The bike routing problem with energy constraints

Yannis Ancele[a], Quang Anh Pham[b], Minh Hoàng Hà[b] , Dante Ben Matellini[a] and Trung Thanh Nguyen[a]

[a]Liverpool Logistics, Offshore and Marine (LOOM) Research Institute, Liverpool John Moores University, Liverpool, UK;
[b]ORLab, Faculty of Computer Science, Phenikaa University, Hanoi, Vietnam

ARTICLE HISTORY
Compiled December 1, 2023

ABSTRACT
As climate change becomes more crucial, transporting products in urban areas by bicycle gains popularity. More companies start using bicycles as an alternative transportation mode and face challenges to efficiently satisfy the needs of their customers and employees. While designing the bike routes for pick up and delivery, it is required to take into account the energy needed by cyclists to move. The energy consumed in a bike route has to be kept under a certain threshold for cyclists to be able to pedal during the whole work shift. This leads to a new variant of the vehicle routing problem called the bike routing problem which aims at tackling constraints arising for bicycle deliveries. We propose a novel Mixed Integer Linear Programming model to determine the bike routes for delivering goods in urban areas. An Evolutionary Local Search algorithm is developed to efficiently solve the problem using new split and local search procedures. Experimental results obtained on random and real instances show the accuracy and stability of the proposed algorithms, as well as the relevance of the new problem.

KEYWORDS
Vehicle Routing Problem, Bike Routing Problem, Energy Constraint, Mixed Integer Linear Programming, Split Procedure, Evolutionary Local Search

1. Introduction

Around 80 percent of European citizens live in urban areas (Allen, Browne, & Holguin-Veras, 2010). Because of the high density of the population in those areas, large quantities of goods have to be transported for commercial and domestic purposes. However, transportation has several impacts on the environment, such as resource consumption, noise, and greenhouse gas emissions (Delfani, Kazemi, SeyedHosseini, & Niaki, 2021; Goodarzian, Shishebori, Bahrami, Abraham, & Appolloni, 2023; Rahman, Rahman, & Tseng, 2022). Among these, gas emissions, in particular, CO_2 emissions, are of great concern as they have consequences on human health and contribute towards climate change. In several countries, transportation is one of the biggest causes of CO_2 emissions. Thus, decision-makers are trying to reduce pollution in their countries or cities (Koning & Conway, 2016). Bike use is promoted and encouraged in countries like the Netherlands, Denmark, and Germany. While bicycles have even become the standard over vehicles in cities like Amsterdam, some other cities (e.g., London) voted

CONTACT Minh Hoàng Hà. Email: hoang.haminh@phenikaa-uni.edu.vn

laws to get entry fees for vehicles. As a result, Supply Chain Management businesses have become more concerned about environmental aspects and tried to tackle related issues. Those growing concerns force to revise planning approaches for road transportation by for example changing last-mile deliveries. One adopted solution is to focus on environment-friendly means such as bikes. Cycling is said to be one of the most energy-efficient and healthy transport modes (Ehrgott, Wang, Raith, & Van Houtte, 2012). Moreover, cargo-based push-bikes as shown in Figure 1 can transport much heavier loads than traditional ones. However, carrying a heavy load on a long distance could make the bikers so tired that they cannot complete the service. Therefore, we need to take into account the human health condition when designing feasible bike routes in practice.

Figure 1.: A cargo-based push-bike of our industry partner

Our purpose in this paper is to introduce a new variant of the Vehicle Routing Problem (VRP), called the Bike Routing Problem (BRP). This problem is quite similar to the typical VRP, except that an additional constraint related to the effort of bikers is taken into account. More precisely, the energy required to move push-bikes with cargo is calculated and checked so that it does not exceed a given power of the cyclist. In the following, we call this requirement as the energy constraint. This new constraint can make the BRP solution quite different from the typical VRP solution. In the VRP solution, a route and its reverse can be considered to be similar. But this is not true for the BRP solution. In fact, the more load accumulated in the beginning of the route, the more tired the biker.

To the best of our knowledge, as shown in the following section, there is no work that considers VRP-like problems using push-bikes with the constraint on the physical limit of bikers. By studying the BRP, our research fills this gap of the literature. The contributions of our paper are as follows. First, we introduce a new variant of the VRP in which bikes are used as the transportation mode. A new constraint that takes into account the health condition of bikers is introduced. This constraint relates to the cumulative load of the bike route and is originally non-linear. Second, we propose a linearization technique for the energy constraint to propose a mixed integer linear programming model so that we can use powerful MILP solvers to solve the problem. Third, an efficient metaheuristic based on Evolutionary Local Search (Prins, 2009) with problem-tailored components is designed. More precisely, our metaheuristic includes a new dynamic programming-based split procedure that optimally splits any given order of deliveries into a complete BRP solution. In addition, we propose additional data structures that allow checking the feasibility of local search moves in $\mathcal{O}(1)$. Finally, experiments carried out on random and real instances show the performance of our methods and the relevance of the new problem.

The remainder of this paper is organised as follows. Section 2 gives a literature review of different aspects of the problem while Section 3 presents the problem with a mathematical model. An evolutionary algorithm is presented in Section 4. Section 5 contains computational results and analyses of the algorithm. Finally, Section 6 concludes the paper by explaining the contribution and presenting future research directions.

2. Literature review

To tackle the pollution problem, researchers are trying to solve the green vehicle routing problem and its variants (Bektaş & Laporte, 2011; Erdoğan & Miller-Hooks, 2012; Kara, Kara, & Yetis, 2007; Lin, Choy, Ho, Chung, & Lam, 2014; Wang, Peng, Zhou, Mahmoudi, & Zhen, 2020). Green logistics is a concept aiming at delivering efficient supply chain management while considering environment-related factors. The traditional objective of distribution management was upgraded to minimising system-wide costs related to economic and environmental issues. In urban areas, trucks, on top of the pollution they cause, have difficulty of achieving door-to-door deliveries. In densely populated urban areas, bike-couriers or bike-messengers offer delivery advantages as they are considered to be environmentally friendly. Although bikes are seen as a good start towards green transportation (Tipagornwong & Figliozzi, 2014), very few papers in the literature deal with the delivery of goods using push-bikes with cargo. Instead, when it comes to optimisation models for bikes, researchers mainly focus on e-bikes (Caggiani, Colovic, Prencipe, & Ottomanelli, 2021; Elbert & Friedrich, 2020), the itinerary for cycling or bike-sharing re-balancing problems (Chemla, Meunier, & Wolfler Calvo, 2013; Li & Liu, 2021). Bike-sharing systems offer a mobility service whereby public bicycles, located at different stations across an urban area, are available for shared use. These systems contribute towards obtaining more sustainable mobility and decreasing traffic and pollution caused by car transportation (Dell'Amico, Hadjicostantinou, Iori, & Novellani, 2014).

In the literature of green vehicle routing problems, Kara et al. (2007) introduce the so-called Energy-Minimising Vehicle Routing Problem. This problem is an extension of the VRP where a weighted load function (load multiplied by distance), rather than just the distance, is minimised. Bektaş and Laporte (2011) introduce the Pollution-Routing Problem (PRP). The PRP is an extension of the classical VRP with a broader and more comprehensive objective function. On top of the travel distance, a few other constraints such as the amount of greenhouse emissions, fuel, travel times and their costs are considered. The main difference with our paper is that we consider bikes with no emission. Therefore, the calculation of the CO_2 is not necessary. Also, Bektaş and Laporte (2011) calculate the amount of CO_2 generated by vehicles in the objective function, while our problem requires the computation of the energy in constraints. Moreover, in the BRP, the total load of a bike is considered and influences the energy constraint which makes the problem more challenging. Bektaş and Laporte (2011) conclude that a weighted load-minimising solution can have counter-intuitive results on energy consumption due to a possible increase of distance travelled. The vehicle routing problems with fuel consumption are studied in Karagul, Sahin, Aydemir, and

Oral (2019); Xiao, Zhao, Kaku, and Xu (2012); J. Zhang, Zhao, Xue, and Li (2015); Z. Zhang, Wei, and Lim (2015) where the energy consumed by the vehicles is minimized in the objective function. Our model, however, only provides distance-minimising solutions where the energy is checked in a constraint. Therefore, such results are not obtained by our proposed problem model.

A significant number of papers deal with route choice models for cyclists. For example, Ehrgott et al. (2012) study such a problem by considering a bi-objective variant. It is said and acknowledged that cyclists choose their route differently to drivers of private vehicles. They optimise routes for bikes by calculating a suitability score considering safety and comfort such as motor traffic volume, motor traffic speed, road lane width, presence of on-street parking, road gradient, percentage of heavy commercial vehicles, presence of cycle facilities (cycle lanes or shared bus/cycle lanes), pavement condition, etc. This score is then evaluated and considered alongside the route distance. It is also argued that in contrast to car drivers, cyclists consider a significantly broader range of factors while deciding on their routes. Hrncír, Zilecky, Song, and Jakob (2015) and Song, Zilecky, Jakob, and Hrncír (2014) also consider the bicycle routing problem with road characteristics. They emphasise the importance of slope, turn frequency, junction control, noise, pollution, scenery, and traffic volumes in addition to travel time and distance for cyclists. The paper from Silbernagl, Krismer, Malfertheiner, and Specht (2016) is another example of cycling routing where the elevation is considered. However, just like the previously mentioned papers, their work does not consider delivery purposes and therefore does not compute the energy required.

The problem of delivering goods using bikes is studied by Ghiani, Manni, Quaranta, and Triki (2009). They solve a dynamic vehicle dispatching problem with pickups and deliveries. Their model considers the use of bicycles for same-day courier services. This problem is closer to the BRP than the aforementioned ones. However, they do not take into account any cycling-related constraints. Another paper dealing with a delivery problem using bicycles is from Arango Serna, Adarme Jaimes, and Zapata Cortes (2010). Their problem is in fact based on the VRP but they do not consider the energy constraints that partly depend on the bike's load and road characteristics. The paper from Ćirović, Pamučar, and Božanić (2014) also solves the routing problem for light delivery vehicles while considering the environment. However, their model only accounts for motorcycles and does not consider the bike-related energy constraint. Optimisation problems of using bike-messengers (Maes & Vanelslander, 2012) and bike-couriers (Lee, Chae, & Kim, 2019) for delivery are also studied in the literature. However, unlike push-bicycles, using electrical bikes for deliveries can neglect the energy required by the driver to complete his/her route.

In di Prampero, Cortili, Mognoni, and Saibene (1979), an equation of motion for cyclists is provided. This equation considers several metrics such as air resistance/temperature, altitude, body size, etc. to provide the power required to move the bicycle. Their aim is to predict the speed attained under a given set of conditions, provided the metabolic power of the subject is known. Since this paper, literature has also attempted to improve models including power (Barth & Boriboonsomsin, 2009; Ross, 1997) via a similar formula. Martin, Milliken, Cobb, McFadden, and Coggan (1998) derive a mathematical model of cycling power from the same formula and provide values for the model parameters. A bicycle-mounted power measurement system is validated by comparison with a laboratory ergo-meter. They provide an analysis of the parameter effect by altering their values. Their results demonstrate the accuracy of the formula which emphasises that cycling power can be predicted by a mathematical model.

There are several papers dealing with the energy constraint for electric vehicles. In Fontana (2013), the authors consider the problem that finds the shortest path satisfying the energy constraint. This problem belongs to the class of the route choice problems, not the vehicle routing problems as in our proposed model. Our studied problem shares some characteristics with the Electric Vehicle Routing Problems in Tahami, Rabadi, and Haouari (2020) (and also with some variants of the Green Vehicle Routing Problems in Erdoğan and Miller-Hooks (2012); Leggieri and Haouari (2017)), and they consider similar constraints named energy level constraints. However, the energy consumption of each arc in these problems is often simply calculated by multiplying its distance with a fixed energy consumption rate, thus ignoring the impact of vehicle load. Additionally, they allow vehicles to recharge its energy at some charging stations which is not available for push-bikes. Pelletier, Jabali, and Laporte (2019) introduce a new variant of the electric vehicle routing problem in which the energy consumption of a vehicle depends on its load. However, the studied problem belongs to the class of robust optimization problems. Using electric vehicles can lead to the presence of strong uncertainties surrounding the energy consumption. Indeed, Asamer, Graser, Heilmann, and Ruthmair (2016) show that several parameters in the calculation of consumed energy for electric vehicle routing problems are difficult to measure or depend on uncontrolable factors. In addition, environmental factors such as wind speed and weather can significantly change the achievable range of electric vehicles. Using bikes makes our model more deterministic. For example, a rolling resistance coefficient, a wind resistance coefficient and an acceleration in the energy consummation computation of bikes are relatively constant. In cases of bad weather conditions (strong wind or heavy rain), the routing plans should be cancelled to ensure the safety of cyclists. Therefore, it is reasonable to consider the deterministic bike routing problem as proposed in this paper.

3. Problem definition and formulation

3.1. Problem definition

Formally, the BRP is defined on a complete graph $G = (N, A)$ with $N = \{0, 1, 2, ..., n\}$ as the set of nodes and A as the set of arcs defined between each pair of nodes. Node 0 is the depot where a homogeneous set of bikes $K = \{1, 2, ..., m\}$ is located. Each bike has a capacity of Q. The remaining nodes represent a set of customers $N_0 = N \setminus \{0\}$. Each customer $i \in N_0$ has a positive demand q_i representing the amount of product that needs to be picked up and transported to the depot. The distance from node i to node j is denoted by d_{ij}. Each cyclist k consumes a certain amount of energy e^k_{ij} to travel over an arc (i, j). This amount depends on a number of factors, such as bike load, speed and slope. The BRP consists in findings at most m bike routes such that the total travel distance is minimised and

- Each bike begins and ends at the depot;
- Each customer node is visited exactly once;
- The total load on each bike is less than its capacity Q;
- The total energy spent by each cyclist is less than a given value E.

3.2. Computation of the energy

The power requirements of a vehicle to move along a road segment depend on not only the dynamic parameters (e.g., speed and acceleration), but also the static ones such as vehicle weight, aerodynamic drag, rolling resistance, and road grade, etc. We use the formula proposed in Barth and Boriboonsomsin (2009) and Martin et al. (1998) to compute the energy $p^k_{ij}(l)$ (in *joule/second*) generated by cyclist $k \in K$ per unit of time to travel from node $i \in N$ to node $j \in N$ with load l:

$$p^k_{ij}(l) = (w_k + l)av + (w_k + l)gv\sin(\theta_{ij}) + 0.5c_w f\rho v^3 + (w_k + l)gc_r\cos(\theta_{ij})v \tag{1}$$

This equation is also used in other studies on routing electric vehicles Wang et al. (2023) and unmanned aerial vehicles (UAV) Nguyen and Hà (2023). The parameters of this formula and their default values are provided as follows:

- c_r: rolling resistance coefficient ($c_r = 0.005$ for a typical bitumen road on clinchers)
- c_w: wind resistance coefficient (c_w= 0.5 if there is no headwind nor tailwind)
- ρ: air density ($\rho = 1.226\ kg/m^3$ at sea level)
- g: gravitational constant (g= 9.8 m/s^2)
- w_k: the total weight of bike and cyclist (w_k= 100 kg)
- a: the acceleration ($a = 0\ m/s^2$ for simplicity)
- v: the speed of the bike on the road ($v = 20\ km/h$ or 5.56 m/s)
- f: the frontal area of the bike and rider ($f = 1.5\ m^2$)
- θ_{ij}: the slope of the road from node i to node j ($\theta_{ij} = 0$ for simplicity).

For simplicity, we assume that the speed of bike is constant ($a = 0$). We also suppose that the working area has a relatively flat terrain and the slope of all the roads can be considered as zero ($\theta_{ij} = 0\ \forall i, j \in N$). Therefore, the formula (1) can be rewritten as follows:

$$\begin{aligned} p^k_{ij}(l) &= (w_k + l)0v + (w_k + l)gv\sin(0) + 0.5c_w f\rho v^3 + (w_k + l)gc_r\cos(0)v \\ &= 0.5c_w f\rho v^3 + (w_k + l)gc_r v \quad \forall k \in K\ \forall i, j \in N \end{aligned} \tag{2}$$

3.3. A mixed integer linear programming formulation

We now formulate the problem as a Mixed Integer Linear Programming (MILP). Our formulation allows to define mathematically the problem and can be used to solve small-size instances to optimality with MILP solvers. There are three types of variables as follows:

- x^k_{ij}: binary variables equal to 1 if bike k travels from i to j
- l^k_i: real variables representing the load weight on bike k when it leaves node i
- e^k_{ij}: real variables representing the energy spent on arc (i, j) by bike k.

The objective function is to minimise the total distance travelled by bikes:

$$\textbf{Minimise} \sum_{k\in K} \sum_{i,j\in N} d_{ij}x^k_{ij} \tag{3}$$

A feasible solution must satisfy the following constraints:

- Each customer is visited by exactly one bike:

$$\sum_{k\in K}\sum_{j\in N} x_{ij}^k = 1 \;\; \forall i \in N_0 \tag{4}$$

- Flow conservation constraint: Each bike must continue its tour until the depot:

$$\sum_{i\in N} x_{ij}^k = \sum_{i\in N} x_{ji}^k \;\; \forall k \in K \; \forall j \in N \tag{5}$$

- Each bike must depart from the depot at most once.

$$\sum_{i\in N} x_{0i}^k \leq 1 \;\; \forall k \in K \tag{6}$$

- Capacity constraint: Each bike load must respect its capacity

$$l_i^k \leq Q \;\; \forall i \in N_0 \; \forall k \in K \tag{7}$$

- Relationship between l_i^k and x_{ij}^k: This constraint also removes sub-tours:

$$l_j^k \geq l_i^k + q_j - M_1(1 - x_{ij}^k) \;\; \forall k \in K \; \forall i \in N \; \forall j \in N_0 \tag{8}$$

$$l_j^k \leq l_i^k + q_j + M_1(1 - x_{ij}^k) \; \forall k \in K \; \forall i \in N \; \forall j \in N_0 \tag{9}$$

  where M_1 is a large number and can be estimated by Q.
- Load of bike k at the depot is null:

$$l_0^k = 0 \;\; \forall k \in K \tag{10}$$

- Energy constraint: The energy generated by a biker is limited to a given value:

$$\sum_{i,j\in N} e_{ij}^k x_{ij}^k \leq E \;\; \forall k \in K \tag{11}$$

where e_{ij}^k is the amount of energy that cyclist k generates to move from i to j. By taking into account the traveling time and the load weight when the bike leaves node i, we have:

$$\begin{aligned} e_{ij}^k x_{ij}^k &= x_{ij}^k \frac{d_{ij}}{v} p_{ij}^k(l_i^k) = \frac{x_{ij}^k d_{ij}}{v} \left[0.5 c_w f \rho v^3 + (l_i^k + w_k) g c_r v\right] \\ &= x_{ij}^k d_{ij} \left[0.5 c_w f \rho v^2 + (l_i^k + w_k) g c_r\right] \\ &= x_{ij}^k d_{ij} l_i^k g c_r + x_{ij}^k d_{ij} \left(0.5 c_w f \rho v^2 + w_k g c_r\right) \qquad \forall k \in K \; \forall i, j \in N \end{aligned} \tag{12}$$

Equation (12) is non-linear and can be linearised by using an additional variable $y_{ij}^k = x_{ij}^k l_i^k$. It equals 0 if $x_{ij}^k = 0$ and equals l_i^k if $x_{ij}^k = 1$. We then can replace (12) by the following constraints:

$$x_{ij}^k e_{ij}^k = gc_r y_{ij}^k d_{ij} + x_{ij}^k d_{ij}\big(0.5c_w f \rho v^2 + w_k gc_r\big)\ \forall k \in K\ \forall i,j \in N \tag{13}$$

$$y_{ij}^k \leq M_2 x_{ij}^k\ \ \forall k \in K\ \forall i,j \in N \tag{14}$$

$$y_{ij}^k \leq l_i^k\ \ \forall k \in K\ \forall i,j \in N \tag{15}$$

$$y_{ij}^k \geq l_i^k + M_3(x_{ij}^k - 1)\ \ \forall k \in K\ \forall i,j \in N \tag{16}$$

Constraints (13) express the relationship between the expression $x_{ij}^k e_{ij}^k$ and variable y_{ij}^k. Constraints (14) ensure that if $x_{ij}^k = 0$, the variable y_{ij}^k must be set to zero. The combination of constraints (15) and (16) requires that if $x_{ij}^k = 1$, the variable y_{ij}^k is set to l_i^k. Here, M_2 and M_3 are sufficiently large numbers and can be set to Q.

4. Metaheuristic

The formulation presented in the last section, as shown in the experimental section, can help to solve only small-size instances to optimality. It cannot efficiently handle medium and large instances of practical size. Therefore, in this section, we introduce a metaheuristic that can provide good solutions for practical-size instances in reasonable running time. One of its main components is a split procedure that converts a giant tour encoded as a Traveling Salesman Problem (TSP) tour into a BRP solution. This procedure is embedded in the framework of Evolutionary Local Search (ELS) proposed in Prins (2009). We decide to select this metaheuristic because it is simple, fast and has been used widely in the literature to successfully solve a variety of VRP variants (e.g., Hà, Bostel, Langevin, and Rousseau (2013, 2014)). In the ELS method, a single solution is mutated to obtain several children that are then improved by local search operators. The next generation is the best solution among the parent and its children.

Algorithm 1: ELS algorithm

Input: problem data;
Output: best solution S_b found;
$S_b \leftarrow$ `init_solution`();
$\overline{T} \leftarrow$ `concat`(S_b);
for i ***from*** *1* ***to*** ni **do**
    $\overline{f} \leftarrow cost(S_b)$;
    for j ***from*** *1* ***to*** nc **do**
        $T \leftarrow$ `mutate`($\overline{T}$);
        $S \leftarrow$ `split`(T);
        $S \leftarrow$ `local_search`(S);
        if $cost(S) < \overline{f}$ **then**
            $S_c \leftarrow S$;
            $\overline{f} \leftarrow cost(S)$
    if $cost(S_c) < cost(S_b)$ **then**
        $S_b \leftarrow S_c$;
        $\overline{T} \leftarrow$ `concat`(S_b);

Algorithm 1 describes the overall ELS algorithm starting with an initial solution generated by `init_solution`() function. A solution S is converted to a giant tour T via `concat`() function, which is then mutated to generate a new giant tour with some modifications by `mutate`() function. A new solution is generated with `split`() function and is improved via local search operators (lines 9-10). The current best solution S_b is updated only when it is outperformed by the best child S_c (lines 14-16). Here, `cost`(S) function is used to get the objective value of solution S. The solution S_b is then used for the next ELS iteration. The algorithm iterates over ni times and in each iteration, there are nc generated children.

We now describe in details the main components of our algorithm: initial solution, concatenation, split, local search, and mutation procedures.

4.1. Initial solution and concatenation procedures

The well-known savings heuristic from Clarke and Wright (Clarke & Wright, 1964) is used to initialise our ELS. The heuristic starts from a trivial solution with one dedicated trip per customer. Each iteration evaluates all constraint-feasible mergers (concatenations) of two trips and executes the one with the largest positive saving. The process stops when no such merger can be found.

The concatenation procedure `concat`() simply concatenates BRP routes into a giant tour T and is presented in Algorithm 2. The sequences of customers of bike trips are combined and each copy of the depot node is then removed. Function `insert`(T, i) is used to insert a node i at a given position in the constructing giant tour T.

Algorithm 2: Concatenation procedure

```
Input: a solution S with copies of the depot node;
Output: a giant tour (TSP tour) T;
T ← ∅;
foreach i in S do
    if i ≠ depot then
        insert(T, i);
return T;
```

4.2. Adapted split procedure

As described in Prins (2009), our algorithm alternates between solutions encoded as TSP tours, called giant tours, and genuine BRP solutions. Given an order of nodes defined by the giant tour, this procedure creates an optimal BRP solution. It adds the depot copies as delimiters to construct a complete solution that satisfies the capacity and energy constraints. The overall process is described as follows.

Given a giant tour of n customers $T = (T_1, T_2, ..., T_n)$, the split procedure builds a weighted directed graph $H = (X, A)$. The nodes in X are indexed from 0 to n. Node 0 is a dummy node while each node $i \neq 0$ represents a customer T_i. Each subsequence of customers $(T_i, T_{i+1}, ..., T_j)$ of the giant tour is evaluated to see if the trip $(0, T_i, T_{i+1}, ..., T_j, 0)$ is feasible. This evaluation not only includes the bike capacity, but also the total energy required for the trip. If the trip is feasible, it is modelled in the arc-set A by an arc $(i-1, j)$, with a weight equal to the trip cost. An optimal splitting of T into feasible trips corresponds to a min-cost path from node 0 to node n in the created graph. The Bellman algorithm is then used to solve the shortest path problem which indicates how to separate the giant tour. Each selected arc on the shortest path represents a bike trip which is included in the returned solution.

Algorithm 3: Adapted split procedure

Input: a maximal number of bikes m, an order of n customer nodes;
Output: an optimal solution regarding the given order of customers;
for k ***from*** *0* ***to*** $m+1$ **do**
  $p[k][0] \leftarrow 0;$
  for i ***from*** *1* ***to*** n **do**
    $p[k][i] = \infty;$
for i ***from*** *0* ***to*** $n-1$ **do**
  $load \leftarrow 0;$
  $j \leftarrow i+1;$
  while $j \leq n$ **do**
    if $j = i+1$ **then**
      $distance \leftarrow d_{0,j};$
      $energy \leftarrow e_{0,j}$ //computed by Eq. 1 with current load;
    else
      $distance \leftarrow distance + d_{j-1,j};$
      $energy \leftarrow energy + e_{j-1,j}$ //computed by Eq. 1 with current load;
    $load \leftarrow load + q_j;$
    if $load \leq Q$ *and* $energy + e_{j,0} \leq E$ **then**
      for k ***from*** *0* ***to*** m **do**
        if $p[k][i] \neq \infty$ ***and*** $p[k][i] + distance + d_{j0} < p[k+1][j]$ **then**
          $p[k+1][j] \leftarrow p[k][i] + distance + d_{j0};$
          $pred[k+1][j] = i;$
      $j \leftarrow j+1;$

Algorithm 3 shows the detailed steps of our split procedure. The main difference with the split procedure proposed for the VRP in Prins (2009) lies in the way to tackle the energy constraints. To check if a trip is feasible w.r.t energy constraints in $\mathcal{O}(1)$, we need to use an additional data structure *energy* to record the accumulative energy generated along the trips. At the end of the procedure, array *pred* contains several paths to reach the last node T_n in the graph H. Given a bike k and a node i, value $pred[k][i]$ represents the precedent node in the graph path. Therefore, reversely looping on *pred* with a decreasing k gives us the last customer node of each bike trip bounded by k trips. Because computing the energy required to complete any trip is done in $\mathcal{O}(1)$, the complexity of our split procedure can be estimated by $\mathcal{O}(nb)$, the same as in Prins (2009), where b is the maximum number of customers per trip. However, the number of bikes in the BRP is limited to the fleet size m, we need to compute a shortest path with at most m arcs. Thus, the overall complexity of our split procedure can be rewritten as $\mathcal{O}(nbm)$. Since graph H is directed acyclic by construction and presents other characteristics as described in Vidal (2016), other faster algorithms could be used.

4.3. Local search

Algorithm 4: Local search procedure

Input: a solution S, string length λ;
Output: a better solution S;
while $(i_1$ ***or*** i_2 ***or*** i_3 ***or*** i_4 ***or*** $i_5) = true$ **do**
    `2-Opt`(S, i_1);
    `CrossoverMove`(S, i_2);
    `SwapTwoNodes`(S, i_3);
    `OrOptMove`(S, λ, i_4);
    `StringExchange`(S, λ, i_5);
return S;

As proposed in Prins (2004), the Local Search (LS) described in Algorithm 4 includes five classical moves. Two-opt moves `2-Opt`() take a sub-sequence of nodes in a solution and invert its order. Crossover moves `Crossover`() replace two edges by two other edges from two different trips. Swap moves `SwapTwoNodes`() exchange the location of two nodes. Or-opt moves `OrOptMove`() change the position of a sub-sequence of nodes. And finally, string exchange moves `StringExchange`() swap the location of two sub-sequences of customers from two different trips. To limit the running time of the procedure, the length of a string is restricted to $\lambda \in \{1, 2, 3\}$. We also use a technique named *Sequential Search* that was presented in Irnich, Funke, and Grünert (2006) to speed up the LS procedure. Our local search operators are executed in the best-improvement fashion. After a best-improvement move is found, the procedure executes it, which modifies S, and sets a Boolean flag (last argument) to true. Otherwise, the flag is set to false. All neighbourhoods are searched again if at least one of them brings an improvement. Every procedure is given a solution S encoded in a list including several depots. The first and last nodes are depots while the others delimit the different trips.

To check a local search move is feasible in $\mathcal{O}(1)$, we adapt the efficient route evaluations from Vidal, Crainic, Gendreau, and Prins (2014). First, we reintroduce the following formula for computing the energy generated on arc (i, j) of bike k based on Equation (12):

$$e_{ij}^k = d_{ij} l_i^k g c_r + d_{ij} w_k \big(0.5 c_w f \rho v^2 + g c_r\big) \tag{17}$$

In this equation, only the first term depends on the bike load. The remaining term depends only on the arc length. For convenience, we can simplify the equation as follows:

$$e_{ij}^k = l_i^k A_{ij} + B_{ij} \tag{18}$$

in which $A_{ij} = d_{ij} g c_r$ and $B_{ij} = d_{ij} w_k (0.5 c_w f \rho v^2 + g c_r)$.

In the following, we denote a subsequence σ as $\sigma = [\sigma(1), \sigma(2), ..., \sigma(|\sigma|)]$ where $\sigma(i)$ represents the i^{th} visit of the subsequence (a visit can be a depot or a customer in a route). It can be observed that any described local search moves can be viewed as a separation of routes into subsequences, which are then concatenated to create new routes. For instance, let two routes r and r' be represented by two visit sequences $\sigma_r = [\sigma_r(1), \sigma_r(2), .., \sigma_r(|\sigma_r|)]$ and $\sigma_{r'} = [\sigma_{r'}(1), \sigma_{r'}(2), .., \sigma_{r'}(|\sigma_{r'}|)]$, respectively. A

Swap move applied on these two routes that exchanges the location of two visits $\sigma_r(i)$ and $\sigma_{r'}(j)$ can be seen as the combination of two sequences $\varphi = [\sigma_r(1), ..., \sigma_r(i-1)] \oplus [\sigma_{r'}(j)] \oplus [\sigma_r(i+1), ..., \sigma_r(|\sigma_r|)]$ and $\varphi' = [\sigma_{r'}(1), ..., \sigma_{r'}(j-1)] \oplus [\sigma_r(i)] \oplus [\sigma_{r'}(j+1), ..., \sigma_{r'}(|\sigma_{r'}|)]$. More precisely, these generated sequences represent two new routes obtained from the Swap move on r and r'. Each of them contains a concatenation of three subsequences.

For each such subsequence σ, we compute the cumulative energy $E(\sigma) = \sum_{i=1}^{|\sigma|-1} e^{veh(\sigma)}_{\sigma(i)\sigma(i+1)}$ $\big(veh(\sigma)$ denotes the bike used to serve the customers in $\sigma\big)$, load $Q(\sigma) = \sum_{i=1}^{|\sigma|} q_{\sigma(i)}$, and cost $C(\sigma) = \sum_{i=1}^{|\sigma|-1} d_{\sigma(i)\sigma(i+1)}$. We also define $A(\sigma) = \sum_{i=1}^{|\sigma|-1} A_{\sigma(i)\sigma(i+1)}$ and $B(\sigma) = \sum_{i=1}^{|\sigma|-1} B_{\sigma(i)\sigma(i+1)}$. This data can be computed directly for a sequence σ^0 involving a single vertex u, as $E(\sigma^0) = 0$, $Q(\sigma^0) = q_u$, $C(\sigma^0) = 0$, $A(\sigma^0) = 0$ and $B(\sigma^0) = 0$. These following equations then enable to compute the same data for a concatenation of two sequences:

$$C(\sigma \oplus \sigma') = C(\sigma) + C(\sigma') + d_{\sigma(|\sigma|)\sigma'(1)} \tag{19}$$

$$Q(\sigma \oplus \sigma') = Q(\sigma) + Q(\sigma') \tag{20}$$

$$A(\sigma \oplus \sigma') = A(\sigma) + A(\sigma') + A_{\sigma(|\sigma|)\sigma'(1)} \tag{21}$$

$$B(\sigma \oplus \sigma') = B(\sigma) + B(\sigma') + B_{\sigma(|\sigma|)\sigma'(1)} \tag{22}$$

$$E(\sigma \oplus \sigma') = E(\sigma) + E(\sigma') + Q(\sigma)[A(\sigma') + A_{\sigma(|\sigma|)\sigma'(1)}] + B_{\sigma(|\sigma|)\sigma'(1)} \tag{23}$$

Our proposed move evaluation procedure is based on equations above to first develop data on relevant consecutive visit subsequences (and their reversal) in a preprocessing phase, and then to evaluate the feasibility and cost of routes generated by the moves. Hence, given the prepared data on subsequences, any move evaluation can be done in constant time.

Note that, the processing will take $\mathcal{O}(n^2)$ if all subsequences are computed. To accelerate the procedure, it is sufficient to limit the preprocessing to either 'prefix' (respectively, 'suffix') subsequences containing the first (respectively, the last) node of a route. This leads to the $\mathcal{O}(n)$ procedure but we can still access data of the remaining subsequences by using the information of 'prefix' ('suffix') subsequences. For instance, in equation (19), assuming that $\sigma \oplus \sigma'$ is a 'prefix' subsequence, we can easily see that σ is also a 'prefix' subsequence and $C(\sigma')$ can be computed via $C(\sigma \oplus \sigma')$ and $C(\sigma)$.

Equations (19)-(20) are frequently used in the VRP literature to calculate loads and costs of subsequences (see for example (Vidal, Crainic, Gendreau, & Prins, 2013; Vidal et al., 2014)). In the following, we prove Equation (23) that computes the energy of concatenated subsequences. The energy of a visit sequence σ can be computed as:

$$\begin{aligned} E(\sigma) &= \sum_{i=1}^{|\sigma|-1} e^{veh(\sigma)}_{\sigma(i)\sigma(i+1)} \\ &= \sum_{i=1}^{|\sigma|-1} (l^{veh(\sigma)}_{\sigma(i)} A_{\sigma(i)\sigma(i+1)} + B_{\sigma(i)\sigma(i+1)}) \\ &= \sum_{i=1}^{|\sigma|-1} B_{\sigma(i)\sigma(i+1)} + \sum_{i=1}^{|\sigma|-1} (A_{\sigma(i)\sigma(i+1)} \sum_{j=1}^{i} q_{\sigma(j)}) \end{aligned} \tag{24}$$

Based on the equation (24), we can compute the cumulative energy of sequence $\sigma \oplus \sigma'$ as follows:

$$E(\sigma \oplus \sigma') = \sum_{i=1}^{|\sigma|-1} e^k_{\sigma(i)\sigma(i+1)} + e^k_{\sigma(|\sigma|)\sigma'(1)} + \sum_{i=1}^{|\sigma'|-1} e^k_{\sigma'(i)\sigma'(i+1)} \quad \big(k = veh(\sigma \oplus \sigma')\big)$$

$$\begin{aligned}
&= \sum_{i=1}^{|\sigma|-1} B_{\sigma(i)\sigma(i+1)} + \sum_{i=1}^{|\sigma|-1} (A_{\sigma(i)\sigma(i+1)} \sum_{j=1}^{i} q_{\sigma(j)}) \\
&+ A_{\sigma(|\sigma|)\sigma'(1)} \sum_{j=1}^{|\sigma|} q_{\sigma(j)} + B_{\sigma(|\sigma|)\sigma'(1)} \\
&+ \sum_{i=1}^{|\sigma'|-1} B_{\sigma'(i)\sigma'(i+1)} + \sum_{i=1}^{|\sigma'|-1} \big[A_{\sigma'(i)\sigma'(i+1)}(\sum_{j=1}^{i} q_{\sigma'(j)} + \sum_{j=1}^{|\sigma|} q_{\sigma(j)})\big] \\
&= E(\sigma) + Q(\sigma)A_{\sigma(|\sigma|)\sigma'(1)} + B_{\sigma(|\sigma|)\sigma'(1)} \\
&+ Q(\sigma) \sum_{i=1}^{|\sigma'|-1} A_{\sigma'(i)\sigma'(i+1)} + (\sum_{i=1}^{|\sigma'|-1} B_{\sigma'(i)\sigma'(i+1)} + \sum_{i=1}^{|\sigma'|-1} A_{\sigma'(i)\sigma'(i+1)} \sum_{j=1}^{i} q_{\sigma'(j)}) \\
&= E(\sigma) + Q(\sigma)A_{\sigma(|\sigma|)\sigma'(1)} + B_{\sigma(|\sigma|)\sigma'(1)} + Q(\sigma)A(\sigma') + E(\sigma') \\
&= E(\sigma) + E(\sigma') + Q(\sigma)[A(\sigma') + A_{\sigma(|\sigma|)\sigma'(1)}] + B_{\sigma(|\sigma|)\sigma'(1)} \qquad \square
\end{aligned}$$

4.4. Mutation

The objective of the mutation procedure is to diversify the search. Here, we do not operate this procedure directly on a complete BRP solution, but indirectly on a giant tour. Thus, neither capacity nor energy constraint needs to be considered. We use two operators `Move` and `Exchange` which are randomly selected at each iteration. The first one relocates a random node while the second one consists in swapping two different nodes randomly selected in the giant tour. After an operator is selected, p successive moves are executed. The value of p is modified during the search process. It is set to a minimum value p_{min} at the beginning and each time a new best solution is found. It is incremented whenever the mutation followed by local search returns a degraded solution, but never exceeding a maximum value p_{max}.

5. Computational results

All the computational results are carried on a computer with an AMD Ryzen 7 3700X 8-Core Processors @ 3.6 GHz and 16.0 GB RAM under Window 10. The ELS algorithm and MILP model are implemented in C++ and compiled with MSVC 14.27. CPLEX 12.10 is used for solving the MILP model.

5.1. Data and Parameters

Two sets of BRP instances denoted by A1 and A2 are randomly generated to investigate the performance of the methods. Each set contains instances with the number of depots and customers n equal to 10, 20, and 100. For each value of n, three instances are created. The location and request load of each customer are randomly generated within certain boundaries. The latitude and longitude of customers are varied in range 0-15000 meters. The demand q_i of customer i is a random integer from 1 kg to 5 kg. The bike capacity is fixed to 50 kg. To ensure the feasibility of created instances, the maximum number of bikes is set as in the initial solution found by our metaheuristic.

The selection of the maximum energy parameter E is based on the real-life scenarios. By assuming that a biker can only go a certain distance represented by $d_{max} = 40\ km$ without any load with the speed of 20 km/h, we then use equation (2) to calculate E, that is equal to 764,501. Let A1 be the set of these instances. To observe the impact of the energy constraint on the BRP solution, for each instance in set A1, we generate a new instance in set A2 by increasing the value of E by a quarter, i.e., $E = 955,626$. We also relax the energy constraint to create the instance set A3 with $E = +\infty$. The instances are labeled as BRP-*string*-*n*-*k*, where *string* is the name of instance set (A1, A2 or A3), n is the number of customers, and k $(= 1, 2, 3)$ is the instance number.

ni	nc	p_{min}	p_{max}	λ	Obj	$time$
150	200	1	1	1	96116.50	2.87
150	200	1	1	2	96084.54	3.24
150	200	1	2	1	96082.87	2.57
150	**200**	**1**	**2**	**2**	**96065.00**	**2.74**
150	200	1	5	1	97360.54	3.12
150	200	1	5	2	97203.94	3.38

Table 1.: Comparison between different configurations of ELS

For ELS parameters, the values of ni and nc are first set to 150 and 200, respectively, corresponding to 30,000 calls to the local search. We then find the suitable values for other parameters $(\lambda, p_{min}, p_{max})$ by conducting preliminary experiments. We test different combinations of these parameters on all instance sets and choose the best one. Table 1 shows the results of these calibration tests. Column '*Obj*' indicates the average objective value of solutions obtained on all the instances; each of them is solved 10 runs by the algorithm. Column '*time*' represents the average running time in seconds. The selected values for the parameters are shown in the line in bold $(\lambda, p_{min}, p_{max}) =$ (2, 1, 2).

5.2. Comparison between MILP formulation and ELS

We now analyse the results obtained by MILP formulation and ELS. For each instance, CPLEX is run only once with a time limit of 1 hour and ELS is run 10 times. Tables 2 and 3 show the results for two sets of instances A1 and A2. For the MILP-based exact method, Columns '*Obj*', '*Gap*', '*Veh*', and '*Time*' report objective values, gaps (in percentage), the number of vehicles of final solutions returned by CPLEX, and the running time (in seconds), respectively. For the ELS metaheuristic, Column 'Obj^*'

		CPLEX					ELS				
Instance	m	LB	Obj	Veh	Gap	$Time$	Obj^*	Veh^*	$\overline{Obj}$	$\overline{Time}$	$\overline{Gap}$
BRP-A1-10-1	2	47804.42	47804.42	2	optimal	1.07	47804.42	2	47804.42	0.25	0
BRP-A1-10-2	2	53886.39	53886.39	2	optimal	0.56	53886.39	2	53886.39	0.22	0
BRP-A1-10-3	2	46432.14	46432.14	2	optimal	0.27	46432.14	2	46432.14	0.29	0
BRP-A1-20-1	3	59192.2	88110.95	3	32.82	3600	75796.46	2	75796.46	0.69	0
BRP-A1-20-2	3	72496.28	88315.94	3	17.91	3600	87555.62	3	87555.62	0.39	0
BRP-A1-20-3	2	56979.47	70567.45	2	19.26	3600	64725.4	2	64725.4	0.18	0
BRP-A1-100-1	5	106880.54				-	164122	5	164368.6	3.15	0.15
BRP-A1-100-2	8	99512.24				OOM	254158.66	7	254192.4	5.08	0.01
BRP-A1-100-3	7	110949.8				-	205903.48	6	205907.4	5.71	0

Table 2.: Results for instance set A1

		CPLEX					ELS				
Instance	m	LB	Obj	Veh	Gap	$Time$	Obj^*	Veh^*	$\overline{Obj}$	$\overline{Time}$	$\overline{Gap}$
BRP-A2-10-1	1	40635.77	40635.77	1	optimal	0.06	40635.77	1	40635.77	0.06	0.00
BRP-A2-10-2	1	44674.37	44674.37	1	optimal	0.08	44674.37	1	44674.37	0.05	0.00
BRP-A2-10-3	1	43207.77	43207.77	1	optimal	0.06	43207.77	1	43207.77	0.05	0.00
BRP-A2-20-1	2	63119.45	67910.12	2	7.05	3600.00	67858.16	2	67858.16	0.31	0.00
BRP-A2-20-2	2	72047.52	79101.77	2	8.92	3600.00	77283.25	2	77283.25	0.18	0.00
BRP-A2-20-3	2	59910.69	60953.15	2	1.71	3600.00	60953.15	2	60953.15	0.66	0.00
BRP-A2-100-1	5	109390.58				OOM	153266.06	5	153266.06	6.94	0.00
BRP-A2-100-2	5	113114.91				OOM	168561.45	4	168606.46	5.31	0.03
BRP-A2-100-3	4	111978.18				OOM	155983.73	4	156063.00	3.10	0.05

Table 3.: Results for instance set A2

gives the objective value of the best solutions found among 10 runs. The number of vehicles used in the best solutions is reported in Column 'Veh^*'. Columns '$\overline{Obj}$', '$\overline{Time}$' and '$\overline{Gap}$' provide the average measures over 10 runs of the objective value, the computation time (in seconds), and the deviation in percentage from the best solution Obj^*.

Column '$Time$' can sometimes contain 'OOM' which stands for 'Out Of Memory' status of CPLEX and '-' which implies that CPLEX cannot find a feasible solution in the time limit. The blanks in the result tables indicate scenarios the data is unavailable. As can be seen, CPLEX can solve all instances with 10 customers to optimality. It cannot prove the optimality for instances with 20 customers but can provide feasible solutions. On 100-customer instances, it fails to provide any feasible solution. The results demonstrate the good performance of the ELS algorithm. It is able to find all the optimal solutions found by CPLEX. On open instances, ones that CPLEX cannot prove the optimality, ELS often finds better solutions. The average gap between solution of each run and the best solution is less than 0.15%, partly indicating the good stability of the ELS algorithm.

5.3. Comparison between ELS and a state-of-the-art metaheuristic

In this section, we compare the performance of ELS with the Hybrid Genetic Search (HGS) (Vidal et al., 2014) on the BRP instances. HGS is a state-of-the-art method for many important VRP variants such as Capacitated VRP (CVRP) (Vidal, 2022) or VRP with time windows (VRPTW) (Kool et al., 2022). In general, the basis of HGS is a genetic algorithm where each chromosome is represented by a giant tour (Section 4). Then a split procedure is applied to this giant tour to provide an explicit solution, which is optimal with respect to the request order of the giant tour. In HGS, two separate populations for feasible and infeasible solutions are created and evolved during the search. The initial size of each population is denoted as μ. The initial solutions are randomly generated by using the split algorithm on random customer orders. The selection process in each iteration involves choosing two parents from either population using a *binary tournament* and then merging them through an *ordered crossover (OX)* (Oliver, Smith, & Holland, 1987) to create a new offspring solution. This new solution is then intensified using local search and inserted into the appropriate population. HGS keeps each population at a maximum size of $\mu + \lambda$. When this value is reached, λ worst solutions in terms of diversity and quality are removed from the population.

We adapt HGS to deal with the BRP by making some modifications to its open-source implementation for solving CVRP [1]. We first replace the split and local search parts with our designed ones for the BRP (Section 4). Because our split algorithm does not feature a penalty mechanism allowing a solution to violate some constraints, HGS maintains only one population which contains feasible solutions. Finally, we follow the initialization phase of Pacheco, Martinelli, Subramanian, Toffolo, and Vidal (2023) for the single-population version of HGS. It uses a random greedy insertion instead of split for constructing solutions from randomly generated customer orders. The idea of this procedure is to iteratively insert each customer into their best position with the least increased cost. The values of μ and λ are set to 25 and 40, respectively, following the default setting of the open-source version. Like ELS, we run HGS 10 times on each instance. The number of iterations of HGS is set to 30,000, similarly to ELS. Tables 4 and 5 show the comparison results between the two methods. The metrics used in

[1] `https://github.com/vidalt/HGS-CVRP`

these tables are described in the previous subsection.

From these tables, it is clear that HGS is significantly slower than ELS. In terms of stability, the $\overline{Gap}$ values of ELS are always smaller than the ones of HGS on all instances. Furthermore, the maximal value of $\overline{Gap}$ value in the case of ELS is only 0.15 % (BRP-A1-100-1) while this figure for HGS is up to 2.47 % (BRP-A1-100-3). The best objective values and their associated number of vehicles found in ten runs of both algorithms are identical in most instances except the 100-customer instances of set A1. ELS produces slightly better $Obj*$ values on BRP-A1-100-1 and BRP-A1-100-2 instances but fails to outperform HGS on the BRP-A1-100-1 instance. However, HGS is nearly twice slower than ELS on this instance. The $\overline{Gap}$ value is 2.47% which is very large compared to ELS ($\overline{Gap}$ = 0.00%). Overall, we still opt for ELS to conduct experiments in the next subsections due to its good stability, fast running time, and reasonable solution quality.

		HGS					ELS				
Instance	m	Obj^*	Veh^*	$\overline{Obj}$	$\overline{Time}$	$\overline{Gap}$	Obj^*	Veh^*	$\overline{Obj}$	$\overline{Time}$	$\overline{Gap}$
BRP-A1-10-1	2	47804.42	2	47804.42	1.22	0.00	47804.42	2	47804.42	0.25	0.00
BRP-A1-10-2	2	53886.39	2	53886.39	1.13	0.00	53886.39	2	53886.39	0.22	0.00
BRP-A1-10-3	2	46432.14	2	46432.14	1.13	0.00	46432.14	2	46432.14	0.29	0.00
BRP-A1-20-1	3	75796.46	2	75796.46	1.52	0.00	75796.46	2	75796.46	0.69	0.00
BRP-A1-20-2	3	87555.62	3	87555.62	1.83	0.00	87555.62	3	87555.62	0.39	0.00
BRP-A1-20-3	2	64725.40	2	64725.40	1.16	0.00	64725.40	2	64725.40	0.18	0.00
BRP-A1-100-1	5	164428.70	5	165103.76	6.28	0.41	**164122.00**	5	164368.60	3.15	0.15
BRP-A1-100-2	8	254496.04	7	254746.61	12.82	0.10	**254158.66**	7	254192.40	5.08	0.01
BRP-A1-100-3	7	**189897.35**	**5**	194701.15	12.31	2.47	205903.48	6	205907.40	5.71	0.00

Table 4.: Comparison results for instance set A1 between HGS and ELS

		HGS					ELS				
Instance	m	Obj^*	Veh^*	$\overline{Obj}$	$\overline{Time}$	$\overline{Gap}$	Obj^*	Veh^*	$\overline{Obj}$	$\overline{Time}$	$\overline{Gap}$
BRP-A2-10-1	1	40635.77	1	40635.77	0.87	0.00	40635.77	1	40635.77	0.06	0.00
BRP-A2-10-2	1	44674.37	1	44674.37	0.88	0.00	44674.37	1	44674.37	0.05	0.00
BRP-A2-10-3	1	43207.77	1	43207.77	0.82	0.00	43207.77	1	43207.77	0.05	0.00
BRP-A2-20-1	2	67858.16	2	67858.16	1.23	0.00	67858.16	2	67858.16	0.31	0.00
BRP-A2-20-2	2	77283.25	2	77283.25	1.10	0.00	77283.25	2	77283.25	0.18	0.00
BRP-A2-20-3	2	60953.15	2	60953.15	1.45	0.00	60953.15	2	60953.15	0.66	0.00
BRP-A2-100-1	5	153266.06	5	153274.24	9.22	0.01	153266.06	5	153266.06	6.94	0.00
BRP-A2-100-2	5	168561.45	4	169087.76	7.94	0.31	168561.45	4	168606.46	5.31	0.03
BRP-A2-100-3	4	155983.73	4	156135.58	5.66	0.10	155983.73	4	156063.00	3.10	0.05

Table 5.: Comparison results for instance set A2 between HGS and ELS

5.4. The impact of energy constraint

In this section, the importance of the energy constraint is discussed. We explain why it is necessary to study the BRP by experiments. To achieve the goal, we only keep the capacity constraint and relax the energy constraint by setting the maximum energy value E to $+\infty$. All the settings of both exact and metaheuristic methods are kept unchanged. The results for the instances without the energy constraint are shown in Table 6 with the column headings similar to those mentioned in subsection 5.2.

		CPLEX					ELS				
Instance	m	LB	Obj	Veh	Gap	$Time$	Obj^*	Veh^*	$\overline{Obj}$	$\overline{Time}$	$\overline{Gap}$
BRP-A3-10-1	1	40635.77	40635.77	1	optimal	0.05	40635.77	1	40635.77	0.25	0.00
BRP-A3-10-2	1	44674.37	44674.37	1	optimal	0.05	44674.37	1	44674.37	0.25	0.00
BRP-A3-10-3	1	43207.77	43207.77	1	optimal	0.06	43207.77	1	43207.77	0.25	0.00
BRP-A3-20-1	1	57597.17	57597.17	1	optimal	0.55	57597.17	1	57597.17	0.59	0.00
BRP-A3-20-2	1	69209.44	69209.44	1	optimal	0.41	69209.44	1	69209.44	0.58	0.00
BRP-A3-20-3	1	50798.45	50798.45	1	optimal	0.31	50798.45	1	50798.45	0.53	0.00
BRP-A3-100-1	5	108317.31				OOM	151811.37	5	151811.37	12.71	0.00
BRP-A3-100-2	4	112782.24				OOM	167025.31	4	167025.31	14.79	0.00
BRP-A3-100-3	4	111224.17				OOM	155578.43	4	155578.43	11.33	0.00

Table 6.: Results for instance set A3

From Table 6, it can be observed that when the energy constraint is ignored, the instances tend to be easier for CPLEX but more difficult for ELS. The MILP-based exact method can now solve successfully all instances with 20 customers but still can not provide any feasible solutions for the 100-customer instances. The ELS algorithm often requires more time to handle the instances without the energy constraints. We believe the reason for this lies in the fact that the wider solution space makes the local search operators of the algorithm to be called more frequent.

We now analyze further the obtained solutions when the energy constraint is relaxed. Two characteristics of the best solutions found by all the methods are investigated: the objective value and the energy consumed by the most tired biker. Table 7 illustrates our obtained results.

	set A1		set A2	
Instance	gap_{Obj}	gap_E	gap_{Obj}	gap_E
10c1	15.00	5.53	0.00	-18.09
10c2	17.10	15.14	0.00	-6.08
10c3	6.94	11.61	0.00	-10.48
20c1	24.01	33.94	15.12	17.42
20c2	20.95	44.96	10.45	31.20
20c3	21.52	26.05	16.66	7.56
100c1	7.50	26.16	0.95	7.69
100c2	34.28	25.40	0.91	6.75
100c3	24.44	21.45	0.26	1.82

Table 7.: Impact of energy constraint on the BRP solution

In Table 7, Column 'gap_{Obj}' represents the gap (in percentage) between the best solutions found by all the methods in two scenarios: with and without the energy constraint. Its value is calculated as $gap_{Obj} = 100\frac{Obj_E - Obj_{NoE}}{Obj_E}$, where Obj_E and Obj_{NoE} are the objective values of the solutions with and without the energy constraint, respectively. Column 'gap_E' reports the deviation from the maximal allowed energy E of the energy E_{max} consumed by the most energy-consumed biker in case the energy constraint is not taken into account. The value of gap_E is then measured as $gap_E = 100\frac{E_{max} - E}{E_{max}}$. We note again that E is set to 764,501 in instance set A1 and 955,626 in instance set A2.

As expected, when the energy constraint is removed, less bikes are required and less cost bike routes are found. The energy constraint makes the transportation cost increase from 6.94% to 34.28% on set A1 and from 0% to 16.66% on set A2. However, the maximum energy consumed by a biker always exceeds the allowed value E representing the physical fitness of bikers on set A1. The gaps in some cases can be up to 44.96%, which are relatively high. Even when the energy constraint is more relaxed as in instance set A2, the value of gap_E increases up to 31.2%. These demonstrate the importance of the energy constraint in the BRP, without which the BRP solution can be infeasible due to the limited physical condition of bikers. In the following, we continue to analyse the impact of energy constraint in a real-world scenario.

5.5. Real-world case

The need for the BRP model is reinforced by our collaboration with Peloton, a company which delivers and picks up goods using bikes in Liverpool. In the collaboration project, the company collects the food donated by restaurants to food banks which are charities that provide food for people in need. As their customer identities and locations are private, 50 restaurants in Liverpool were randomly selected to create a real instance. The depot is Asda Breck Road Superstore. The bike number is bounded to 5 and the bike capacity is 50 kg. The customer demand is still varied between 1 and 5. An OpenStreetMap API is used to get the coordinates (latitude, longitude) of the customers. The real distances are also got from this API by requesting the shortest path for each pair of customers.

Because the MILP model cannot handle efficiently the instance with 50 customers, we use the ELS algorithm for this experiment. The ELS algorithm is run 10 times and the best solution is recorded. Two settings are considered: with and without the energy constraint. In case of using the energy constraint, the maximum energy level E is set to 764,501 and 955,626, which is the same as in Section 5.1.

Table 8 presents the detailed solutions of this experiment. We report the distance, consumed energy, and load of each mobilized bike route of the best solution found. As can be seen, when E is set to 764,501, four bikes are mobilized while in two remaining scenarios, only three bikes are required. The energy constraint makes the increase of travel cost by 1.53% and 0.98% for E = 764,501 and 955,626, respectively. In the solution obtained when the energy constraint is ignored, two bike routes violate the energy constraint with $E = 764,501$ (the violations are 28.94% and 3.26%, respectively), and one bike route violates the energy constraint with $E = 955,626$ (the violation is 11.18%). This again confirms the importance of the energy constraint in the BRP problem. Finally, Figures 2, 3, and 4 visualize the obtained solutions for three different scenarios. The different bike routes are represented by different color lines.

	Bike 1	Bike 2	Bike 3	Bike 4	Total
E = 764,501					
distance	13992.78	31955.53	30368.90	33962.71	110279.93
energy	277720.58	647100.73	600640.21	673884.01	2199345.54
load	32	45	28	30	135
E = 955,626					
distance	32779.03	31955.53	44934.14	0.00	109668.71
energy	671730.79	647100.73	912048.55	0.00	2230880.08
load	43	45	47	0	135
$E = \infty$					
distance	16379.48	53111.75	39116.99	0.00	108608.23
energy	325498.05	1075909.19	790293.88	0.00	2191701.14
load	35	50	50	0	135

Table 8.: Details of the bike trips for the real instance

Figure 2.: BRP solution for the real instance with the energy constraint E = 764,501

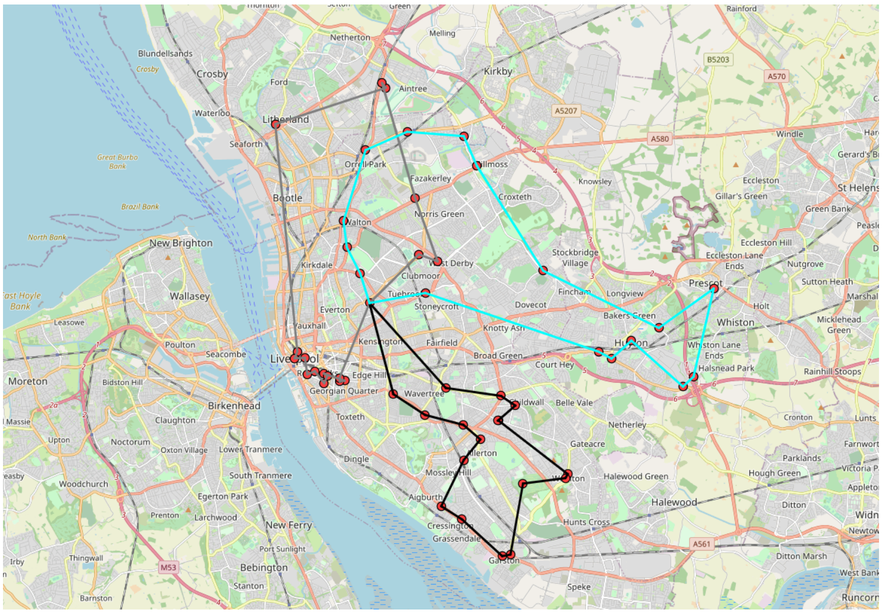

Figure 3.: BRP solution for the real instance with the energy constraint $E = 955{,}626$

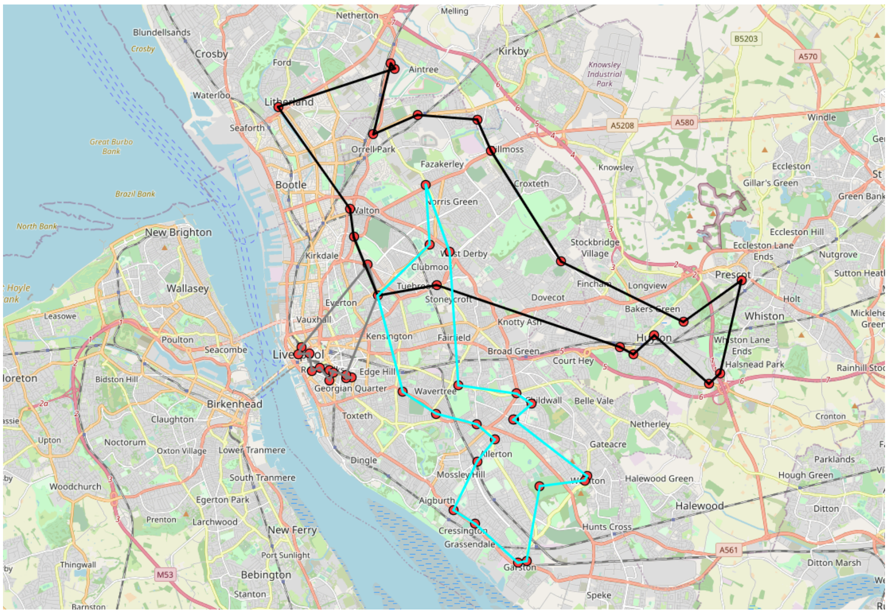

Figure 4.: BRP solution for the real instance without the energy constraint

Last but not least, we investigate the impact of different capacity settings on the solutions of the real-world instance. Three values of the capacity $Q \in \{30, 50, 70\}$ are taken into account. Tables 9 and 10 show the results of the ELS algorithm for the two maximum energy values as described in Section 5.1. When $E = 764, 501$, increasing Q from 30 to 50 and 70 allows to reduce one bike. Two routes are also excluded if we increase the value of Q from 30 in case of $E = 955, 626$. As expected, when reducing the capacity Q, the objective value is increased in most cases. The only exception is for $E = 764, 501$ where the solutions of $Q = 50$ and $Q = 70$ are identical. This can be due to the energy bound which makes the biker unable to service more requests when increasing the capacity.

	Bike 1	Bike 2	Bike 3	Bike 4	Bike 5	Total
$Q = 30$						
distance	12442.67	19007.79	30368.9	33962.71	28960.68	124742.76
energy	244616.78	373649.99	600640.21	673884.01	572660.79	2465451.8
load	22	26	28	30	29	135
$Q = 50$						
distance	13992.78	31955.53	30368.9	33962.71	0	110279.93
energy	277720.58	647100.73	600640.21	673884.01	0	2199345.54
load	32	45	28	30	0	135
$Q = 70$						
distance	13992.78	31955.53	30368.9	33962.71	0	110279.93
energy	277720.58	647100.73	600640.21	673884.01	0	2199345.54
load	32	45	28	30	0	135

Table 9.: Details of the bike trips for different capacities with $E = 764,501$

	Bike 1	Bike 2	Bike 3	Bike 4	Bike 5	Total
$Q = 30$						
distance	12442.67	19007.79	30368.9	33962.71	28960.68	124742.76
energy	244616.78	373649.99	600640.21	673884.01	572660.79	2465451.8
load	22	26	28	30	29	135
$Q = 50$						
distance	32779.03	31955.53	44934.14	0	0	109668.71
energy	671730.79	647100.73	912048.55	0	0	2230880.08
load	43	45	47	0	0	135
$Q = 70$						
distance	37461.97	38354.14	30368.9	0	0	106185.02
energy	765786.66	769008.99	601878.79	0	0	2136674.46
load	64	43	28	0	0	135

Table 10.: Details of the bike trips for different capacities with $E = 955,626$

6. Conclusion

This paper introduced a bike routing problem and made several contributions. First, a new MILP model was proposed to tackle the routing problem using push-bikes while taking into account the energy required to move the bike as the load accumulates during the trip. A linearization technique was applied to avoid using non-linear solvers. Second, an adapted Evolutionary Local Search was introduced to solve large instances in a reasonable amount of time. A new split procedure was designed to handle energy constraints. New data structures were proposed to accelerate the local search procedures. The computational results demonstrated the efficiency of our method compared to CPLEX. We also showed the importance of the new energy constraint considered in the BRP problem. Moreover, an instance based on real-world customers was created to better demonstrate the relevance of the method and the new problem.

Further research directions can be investigated. For the simplicity purpose, a number of model parameters have been neglected in this research. Grappe, Candau, Belli, and Rouillon (1997) study the power required from a cyclist to move given a certain position. They conclude that an important part of the energy is due to the aerodynamic drag of air. Due to the cargo-based bikes that are being used, the frontal surface area is not negligible when facing significant winds. Therefore, studying the BRP while considering the wind speed could lead to better itineraries. Another improvement could be to consider real characteristics of the roads such as slope, different materials for road construction, etc,. Two customers could be linked by several paths of different elevation profiles which could require different energy from the cyclist. In that case, the model could be extended to an arc routing problem for another type of delivery. Another research direction could allow the variation of the bike speed on different edges, leading to the more complex speed optimization problem.

Acknowledgment

The manuscript of the paper was finished during the research stay of the corresponding author Minh Hoàng Hà at the Vietnamese Institute for Advanced Studies in Mathematics (VIASM). He wishes to thank this institution for their kind hospitality and support.

Disclosure statement

No potential conflict of interest was reported by the authors.

Funding

Data availability statement

The data and detailed results that support the findings of this study are openly available in a repository which will be published after the paper is accepted.

Notes on contributor(s)

Yannis Ancele received a PhD degree at Liverpool John Moores University. His research interests include operations research, mathematical modeling, and machine learning.

Quang Anh Pham is a research assistant at ORLab in Phenikaa University. He obtained a BSc in Computer Science from VNU University of Engineering and Technology. His research interests are operations research, artificial intelligence, and combinatorial optimization techniques such as metaheuristic and integer & dynamic programming.

Minh Hoàng Hà is a researcher/lecturer at ORLab, Phenikaa University. He received his PhD from Ecole des Mines de Nantes, France. His research appears in Transportation Science, European Journal of Operational Research, Computers and Operations Research, Networks, Transportation Research Part E: Logistics and Transportation Review, Transportation Research Part C: Emerging Technologies, etc.

Dante Ben Matellini is a senior industrial lecturer within the School of Engineering, Technology and Maritime Operations. Ben leads several modules within engineering programmes, which focus on risk and safety management, and industrial management. Ben's area of expertise and research interest is in risk and safety management. Ben is interested in the development and application of risk management techniques within industry. He has previously worked on projects within the fire and rescue services, process industries, and Arctic shipping. Ben is director of study for several PhD projects linked to these areas.

Trung Thanh Nguyen is currently a Reader in operational research (OR) with Liverpool John Moores University and the Co-Director of the Liverpool Offshore and Marine Research Institute. He has an international standing in operational research for logistics/transport. He has led over 20 research projects in transport/logistics, most with close industry collaborations. He has published about 50 peer-reviewed articles. All of his journal articles are in leading journals (ranked 1st–20th in their fields). He has edited eight books and gave speeches to many conferences/events. He co-organized six leading conferences. He was a TPC member of more than 30 international conferences.

References

Allen, J., Browne, M., & Holguin-Veras, J. (2010). Sustainability strategies for city logistics. In A. McKinnon, S. Cullinane, M. Browne, & A. Whiteing (Eds.), *Green logistics: Improving the environmental sustainability of logistics* (pp. 282–305). Kogan Page London.

Arango Serna, M. D., Adarme Jaimes, W., & Zapata Cortes, J. A. (2010). Commodities distribution using alternative types of transport. A study in the Colombian Bread SMEs. *Dyna*, *77*(163), 222–233.

Asamer, J., Graser, A., Heilmann, B., & Ruthmair, M. (2016). Sensitivity analysis for energy demand estimation of electric vehicles. *Transportation Research Part D: Transport and Environment*, *46*, 182-199.

Barth, M., & Boriboonsomsin, K. (2009). Energy and emissions impacts of a freeway-based dynamic eco-driving system. *Transportation Research Part D: Transport and Environment*, *14*(6), 400–410.

Bektaş, T., & Laporte, G. (2011). The pollution-routing problem. *Transportation Research Part B: Methodological*, *45*(8), 1232–1250.

Caggiani, L., Colovic, A., Prencipe, L. P., & Ottomanelli, M. (2021). A green logistics solution for last-mile deliveries considering e-vans and e-cargo bikes. *Transportation Research Procedia*, *52*, 75-82. (23rd EURO Working Group on Transportation Meeting, EWGT 2020, 16-18 September 2020, Paphos, Cyprus)

Chemla, D., Meunier, F., & Wolfler Calvo, R. (2013). Bike sharing systems: Solving the static rebalancing problem. *Discrete Optimization*, *10*(2), 120-146.

Ćirović, G., Pamučar, D., & Božanić, D. (2014). Green logistic vehicle routing problem: Routing light delivery vehicles in urban areas using a neuro-fuzzy model. *Expert Systems with Applications*, *41*(9), 4245–4258.

Clarke, G., & Wright, J. W. (1964). Scheduling of vehicles from a central depot to a number of delivery points. *Operations Research*, *12*(4), 568–581.

Delfani, F., Kazemi, A., SeyedHosseini, S. M., & Niaki, S. T. A. (2021). A novel robust possibilistic programming approach for the hazardous waste location-routing problem considering the risks of transportation and population. *International Journal of Systems Science: Operations & Logistics*, *8*(4), 383-395.

Dell'Amico, M., Hadjicostantinou, E., Iori, M., & Novellani, S. (2014). The bike sharing rebalancing problem: Mathematical formulations and benchmark instances. *Omega*, *45*, 7–19.

di Prampero, P. E., Cortili, G., Mognoni, P., & Saibene, F. (1979). Equation of motion of a cyclist. *Journal of Applied Physiology*, *47*(1), 201-206.

Ehrgott, M., Wang, J. Y., Raith, A., & Van Houtte, C. (2012). A bi-objective cyclist route choice model. *Transportation Research Part A: Policy and Practice*, *46*(4), 652–663.

Elbert, R., & Friedrich, C. (2020). Urban consolidation and cargo bikes: a simulation study. *Transportation Research Procedia*, *48*, 439-451.

Erdoğan, S., & Miller-Hooks, E. (2012). A green vehicle routing problem. *Transportation Research Part E: Logistics and Transportation Review*, *48*(1), 100–114.

Fontana, M. W. (2013). *Optimal routes for electric vehicles facing uncertainty, congestion, and energy constraints* (Unpublished doctoral dissertation). Massachusetts Institute of Technology.

Ghiani, G., Manni, E., Quaranta, A., & Triki, C. (2009). Anticipatory algorithms for same-day courier dispatching. *Transportation Research Part E: Logistics and Transportation Review*, *45*(1), 96–106.

Goodarzian, F., Shishebori, D., Bahrami, F., Abraham, A., & Appolloni, A. (2023). Hybrid meta-heuristic algorithms for optimising a sustainable agricultural supply chain network considering co2 emissions and water consumption. *International Journal of Systems Science: Operations & Logistics*, *10*(1), 2009932.

Grappe, F., Candau, R., Belli, A., & Rouillon, J. D. (1997). Aerodynamic drag in field cycling with special reference to the obree's position. *Ergonomics*, *40*(12), 1299–1311.

Hà, M. H., Bostel, N., Langevin, A., & Rousseau, L. (2013). An exact algorithm and a metaheuristic for the multi-vehicle covering tour problem with a constraint on the number of vertices. *European Journal of Operational Research*, *226*(2), 211–220.

Hà, M. H., Bostel, N., Langevin, A., & Rousseau, L. (2014). An exact algorithm and a metaheuristic for the generalized vehicle routing problem with flexible fleet size. *Computers & Operations Rearch*, *43*, 9–19.

Hrncír, J., Zilecky, P., Song, Q., & Jakob, M. (2015). Speedups for multi-criteria urban bicycle routing. In G. F. Italiano & M. Schmidt (Eds.), *15th workshop on algorithmic approaches for transportation modelling, optimization, and systems, ATMOS 2015, september 17, 2015, patras, greece* (Vol. 48, pp. 16–28). Schloss Dagstuhl - Leibniz-Zentrum für Informatik.

Irnich, S., Funke, B., & Grünert, T. (2006). Sequential search and its application to vehicle-routing problems. *Computers & Operations Research*, *33*(8), 2405-2429.

Kara, I., Kara, B. Y., & Yetis, M. K. (2007). Energy minimizing vehicle routing problem. In A. Dress, Y. Xu, & B. Zhu (Eds.), *International conference on combinatorial optimization and applications* (pp. 62–71). Springer Berlin Heidelberg.

Karagul, K., Sahin, Y., Aydemir, E., & Oral, A. (2019). A simulated annealing algorithm based solution method for a green vehicle routing problem with fuel consumption. In *Lean and green supply chain management* (pp. 161–187). Springer.

Koning, M., & Conway, A. (2016). The good impacts of biking for goods: Lessons from Paris city. *Case Studies on Transport Policy*, *4*(4), 259–268.

Kool, W., Juninck, J. O., Roos, E., Cornelissen, K., Agterberg, P., van Hoorn, J., & Visser, T. (2022). Hybrid genetic search for the vehicle routing problem with time windows: a high-performance implementation. In *12th dimacs implementation challenge workshop.*

Lee, K., Chae, J., & Kim, J. (2019). A courier service with electric bicycles in an urban area: The case in Seoul. *Sustainability*, *11*(5), 1255.

Leggieri, V., & Haouari, M. (2017). A practical solution approach for the green vehicle routing problem. *Transportation Research Part E: Logistics and Transportation Review*, *104*, 97–112.

Li, Y., & Liu, Y. (2021). The static bike rebalancing problem with optimal user incentives. *Transportation Research Part E: Logistics and Transportation Review*, *146*, 102216.

Lin, C., Choy, K. L., Ho, G. T., Chung, S. H., & Lam, H. (2014). Survey of green vehicle routing problem: past and future trends. *Expert Systems with Applications*, *41*(4), 1118–1138.

Maes, J., & Vanelslander, T. (2012). The use of bicycle messengers in the logistics chain, concepts further revised. *Procedia-Social and Behavioral Sciences*, *39*, 409–423.

Martin, J. C., Milliken, D. L., Cobb, J. E., McFadden, K. L., & Coggan, A. R. (1998). Validation of a mathematical model for road cycling power. *Journal of Applied Biomechanics*, *14*(3), 276–291.

Nguyen, M. A., & Hà, M. H. (2023). The parallel drone scheduling traveling salesman problem with collective drones. *Transportation Science*, *57*(4), 866-888.

Oliver, I. M., Smith, D. J., & Holland, J. R. C. (1987). A study of permutation crossover operators on the traveling salesman problem. In *Proceedings of the second international conference on genetic algorithms on genetic algorithms and their application* (p. 224–230). USA: L. Erlbaum Associates Inc.

Pacheco, T., Martinelli, R., Subramanian, A., Toffolo, T. A., & Vidal, T. (2023). Exponential-size neighborhoods for the pickup-and-delivery traveling salesman problem. *Transportation Science*, forthcoming.

Pelletier, S., Jabali, O., & Laporte, G. (2019). The electric vehicle routing problem with energy consumption uncertainty. *Transportation Research Part B: Methodological*, *126*, 225–255.

Prins, C. (2004). A simple and effective evolutionary algorithm for the vehicle routing problem. *Computers & Operations Research*, *31*(12), 1985–2002.

Prins, C. (2009). A GRASP × evolutionary local search hybrid for the vehicle routing problem. In F. B. Pereira & J. Tavares (Eds.), *Bio-inspired algorithms for the vehicle routing problem* (Vol. 161, pp. 35–53). Springer.

Rahman, M. H., Rahman, M. F., & Tseng, T.-L. B. (2022). Estimation of fuel consumption and selection of the most carbon-efficient route for cold-chain logistics. *International Journal of Systems Science: Operations & Logistics*, *0*(0), 1-17.

Ross, M. (1997). Fuel efficiency and the physics of automobiles. *Contemporary Physics*, *38*(6), 381–394.

Silbernagl, D., Krismer, N., Malfertheiner, M., & Specht, G. (2016). Optimization of digital elevation models for routing. In L. Wiese, H. Bitzmann, & T. Waage (Eds.), *Proceedings of the 28th gi-workshop grundlagen von datenbanken, nörten hardenberg, germany, may 24-27, 2016* (Vol. 1594, pp. 103–108). CEUR-WS.org.

Song, Q., Zilecky, P., Jakob, M., & Hrncír, J. (2014). Exploring pareto routes in multi-criteria urban bicycle routing. In *17th international IEEE conference on intelligent transportation systems, ITSC 2014, qingdao, china, october 8-11, 2014* (pp. 1781–1787). IEEE.

Tahami, H., Rabadi, G., & Haouari, M. (2020). Exact approaches for routing capacitated electric vehicles. *Transportation Research Part E: Logistics and Transportation Review*, *144*, 102126.

Tipagornwong, C., & Figliozzi, M. (2014). Analysis of competitiveness of freight tricycle delivery services in urban areas. *Transportation Research Record*, *2410*(1), 76–84.

Vidal, T. (2016). Split algorithm in O(n) for the capacitated vehicle routing problem. *Computers & Operations Research*, *69*, 40–47.

Vidal, T. (2022). Hybrid genetic search for the cvrp: Open-source implementation and swap* neighborhood. *Computers & Operations Research*, *140*, 105643.

Vidal, T., Crainic, T. G., Gendreau, M., & Prins, C. (2013). A hybrid genetic algorithm with adaptive diversity management for a large class of vehicle routing problems with time-windows. *Computers & Operations Research*, *40*(1), 475-489.

Vidal, T., Crainic, T. G., Gendreau, M., & Prins, C. (2014, May). A unified solution framework for multi-attribute vehicle routing problems. *European Journal of Operational Research*,

234(3), 658–673.
Wang, Y., Peng, S., Zhou, X., Mahmoudi, M., & Zhen, L. (2020). Green logistics location-routing problem with eco-packages. *Transportation Research Part E: Logistics and Transportation Review*, *143*, 102118.
Wang, Y., Zhou, J., Sun, Y., Fan, J., Wang, Z., & Wang, H. (2023). Collaborative multidepot electric vehicle routing problem with time windows and shared charging stations. *Expert Systems with Applications*, *219*, 119654.
Xiao, Y., Zhao, Q., Kaku, I., & Xu, Y. (2012). Development of a fuel consumption optimization model for the capacitated vehicle routing problem. *Computers & Operations Research*, *39*(7), 1419–1431.
Zhang, J., Zhao, Y., Xue, W., & Li, J. (2015). Vehicle routing problem with fuel consumption and carbon emission. *International Journal of Production Economics*, *170*, 234-242.
Zhang, Z., Wei, L., & Lim, A. (2015). An evolutionary local search for the capacitated vehicle routing problem minimizing fuel consumption under three-dimensional loading constraints. *Transportation Research Part B: Methodological*, *82*, 20-35.