Computers & Operations Research 164 (2024) 106557

Contents lists available at ScienceDirect

Computers and Operations Research

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

The pollution-routing problem with speed optimization and uneven topography

David Lai [a,b,*], Yasel Costa [c,d], Emrah Demir [e], Alexandre M. Florio [f], Tom Van Woensel [b]

[a] Southampton Business School, University of Southampton, Southampton, United Kingdom
[b] School of Industrial Engineering, Eindhoven University of Technology, Netherlands
[c] MIT-Zaragoza International Logistics Program, Zaragoza Logistics Center, Spain
[d] Universidad de Manizales, Colombia
[e] PARC Institute of Manufacturing, Logistics and Inventory, Cardiff Business School, Cardiff University, Cardiff, United Kingdom
[f] Amazon.com, Inc., Bellevue, Washington 98004, United States

ARTICLE INFO

Keywords:
Routing
Network topography
Speed optimization
Branch-and-price

ABSTRACT

This paper considers a joint pollution-routing with time windows and speed optimization problem (PRP-SO) where vehicle speed, payload, and road grade influence fuel costs and CO_2e emissions. We present two advanced optimization methods (i.e., approximate and exact) for solving the PRP-SO. The approximate strategy solves large-scale instances of the problem with a Tabu search-based metaheuristic coupled with an efficient fixed-sequence speed optimization algorithm. The second strategy consists of a tailored branch-and-price (BP) algorithm to manage speed optimization within the pricing problem. We test both methods on modified Solomon benchmarks and newly constructed real-life instance sets. Our BP algorithm successfully solves the majority of instances involving up to 50 customers and many instances with 75 and 100 customers. The heuristic can find near-optimal solutions to all instances and requires less than one minute of computational time per instance. Results on real-world instances suggest several managerial insights. First, fuel savings of up to 53% can be achieved when explicitly considering arc payloads and road grades. Second, fuel savings and emission reduction can be achieved by scheduling uphill customers later along the routes. Lastly, we show that ignoring elevation information when planning routes leads to highly inaccurate fuel consumption estimates.

1. Introduction

Road freight transportation is vital to the functioning of the economy and the supply chain. However, significant negative impacts on people and the environment must be considered due to excessive energy usage and considerable greenhouse emissions. According to the International Energy Agency (IEA, 2020), global transportation is still responsible for 24% of direct CO_2-equivalent (CO_2e) emissions from fuel combustion. This is especially true in different application contexts such as long-haul freight transportation (Rahman et al., 2013; Koç et al., 2016a), traditional last mile logistics (Savelsbergh and Van Woensel, 2016; Demir et al., 2022), and for large mountainous cities (Giraldo and Huertas, 2019).

Over the past few years, these observations have led to the introduction of pollution- and sustainability-related aspects into traditional Vehicle Routing Problems (VRPs), popularly coined in the literature as the Pollution-Routing Problem (PRP; Bektaş and Laporte, 2011). In the literature, similar problems and definitions can be found as the Emissions Minimizing VRPs (EMVRPs) (Raeesi and Zografos, 2019; Behnke et al., 2021) or Green VRPs (Erdoğan and Miller-Hooks, 2012; Moghdani et al., 2020; Abdullahi et al., 2021). These models use the fact that the amount of transport-related greenhouse gas (GHG) emissions is directly proportional to fuel consumption (Kirby et al., 2000). Multiple factors have already been considered, including slope (Suzuki, 2011), vehicle speed (Demir et al., 2012), the payload (Bektaş and Laporte, 2011), traffic congestion (Franceschetti et al., 2013), driver's operating habits (Bandeira et al., 2013), and the fleet size and mix (Koç et al., 2014).

The majority of PRP models describe the road angle, and thus the network topography, as one of the parameters used to formulate the instantaneous engine-out emission rate (Barth and Boriboonsomsin, 2009), but do not consider this further in the model, in the solution methodology or the results and insights. More specifically, efficient solution methods and extensive computational experiments that analyze the effect of road gradient on fuel consumption and CO_2e emissions are

* Corresponding author at: Southampton Business School, University of Southampton, Southampton, United Kingdom.
E-mail addresses: david.lai@soton.ac.uk (D. Lai), ycosta@zlc.edu.es (Y. Costa), demire@cardiff.ac.uk (E. Demir), aflorio@gmail.com (A.M. Florio), t.v.woensel@tue.nl (T. Van Woensel).

https://doi.org/10.1016/j.cor.2024.106557
Received 20 September 2023; Received in revised form 19 December 2023; Accepted 18 January 2024
Available online 1 February 2024

missing in the literature. One notable exception in the same application domain is the paper by Brunner et al. (2021). However, this work assumes that speed (and also the payload) is a user-defined input parameter, which could lead to significant deviations (up to 20%) in the optimization process of CO_2e emissions (Van Woensel et al., 2001). With regards to vehicle speed, several studies propose various optimization procedures (Norstad et al., 2011; Demir et al., 2012; Kramer et al., 2015a). However, there is still a need for efficient and fast speed optimization algorithms to use in combination with exact routing algorithms. This paper focuses on joint pollution-routing with time windows and speed optimization problems (PRP-SO). Motivated by down-speeding strategies commonly used in freight transportation, we focus on optimizing the speed of vehicles for reducing CO_{2e} emissions and fuel consumption without violating the customers' time windows.

Specifically, the PRP aims to build routes that minimize an objective function, integrating the vehicle's routing cost (e.g., fuel consumption, pollution) and driving costs (e.g., vehicle usage, drivers' wages, and other direct cost aspects). This paper builds upon this literature and considers modeling driving costs as fixed costs of vehicles, which is commonly used in heterogeneous vehicle routing problems (see, e.g., Koç et al., 2016b), but is often ignored and treated as duration-dependent costs in PRPs. Moreover, considering this richer version of the PRP leads to interesting new methodological challenges for speed optimization.

The contributions of this paper are threefold:

- We introduce the road gradient to compute fuel consumption utilizing terrain elevation information. Despite the inclusion of road gradients into the original formulation of fuel consumption, most papers assume a constant road angle for all vehicle trips.
- Several novel solution approaches are presented, including an exact branch-and-price (BP) algorithm and a Tabu Search (TS) metaheuristic for solving medium- and large-scale PRP-SO instances. Key to the efficiency of these approaches is a novel speed optimization algorithm.
- Based on results obtained in a broad set of instances, we derive essential insights regarding the importance of considering road gradients when solving the PRP-SO and the potential benefits of employing optimized down-speeding strategies. Furthermore, we propose a new set of real-life benchmark instances of the problem.

The remainder of this paper is organized as follows. Section 2 briefly reviews the related scientific literature. In Section 3, we describe our PRP-SO model. Section 4 shows the proposed Tabu Search (TS) metaheuristic with a detailed description of the fixed-sequence speed optimization algorithm. In Section 5, we present the major components of our BP algorithm. Section 6 discusses an extensive computational study on several instance sets. Finally, we draw concluding remarks and present further research directions in Section 7.

2. Literature review

The increasing amount of CO_2e emissions from road freight transportation has ignited worldwide concerns. Over the last ten years, this resulted in a large body of literature on emissions-aware transportation problems (see, e.g., Moghdani et al., 2020; Marrekchi et al., 2021). The Pollution-Routing Problem (PRP) is an efficient and comprehensive formulation to reduce carbon emissions. The resulting greenhouse emissions from vehicle fuel consumption are the result of some influential factors beyond the travel distance (Ericsson, 2001; Brundell-Freij and Ericsson, 2005). According to Demir et al. (2014b), vehicle fuel consumption is affected by multiple factors such as speed, road gradient, road congestion, driver's operating habit, size and composition (mix) of vehicle fleet, and payload.

As summarized in Table 1, some critical factors have been less studied in previous vehicle routing research. Specifically, the interaction of load, road gradient, and vehicle speed is missing.

Road gradient. The road grade (slope) has a significant influence on both the conventional-vehicle fuel economy (Suzuki, 2011) and the electric-vehicle energy consumption (Goeke and Schneider, 2015). Transiting a typical light-duty vehicle over a sloping road surface, with a +6 percent grade, could increase fuel consumption by 15–20 percent (Boriboonsomsin and Barth, 2009). The influence of hilly roads is undoubtedly more significant on the fuel consumption of heavy-duty trucks. According to Davis et al. (2009), a minor increase or decrease in road grade (1%–4%) can reduce or increase fuel economy by more than 50%. In the case of electric vehicles, experimental results show the impact of hilly terrain on miles traveled, confirming that the range of electric vehicles in mountainous landscapes is lower (Travesset-Baro et al., 2015).

The road gradient has been mainly studied in PRPs and the so-called Electric Vehicle Routing Problems (E-VRPs). Since the first mathematical formulation of PRPs (Bektaş and Laporte, 2011), the road angle was one of the parameters used to define the instantaneous engine-out emission rate. Table 1 clearly shows that most previous PRP contributions included the road angle in the fuel use rate mathematical formulation. This table also identifies two significant gaps in the area that the proposed work is trying to address: (i) despite the inclusion of road gradient into the original formulation of fuel consumption, most of the papers assume that the road angle remains constant throughout all vehicle trips, and (ii) the majority of previously generated problem instances have not yet considered the road gradient. It is worth mentioning that the present work is the first study in the area to create and optimally solve a set of realistic problem instances, which include elevation information for computing the road slopes along the paths.

As mentioned earlier, we found the paper by Brunner et al. (2021) as the only contribution that studied the influence of road grade on fuel consumption. Although these authors considered the slope in their VRP formulation, they assumed a constant road grade for all arcs that form the directed graph. The above does not accurately reflect the hilly topography profile of a road, which typically connects two nodes (arc) with a sequence of multiple uphill/downhill segments. Another assumption in this contribution is related to vehicle speed. The authors addressed a VRP without time windows, assuming that the vehicles travel through each arc with a given constant speed (input parameter). Last but not least, with respect to the assumptions of the paper, in Brunner et al. (2021), the payload carried by the vehicle is chosen from a prescribed set of values, which means that the authors did not consider the payload as a continuous decision variable.

Few papers outside the application context of PRPs consider the effect of road gradients on fuel consumption. For instance, Tavares et al. (2009) utilize an exponential regression model (COPERT-III method, introduced by Ntziachristos et al., 2000) to estimate the minimum fuel consumption during the waste collection process. They studied two realistic routing problems, showing that the optimal route does not necessarily correspond to the shortest distance. Their computational results demonstrated that significant fuel savings are possible for longer routes with moderate inclination on the road. Moreover, the contribution of Suzuki (2011) proposed a linear regression model to compute the fuel consumption rate for a heavy-duty truck based on the fuel-efficiency study of Davis et al. (2009). The author included the road-gradient factor as one of the objective function components (distance and fuel consumption), designed to formulate a traveling salesman problem with time windows.

Concerning the E-VRPs, Yang et al. (2014) performed a numerical simulation. In their paper, the authors used the electric vehicle's battery (physical model) theory to study the effects of the road's slope on electricity consumption for uphill and downhill paths. They concluded that with the increase of the uphill tilt angle, each electric vehicle's electricity consumption increases significantly. Goeke and Schneider (2015) also assumed not-flat terrain with grades in their energy consumption model for electric vehicles. They intensely focused on the effect of load distribution on the performance of commercial electric

Table 1
Pollution-related Factors Covered by Previous Research on PRPs.

Authors	Pollution-related factors(model formulation)				Factors included in the computational analysis	Type of dataset	Solution approach
	load	speed	slope	others			
Bektaş and Laporte (2011)	√	√	√		load, speed	UK	CPLEX
Demir et al. (2012)	√	√	√		speed	UK	ALNS
Franceschetti et al. (2013)	√	√	√	√	departure time, speed,traffic congestion	UK	CPLEX
Demir et al. (2014a)	√	√	√		driving time, speed	UK	ALNS
Koç et al. (2014)	√	√	√	√	speed, fleet size and mix	UK	HEA
Kramer et al. (2015a)		√		√	departure time, speed	Modified UK	ILS
Kramer et al. (2015b)	√	√		√	departure time, speed	Modified UK	ILS
Fukasawa et al. (2016)	√	√			speed	UK	DCP
Dabia et al. (2017)		√	√	√	start-time, speed	UK	B&P
Rauniyar et al. (2019)	√	√			load	UK	NSGA-II
Raeesi and Zografos (2019)	√	√	√	√	load, fleet size and mix,road congestion	New dataset	CPLEX
Xiao et al. (2020)	√	√		√	travel-arrival-departure-waiting time, speed, load	UK	CPLEX
Brunner et al. (2021)	√		√		arc slope, fixed load	New dataset	Tailored heuristic
Erdoğdu and Karabulut (2022)	√				load	New dataset	ALNS
Proposed work	√	√	√	√	**road gradient**, speed, load	**Noveldataset**	**B&P, TS**

Abbreviations – CPLEX: IBM Commercial Solver; ALNS: Adaptive Large Neighborhood Search; HEA: Hybrid Evolutionary Algorithm; ILS: Iterated Local Search; DCP: Disjunctive Convex Programming; B&P: Branch and Price; NSGA-II: Non-dominated Sorting Genetic Algorithm; TS: Tabu Search.

vehicles. Later, Liu et al. (2017) also investigated the impact of the road gradient on the electricity consumption of electric cars. Using 12 gradient ranges and GPS tracking data with a digital elevation map, the authors showed that, on uphill trips, energy consumption increases almost linearly with the absolute gradient. However, the numerical results also showed the positive effect of the regenerative braking power acquired during downhill trips. Finally, Macrina et al. (2019) modeled a comprehensive energy consumption function considering road gradients. However, in their computational study, the authors set the road angle equal to zero for conventional and electric vehicles.

Vehicle speed. Bektaş and Laporte (2011) first addressed the vehicle's speed in PRPs. In the original formulation of this problem, they explicitly assumed that speed over each arc is chosen from a predefined list of possible values. Koç et al. (2014) also utilized the same discrete speed function but analyzed, for the first time, the effect of fleet size and mix in PRPs. The discretization of vehicle speed was also adopted by Eshtehadi et al. (2017). Here, the authors considered the demand and travel time uncertainty, both aspects addressed by several robust optimization techniques. Moreover, Demir et al. (2012) proposed a specialized speed optimization algorithm (SOA), which computes optimal speeds on a given path to minimize fuel consumption, emissions, and driver costs. The authors modified the original SOA, initially designed to solve the tramp speed optimization problem (Norstad et al., 2011). Franceschetti et al. (2013) also studied the speed and traffic congestion, and named it as the time-dependent PRP. This research considers two phases within the planning horizon: the free-transit phase and the congested phase. An interesting fact of their computational results is that they reduced the emissions cost by waiting at specific locations (stopped vehicles) and avoiding traffic congestion. Kramer et al. (2015a) also modified the original SOA, providing a new speed and departure time optimization algorithm. Fukasawa et al. (2018) consider a joint vehicle routing and speed optimization problem and solve it with a branch-and-price algorithm. In this work, the fuel consumption rate depends on the traveling speed and is assumed to be strictly convex.

Finally, many solution methods have been proposed to solve PRPs. One can find the frequent application of metaheuristics such as ALNS (Demir et al., 2012, 2014a), evolutionary and genetic algorithms (Koç et al., 2014; Rauniyar et al., 2019), and hybrid approaches (Tirkolaee et al., 2020). Table 1 shows the methods used for solving different variants of PRPs. As can be seen, the utilization of exact algorithms in PRPs is minimal. Existing exact methods often approximated or discretized the vehicle speeds to reduce the complexity. Fukasawa et al. (2016) resolved the issues of speed discretization by introducing a formulation framework to directly incorporate the nonlinear relationship between cost and speed into the PRP. They employed different tools from disjunctive convex programming to find a set of vehicle speeds over the routes, minimizing the total cost (operational and environmental) and respecting the constraints on time and vehicle capacities. More recently, vehicle speed has been computed using continuous optimization on PRP. Here Xiao et al. (2020) introduced the continuous PRP (ϵ-CPRP), where the travel time, load flow, departing/arrival/waiting times, and driving speed were treated as continuous decision variables. The authors developed an ϵ-accurate inner polyhedral approximation method for linearizing the original fuel consumption equation (Bektaş and Laporte, 2011) and solved the PRP instances with up to 25 customers.

To our knowledge, Dabia et al. (2017) is the only previous contribution that addressed a complex variant of the PRP and developed an exact branch-and-price algorithm. To address speed optimization within the pricing subproblem, the authors introduced a ready-time function for updating the speed-dependent routing costs within a bidirectional labeling algorithm. More precisely, in Dabia et al. (2017), a complex ready time function has to be solved recursively to determine the optimal speed that minimizes the routing cost of a partial path, which can be computationally expensive. In our proposed labeling algorithm, the optimal speed can be determined efficiently and without complete "backtracking" of the partial path. In addition, we have also developed completion bounds that further improved the efficiency of solving the pricing subproblem.

3. Problem formulation

In Section 3.1, we formulate the carbon dioxide equivalent (CO_2e) emission using the comprehensive modal emissions model (CMEM). In Section 3.2, we introduce a mixed-integer linear programming formulation for the joint Pollution Routing and Speed Optimization problem.

3.1. Modeling CO_2e emissions

We model CO_2e emissions using the CMEM (see, e.g., Barth and Boriboonsomsin (2009), Boriboonsomsin and Barth (2009), Demir et al. (2012)). The parameters and the values for light, medium, and heavy-duty vehicles (LDV, MDV, HDV) we used in our experiments are shown in Table 2.

According to the CMEM model, the instantaneous fuel use rate of a vehicle k when traveling at a constant speed v_k with payload x on a path with the road angle ϕ is given by

$$\frac{\xi}{\kappa\psi}\Big(F_k N_k V_k + \frac{0.5C_k^d A_k \rho v_k^3 + (w_k + x)v_k(r_k + g\sin\phi + gC_k^r\cos\phi)}{1000\varepsilon\varpi}\Big).$$

Table 2
Comprehensive Modal Emissions Model (CMEM)

Symbol	Description	LDV	MDV	HDV
F_k	Engine friction factor (kJ/rev/liter)	0.23	0.20	0.17
N_k	Engine speed (rev/s)	35	34	33
V_k	Engine displacement (liters)	3	7	11
A_k	Frontal surface area of a vehicle (m^2)	5	7.6	8.2
C_k^d	Aerodynamic drag coefficients	0.32	0.55	0.70
C_k^r	Rolling resistance coefficients	0.01	0.009	0.008
r_k	Vehicle acceleration (m/s^2)	0	0	0
w_k	Curb weight (kg)	2,300	5,500	13,000
κ	Heating value for diesel fuel (kJ/g)	45	45	45
ε	Vehicle drive train efficiency	0.4	0.4	0.4
ϖ	Efficiency parameter for diesel engines	0.9	0.9	0.9
ξ	Fuel-to-air mass ratio	1	1	1
ψ	Conversion factor from grams to liters	737	737	737
ρ	Air density (kg/m^3)	1.2041	1.2041	1.2041
g	Gravity (m/s^2)	9.81	9.81	9.81

When traversing a distance of d meters, the amount of fuel consumption is therefore given by

$$\frac{\xi F_k N_k V_k}{\kappa\psi}\frac{d}{v_k}+\frac{1}{1000\varepsilon\varpi}(r_k+g\ sin\ \phi+g\ C_k^r\ cos\ \phi)(w_k+x)(d)+\frac{0.5C_k^d A_k\rho}{1000\varepsilon\varpi}d v_k^2.$$

Note that a sequence of road segments is associated with each arc e, denoted by S_e. Let d_{es} denote the travel distance of segment $s \in S_e$, and x_{ke} denote the payload of vehicle k when traversing arc e. For traversing the sequence of road segments associated with arc e, the amount of fuel consumption of vehicle k can then be determined by

$$\alpha_{ke}\frac{1}{v_k}+\beta_{ke}(w_k+x_{ke})+\gamma_{ke}v_k^2, \tag{1}$$

where

$$\alpha_{ke}=\frac{\xi F_k N_k V_k \sum_{s\in S_e} d_{es}}{\kappa\psi}, \tag{2}$$

$$\beta_{ke}=\frac{\xi\sum_{s\in S_e} d_{es}(r_k+g\ sin\ \phi_{es}+g\ C_k^r\ cos\ \phi_{es})}{1000\varepsilon\varpi\kappa\psi}, \tag{3}$$

$$\gamma_{ke}=\frac{0.5\xi C_k^d A_k\rho\sum_{s\in S_e} d_{es}}{1000\varepsilon\varpi\kappa\psi}. \tag{4}$$

To speed up the CO_2e emission calculations, we pre-compute the parameters α_{ke}, β_{ke}, and γ_{ke} for all the vehicles k and arcs e. These parameters are used when formulating the mathematical model in Section 3.2 and developing the solution approaches in Sections 4 and 5.

3.2. Mathematical model

Let $\mathcal{K}$ be the set of vehicles. Let $G(\mathcal{N},\mathcal{A})$ be the underlying directed graph. A set of nodes $\mathcal{N}=\{0,1,\dots,n\}$ contains n customers and a depot (represented by node 0). The set of arcs $\mathcal{A}$, defined as $\{(i,j)\in\mathcal{N}\times\mathcal{N} : i\neq j\}$, represents the paths between the nodes. Each node $i\in\mathcal{N}$ is associated with a demand q_i and a time window $[e_i, l_i]$. Each vehicle $k\in\mathcal{K}$ is associated with a fixed cost f_k, a variable cost c_k (including the costs for CO_2e emission and fuel consumption), a vehicle capacity Q_k, vehicle curb weight w_k, and speed limits $[a_k, b_k]$. Each arc $e\in\mathcal{A}$ is associated with a distance d_e, and the parameters α_{ke}, β_{ke} and γ_{ke} described in Section 3.1 for estimating fuel consumption.

For all $k\in\mathcal{K}$ and $i\in\mathcal{N}$, let z_{ki} be a binary decision variable, with $z_{ki}=1$ if and only if customer $i\in\mathcal{N}\setminus\{0\}$ is served by vehicle k; and with $z_{k0}=1$ if and only if vehicle k is in use. For all $k\in\mathcal{K}$ and $e\in\mathcal{A}$, let y_{ke} be a binary decision variable with $y_{ke}=1$ if and only if vehicle k traverses arc e, and let x_{ke} denote the corresponding payload when vehicle k traverses arc e. For all $k\in\mathcal{K}$ and $i\in\mathcal{N}$, let t_{ki} denote the time at which vehicle k starts serving customer $i\in\mathcal{N}\setminus\{0\}$; and let t_{k0} denote the time vehicle k returns to the depot. For all $k\in\mathcal{K}$, let v_k denote the vehicle's speed k. We acknowledge the broader problem setting where variable speeds on different arcs within the path could be explored, we opted for the simplicity of a constant speed assumption. This decision was guided by both practical considerations and insights from related studies, such as the work by Dabia et al. (2017), who demonstrated that the potential improvement in solution quality by optimizing vehicle speed on each arc is rather limited. Table 3 provides the list of notations.

The objective is to minimize the total CO_2e emission costs, fuel costs, and vehicle fixed costs, subject to the following constraints: (i) the total demand for a vehicle does not exceed the vehicle capacity; (ii) every route starts and ends at the vehicle's home depot; (iii) every customer is visited precisely once by exactly one vehicle; (iv) all vehicles should return to their home depot within a time limit; (v) every vehicle travels at a speed within the speed limit.

The PRP-SO can now be formulated as the following nonlinear integer programming model:

(PRP-SO) :

$$\min \sum_{k\in\mathcal{K}} f_k z_{k0}+\sum_{k\in\mathcal{K}}\sum_{e\in\mathcal{A}} c_k y_{ke}\Big(\frac{\alpha_{ke}}{v_k}+\beta_{ke}w_k+\beta_{ke}x_{ke}+\gamma_{ke}v_k^2\Big), \tag{5}$$

$$\text{s.t.} \sum_{k\in\mathcal{K}} z_{ki}=1, \quad \forall i\in\mathcal{N}\setminus\{0\}, \tag{6}$$

$$\sum_{e\in\delta^+(i)} y_{ke}=\sum_{e\in\delta^-(i)} y_{ke}=z_{ki}, \quad \forall k\in\mathcal{K}, i\in\mathcal{N}, \tag{7}$$

$$\sum_{e\in\delta^-(i)} x_{ke}-\sum_{e\in\delta^+(i)} x_{ke}=q_i z_{ki}, \quad \forall k\in\mathcal{K}, i\in\mathcal{N}\setminus\{0\}, \tag{8}$$

$$x_{ke}\le(Q_k-q_i)y_{ke}, \quad \forall k\in\mathcal{K}, e=(i,j)\in\mathcal{A}, \tag{9}$$

$$t_{kj}-t_{ki}\ge s_i+d_e\frac{y_{ke}}{v_k}-l_0\Big(1-y_{ke}\Big), \forall k\in\mathcal{K}, e=(i,j)\in\mathcal{A} : j\neq 0, \tag{10}$$

$$e_i z_{ki}\le t_{ki}\le l_i z_{ki}, \quad \forall k\in\mathcal{K}, i\in\mathcal{N}\setminus\{0\}, \tag{11}$$

$$t_{ki}+s_i+d_e\frac{y_{ke}}{v_k}\le l_0, \quad \forall k\in\mathcal{K}, i\in\mathcal{N}\setminus\{0\}, e=(i,0) \tag{12}$$

$$a_k\le v_k\le b_k, \quad \forall k\in\mathcal{K}, \tag{13}$$

$$v_k\in\mathbb{R}^+, \quad \forall k\in\mathcal{K}, \tag{14}$$

$$t_{ki}\in\mathbb{R}^+, \quad \forall k\in\mathcal{K}, i\in\mathcal{N}, \tag{15}$$

$$x_{ke}\in\mathbb{R}^+, \quad \forall k\in\mathcal{K}, e\in\mathcal{A}, \tag{16}$$

$$y_{ke}\in\{0,1\}, \quad \forall k\in\mathcal{K}, e\in\mathcal{A}, \tag{17}$$

$$z_{ki}\in\{0,1\}, \quad \forall k\in\mathcal{K}, i\in\mathcal{N}. \tag{18}$$

The objective function (5) minimizes the total vehicle fixed costs, fuel consumption costs, and CO_2e emission costs. Constraints (6)–(7) ensure that each customer is visited once by exactly one vehicle. Constraints (8) are the flow conservation constraints. Constraints (9) ensure that the payload does not exceed vehicle capacity. Constraints (10)–(12) ensure that customers are visited within the given time windows. Constraints (13) ensure that vehicles travel at a speed within the limits.

Table 3
Notation for the MINLP Model.

Symbol	Description	Domain
$\mathcal{K}$	Set of vehicles	Index set
$\mathcal{N}$	Set of nodes representing n customers and the depot	Index set
$\mathcal{A}$	Set of arcs	Index set
q_i	Demand of customer i	$\mathbb{Z}^+$
w_k	Curb weight of vehicle k	$\mathbb{Z}^+$
Q_k	Capacity of vehicle k	$\mathbb{Z}^+$
e_i	Earliest start time at customer i	$\mathbb{R}^+$
l_i	Latest start time at customer i	$\mathbb{R}^+$
s_i	Service time at customer i	$\mathbb{R}^+$
f_k	Fixed cost of vehicle k	$\mathbb{R}^+$
c_k	Variable cost of vehicle k including fuel and CO_2e emission costs	$\mathbb{R}^+$
a_k	Lower limit of the speed of vehicle k	$\mathbb{R}^+$
b_k	Upper limit of the speed of vehicle k	$\mathbb{R}^+$
d_e	Distance of arc e	$\mathbb{R}^+$
α_{ke}	A constant for estimating the fuel consumption of vehicle k on arc e	$\mathbb{R}^+$
β_{ke}	A constant for estimating the fuel consumption of vehicle k on arc e	$\mathbb{R}^+$
γ_{ke}	A constant for estimating the fuel consumption of vehicle k on arc e	$\mathbb{R}^+$
z_{ki}	Decision variable, to allocate customers to vehicles	$\mathbb{B}$
y_{ke}	Decision variable, to allocate arcs to vehicles	$\mathbb{B}$
x_{ke}	Decision variable, payload of vehicle k when traversing on arc e	$\mathbb{R}^+$
t_{ki}	Decision variable, time at which vehicle k starts serving customer i	$\mathbb{R}^+$
v_k	Decision variable, speed of vehicle k	$\mathbb{R}^+$

4. Metaheuristic

Metaheuristic approaches require frequently evaluating solutions with fixed vehicle routes. We present an efficient algorithm for finding the optimal vehicle speed of a given route to compute the costs of CO_2e emissions and fuel consumption efficiently. In Section 4.1, we present a novel polynomial-time algorithm for solving the fixed-sequence speed optimization subproblem — determining the optimal speed of a vehicle when the customer sequence is fixed. To demonstrate the effectiveness of the speed optimization algorithm, it is embedded into a Tabu Search (TS) metaheuristic for our experiments. The TS algorithm was initially proposed by Glover (1986) and has been widely used in the literature. TS has been widely used and is effective for finding near-optimal solutions to vehicle routing problems (see, e.g., Gendreau et al. (1994), Toth and Vigo (2003), Lai et al. (2016)). The interested reader is referred to Gendreau and Potvin (2019) for more details on the TS algorithm. The major components of the proposed TS include the fixed-sequence speed optimization subproblem in Section 4.1, the penalized objective function in Section 4.2, the initial solutions in Section 4.3, the neighborhood structure in Section 4.4, the intra-route improvement procedure in Section 4.5, and the search procedure in Section 4.6.

4.1. Fixed-sequence speed optimization subproblem

The fixed-sequence speed optimization subproblem determines the optimal speed of a vehicle for a given customer sequence. Let $R = (v_1, v_2, \dots, v_n)$ denote a fixed sequence of nodes in a vehicle route where v_1 and v_n represent the home depot.

Without time-window constraints, vehicles should always travel at a speed that is most cost-efficient and within the speed limits of the vehicle. The most cost-efficient speed of vehicle k when there are no time window constraints is given by

$$\bar{v}_k = \sqrt[3]{\frac{\xi_k F_k N_k V_k 1000 \varepsilon_k \varpi_k}{C_k^d A_k \rho_k}}. \tag{19}$$

Since $\bar{v}_k$ is independent of the vehicle route when there are no time window constraints, the optimal speed of vehicle k within speed limits $[a_k, b_k]$ can be computed by

$$v_k^* = \begin{cases} \bar{v}_k, & \text{if } a_k \le \bar{v}_k \le b_k, \\ a_k, & \text{if } \bar{v}_k \le a_k, \\ b_k, & \text{if } \bar{v}_k \ge b_k. \end{cases} \tag{20}$$

With time window constraints, a vehicle should travel at a speed that satisfies all constraints and incurs a minimal cost. Let $\sigma(R)$ denote the lowest vehicle speed without violating any time windows associated with the route nodes R. When a vehicle travels at the $\sigma(R)$ speed in route R, all time window constraints will be satisfied. We will show in Proposition 1 that $\sigma(R)$ can be computed efficiently by

$$\sigma(R) = \max_{i,j \in \{1,2,\dots,|R|\}: i<j} \frac{\Delta(j) - \Delta(i)}{l_{v_j} - e_{v_i} - S_{ij}}, \tag{21}$$

where $\Delta(i) = \sum_{l=1}^{i-1} d_{v_l, v_{l+1}}$ denotes the total distance from the depot to node i along vehicle route R, $S_{ij} = \sum_{k=i}^{j-1} s_{v_k}$ denotes the total service time from node v_i to node v_{j-1} along vehicle route R, and that $[e_v, l_v]$ is the time window associated on node v. If $\sigma(R) \le 0$, no feasible speed exists due to conflicting time window constraints.

Since a vehicle can wait at the customer node if the vehicle arrives earlier than the lower bound of a time window, any speed greater than $\sigma(R)$ satisfies the time-window constraints in route R. Thus, the optimal speed of vehicle k when traversing on route R is given by

$$\max(v_k^*,\ \sigma(R)). \tag{22}$$

The total cost (as defined in the objective function (5)) of a given solution can be evaluated straightforwardly when the optimal speeds of the vehicle routes have been found by using (21) and (22).

Proposition 1. *The minimum speed without violating any time window constraints of a vehicle route R is given by $\sigma(R)$ when $\sigma(R) > 0$ and can be obtained in $O(n^2)$ time complexity where n is the number of nodes in route R.*

Proof. Let $R = (v_1, v_2, \dots, v_n)$ denote the sequence of nodes in a vehicle route R with both v_1 and v_n representing the home depot, and there is at least one customer node in R i.e. $n \ge 3$. Furthermore, we assume that the nodes in R are not all located at the same location. For $i = 2, \dots, n$, let $\Delta(i) = \sum_{l=1}^{i-1} d_{v_l, v_{l+1}}$ denote the total distance from node v_1 (the depot) to node v_i along vehicle route R. Set $\Delta(1) = 0$. For all $i = 1, 2, \dots, n$ and $j = 2, \dots, n$, with $i < j$, let $S_{ij} = \sum_{k=i}^{j-1} s_{v_k}$ denote the total service time from node v_i to node v_{j-1}. For every distinct pair of nodes $i, j \in \{1, 2, \dots, n\}$, with $i < j$, the following lower bound of $\sigma(R)$ can be derived:

$$\frac{\Delta(j) - \Delta(i)}{l_{v_j} - e_{v_i} - S_{ij}}. \tag{23}$$

Any vehicle speed higher than the above lower bound induced from nodes i and j would violate one of the time window constraints associated with nodes v_i and v_j.

The maximum lower bounds give the minimum speed without violating any time window constraints in R for all distinct pairs of nodes, and thus we have

$$\sigma(R) = \max_{i,j\in\{1,2,\ldots,|R|\}:i<j} \frac{\Delta(j)-\Delta(i)}{l_{v_j} - e_{v_i} - S_{ij}}. \tag{24}$$

Since there are $\frac{n(n-1)}{2}$ distinct pairs of nodes in a vehicle route of length n, $\sigma(R)$ can be computed in $O(n^2)$.

To complete the proof, we will show two conflicting time window constraints exist in route R if and only if $\sigma(R) \le 0$. Consider two cases: (i) $\sigma(R) < 0$; (ii) $\sigma(R) = 0$. Case i: By definition we have $\Delta(j) - \Delta(i) \ge 0$ for all $i, j = 1,\ldots,n$ with $i < j$. Therefore, $\sigma(R) < 0$ iff there exist v_i and v_j with $l_{v_j} - e_{v_i} < S_{ij}$ which implies a violation on one of the time window constraints associated on nodes v_i and v_j. Case ii: Suppose, on the contrary, we consider a vehicle with a speed equal to zero and that all the time window constraints in R are satisfied. Since the due date $l_v \ne \infty$ for all v, we have $\sigma(R) = 0$ iff $\Delta(j) - \Delta(i) = 0$ for all $i, j = 1,\ldots,n$ with $i < j$. Since the nodes in R are not all located at the same location, there exist two distinct nodes v_i and v_j in R with $d_{v_i,v_j} > 0$ and $i < j$. This implies that, there exist v_i and v_j in R with $i < j$ and $\Delta(j) - \Delta(i) > 0$ which leads to a contradiction. □

4.2. Penalized objective function

We allow infeasible solutions in the search space. It is implemented as a penalized objective function obtained by relaxing some constraints and incorporating them into the objective function using self-adjusting penalty parameters.

If a vehicle $k \in \mathcal{K}$ is assigned to a non-empty route R, and is traveling at a speed of v_k, the travel cost $c(R)$ can be written as $c(R) = f_k + \sum_{e\in\mathcal{A}(R)} c_k\left(\frac{\alpha_{ke}}{v_k} + \beta_{ke}w_k + \beta_{ke}\bar{x}_{ke} + \gamma_{ke}v_k^2\right)$ where $\bar{x}_{ke}$ is the payload of vehicle k on arc e and $\mathcal{A}(R)$ denotes the arcs on route R. The overload $\mathrm{P}(R)$ representing the violation of the vehicle capacity constraints is defined as $\mathrm{P}(R) = \left[\sum_{i\in\mathcal{N}(R)} q_i - Q_k\right]^+$ where $\mathcal{N}(R)$ denotes the customer nodes on route R. After incorporating the penalties for possible violations, the penalized objective function value is computed by $z(R) = c(R) + \rho\mathrm{P}(R)$ where $\rho \in \mathbb{R}^+$ is the penalty weight that is self-adjusting in the search. The penalty weight ρ is initialized as 1, and updated in every δ iterations as follows. If $\mathrm{P}(R) = 0$ for all vehicle routes R, then update ρ to 0.5ρ; otherwise, update ρ to 2ρ.

4.3. Initial solution

The initial solution is created by applying a stochastic insertion-based heuristic and then improved by using the TS procedure described in Algorithm 1 with a limited number of iterations. The number of iterations is set to $\lceil I_1 n\rceil$ where I_1 is a user-controlled parameter and n is the number of customers. After creating ten such initial solutions, the best one is returned as the initial solution for the main search procedure described in Section 4.6.

The following stochastic insertion-based heuristic is applied. It begins by assigning exactly one arbitrarily selected customer to each randomly selected l vehicle, where l is a randomly generated integer between 1 and the total number of vehicles available. Then, the remaining customers are considered individually, following a randomized order. For each customer, all possible locations in all routes are evaluated for insertion, and the customer is subsequently inserted into a vehicle route at the position that minimizes the insertion cost (the incremental change in the penalized objective function value). We determine the insertion costs using the algorithm described in Section 4.1 that performs speed optimization for a fixed sequence of nodes on a route.

4.4. Neighborhood structure

Let $\mathcal{X}$ denote the set of all feasible solutions for the instance. We define the solutions in the neighborhood of a given solution $x \in \mathcal{X}$ as $\mathbf{N}(x)$. All possible move operations for all customers are evaluated at each iteration, and the best one is subsequently performed. The move operation involves relocating a customer from its current route to another route at the location that minimizes the insertion cost. The insertion cost is determined by applying the speed optimization algorithm described in Section 4.1. The best move operation is the one that leads to the lowest total penalized objective function value and the following diversification penalty. For a given solution $x \in \mathcal{X}$, we define the diversification penalty as $\phi(x) = \lambda c(x)\sqrt{n}\vartheta_{ir}$ where n is the number of customers, ϑ_{ir} counts the number of times customer i has been moved to route r so far in the search, and λ is a positive parameter that controls the intensity of diversification. Readers can refer to Soriano and Gendreau (1996) for extensive discussion on diversification schemes.

To prevent cycling, if a customer has been moved from route r to route s in a given iteration, then moving the same customer back to route r is declared tabu which implies that this reverse operation is forbidden for the next $\lceil h \log_{10}(n)\rceil$ iterations where h is a user-controlled parameter and n is the number of customers. To prevent the search from stagnating, the following aspiration criterion is applied: a Tabu move is allowed only when the resulting solution is feasible and has an objective function value better than the current best feasible solution found by the search.

4.5. Intra-route improvement procedure

The following procedure is applied to improve the solution by modifying customers' position within the same route: a customer is randomly picked and then reinserted into the best location of the same route until no further improvement is possible. The intra-route improvement procedure is invoked after the selected move operation is performed, the parameter for overload penalty is updated, or when the best feasible solution is improved.

4.6. Search procedure

The TS routine is presented in Algorithm 1. The search procedure consists of two phases. The first phase of the procedure starts by constructing an initial solution as described in Section 4.3. Next, the best solution found in the first phase is improved in the second phase by executing I_2 iterations of the TS routine.

Algorithm 1 Tabu Search

1: **input:** initial solution x_0
2: Set $x = x_0$. If x is feasible, set $z^* = c(x)$ and $x^* = x$; otherwise, set $z^* = \infty$ and $x^* = x$.
3: determine $z(\bar{x})$ and $\phi(\bar{x})$ for all $\bar{x} \in \mathcal{N}(x)$
4: **while** stopping condition is not satisfied **do**
5:     **select** $\bar{x} \in \mathcal{N}(x)$ **that**
6:     - minimizes $z(\bar{x}) + \phi(\bar{x})$
7:     - $\bar{x}$ is non-tabu or it satisfies the aspiration criteria
8:     set the reverse move tabu for θ iterations
9:     perform the intra-route improvement procedure on $\bar{x}$.
10:     **if** $\bar{x}$ is feasible and $z(\bar{x}) < z^*$ **then** set $x^* = \bar{x}$ and $z^* = z(\bar{x})$.
11:     set $x = \bar{x}$, and update the penalty weight of overload for every δ iterations
12:     update $z(\bar{x})$ and $\phi(\bar{x})$ for all $\bar{x} \in \mathbf{N}(x)$
13: **return** x^*

Algorithm 1 shows the TS procedure. In Line 2, the algorithm starts with an initial solution x_0 obtained by running the TS routine with I_1 iterations as described in Section 4.3. In Line 3, the penalized objective function and diversification penalty associated with all the move operations are updated. The loop in Lines 4–12 is invoked and stops until after I_2 iterations. In Lines 5–7, the best neighborhood solution considers the diversification penalty, overload penalty, solutions in the Tabu list, and the aspiration criteria. As described in Section 4.4, the diversification penalty $\phi(x)$ depends on the intensification parameter λ. In Line 8, the reverse move is forbidden for the subsequent $\lceil h \log_{10}(n) \rceil$ iterations. As shown in Lines 9–10, the intra-route improvement procedure described in Section 4.5 is performed on the two routes modified in the best neighborhood solution. The incumbent is updated if a new, best feasible solution is identified. In Lines 11–12, the search moves to the selected neighboring solution. The penalty weight of overload is updated in every δ iterations. Whenever the search moves to a neighboring solution, the penalized objective function and diversification penalty associated with each of the move operations are again updated by using the speed optimization algorithm described in Section 4.1 and the penalized objective function defined in Section 4.2. The computational time is manageable since we only need to update the move operations costs involved with the modified routes. The values for the algorithmic parameters in the search procedure include I_1, I_2, λ, h, and δ, which are set according to the parameter tuning experiment described in Section 6.3.

5. Branch-and-price algorithm

The branch-and-price algorithm is a successful exact method for solving several VRP variants. It relies on reformulating the problem as a set-partitioning (SP) problem and applying branch-and-bound to solve the SP formulation, which provides tight linear bounds. As the number of variables in the SP model grows exponentially with the instance size, column generation is applied to identify profitable variables dynamically and add them to the model. For a review of branch-and-price algorithms used for vehicle routing, we refer to Costa et al. (2019). This section presents the SP-based PRP-SO formulation and describes our branch-and-price solution method, focusing on the specialized algorithm for solving the non-linear column generation subproblem.

5.1. Set-partitioning formulation

A route p is feasible for vehicle k when (i) there exists a value $v \in [a_k, b_k]$ such that no time-window constraint is violated when vehicle k executes route p at speed v; and (ii) the total demand of the customers served along p does not exceed the vehicle capacity Q_k. Let Ω_k be the set of feasible routes for vehicle k. Furthermore, for each $p \in \Omega_k$, let c_p^k be the cost incurred when vehicle k executes route p at the optimal speed, as defined in Section 4.1. Finally, for each route $p \in \Omega_k$ and customer $i \in \mathcal{N} \setminus \{0\}$, let

$$\delta_{ip} = \begin{cases} 1, & \text{if route } p \text{ visits customer } i, \\ 0, & \text{otherwise.} \end{cases}$$

We define binary decision variables λ_p^k for all $k \in \mathcal{K}$ and $p \in \Omega_k$, such that $\lambda_p^k = 1$ if and only if vehicle k executes route p in the solution. Then, the PRP-SO is formulated as the following SP problem:

$$\text{(SP)}: \quad \min \sum_{k\in\mathcal{K}} \sum_{p\in\Omega_k} c_p^k \lambda_p^k, \tag{25}$$

$$\text{s.t.} \sum_{k\in\mathcal{K}} \sum_{p\in\Omega_k} \delta_{ip}\lambda_p^k = 1, \qquad \forall i \in \mathcal{N} \setminus \{0\}, \tag{26}$$

$$\sum_{p\in\Omega_k} \lambda_p^k \le 1, \qquad \forall k \in \mathcal{K}, \tag{27}$$

$$\lambda_p^k \in \{0,1\}, \qquad \forall k \in \mathcal{K}, \forall p \in \Omega_k. \tag{28}$$

The objective function (25) minimizes the total costs. Constraints (26) ensure that each customer is visited exactly once, and constraints (27) enforce a maximum of one route per vehicle.

5.2. Column generation subproblem

The restricted master problem (RMP) refers to the linear relaxation of (25)–(28) with a restricted set of vehicle routes. Let π_i, $i \in \mathcal{N}\setminus\{0\}$, be the dual prices associated with constraints (26) after solving the RMP. The pricing problem for vehicle k is defined as follows:

$$\text{(PP)}: \quad \min f_k + c_k \sum_{e\in\mathcal{A}} \left(\frac{\alpha_{ke}}{v} + \beta_{ke}(w_k + x_e) + \gamma_{ke} v^2 \right) y_e - \sum_{i\in\mathcal{N}\setminus\{0\}} \pi_i z_i, \tag{29}$$

$$\text{s.t.} \sum_{e\in\delta^+(i)} y_e = \sum_{e\in\delta^-(i)} y_e = z_i, \qquad \forall i \in \mathcal{N} \setminus \{0\}, \tag{30}$$

$$\sum_{e\in\delta^+(0)} y_e = \sum_{e\in\delta^-(0)} y_e = 1, \tag{31}$$

$$\sum_{e\in\delta^-(i)} x_e - \sum_{e\in\delta^+(i)} x_e = q_i z_i, \qquad \forall i \in \mathcal{N} \setminus \{0\}, \tag{32}$$

$$x_e \le (Q_k - q_i) y_e, \qquad \forall e = (i,j) \in \mathcal{A}, \tag{33}$$

$$\sum_{i\in\mathcal{N}\setminus\{0\}} q_i z_i \le Q_k, \tag{34}$$

$$t_j - t_i \ge s_i + d_e \frac{y_e}{v} - l_0 \left(1 - y_e\right), \qquad \forall e = (i,j) \in \mathcal{A} : j \ne 0, \tag{35}$$

$$e_i z_i \le t_i \le l_i z_i, \qquad \forall i \in \mathcal{N} \setminus \{0\}, \tag{36}$$

$$t_i + s_i + d_e \frac{y_e}{v} \le l_0, \qquad \forall i \in \mathcal{N} \setminus \{0\}, e = (i,0), \tag{37}$$

$$v \in [a_k, b_k], \tag{38}$$

$$t_i \in \mathbb{R}^+, \qquad \forall i \in \mathcal{N}, \tag{39}$$

$$x_e \in \mathbb{R}^+, \qquad \forall e \in \mathcal{A}, \tag{40}$$

$$y_e \in \{0,1\}, \qquad \forall e \in \mathcal{A}, \tag{41}$$

$$z_i \in \{0,1\}, \qquad \forall i \in \mathcal{N} \setminus \{0\}. \tag{42}$$

The objective function (29) minimizes the route-dependent costs (including vehicle fixed cost and CO_2e emissions cost) and the dual prices of the RMP. The flow conservation constraints are constraints (30)–(31). Constraints (32)–(33) keep track of the payload in the vehicle along each arc traversed. Constraint (34) is the vehicle capacity constraint. Constraints (35)–(37) are the time window constraints. Finally, constraint (38) ensures that the vehicle travels at speed within the prescribed limits.

5.3. Pricing algorithm

The pricing algorithm identifies columns with a negative reduced cost. It finds solutions to (29)–(42) with a negative objective value. As standard in branch-and-price algorithms for vehicle routing, we solve the pricing problem with a labeling algorithm. A label represents a partial route from the depot to a customer. Label extensions are created by extending labels to all feasible customers. Table 4 summarizes the attributes of a label alongside their corresponding initialization values and updating rules.

A key component of our pricing algorithm is the set of label extension procedures that do not require full backtracking to determine the cost of a route under the optimal speed. The idea is formalized with the following definition and proposition.

Definition 2 (*Cost of a path*). Let Q be a path with arcs $(e_1, e_2, \dots, e_L)$ and nodes $(v_0, v_1, v_2, \dots, v_L)$ where $v_0 = v_L = 0$ is the depot. The *cost of*

Table 4
Pricing algorithm: Label attributes, initialization values, and updating rules.

Attribute[a]	Description	Initialization[b]	Updating rule[c]
N_P	Set of customers	$\{i\}$	$N_Q = N_P \cup \{j\}$
n_P	Last customer served	i	j
q_P	Total demand	q_i	$q_Q = q_P + q_j$
D_P	Total distance traveled	d_e	$D_Q = D_P + d_e$
τ_P	Earliest departure time assuming maximum speed	$\max(e_i, d_e/b_k) + s_i$	$\tau_Q = \max(e_j, \tau_P + d_e/b_k) + s_j$
S_P	Total service time before n_P	0	$S_Q = S_P + s_{n_P}$
$\mathcal{T}_P$	Triples of distance, total service time and earliest service time for calculating σ_P	$\{(d_e, 0, e_i)\}$	$\mathcal{T}_Q = \mathcal{T}_P \cup \{(D_Q, S_Q, e_j)\}$
σ_P	Minimum vehicle speed such that all time windows are respected	d_e/l_i	$\sigma_Q = \max_{(d,s,t)\in\mathcal{T}_P} \frac{D_Q - d}{l_j - t - (S_Q - s)}$
v_P	Optimal speed	$\max(v_k^*, d_e/l_i)$	$v_Q = \max(v_k^*, \sigma_Q)$
α_P		α_{ke}	$\alpha_Q = \alpha_P + \alpha_{ke}$
β_P	Coefficients for CO_2e emissions	β_{ke}	$\beta_Q = \beta_P + \beta_{ke}$
γ_P		γ_{ke}	$\gamma_Q = \gamma_P + \gamma_{ke}$
δ_P		$\beta_P q_i$	$\delta_Q = \delta_P + \beta_Q q_j$

[a] Assuming label P.
[b] Assuming vehicle k and a label representing the partial route along arc $e = (0, i)$.
[c] Assuming label Q obtained by extending P along arc $e = (i, j)$.

path Q when traveling at a speed of v is defined as

$$c_k\Big(\frac{\alpha_Q}{v} + \delta_Q + \beta_Q w_k + \gamma_Q v^2\Big) - \pi_Q, \tag{43}$$

where $\alpha_Q = \sum_{l=1}^{L} \alpha_{ke_l}$, $\beta_Q = \sum_{l=1}^{L} \beta_{ke_l}$, $\gamma_Q = \sum_{l=1}^{L} \gamma_{ke_l}$, $\pi_Q = \sum_{l=1}^{L} \pi_{e_l}$, $\delta_Q = \sum_{i=1}^{L-1} \beta_{P_i} q_i$, and P_i denote the path $(v_0, v_1, v_2, \dots, v_i)$.

Proposition 3. *Let P be a complete path that ends at the depot, and let p be the route induced by P. Then, the cost of p executed under the optimal vehicle speed, given by c_p, is equal to P as determined by (43) where $v = v_P$.*

Proof. Let Q be a partial path of vehicle k with arcs $(e_1, e_2, \dots, e_L)$ and nodes $(v_0, v_1, v_2, \dots, v_L)$ where $v_0 = v_L = 0$ is the depot with demand q_0 set to 0. The travel cost of the route induced by Q, according to the objective function (29), is given by

$$c_k \sum_{l=1}^{L}\Big(\frac{\alpha_{ke_l}}{v} + \beta_{ke_l}(w_k + x_{e_l}) + \gamma_{ke_l} v^2\Big) - \sum_{l=1}^{L} \pi_{v_l}, \tag{44}$$

where $x_{e_l} = \sum_{m=l}^{L-1} q_m$ is the sum of the demand of the customers in the remaining part of the route, that is the payload of arc e_l. As shown below, the cost calculation by (44) is equivalent to (43).

The travel cost (44) can be rewritten as

$$c_k\Big(\frac{\sum_{l=1}^{L}\alpha_{ke_l}}{v} + \sum_{l=1}^{L}\sum_{m=l}^{L}\beta_{ke_l} q_m + \sum_{l=1}^{L}\beta_{ke_l} w_k + \sum_{l=1}^{L}\gamma_{ke_l} v^2\Big) - \sum_{l=1}^{L}\pi_{v_l}$$
$$= c_k\Big(\frac{\alpha_Q}{v} + \sum_{l=1}^{L}\sum_{m=l}^{L}\beta_{ke_l} q_m + \beta_Q w_k + \gamma_Q v^2\Big) - \pi_Q.$$

Note that $\sum_{l=1}^{L}\sum_{m=l}^{L}\beta_{ke_l} q_m = \sum_{l=1}^{L}\sum_{m=1}^{l}\beta_{ke_m} q_l = \sum_{l=1}^{L}\beta_{P_l} q_l = \delta_Q$, and hence equivalent to the routing costs defined in the objective function (29). □

The pricing problem can be considered as a variant of the resource-constrained shortest-path (RCSP) problem (see, e.g. Feillet et al., 2004). Typically, one finds the RCSP with a labeling procedure where dominance rules are employed to discard non-promising partial paths. In our case, there are no existing effective dominance rules that are applicable. More specifically, in addition to the usual resources to handle vehicle capacity and time windows, in our case, one must also observe dominance conditions on the allowed vehicle speed range and each cost component individually. Therefore, in our pricing algorithm, we decide to control the combinatorial growth of labels exclusively with completion bounds, which are detailed next.

5.3.1. Completion bounds

A completion bound is a lower bound on the reduced cost of all routes generated from a label. Completion bounds accelerate the solution of the pricing problem since partial paths with nonnegative bounds are discarded during the labeling procedure.

Consider a partial path P with arcs $(e_1, e_2, \dots, e_L)$ and nodes $(v_0, v_1, v_2, \dots, v_L)$. The precise cost along P depends on the customers visited after v_L, since the cost along each arc depends on the arc payload. A lower bound on the cost along P, however, can be computed as follows:

$$\Phi_P = f_k + c_k \sum_{l=1}^{L}\Big(\frac{\alpha_{ke_l}}{v} + \beta_{ke_l}(w_k + \bar{x}_{e_l}) + \gamma_{ke_l} v^2\Big), \tag{45}$$

where

$$\bar{x}_{e_l} = \begin{cases} q_P - \sum_{m=1}^{l-1} q_{v_m}, & \text{if } \beta_{ke_l} \ge 0, \\ Q_k - \sum_{m=1}^{l-1} q_{v_m}, & \text{if } \beta_{ke_l} < 0. \end{cases} \tag{46}$$

Eq. (46) considers a best-case (i.e., cost-minimizing) scenario concerning the payload along each arc. If the corresponding β is nonnegative, the vehicle is assumed to travel along arc e as lightly as possible. Otherwise, the vehicle is assumed to travel as loaded as possible.

Given the lower bound (45), we propose two completion bounds for a label P. The first bound is based on an RCSP and explores the capacity resource to find a lower bound on the reduced cost of any extension of P. The second bound is based on a knapsack problem. It explores not only the capacity resource but also the "timing" resources, that is, customers cannot be visited after their time windows, and the vehicle must return to the depot no later than instant l_0.

We start with the RCSP-based completion bound, which adapts the bound proposed by Florio et al. (2021) for solving the elementary RCSP. First, we associate to each arc $e = (i,j) \in \mathcal{A}$ a lower bound $\bar{\phi}_e$ on the reduced cost change when partial path P is extended along $e = (i,j)$:

$$\bar{\phi}_e = \frac{\alpha_{ke}}{v_k^*} + \beta_{ke}(w_k + x'_e) + \gamma_{ke}(v_k^*)^2 - \pi_j, \tag{47}$$

where $x'_e = 0$ if $\beta_e \ge 0$ and $x'_e = Q_k$ otherwise.

We denote by $S_i^*(Q)$ the value of the RCSP from node $i \in \mathcal{N} \setminus \{0\}$ to node 0 in a graph with arc costs given by (47), in which the initial resource limit is Q and an amount q_j of resource is consumed each time node $j \ne 0$ is visited. Then, the RCSP-based completion bound is given by:

$$\Phi_P - \sum_{i \in N_P} \pi_i + S_{n_P}^*(Q_k - q_P). \tag{48}$$

Eq. (48) yields a valid bound because $\Phi_P - \sum_{i\in N_P} \pi_i$ is a lower bound on the reduced cost of partial path P, and $S^*_{n_P}(Q_k - q_P)$ is a lower bound on the reduced cost of any feasible extension to P. To enable evaluations of (48) in constant time for any label P, at the beginning of an iteration of the pricing problem we pre-compute $S^*_i(Q)$ for all $i \in \mathcal{N} \setminus \{0\}$ and $Q \in \{0, \dots, Q_k\}$. The (non-elementary) RCSPs can be solved efficiently by dynamic programming.

While the RCSP bound is computationally efficient, it does not explore time window constraints nor the fact that we price elementary routes. In the knapsack bound, we set up a $\{0,1\}$-knapsack problem with capacity of $l_0 - \tau_P$, which corresponds to the maximum remaining routing time after customer n_P is served. Then, we define a set of knapsack items

$$\mathcal{I} = \{i \in \mathcal{N} \setminus (\{0\} \cup N_P) : q_i + q_P \le Q_k \wedge \tau_P + d_{n_P,i}/b_k \le l_i\}.$$

Each element of $\mathcal{I}$ is a customer that can be visited after n_P, considering time window and vehicle capacity constraints. With each item $i \in \mathcal{I}$ a value $v(i)$ and a weight $w(i)$ are associated:

$$v(i) = \max_{e=(j,i)\in \mathcal{A}} -\bar{\phi}_e,$$
$$w(i) = \min_{(j,i)\in \mathcal{A}} d_{ji}/b_k.$$

The value and weight of an item i correspond to the maximum reduced cost decrease and minimum amount of the time resource consumed, respectively, when customer i is visited in an extension of partial path P. We let K^*_P be the optimal solution value of the abovementioned knapsack problem. Then, the knapsack completion bound is given by:

$$\Phi_P - \sum_{i\in N_P} \pi_i - K^*_P. \tag{49}$$

Each time a label P is generated, we evaluate (49) and discard P if the bound is nonnegative. This evaluation requires solving the knapsack problem to optimality, which dynamic programming can also achieve efficiently.

5.4. Branching rules

The implemented branch-and-bound framework finds an optimal integer solution to (25)–(28) by branching on variables y_e, $e \in \mathcal{A}$, that take fractional values. We apply a semi-strong branching rule where each potential branching variable is evaluated under the current pool of columns, and the variable on which branching leads to the highest lower bound is chosen. More precisely, at a given branch-and-bound node, we let Ω_{R} be the set of all columns generated and $\mathcal{Y}$ the set of arc variables that assume fractional values in the solution to the RMP. Then, we evaluate

$$\min\{\mathrm{RMP}(\Omega_{\mathrm{R}}, \{y_e\}, \{\}), \mathrm{RMP}(\Omega_{\mathrm{R}}, \{\}, \{y_e\})\} \tag{50}$$

for each $y_e \in \mathcal{Y}$, where $\mathrm{RMP}(\Omega, \mathcal{Y}_0, \mathcal{Y}_1)$ corresponds to the optimal solution value of the RMP restricted to columns Ω and enforcing constraints $y_e = 0$ for all $y_e \in \mathcal{Y}_0$ and $y_e = 1$ for all $y_e \in \mathcal{Y}_1$ in addition to the branching constraints of the parent branch-and-bound node. Finally, we branch on the variable $y_e \in \mathcal{Y}$ such that (50) is maximum. Note that evaluating (50) for each potential branching variable can be implemented efficiently by loading a single linear program and (re)solving it for each variable after adjusting the cost vector accordingly by penalizing the cost of routes that do not comply with the candidate branching decision.

6. Computational experiments

In this section, we will evaluate the performance of Tabu search heuristic described in Section 4 (denoted as **TS**) and the branch-and-price approach described in Section 5 (denoted as **BP**). Afterwards, we use the heuristic in an empirical study to evaluate the potential benefits of using elevation data for optimizing vehicle routes.

We organize the remaining subsections as follows. Section 6.1 describes the test instances constructed using data from the literature, and Section 6.2 describes the test instances constructed using real-world data. Section 6.3 is about the parameter tuning experiment. In Section 6.4, we evaluate the efficiency of BP and TS. In Section 6.5, we evaluate the potential benefits of using elevation data for planning vehicle routes.

Table 5
Vehicle information.

Original capacity	Vehicle type	Capacity	Demand unit
200	LDV	1,200 kg	6 kg per unit
700	MDV	12,600 kg	18 kg per unit
1,000	HDV	31,000 kg	31 kg per unit

6.1. Test instances using data from literature

Since there are no existing benchmark datasets of PRP-SO, we constructed test instances based on the instances of Solomon (1987), which are initially created for VRPTW. The VRPTW instances are adapted into PRP-SO instances by associating randomly generated elevation information on the nodes. The units for time and demand are also scaled to match the realistic instances.

The VRPTW instances have four sets of instances involving 25, 50, 75, and 100 customers, respectively. Instances are divided into three classes according to the customer location distribution: clustered distribution (C class), scattered distribution (R class), and partially scattered and partially clustered distribution (RC class). Each class is further subdivided into the narrower time window class and the wider time window class. Our experiments will test on the instances with narrower time windows. In total, 116 instances of PRP-SO are constructed using the instances of Solomon (1987). The remaining part of this subsection describes how the VRPTW instances are adapted into the PRP-SO instances.

The elevation of the nodes is randomly generated with a uniform distribution between 0 and 1000 m. The distance in kilometers between node i and node j is given by

$$D_{ij} = \sqrt{(X_i - X_j)^2 + (Y_i - Y_j)^2 + \left(\frac{Z_i - Z_j}{1000}\right)^2},$$

rounded to the nearest meter, where (X_i, Y_i) and (X_j, Y_j) are the coordinates, and Z_i and Z_j are the elevations of node i and j, respectively. With the rounded values of distances, the road angle between two nodes is given by $\tan^{-1}\left(\frac{Z_i - Z_j}{D_{ij}}\right)$.

The units for time and demand are also scaled to match the realistic instances. For our experiments, service times and the time windows have a unit of 0.02 h. For example, a due date of 1236 from Solomon's instances represents a due date of 24.72 h (given by 1236*0.02) after the planning horizon starts. This implies that if vehicles always travel at 50 km per hour, the time window constraints would remain the same as the ones in the original VRPTW instances.

Three types of vehicles appeared in Solomon's instances: 200, 700, and 1000 units. We scale the demand and the vehicle capacity accordingly so that it is equivalent to the original constraints for vehicle capacity and, at the same time, matches typical truck classifications: LDV, MDV, and HDV for the 200, 700, and 1000 capacity units, respectively.

Table 5 summarizes our experiments' vehicle capacity and demand unit. For example, the vehicles in Solomon's instances with a capacity of 200 units correspond to the LDV vehicles of the PRP-SO instances. Therefore, a demand of 20 units in those instances represents a demand of 120 kg (given by 20 × 6) in the PRP-SO instances.

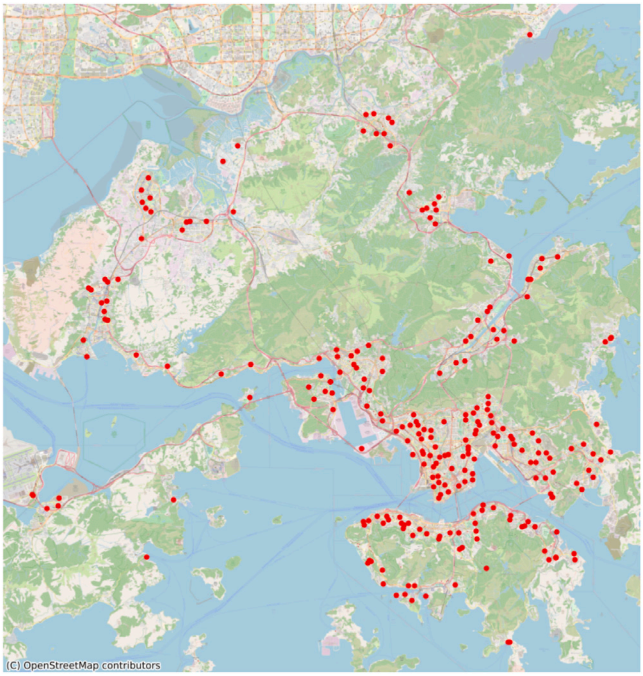

Fig. 1. Customer locations.

All the vehicles have a fixed cost of 100 EUR, a fuel cost of 1.42 EUR per liter, and a maximum speed of 80 km per hour. Other parameters used for the CO_2e consumption calculations are summarized in Table 2.

6.2. Test instances using real-world data

Another dataset is constructed based on the distribution network of a large international health and beauty retailer. There are 248 customers considered in our experiments, which represent the retailer's stores in Hong Kong. Geometric information is obtained using Google Maps APIs, including the coordinates, elevations, and suggested paths between the stores. Fig. 1 illustrates the geographical locations of these stores. The landscape varies from moderately hilly to mountainous, with steep slopes. The stores' locations are clustered and densely populated in the central areas. We use this dataset to evaluate the potential benefits of using elevation information in optimizing vehicle routes.

We preprocess the geographic data from the Google Maps API and the Elevation API into the α, β, and γ values associated with the arcs (defined in Section 3) so that the proposed solution approaches can compute the fuel consumption and CO_2e emission efficiently. To begin with, for every distinct pair of the stores, we obtain a suggested path by using the Google Maps API and find out the elevation for all the coordinates along the suggested path by using Google Elevation API. Afterward, to construct arc segments coordinates along a path are divided into segments. With each segment, the distance is no longer than 1000 m. An arc segment can be viewed as a slope along the suggested path. In our experiments, there are a total of 155,333 such slopes. Lastly, the angles and distance of all these slopes are determined using the coordinates and elevation data, and thus we have the α, β, and γ values of all the arcs connecting the stores.

In our experiments, nine instances are constructed. Each instance consists of 100 customers randomly selected from the 248 stores. The ready time, due time, demand, vehicle number, and capacity are data from Solomon's C class instances. The depot is located at the Kwai Tsing Container Terminal, which is the busiest port in Hong Kong.

6.3. Parameter tuning

The best parameters amongst the values specified in Table 6 are chosen for each instance class.

All experiments have been conducted on an Ubuntu server with an Intel Xeon CPU E5-2698 v3 @ 2.30 GHz, 16 cores, and 16 GB of main memory. Linear programs were solved using IBM® CPLEX® version 12.10. Algorithms have been implemented in C++ and compiled using GNU g++ version 10.2.0 with the -O2 flag. The algorithms run on a single core per instance.

6.4. Efficiency of the approaches

Table 7 summarizes the computational results of BP and TS on the instances described in Section 6.1 with 25, 50, 75, and 100 customers, respectively. Appendix A shows individual instances' results. Column **NI** is the number of instances in the dataset class, including the ones that no feasible solution can be found using BP within 3 h. Column **NO** is the number of instances that can be solved optimally by using BP within the time limit. Column **NV** is the average number of vehicle routes, column **TD** is the average total travel cost, column **AT** is the average CPU time (in seconds), and column **AG** is the average optimality gap (in percentage). When reporting the average values, we excluded instances where no feasible solution could be found by using BP within 3 h. If BP is terminated due to the time limit, the best feasible solution is used for computing the average values.

As shown in Table 7, BP can solve instances up to 100 customers, with 61 instances (53%) solved to optimality. BP can find all the optimal solutions of the dataset with 25 customers within a reasonable time and the most optimal solutions with 50 customers within 3 h. Compared to TS, BP can determine better solutions on smaller instances but at the expense of significantly more CPU time. TS can find near-optimal solutions for all instances within one minute and outperforms BP (with a time limit of 3 h) on solution quality for the dataset with 100 customers.

6.5. The value of using elevation information

The real-world instances described in Section 6.2 are solved by using TS to evaluate the potential benefits of using elevation information in optimizing vehicle routes. Table 8 summarizes the total distance (in km), average speed (in km/h), total fixed cost (in EUR), total fuel cost (in EUR), and the total elevation (in meters). Fig. 2 illustrates an example vehicle route. The same dataset is solved again with elevation information ignored (disregarding slopes). i.e. Setting $\phi_{es} = 0$ when we precompute the parameters using (2)–(4). The results are reported in Table 9. This is achieved in our experiment by setting the angles of all slopes to zero when optimizing the vehicle routes by using TS and evaluating the solutions with the correct elevations and slopes after the vehicle routes have been decided.

We can observe the impacts on solutions when elevation information is used for planning the vehicle routes by comparing the results in Tables 8 and 9. As shown in the experimental results, when elevation information is considered in planning, fuel consumption decreased by 54% on average. In contrast, the average vehicle speed and elevation increased slightly, and the total travel distance increased by 31%.

More considerable travel distances and lower fuel consumption are contradictory for typical vehicle routing problems. In the PRP-SO, fuel consumption depends on distance, slopes, payload, and vehicle speeds. Optimal vehicle routes should save fuel costs by avoiding going uphill at a high speed with a large payload. As a result, vehicles tend to visit more customers first before going uphill. Although this will increase travel distance, fuel consumption can be saved by going uphill with a smaller payload.

For typical vehicle routing problems or when vehicle routes are planned manually, travel distance is usually minimized in the objective

Table 6
Parameter values.

Parameter	Possible values	Description
I_1	1, 2	Number of iterations in the first phase
I_2	100,000	Number of iterations in the second phase
λ	10^{-6}, 5×10^{-6}, 10^{-7}, 5×10^{-7}	Diversification intensity
h	3, 4, 5, 6	Parameter for setting the tabu tenure
δ	10, 20, 30, 40	Penalty update frequency

Table 7
Computational results of the PRP-SO instances.

Class		NO/NI	BP				TS		
			NV	TD	AT (s)	AG (%)	NV	TD	AT (s)
N25	C	9/9	3.000	24.059	215.90	0%	3.000	24.597	0.41
	RC	8/8	3.250	45.314	30.28	0%	3.250	45.331	0.98
	R	12/12	4.667	60.240	30.02	0%	4.750	59.306	2.49
N50	C	7/7	5.000	45.641	3608.80	0.000%	5.000	46.612	2.65
	RC	4/8	6.500	94.850	6933.27	0.087%	6.500	97.514	10.26
	R	9/10	7.500	107.889	2707.29	0.018%	8.000	109.131	8.17
N75	C	2/4	8.000	83.020	8314.54	0.079%	8.000	82.994	3.85
	RC	1/6	9.667	154.001	9262.17	4.088%	10.333	153.598	10.03
	R	5/6	11.833	152.447	3956.78	0.018%	12.500	160.035	8.23
N100	C	1/1	10.000	107.742	3255.73	0.000%	10.000	107.742	2.21
	RC	1/4	12.750	210.165	8957.57	2.440%	13.000	195.959	12.93
	R	2/3	16.667	194.144	4489.88	1.983%	13.500	170.702	42.31
Total:		61/78	98.8	1279.5	51 762.2		97.8	1253.5	104.5

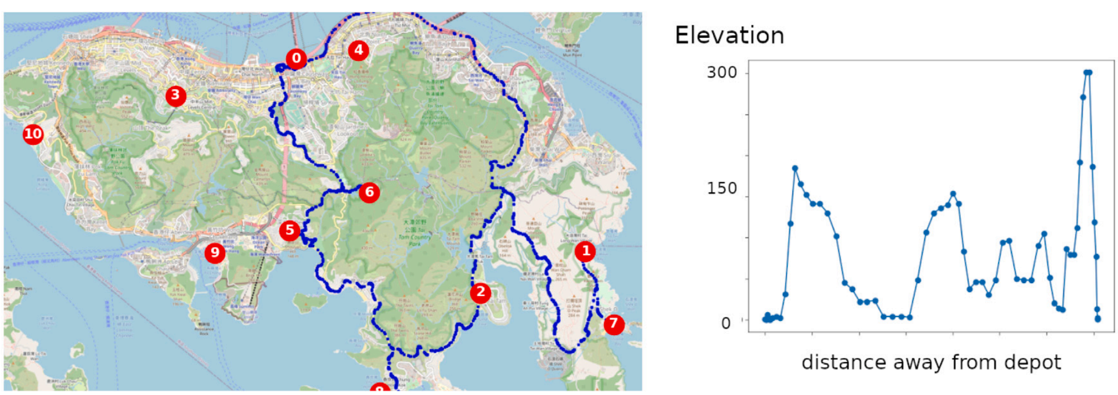

Fig. 2. An example vehicle route.

Table 8
Results on real-world instances.

Instance	Distance	Speed	Fixed cost	Fuel cost	Elevation
HK01	1638.6	62.3	1500	109.5	1816
HK02	1460.1	62.3	1400	163.6	1436
HK03	990.2	61.1	1200	196.7	1070
HK04	1153.7	63.1	1000	50.0	1440
HK05	1209.0	63.5	1400	197.5	1185
HK06	1441.9	64.9	1300	89.4	1748
HK07	1159.5	65.5	1300	185.4	1309
HK08	1384.8	64.6	1200	110.3	1359
HK09	1088.5	63.7	1100	81.5	1495
Average:	1280.7	63.5	1266.7	131.5	1428.7

Table 9
Results on real-world instances when slopes are ignored.

Instance	Distance	Speed	Fixed cost	Fuel cost	Elevation
HK01	1158.7	60.6	1500	405.1	1582
HK02	1015.1	60.7	1400	331.1	1152
HK03	953.0	60.9	1200	184.5	1043
HK04	765.2	60.5	1100	114.5	1443
HK05	1040.7	62.9	1400	409.8	1140
HK06	1071.8	62.8	1300	518.3	1536
HK07	1004.4	62.3	1300	278.1	1311
HK08	945.3	64.0	1300	197.7	1307
HK09	831.7	59.5	1100	138.0	1331
Average:	976.2	61.6	1288.9	286.3	1316.1

function. Our experimental result reveals that this can ultimately lead to a suboptimal solution. With the elevation information, the optimal solution can balance the tradeoff between the energy consumption due to longer distances and the higher payload when going uphill. To avoid high fuel consumption when vehicles go uphill, heavier items tend to be delivered first before going uphill. Customer time windows must be considered so that vehicles do not need to speed up to meet the due times, which can result in high fuel consumption. If elevation information is ignored when planning the vehicle routes, fuel consumption due to payload is underestimated when vehicles go uphill, leading to poor solutions. With the significant savings we observed from the experimental results, logistic service providers should consider using elevation data for planning their vehicle routes in practice.

6.6. Impact of payloads and slopes

For evaluating the impact of payloads and slopes on the optimized solutions, the real-world instances described in Section 6.2 are modified

Table 10
Impact of payloads and slopes on routing costs.

		Slope factor (r_2)						
		0	0.2	0.4	0.6	0.8	1	Average
Payload factor (r_1)	0	1516.0	1543.9	1527.4	1458.7	1456.2	1361.8	1477.3
	0.2	1514.3	1519.5	1527.3	1472.0	1475.4	1374.8	1480.6
	0.4	1530.7	1563.0	1540.4	1466.4	1478.4	1363.9	1490.5
	0.6	1545.9	1565.5	1556.6	1475.9	1512.0	1373.8	1505.0
	0.8	1553.0	1532.5	1582.0	1511.1	1494.6	1404.7	1513.0
	1	1575.1	1542.8	1586.7	1516.6	1496.1	1413.3	1521.8
	Average	1539.2	1544.6	1553.4	1483.5	1485.5	1382.0	

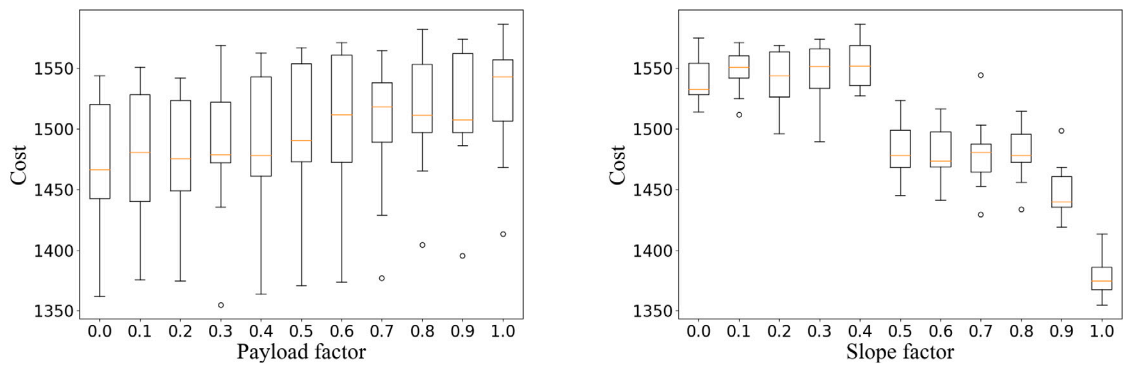

Fig. 3. Impact of payloads and slopes on routing costs.

by scaling the payloads by a factor $r_1 \in \{0, 0.1, \dots, 1\}$ when calculating the costs, and scaling the slopes by a factor $r_2 \in \{0, 0.1, \dots, 1\}$. In our experiments, we replace the payload x_{ke} in (1) by a factor $r_1 x_{ke}$ when calculating the costs and replace the slope ϕ_{es} in (3) by $r_2\phi_{es}$ when preprocessing the data. A higher value of the payload factor (r_1) represents scenarios when relatively heavier items are shipped, and a higher value of the slope factor (r_2) represents more hilly areas where steep slopes commonly appear. There are 1089 instances tested in total, and each is solved using TS with a time limit of 300 s.

Table 10 summarizes the average costs (see Appendix B for the complete results). Fig. 3 shows the average cost with varying payloads (r_1) and slopes (r_2), respectively.

The experimental results show that the shipping cost increases with the payload (r_1) and decreases with the slope (r_2). It is more costly to ship with a heavier payload (higher value of r_1) with a percentage increase in total costs up to 3.49%, on average, regardless of the slopes. It is less costly to ship in more hilly areas (higher value of r_2), saving up to 11.71% of the total costs on average, regardless of the payloads. The impact of slopes is significantly higher than that due to payloads, which reveals the importance of considering slopes when optimizing vehicle routes.

7. Conclusions

This paper formulates and proposes efficient solution methods for a joint Pollution-Routing and Speed Optimization Problem (PRP-SO), where the total travel cost is a function of fuel consumption and CO_2e emissions and depends, simultaneously, on road grades, arc payloads, and vehicle speed. The introduction of vehicle speed as a continuous decision variable results in more complicated optimization subproblems in the presence of time window constraints. For the fixed-sequence speed optimization problem where the vehicle route is known, the proposed approach is conceptually simple and computes the optimal vehicle speed (with and without time windows) in quadratic time. In the speed optimization with variable routes, we introduced a novel labeling algorithm, without full backtracking searching, that efficiently determines the cost of a vehicle route that travels with optimal speed.

Based on the proposed speed optimization algorithms, we present two general solution approaches for solving the PRP-SO. An approximate solution strategy aims to solve large instances in a short computational time. For this purpose, we integrated the fixed-sequence speed optimization algorithm into Tabu search metaheuristic. The second approach consists of an exact branch-and-price algorithm, in which the variable-route speed optimization is managed within the pricing problem.

We carried out extensive computational experiments on modified Solomon benchmarks and newly constructed real-life instances. Numerical results show that the exact solution methodology performs very well in terms of solution quality: 61 out of 116 instances are solved to optimality. Contrary to the computational outcomes presented by Dabia et al. (2017), in which several instances with only 25 customers cannot be solved, we are able to solve all small-scale problem instances (25 customers) within a reasonable time. Our BP algorithm solved most of the problem instances with 50 customers and reached optimal solutions for some larger benchmark instances with up to 100 customers. We show that our metaheuristic works very effectively for all instances solved. The heuristic consumes less than one minute to find near-optimal solutions in all instances and improves best-known solutions where the exact algorithm did not reach optimality.

Our computational results on real-world instances provide sufficient evidence to suggest some essential managerial insights. First, significant savings (53%) in fuel consumption and CO_2e emissions are observed, especially when shipping heavy items in hilly areas. Second, vehicle routes included a larger number of customer visits (located on flatter terrain) before going to uphill destinations, also significantly reducing fuel consumption. Third, if elevation information is ignored when planning vehicle routes, fuel consumption estimation is inaccurate.

Finally, it is important to highlight that further research is needed. Specifically, addressing the challenges posed by route security in networks situated in hilly topography, considering the constraints imposed on heavy vehicles navigating steep uphill and downhill paths, proves to be a challenging task within the domain of vehicle routing. Subsequent research could focus on extending the proposed methodological approaches to a new context, where optimization of two key performance metrics – fuel consumption and vehicle route security – takes precedence.

CRediT authorship contribution statement

David Lai: Conceptualization, Methodology, Software, Writing – original draft, Writing – review & editing. **Yasel Costa:** Conceptualization, Writing – original draft, Writing – review & editing. **Emrah Demir:** Conceptualization, Writing – original draft, Writing – review & editing. **Alexandre M. Florio:** Conceptualization, Software, Writing – original draft, Writing – review & editing. **Tom Van Woensel:** Conceptualization, Writing – original draft, Writing – review & editing.

Declaration of competing interest

The authors have no competing interests to declare relevant to this article's content.

Data availability

Data will be made available on request.

Appendix A & B. Supplementary data

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.cor.2024.106557.

References

Abdullahi, H., Reyes-Rubiano, L., Ouelhadj, D., Faulin, J., Juan, A.A., 2021. Modelling and multi-criteria analysis of the sustainability dimensions for the green vehicle routing problem. European J. Oper. Res. 292 (1), 143–154.

Bandeira, J., Almeida, T.G., Khattak, A.J., Rouphail, N.M., Coelho, M.C., 2013. Generating emissions information for route selection: Experimental monitoring and routes characterization. J. Intell. Transp. Syst. 17 (1), 3–17.

Barth, M., Boriboonsomsin, K., 2009. Energy and emissions impacts of a freeway-based dynamic eco-driving system. Transp. Res. D 14 (6), 400–410.

Behnke, M., Kirschstein, T., Bierwirth, C., 2021. A column generation approach for an emission-oriented vehicle routing problem on a multigraph. European J. Oper. Res. 288 (3), 794–809.

Bektaş, T., Laporte, G., 2011. The pollution-routing problem. Transp. Res. B 45 (8), 1232–1250.

Boriboonsomsin, K., Barth, M., 2009. Impacts of road grade on fuel consumption and carbon dioxide emissions evidenced by use of advanced navigation systems. Transp. Res. Rec. 2139 (1), 21–30.

Brundell-Freij, K., Ericsson, E., 2005. Influence of street characteristics, driver category and car performance on urban driving patterns. Transp. Res. D 10 (3), 213–229.

Brunner, C., Giesen, R., Klapp, M.A., Flórez-Calderón, L., 2021. Vehicle routing problem with steep roads. Transp. Res. A 151, 1–17.

Costa, L., Contardo, C., Desaulniers, G., 2019. Exact branch-price-and-cut algorithms for vehicle routing. Transp. Sci. 53 (4), 946–985.

Dabia, S., Demir, E., Van Woensel, T., 2017. An exact approach for a variant of the pollution-routing problem. Transp. Sci. 51 (2), 607–628.

Davis, S.C., Diegel, S.W., Boundy, R.G., et al., 2009. Transportation Energy Data Book. Technical Report, Oak Ridge National Laboratory.

Demir, E., Bektaş, T., Laporte, G., 2012. An adaptive large neighborhood search heuristic for the pollution-routing problem. European J. Oper. Res. 223 (2), 346–359.

Demir, E., Bektaş, T., Laporte, G., 2014a. The bi-objective pollution-routing problem. European J. Oper. Res. 232 (3), 464–478.

Demir, E., Bektaş, T., Laporte, G., 2014b. A review of recent research on green road freight transportation. European J. Oper. Res. 237 (3), 775–793.

Demir, E., Syntetos, A., van Woensel, T., 2022. Last mile logistics: Research trends and needs. IMA J. Manag. Math. 33 (4), 549–561.

Erdoğan, S., Miller-Hooks, E., 2012. A green vehicle routing problem. Transp. Res. E 48 (1), 100–114.

Erdoğdu, K., Karabulut, K., 2022. Bi-objective green vehicle routing problem. Int. Trans. Oper. Res. 29 (3), 1602–1626.

Ericsson, E., 2001. Independent driving pattern factors and their influence on fuel-use and exhaust emission factors. Transp. Res. D 6 (5), 325–345.

Eshtehadi, R., Fathian, M., Demir, E., 2017. Robust solutions to the pollution-routing problem with demand and travel time uncertainty. Transp. Res. D 51, 351–363.

Feillet, D., Dejax, P., Gendreau, M., Gueguen, C., 2004. An exact algorithm for the elementary shortest path problem with resource constraints: Application to some vehicle routing problems. Networks 44 (3), 216–229.

Florio, A.M., Absi, N., Feillet, D., 2021. Routing electric vehicles on congested street networks. Transp. Sci. 55 (1), 238–256.

Franceschetti, A., Honhon, D., Van Woensel, T., Bektaş, T., Laporte, G., 2013. The time-dependent pollution-routing problem. Transp. Res. B 56, 265–293.

Fukasawa, R., He, Q., Santos, F., Song, Y., 2018. A joint vehicle routing and speed optimization problem. INFORMS J. Comput. 30 (4), 694–709.

Fukasawa, R., He, Q., Song, Y., 2016. A disjunctive convex programming approach to the pollution-routing problem. Transp. Res. B 94, 61–79.

Gendreau, M., Hertz, A., Laporte, G., 1994. A tabu search heuristic for the vehicle routing problem. Manage. Sci. 40 (10), 1276–1290.

Gendreau, M., Potvin, J.-Y., 2019. Tabu search. In: Handbook of Metaheuristics. Springer, pp. 37–56.

Giraldo, M., Huertas, J.I., 2019. Real emissions, driving patterns and fuel consumption of in-use diesel buses operating at high altitude. Transp. Res. D 77, 21–36.

Glover, F., 1986. Future paths for integer programming and links to artificial intelligence. Comput. Oper. Res. 13 (5), 533–549.

Goeke, D., Schneider, M., 2015. Routing a mixed fleet of electric and conventional vehicles. European J. Oper. Res. 245 (1), 81–99.

IEA, 2020. Tracking transport 2020. The international energy agency. https://www.iea.org/reports/tracking-transport-2020.

Kirby, H.R., Hutton, B., McQuaid, R.W., Raeside, R., Zhang, X., 2000. Modelling the effects of transport policy levers on fuel efficiency and national fuel consumption. Transp. Res. D 5 (4), 265–282.

Koç, Ç., Bektaş, T., Jabali, O., Laporte, G., 2014. The fleet size and mix pollution-routing problem. Transp. Res. B 70, 239–254.

Koç, Ç., Bektaş, T., Jabali, O., Laporte, G., 2016a. A comparison of three idling options in long-haul truck scheduling. Transp. Res. B 93, 631–647.

Koç, Ç., Bektaş, T., Jabali, O., Laporte, G., 2016b. Thirty years of heterogeneous vehicle routing. European J. Oper. Res. 249 (1), 1–21.

Kramer, R., Maculan, N., Subramanian, A., Vidal, T., 2015a. A speed and departure time optimization algorithm for the pollution-routing problem. European J. Oper. Res. 247 (3), 782–787.

Kramer, R., Subramanian, A., Vidal, T., Lucídio dos Anjos, F.C., 2015b. A matheuristic approach for the pollution-routing problem. European J. Oper. Res. 243 (2), 523–539.

Lai, D.S., Demirag, O.C., Leung, J.M., 2016. A tabu search heuristic for the heterogeneous vehicle routing problem on a multigraph. Transp. Res. E 86, 32–52.

Liu, K., Yamamoto, T., Morikawa, T., 2017. Impact of road gradient on energy consumption of electric vehicles. Transp. Res. D 54, 74–81.

Macrina, G., Laporte, G., Guerriero, F., Pugliese, L.D.P., 2019. An energy-efficient green-vehicle routing problem with mixed vehicle fleet, partial battery recharging and time windows. European J. Oper. Res. 276 (3), 971–982.

Marrekchi, E., Besbes, W., Dhouib, D., Demir, E., 2021. A review of recent advances in the operations research literature on the green routing problem and its variants. Ann. Oper. Res. 304, 529–574.

Moghdani, R., Salimifard, K., Demir, E., Benyettou, A., 2020. The green vehicle routing problem: A systematic literature review. J. Clean. Prod. 123691.

Norstad, I., Fagerholt, K., Laporte, G., 2011. Tramp ship routing and scheduling with speed optimization. Transp. Res. C 19 (5), 853–865.

Ntziachristos, L., Samaras, Z., Eggleston, S., Gorissen, N., Hassel, D., Hickman, A., et al., 2000. Copert iii. Computer Programme to calculate emissions from road transport, methodology and emission factors (version 2.1), European Energy Agency (EEA), Copenhagen.

Raeesi, R., Zografos, K.G., 2019. The multi-objective steiner pollution-routing problem on congested urban road networks. Transp. Res. B 122, 457–485.

Rahman, S.A., Masjuki, H., Kalam, M., Abedin, M., Sanjid, A., Sajjad, H., 2013. Impact of idling on fuel consumption and exhaust emissions and available idle-reduction technologies for diesel vehicles–A review. Energy Convers. Manage. 74, 171–182.

Rauniyar, A., Nath, R., Muhuri, P.K., 2019. Multi-factorial evolutionary algorithm based novel solution approach for multi-objective pollution-routing problem. Comput. Ind. Eng. 130, 757–771.

Savelsbergh, M., Van Woensel, T., 2016. 50Th anniversary invited article—city logistics: Challenges and opportunities. Transp. Sci. 50 (2), 579–590.

Solomon, M.M., 1987. Algorithms for the vehicle routing and scheduling problems with time window constraints. Oper. Res. 35 (2), 254–265.

Soriano, P., Gendreau, M., 1996. Diversification strategies in tabu search algorithms for the maximum clique problem. Ann. Oper. Res. 63 (2), 189–207.

Suzuki, Y., 2011. A new truck-routing approach for reducing fuel consumption and pollutants emission. Transp. Res. D 16 (1), 73–77.

Tavares, G., Zsigraiova, Z., Semiao, V., Carvalho, M.d.G., 2009. Optimisation of MSW collection routes for minimum fuel consumption using 3D GIS modelling. Waste Manag. 29 (3), 1176–1185.

Tirkolaee, E.B., Goli, A., Faridnia, A., Soltani, M., Weber, G.-W., 2020. Multi-objective optimization for the reliable pollution-routing problem with cross-dock selection using Pareto-based algorithms. J. Clean. Prod. 276, 122927.
Toth, P., Vigo, D., 2003. The granular tabu search and its application to the vehicle-routing problem. INFORMS J. Comput. 15 (4), 333–346.
Travesset-Baro, O., Rosas-Casals, M., Jover, E., 2015. Transport energy consumption in mountainous roads. A comparative case study for internal combustion engines and electric vehicles in andorra. Transp. Res. D 34, 16–26.
Van Woensel, T., Creten, R., Vandaele, N., 2001. Managing the environmental externalities of traffic logistics: The issue of emissions. Prod. Oper. Manage. 10 (2), 207–223.
Xiao, Y., Zuo, X., Huang, J., Konak, A., Xu, Y., 2020. The continuous pollution routing problem. Appl. Math. Comput. 125072.
Yang, S., Li, M., Lin, Y., Tang, T., 2014. Electric vehicle's electricity consumption on a road with different slope. Physica A 402, 41–48.