpha = 25\%$.

direct trips. This allows for accommodating a significant increase in loading/unloading time without substantial cost increases.

7.5. Network size

Finally, it is explored how the number of transfer hubs affects the ATHN. Fig. 19 summarizes the results. Fig. 19(a) shows that as the number of hubs increases, more loads will be served autonomously. This effect is due to the shorter first and last mile distances that make it more attractive to use the ATHN. Fig. 19(b) shows that the ATHN remains at least 19.7% cheaper than the current system, even when the number of hubs is reduced to 50. This is an important fact for practical implementation: it is not necessary to deploy the full network at once to obtain substantial savings. For the case study, additional benefits can be obtained by further extending to 200 hubs. In this case the benefits mainly derive from making the first and last miles shorter, rather than capturing additional loads.

8. Conclusion

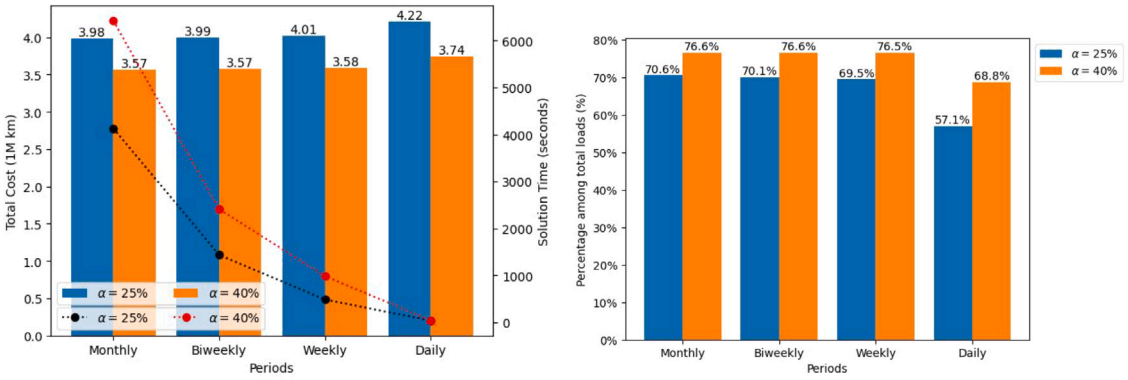
Autonomous trucks are expected to fundamentally transform the freight transportation industry. In particular, Autonomous Transfer Hub Networks (ATHNs), which combine autonomous trucks on middle miles with human-driven trucks on the first and last miles, are seen as the most likely deployment pathway for this technology. This paper presented a framework to optimize ATHN operations based on a flow-based optimization model. The optimization model exploits the problem structure and can be solved by blackbox solvers in a matter of hours. Tools were also provided to perform regional and temporal decomposition in the same framework.

A realistic case study was performed to demonstrate the capability of the optimization model to handle large-scale systems. The case study is based on realistic data provided by Ryder System, Inc., that spans all of the United States over a four-week horizon. Computational results were presented for a system of 100 hubs, 6842 loads, and up to 250 vehicles. The results show that a blackbox solver can solve these instances to within 0.18% of optimality within three hours of computation time.

The case study itself provided many insights into the impact of self-driving trucks on freight transportation. The baseline results indicate cost savings in the range of 20% to 37% for the most challenging orders that are currently served by direct trips. This corresponds to upward of $16.9M per year on the case study. A detailed sensitivity analysis was provided to study various changes to the system. It was shown that increasing the pickup-time flexibility leads to additional potential savings that cannot yet be exploited by the current algorithm, which motivates future work. Using a MIP start significantly improved the performance for these difficult instances. The analysis of the number of trucks demonstrates the trade-off between cost reduction and truck inactivity. It also shows the recommended path of implementation for the case study, starting with trucks that exclusively serve the busy East and West, and moving to routes that cover the whole country as the number of trucks increases.

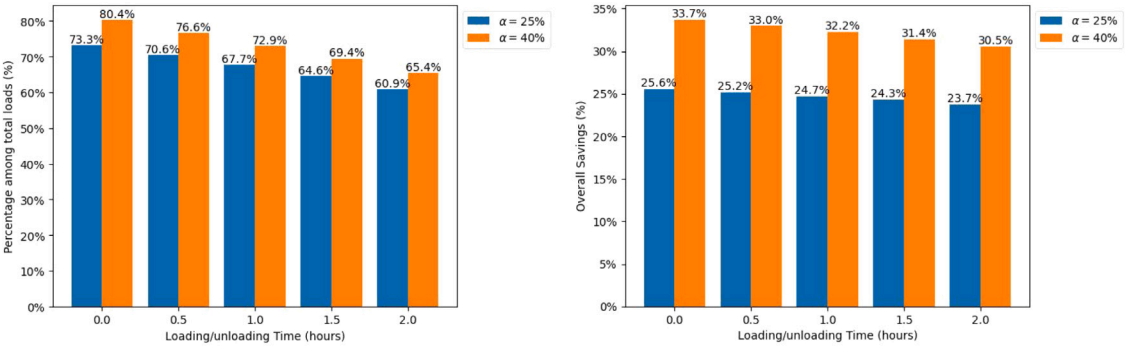
Decomposing the optimization problems into regions and into smaller time periods led to some interesting practical insights. For the case study, it was shown that planning on the regional level maintains many of the benefits of planning on the national level. Similarly, it was found that planning one week ahead on a rolling horizon leads to surprisingly good results and also significantly reduced solving time by 88%. Daily planning is extremely fast and may be useful to adjust schedules when unforeseen events happen. The analysis of the loading/unloading time showed that long loading/unloading times of up to two hours (e.g., to perform inspections) can be accommodated with minimal impact on the system's cost-effectiveness. And finally, varying the size of the network demonstrated that the ATHN can still be run efficiently when the number of hubs is reduced from 100 to 50.

In conclusion, it is found that the optimization framework for ATHN is an effective tool to study large full truckload systems and gain insights from real data. Interesting directions for future work include studying the effect of ATHN on human labor beyond the drivers,



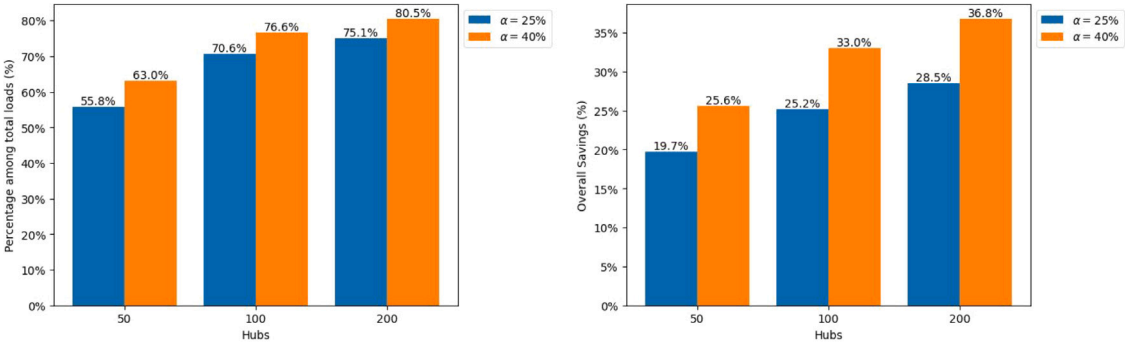
(a) Total Cost and Computational Time (b) Loads

Fig. 17. Statistics for temporal decomposition.



(a) Loads (b) Overall Savings

Fig. 18. Statistics for varying loading/Unloading time.



(a) Loads (b) Overall Savings

Fig. 19. Statistics for varying number of hubs.

e.g., support staff at the transfer hubs. A first step in that direction has been taken by Lee et al. (2023), which presents a constraint programming method to optimize the size of each hub. Another future direction could be to incorporate the possibility of platooning into the model, to save additional costs. Finally, it would be interesting to extend the framework to less-than-truckload transportation.

CRediT authorship contribution statement

Chungjae Lee: Data curation, Formal analysis, Investigation, Methodology, Software, Validation, Visualization, Writing – original draft, Writing – review & editing. **Kevin Dalmeijer:** Conceptualization, Data curation, Formal analysis, Investigation, Methodology, Project administration, Software, Supervision, Validation, Visualization, Writing – original draft, Writing – review & editing. **Pascal Van Hentenryck:** Conceptualization, Data curation, Formal analysis, Funding acquisition, Investigation, Methodology, Project administration, Supervision, Validation, Writing – original draft, Writing – review & editing. **Peibo Zhang:** Data curation, Formal analysis, Investigation, Methodology, Software, Visualization, Writing – original draft.

Declaration of AI and AI-assisted technologies in the writing process

During the preparation of this work the authors used ChatGPT to improve readability. After using this service, the authors reviewed and edited the content as needed and take full responsibility for the content of the publication.

Acknowledgments

This research was partly funded through a gift from Ryder and partly supported by the NSF AI Institute for Advances in Optimization (Award 2112533). Special thanks to the Ryder team for their invaluable support, expertise, and insights.

References

Ahuja, Ravindra K., Magnanti, Thomas L., Orlin, James B., 1993. Network Flows: Theory, Algorithms, and Applications. Prentice-Hall, ISBN: 0-13-617549-X.

Al Hajj Hassan, Lama, Hewitt, Mike, Mahmassani, Hani S., 2022. Daily load planning under different autonomous truck deployment scenarios. Transp. Res. E 166, 102885. http://dx.doi.org/10.1016/j.tre.2022.102885.

Arizona Bank & Trust, 2022. How far are we from automated and electric semi-trucks? In: Experts Weigh in. https://www.arizbank.com/business-insights/automated-electric-semi-trucks-experts.

Arunapuram, Sundararajan, Mathur, Kamlesh, Solow, Daniel, 2003. Vehicle routing and scheduling with full truckloads. Transp. Sci. 37 (2), 170–182. http://dx.doi.org/10.1287/trsc.37.2.170.15248.

Berger, Roland, 2018. Shifting up a gear – automation, electrification and digitalization in the trucking industry. https://www.rolandberger.com/publications/publication_pdf/roland_berger_trucking_industry.pdf.

Black, Dan, Eglese, Richard, Wøhlk, Sanne, 2013. The time-dependent prize-collecting arc routing problem. Comput. Oper. Res. 40 (2), 526–535. http://dx.doi.org/10.1016/j.cor.2012.08.001.

Campbell, Ann, Savelsbergh, Martin, 2004. Efficient insertion heuristics for vehicle routing and scheduling problems. Transp. Sci. 38 (3), 369–378. http://dx.doi.org/10.1287/trsc.1030.0046.

Chen, Shukai, Wang, Hua, Meng, Qiang, 2021. Autonomous truck scheduling for container transshipment between two seaport terminals considering platooning and speed optimization. Transp. Res. B 154, 289–315. http://dx.doi.org/10.1016/j.trb.2021.10.014.

Dalmeijer, Kevin, Van Hentenryck, Pascal, 2021. Optimizing freight operations for autonomous transfer hub networks. arXiv:2110.12327.

de Almeida Correia, Gonçalo Homem, van Arem, Bart, 2016. Solving the user optimum privately owned automated vehicles assignment problem (UO-POAVAP): A model to explore the impacts of self-driving vehicles on urban mobility. Transp. Res. B 87, 64–88. http://dx.doi.org/10.1016/j.trb.2016.03.002.

Desrosiers, Jacques, Dumas, Yvan, Solomon, Marius M., Soumis, François, 1995. Time constrained routing and scheduling. In: Handbooks in Operations Research and Management Science, Vol. 8. Elsevier, pp. 35–139. http://dx.doi.org/10.1016/s0927-0507(05)80106-9.

Dezső, Balázs, Jüttner, Alpár, Kovács, Péter, 2011. LEMON – an open source C++ graph template library. Electron. Notes Theor. Comput. Sci. 264 (5), 23–45. http://dx.doi.org/10.1016/j.entcs.2011.06.003.

Engholm, Albin, Pernestål, Anna, Kristoffersson, Ida, 2020. Cost analysis of driverless truck operations. Transp. Res. Rec. 2674 (9), 511–524. http://dx.doi.org/10.1177/0361198120930228.

FedEx, 2022. Fedex and aurora expand autonomous commercial linehaul trucking pilot in texas ahead of schedule. FedEx Newsroom https://newsroom.fedex.com/newsroom/global-english/fedex-and-aurora-expand-autonomous-commercial-linehaul-trucking-pilot-in-texas-ahead-of-schedule.

Flämig, Heike, 2016. Autonomous vehicles and autonomous driving in freight transport. In: Markus Maurer, J., Gerdes, Christian, Lenz, Barbara, Winner, Hermann (Eds.), Autonomous Driving: Technical, Legal and Social Aspects. Springer, pp. 365–385. http://dx.doi.org/10.1007/978-3-662-48847-8_18.

FleetOwner, 2021. Tusimple among autonomous truck companies to join self-driving coalition. https://www.fleetowner.com/technology/autonomous-vehicles/article/21152006/tusimple-among-autonomous-truck-companies-to-join-selfdriving-coalition.

Forbes, 2021. Plus partners with IVECO to develop automated trucks for global deployment. https://www.forbes.com/sites/richardbishop1/2021/04/12/plus-partners-with-iveco-to-develop-automated-trucks-for-global-deployment.

FreightWaves, 2021. Gatik, Isuzu to partner on autonomous truck platform. https://www.freightwaves.com/news/gatik-isuzu-to-partner-on-autonomous-truck-platform.

Freling, Richard, Wagelmans, Albert P.M., Paixão, José M. Pinto, 2001. Models and algorithms for single-depot vehicle scheduling. Transp. Sci. 35 (2), 165–180. http://dx.doi.org/10.1287/trsc.35.2.165.10135.

Greene, William, 2013. Autonomous freight vehicles: They're heeeeere! In: Shanker, Ravi, Jonas, Adam, Devitt, Scott, Huberty, Katy, Flannery, Simon, Greene, William, et al. (Eds.), Autonomous Cars – Self-Driving the New Auto Industry Paradigm. Morgan Stanley & Co. LLC, pp. 85–89.

Gronalt, Manfred, Hartl, Richard F., Reimann, Marc, 2003. New savings based algorithms for time constrained pickup and delivery of full truckloads. European J. Oper. Res. 151 (3), 520–535. http://dx.doi.org/10.1016/s0377-2217(02)00650-1.

Hadjar, Ahmed, Marcotte, Odile, Soumis, François, 2006. A branch-and-cut algorithm for the multiple depot vehicle scheduling problem. Oper. Res. 54 (1), 130–149. http://dx.doi.org/10.1287/opre.1050.0240.

Lee, Chungjae, Boonbandansook, Wirattawut, Akhlaghi, Vahid Eghbal, Dalmeijer, Kevin, Van Hentenryck, Pascal, 2023. Constraint programming to improve hub utilization in autonomous transfer hub networks. In: Yap, Roland H.C. (Ed.), International Conference on Principles and Practice of Constraint Programming. In: Leibniz International Proceedings in Informatics, vol. 280, pp. 46:1–46:11. http://dx.doi.org/10.4230/LIPIcs.CP.2023.46.

Lee, Chungjae, Dalmeijer, Kevin, Van Hentenryck, Pascal, 2022. Optimization models for autonomous transfer hub networks. arXiv:2201.06137.

Miller, Clair E., Tucker, Albert W., Zemlin, Richard A., 1960. Integer programming formulation of traveling salesman problems. J. ACM 7 (4), 326–329. http://dx.doi.org/10.1145/321043.321046.

OpenStreetMap, 2021. Planet dump. retrieved from https://planet.osm.org. URL https://www.openstreetmap.org.

PACE, 2017. Partnership for an advanced computing environment (PACE). URL http://www.pace.gatech.edu.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., Duchesnay, E., 2011. Scikit-learn: Machine learning in python. J. Mach. Learn. Res. 12, 2825–2830, URL https://jmlr.org/papers/v12/pedregosa11a.html.

Ribeiro, Celso C., Soumis, François, 1994. A column generation approach to the multiple-depot vehicle scheduling problem. Oper. Res. 42 (1), 41–52. http://dx.doi.org/10.1287/opre.42.1.41.

Ryder System, Inc., Socially Aware Mobility Lab, 2021. The Impact of Autonomous Trucking: A Case-Study of Ryder's Dedicated Transportation Network. Ryder Newsroom, https://newsroom.ryder.com/news/news-details/2021/Ryder-Teams-Up-with-Georgia-Tech-for-Industrys-First-Data-Driven-Study-on-Impact-of-Autonomous-Trucking/.

SAE International, 2018. Taxonomy and definitions for terms related to driving automation systems for on-road motor vehicles.

Scherr, Yannick Oskar., Hewitt, Mike, Neumann-Saavedra, Bruno Albert, Mattfeld, Dirk Christian, 2020. Dynamic discretization discovery for the service network design problem with mixed autonomous fleets. Transp. Res. B 141, 164–195. http://dx.doi.org/10.1016/j.trb.2020.09.009.

Scherr, Yannick Oskar, Neumann-Saavedra, Bruno Albert, Hewitt, Mike, Mattfeld, Dirk Christian, 2018. Service network design for same day delivery with mixed autonomous fleets. Transp. Res. Procedia 30, 23–32. http://dx.doi.org/10.1016/j.trpro.2018.09.004.

Shahandasht, Mohsen, Pudasaini, Binaya, McCauley, Sean Logan, 2019. Autonomous Vehicles and Freight Transportation Analysis. Technical report, The University of Texas at Arlington.

Short, Jeffrey, Murray, Dan, 2016. Identifying Autonomous Vehicle Technology Impacts on the Trucking Industry. American Transportation Research Institute, URL http://atri-online.org/2016/11/15/identifying-autonomous-vehicle-technology-impacts-on-the-trucking-industry/.

Slowik, Peter, Sharpe, Ben, 2018. Automation in the Long Haul: Challenges and Opportunities of Autonomous Heavy-Duty Trucking in the United States. The International Council on Clean Transportation, URL https://theicct.org/publications/automation-long-haul-challenges-and-opportunities-autonomous-heavy-duty-trucking-united.

Steinzen, Ingmar, Gintner, Vitali, Suhl, Leena, Kliewer, Natalia, 2010. A time-space network approach for the integrated vehicle- and crew-scheduling problem with multiple depots. Transp. Sci. 44 (3), 367–382. http://dx.doi.org/10.1287/trsc.1090.0304.

TechCrunch, 2023. Tusimple tests removing human driver from self-driving truck in China. https://techcrunch.com/2023/06/19/tusimple-tests-removing-human-driver-from-self-driving-truck-in-china/.

Toth, Paolo, Vigo, Daniele (Eds.), 2014. Vehicle Routing: Problems, Methods, and Applications, second ed. SIAM, http://dx.doi.org/10.1137/1.9781611973594.

Transport Topics, 2023. Torc lays out road map to autonomous truck launch in 2027. https://www.ttnews.com/articles/torc-autonomous-launch-27.

U.S. Census Bureau, 2010. Census regions and divisions of the United States. https://www2.census.gov/geo/pdfs/maps-data/maps/reference/us_regdiv.pdf.

U.S. Department of Transportation, 2019. Truck Stop Parking. ArcGIS Online, https://data-usdot.opendata.arcgis.com/datasets/usdot::truck-stop-parking.

Viscelli, Steve, 2018. Driverless? Autonomous Trucks and the Future of the American Trucker. Center for Labor Research and Education, University of California, Berkeley, and Working Partnerships USA, http://driverlessreport.org/.