

Fuel Emissions Optimization in Vehicle Routing Problems with Time-Varying Speeds

Jiani Qian¹ and Richard Eglese²

Department of Management Science
Lancaster University Management School, Lancaster, LA1 4YX, U.K.

Abstract

The problem considered in this paper is to produce routes and schedules for a fleet of delivery vehicles that minimize the fuel emissions in a road network where speeds depend on time. In the model, the route for each vehicle must be determined, and also the speeds of the vehicles along each road in their paths are treated as decision variables. The vehicle routes are limited by the capacities of the vehicles and time constraints on the total length of each route. The objective is to minimize the total emissions in terms of the amount of Greenhouse Gas (GHG) produced, measured by the equivalent weight of CO₂ (CO₂e).

A column generation based tabu search algorithm is adapted and presented to solve the problem. The method is tested with real traffic data from a London road network. The results are analyzed to show the potential saving from the speed adjustment process. The analysis shows that most of the fuel emissions reduction is able to be attained in practice by ordering the customers to be visited on the route using a distance-based criterion, determining a suitable path between customers for each vehicle and travelling as fast as is allowed by the traffic conditions up to a preferred speed.

Key words: Distribution, fuel emissions optimization, vehicle routing problems, column generation, tabu search

¹Present affiliation and address: Shanghai Lujiazui International Financial Asset Exchange Co. Ltd., 1333 Lujiazui Ring Road, Pudong New District, Shanghai, P.R. China
Email: j.qian0206@gmail.com
Sponsored by LANCS Initiative (EPSRC); Grant number: EP/F033613/1.

² Corresponding author: Email: R.Eglese@lancaster.ac.uk, Tel. +44 1524 593869.

1. Introduction

Technical developments and the growth in road traffic pose new challenges for research in vehicle routing and scheduling for freight transport. Remote vehicle tracking techniques enable the road traffic data for different times of day and different days of the week to be collected, so as to provide detailed information on transit times for different roads by time of day and day of week. This provides an opportunity to plan vehicle routes and schedules taking time-varying speeds into account. In addition, the growth in road traffic and the use of road freight transport also bring problems of environmental pollution. Concerns about the environmental impact of transport activities have led to new vehicle routing models where the objective is to minimize the harmful effects of transportation on the environment.

An increasing number of papers are being published where fuel emissions are explicitly modeled. However many of these simplify the model by assuming that paths between customers are fixed or that the speeds of the vehicles are time-independent. In the model described in this paper, the speed of the traffic on the underlying road network is time dependent. In addition, the path used by a vehicle between a pair of customers and the speeds on the road segments are decision variables. This paper will describe a column generation based tabu search algorithm, which can work together with a solution method for single paths, in order to minimize fuel emissions for Vehicle Routing Problems (VRPs) with time-varying speeds.

The algorithm is then used for modeling a distribution operation using real traffic data from a road network located in London. The aim of these experiments is to discover how much reduction in CO₂e can be obtained by using the algorithm described in this paper, compared with other approaches that are faster to compute. Experiments are also carried out to determine the effect of allowing more waiting time at customers.

The paper is organized as follows. Section 2 contains a review of relevant literature. The problem is described in Section 3, which is followed by a set-partitioning model for the VRP. Section 4 introduces the framework of the column generation based tabu search algorithm, which is used to find a prospective sequence for a set of customers, and goes on to discuss details of the algorithm. The computational experiments and their results are

then presented in Section 5. Finally, conclusions are drawn in Section 6, and the main findings are highlighted.

2. Literature Review

In recent years, there has been increasing interest in estimating the environmental effects of vehicle routing policies. A survey of recent work in this area can be found in [9]. Various models have been proposed for estimating the fuel used by vehicles when travelling on roads. Examples include one published by the European Commission in the MEET report described by Hickman et al. [17] and the Comprehensive Modal Emissions Model (CMEM) described by Scora and Barth [27]. The CO₂ emissions are normally calculated as being proportional to the fuel used. The fuel consumption and hence the emissions may relate to factors such as the vehicle type, weight and speed. Demir et al. [3] provide a comparison of a number of such models.

Recent research on minimizing emissions in vehicle routing models can be divided into two main categories: the first is the set of models where time-independence is assumed and the second set includes models where the road conditions are subject to traffic congestion and so the time needed to travel along a road segment depends on the time of day.

Among the time-independent models, Palmer [23] developed a model where vehicle speeds are inputs to the model and the approach is tested on a case study of home deliveries for grocery stores in the UK. He found an average saving of 4.8% in CO₂ emissions was possible compared to using routes that minimize time, but at the expense of a 3.8% increase in the time required. His model does not take vehicle loads explicitly into account, but Suzuki [28] uses a model where load is taken into account and finds that delivering relatively heavy items early in a tour can be worthwhile in reducing the fuel consumption. Several case studies have been reported using time-independent approaches with the objective of minimizing fuel consumption and hence emissions. For example, Ubeda et al. [29] consider emission factors in planning routes for a food delivery operation. They show savings of around 25.5% in CO₂ emissions, but this is mainly due to reducing the number of routes needed compared to the original plan.

Other time-independent models allow the speeds of vehicles to be decision variables. The approach adopted by Bektaş and Laporte [1] in their Pollution-Routing Problem (PRP) uses a CMEM-based model and considers both load and speed in estimating a cost function to be minimized. They propose a non-linear mixed integer mathematical programming formulation and show how it can be linearized. The formulation can only solve small PRP instances, but Demir et al. [4] provide an adaptive large neighbourhood search algorithm for much larger PRP instances. Van Woensel et al. [30] develop a model showing how queuing theory can be used to describe traffic flows and calculate emissions using the model described in the MEET report.

In the set of time-dependent models, Eglese et al. [10] make use of traffic speed information collected at different times on sections of a road network to create a Road Timetable showing the quickest times between origins and destinations starting at different times of the day. In Maden et al. [21] the Road Timetable is used with a tabu-search called LANTIME to minimize the total time required for a distribution operation. Vehicles are assumed to travel at the speed which minimizes their emissions per unit distance unless the congestion indicates that this is not possible, when the vehicles travel at the average speed of the traffic recorded for that road segment at that time. Results from a case study based on the distribution plans for an electrical goods wholesaler in the UK show that CO₂ emissions can be reduced by around 7% with this approach. This is because routes with high congestion and hence, enforced low speeds and high emissions, are avoided. Figliozzi [13] also takes into account congestion in minimizing emissions using a model based on the MEET report. An integer programming formulation is presented and a solution algorithm is described which is tested on modified Solomon benchmark problems. In contrast to Maden et al. [21], the model allows vehicles to travel faster than their optimum speed that minimizes emissions if the traffic conditions and speed limits allow. Thus, there are examples where uncongested conditions can lead to increased emissions.

Vehicle speeds may also be used as decision variables in time-dependent models. Jabali et al. [19] use a similar model to Figliozzi [14] but with speed as an additional decision variable, though without the use of time windows. Their model is based on a complete network where the nodes represent the depot and customers, while the maximum speeds on the arcs linking the nodes are subject to similar profiles. They describe a tabu search

heuristic for solving the problem and test it on standard benchmark instances. The results suggest that a reduction of about 11.4% in CO₂ emissions can be achieved, but with a 17.1% increase in travel times. Franceschetti et al. [15] follow a similar approach which also takes costs into account in a similar way to Bektaş and Laporte [1]. A mathematical formulation is produced and provides insights on when it is profitable to wait at customers.

The model presented in this paper is in the last category of models which take into account time-dependent conditions and where vehicle speeds are decision variables. It is designed for use on a road network where information is available on the speed of traffic on individual road segments at different times of the day. The solution provides the path to follow between customers and the speeds to be applied on each road segment. It thus provides a more detailed model than the one used by Jabali et al. [19]; the path used between a pair of customers may change depending on the time of travel in our model. Also, it allows time window constraints for serving customers which are not included in Jabali et al. [19].

If it is assumed that the path used between customers is fixed, then some other recent research on speed optimization is relevant. Fagerholt et al. [12] present the Speed Optimization Problem (SOP) in the context of shipping, provide models to formulate the problem and a solution algorithm. Norstad et al. [22] provide a recursive smoothing algorithm for the SOP that runs fast and has been shown to be optimal by Hvattum et al. [18].

It is often the case that reductions can be made in the emissions resulting from a distribution operation, but at the expense of more time or cost. There are methods explicitly aimed at modelling this issue through a multi-objective approach. One example is provided by Jemai et al. [20] where an evolutionary algorithm is used to solve a bi-objective VRP where one objective minimizes total distance, while the other minimizes CO₂ emissions. Demir et al. [5] consider the bi-objective PRP where fuel consumption and driving time are the two relevant objectives.

Finally, there is an emerging strand of research considering vehicle routing problems for alternatively powered vehicles that are designed to be more environmentally friendly.

Such vehicles may have a more limited range before requiring refuelling and there may be a limited availability of refuelling points. An example is given by Erdoğan and Miller-Hooks [11] in which they define a “Green Vehicle Routing Problem” where there are additional constraints on how far the vehicles may travel without refuelling and the refuelling stations are located at specific places. They formulate a mixed integer program to minimise the total distance and develop heuristics for its solution. Tests are carried out based on the location of stations supplying biodiesel fuel in a part of the USA.

3. Problem Description and Modelling

Let $G = (C, A)$ be a graph, where $C = \{c_0, c_1, \dots, c_n\}$ contains a depot (c_0) and a set of customers to serve. We define $A = \{a(c_i, c_j) : c_i, c_j \in C, i \neq j\}$ as the set of arcs connecting customer nodes. For any pair c_i, c_j there may be more than one arc connecting them, representing different paths through the underlying road network. The time horizon for the delivery operation is divided into time slots. For each road segment, the maximum speed allowed by the traffic conditions is given for each time slot. The demand of customer c_i is $d(c_i)$ and the service time for customer c_i is $b(c_i)$, where $b(c_0)$ represents any initial loading time at the depot after the start time. In some problem sets, each customer $c_i \in C$ is associated with a time window $[e(c_i), l(c_i)]$, where $e(c_i)$ and $l(c_i)$ are the lower and upper bound of the time window respectively (with $e(c_0)$ the earliest start time and $l(c_0)$ the latest finish time of the journey). There is a fleet of homogeneous vehicles available, each with capacity Q . A vehicle route starts and finishes at the depot node, and visits a subset of customers in some order. The path between two customers or between the depot and a customer must follow the road segments of the transportation network. The goal is to minimize the total fuel emissions by assigning the customers to routes and ordering the customers in each route subject to the capacity and time window constraints if they apply. In all problems, the waiting time at each customer after completing service is limited to a maximum value, W . There is no restriction on the number of vehicles.

The fuel emissions optimization VRP can be modelled as a set-partitioning problem. Let \mathcal{P} be the set of feasible vehicle routes. For each route, $p \in \mathcal{P}$, the speed of the vehicle on each road segment and the waiting time at each customer are set to minimize the corresponding emissions, f_p , and are determined by a new heuristic approach referred to

as NHA and described in [26]. The quantity α_{ip} is a binary variable used to denote whether customer c_i is visited by a vehicle travelling along route p , where $\alpha_{ip} = 1$ if customer $c_i \in C$ is included in route p , and $\alpha_{ip} = 0$, otherwise. Let y_p be a decision variable to count the number of times route p is used. In a feasible solution, y_p should be 0 or 1 for all $p \in \mathcal{P}$. The formulation is shown as follows.

$$\min \sum_{p \in \mathcal{P}} f_p y_p, \quad (1)$$

$$s. t. \quad \sum_{p \in \mathcal{P}} \alpha_{ip} y_p = 1 \quad \text{for } i = 1, 2, \dots, n, \quad (2)$$

$$y_p \in \{0, 1\} \quad \forall p \in \mathcal{P}. \quad (3)$$

4. Column Generation Based Tabu Search Algorithm

The proposed solution method uses a column generation algorithm that takes advantage of the power of the branch-and-price technique to solve a set partitioning problem. It is based on a branch-and-price-based large neighborhood search algorithm for the Vehicle Routing Problem with Time Windows (VRPTW) proposed by Prescott-Gagnon et al. [24]. The current solution is destroyed in the destruction step by selecting one of four operators randomly. This leaves a set of partial routes and isolated customers. The large neighbourhood then contains all feasible solutions that are compatible with the partial routes. A heuristic column generation algorithm is used to reconstruct the solution, where tabu search is used to generate columns of negative reduced cost.

The formulation of the restricted master problem (RMP) called within the method is described as follows.

$$\text{(RMP)} \quad \min \sum_{p \in \mathcal{P}'} f_p y_p, \quad (4)$$

$$s. t. \quad \sum_{p \in \mathcal{P}'} \alpha_{ip} y_p = 1 \quad \text{for } i = 1, 2, \dots, n, \quad (5)$$

$$y_p \geq 0 \quad \forall p \in \mathcal{P}'. \quad (6)$$

The set $\mathcal{P}' \subseteq \mathcal{P}$ contains feasible routes that include the fixed parts of the solution that have not been removed in the destruction step. We wish to find whether the objective value of the RMP could be further improved by adding any route that has not been

included in \mathcal{P}' . Any route that improves the solution will have a negative reduced cost when added to the RMP. The reduced cost for any route can be calculated using the values of the dual prices associated with the constraints in (5). If no route exists that was not in \mathcal{P}' with a negative reduced cost, then the solution is optimal. Therefore, the sub-problem of the column generation process is to find the column, which satisfies the time window and capacity constraints, with minimum reduced cost. The sub-problem can be formulated as:

$$\min_{p \in \mathcal{P}} f_p - \sum_{i=1}^n \pi_i \alpha_{ip}, \quad (7)$$

where π_i is the dual price associated with constraint i in (5).

A tabu search algorithm is used to solve the sub-problem heuristically, and the goal is to produce feasible routes with negative reduced costs that have not yet been included in set \mathcal{P}' . The algorithm framework is briefly introduced in the following steps.

Step 1: Generate an initial solution with Clarke and Wright Savings Algorithm (parallel version). Go to Step 2.

Step 2: Apply a destroy operator to determine the neighbourhood. Go to Step 3.

Step 3: Solve the RMP, i.e. a set-partitioning problem formulated as an LP relaxation based on the current set of columns (routes). If the stopping rule for the column generation process is met, go to Step 6; otherwise, go to Step 4.

Step 4: Apply tabu search to generate new columns (routes). Go to Step 5.

Step 5: Find the fuel emissions for the new columns (routes) using NHA. Go to Step 3.

Step 6: If the solution is not integer, apply a branching strategy to get an integer solution. While the number of iterations for the algorithm is less than a specified maximum number, go to Step 2; otherwise, stop.

Details of the algorithm corresponding to these steps are discussed in the following sections. Additional details can be found in the PhD thesis [25].

4.1 The initial solution

Define $EC = \{\tilde{f}(c_i, c_j): c_i, c_j \in C, i \neq j\}$, $ED = \{\tilde{d}(c_i, c_j): c_i, c_j \in C, i \neq j\}$ and $ET = \{\tilde{t}(c_i, c_j): c_i, c_j \in C, i \neq j\}$ as matrices of estimated travel cost (fuel emissions), travel distance and travel time from customer c_i to c_j . The initial values of the matrices are based on the weighted average speeds of arcs, where the weights are the lengths of the time slots for which the average speeds apply. The optimal speed under free flow is used, if the weighted average speed is greater than the optimal speed. The optimal speed is the speed of the vehicle that minimizes the fuel emissions per kilometer. Once a speed for each arc is determined, the emissions for each arc can be calculated and treated as a cost for each arc. The optimal path between customers and between depot and customer in terms of fuel emissions is calculated using Dijkstra's algorithm [8] and the corresponding travel cost (fuel emissions), travel distance and travel time are used as the initial values of the elements in EC , ED and ET .

The initial customer assignment is computed by the Clarke and Wright Savings Algorithm (parallel version) [2]. Each customer is assigned to an individual route at the beginning. If the arrival time at customer c_i , which is approximately computed by $\tilde{t}(c_0, c_i)$ is before the early time window $e(c_i)$, unlimited waiting is allowed. Thus, the starting service time at customer c_i denoted as sst_i is, $sst_i = \max \{e(c_i), e(c_0) + b(c_0) + \tilde{t}(c_0, c_i)\}$, and the corresponding waiting time denoted as wt_i is, $wt_i = sst_i - (e(c_0) + b(c_0) + \tilde{t}(c_0, c_i))$. Then routes are combined using the estimated travel costs based on fuel emissions. Capacity and time window constraints are considered when deciding if two routes can be merged.

4.2 The Destroy Operators

There are four destroy operators used to modify the current solution in order to diversify the search. The destroy operators are analogous to the ones presented by Prescott-Gagnon et al. [24] and more details are given in [25]. With each destroy operator, only part of a complete route will be removed.

The large neighbourhood search is a process where each iteration starts with the application of a destroy operator to the current solution. Only one destroy operator is applied at each iteration through a roulette-wheel selection procedure. Each of the destroy

operators is assigned a value ω_i . The destroy operator is chosen randomly at the beginning of each iteration, where each operator i has probability of $\omega_i / \sum_i \omega_i$ to be selected. Following the method proposed by Prescott-Gagnon et al. [24], the initial value of ω_i is set to 5. Whenever operator i is called and its use improves the current solution, the value of ω_i is increased by 1.

4.3 Solving the RMP

The column generation process has already been outlined at the beginning of Section 3. In Step 3 of the algorithm framework, the stopping rule is that either an improved integer solution has been found or the value of the objective in (4) has not been improved for a certain number of iterations.

A solution pool is kept to store the routes associated with the basic variables obtained by solving the last RMP at each iteration of the large neighbourhood search. After a destroy operator is selected and applied, the remaining customers in the route define the fixed part. Then, the routes in the solution pool with the same structure as the fixed part will be added to set \mathcal{P}' . As a result, the reconstruction process will start with a set of prospective columns, some of which may not be obtained by applying tabu search. Moreover, the fuel emissions for these routes do not need to be recalculated, which saves computational time. However, extra time is required to scan the structure of the routes in the solution pool, so the size of the solution pool is limited in order to control the extra time needed to manage it. Columns are removed by following the ‘first in first out’ rule once the solution pool is full. The solution pool can also be thought of as a long-term memory.

4.4 The tabu search algorithm

The tabu search algorithm is used to solve the sub-problem heuristically, and the goal is to produce feasible routes with negative reduced costs that have not yet been included in set \mathcal{P}' . The costs used in the tabu search $f'(c_i, c_j)$ are based on the approximate costs, but they are adjusted by the dual prices, π_i , corresponding to the constraints in the RMP as follows.

$$f'(c_i, c_j) = \tilde{f}(c_i, c_j) - \pi_i, \text{ for } i = 1, 2, \dots, n. \quad (8)$$

Routes associated with the basic variables after solving the RMP are used as initial solutions, and the tabu search approach is applied individually for each of them. Since the reduced costs of the basic variables are zero, they should be good initial solutions when searching for routes with negative reduced costs. The total number of iterations for the tabu search is at most I_{max}^{tabu} . As a result, the number of iterations allowed for each column is set to $\lfloor \frac{I_{max}^{tabu}}{\# \text{ of basic variables}} \rfloor$.

4.4.1 The tabu search neighbourhood

Two simple operators are used to reconstruct new routes, which are removing and inserting a customer. Every time a customer node is removed/inserted, the reverse move is tabu for the next tb^{max} iterations. Hence, at one iteration, a move is defined as removing a (non-tabu) customer and inserting another (non-tabu) customer at a possible insertion place. All possible removals and all possible insertion places are tested to find the best one. Only feasible moves are allowed in the search process, so the capacity and time window constraints have to be checked for each customer insertion and only the time window constraints for customer removal.

4.4.2 The challenge of neighbourhood move evaluation

One of the advantages of applying a heuristic method to solve a VRP in a static network is that only the sub-routes changed by a neighbourhood move have to be re-evaluated. However, in a time-varying road network, the evaluation of a neighbourhood move is not so straightforward. The fuel emissions from any customer c_i to customer c_j may change according to different departure times from customer c_i , so the re-evaluation process is no longer simply an addition and subtraction operation of a few static values relating to the links between customers that have been changed. The time dimension has to be taken into account which leads to the need to re-evaluate significant parts of the affected routes.

Harwood et al. [16] has examined different ways to estimate the cost of a neighbourhood move within a single tour with time-varying traversal speeds, when the objective is to minimize the total time. The tour is divided into three parts according to the nodes being moved: the pre-change part is the tour from the depot until the first node to be changed;

the post-change part is the tour from the last node to be changed back to the depot; and the remaining section is called the changed part. To determine whether a move leads to an improvement, the pre-change part of the tour does not need to be recalculated and provided the First-In-First-Out (FIFO) property holds, the post-change part of the tour does not need to be recalculated either. Furthermore, the necessary and sufficient condition for an overall improvement to be achieved is that the tour from the depot to just before the post-change part should be improved. Therefore, only the changed part of the routes has to be reevaluated to find out whether a neighbourhood move will improve the cost, if the FIFO property is maintained in a time-varying network. However the post-change part of the route still has to be recalculated in order to obtain the overall improvement, once a neighbourhood move has been identified as leading to an improvement.

When the objective is to minimize emissions or costs that depend on the time of travel, then these results do not apply. Suppose the optimal solution has been found from an origin node c_0 to node c_j which passes through c_i . That solution may not contain the optimal solution for a path from c_0 to c_i . This is because it may be better to travel faster/slower than the optimal speed with lower fuel efficiency from c_0 to c_i , so as to arrive at c_i earlier/later to avoid the congestion from c_i to c_j .

Consequently, any neighbourhood change requires the whole route to be re-evaluated to find the exact change in fuel emissions, which makes the problem of minimizing fuel emissions in a time-varying network more difficult to solve. In order to evaluate neighbourhood moves quickly, approximate values of the fuel emissions between customers are used to evaluate each move in the tabu search procedure. Using these approximate values means that only the links that have changed in the neighbourhood move need to be considered in estimating the change in emissions and so can be calculated quickly.

The estimations resulting from the time-varying speeds are improved from one iteration to the next by using a simple exponential smoothing method to update the values of the estimated travel cost (fuel emissions), travel distance and travel time between customers. After NHA has been applied to estimate the cost of a complete route, the values of $\tilde{f}(c_i, c_j)$, $\tilde{d}(c_i, c_j)$ and $\tilde{t}(c_i, c_j)$, where customer c_j follows customer c_i immediately in

the complete route, are updated as follows:

$$\tilde{f}(c_i, c_j) = w_1 \tilde{f}^*(c_i, c_j) + (1 - w_1) \tilde{f}(c_i, c_j) \quad (9)$$

$$\tilde{d}(c_i, c_j) = w_2 \tilde{d}^*(c_i, c_j) + (1 - w_2) \tilde{d}(c_i, c_j) \quad (10)$$

$$\tilde{t}(c_i, c_j) = w_3 \tilde{t}^*(c_i, c_j) + (1 - w_3) \tilde{t}(c_i, c_j) \quad (11)$$

where $\tilde{f}^*(c_i, c_j)$, $\tilde{d}^*(c_i, c_j)$ and $\tilde{t}^*(c_i, c_j)$ are the corresponding fuel emissions, travel time and travel distance from c_i to customer c_j in the complete route; and w_1, w_2 and w_3 are weights with values between 0 and 1. After some testing with different values, w_1, w_2 and w_3 are each set to 0.8 in subsequent runs.

Under the following two circumstances, NHA is applied to evaluate fuel emissions for the current complete route, so that more accurate values can be used:

- The estimated reduced cost of a new column is better than the best solution to the sub-problem associated with the RMP (7) so far. In this case, NHA is used to calculate the actual fuel emissions of the new column, in order to obtain the actual reduced cost. If the reduced cost is smaller than the best solution so far, the aspiration criterion is met, and the tabu restriction is overridden.
- After $\left\lfloor \frac{I_{max}^{tabu}}{\# \text{ of basic variables}} \right\rfloor$ iterations of tabu search have been applied to a column, the fuel emissions of the column will be calculated by NHA. If the reduced cost based on the re-estimated fuel emission is negative, this column is added into set \mathcal{P}' .

If there is an improved integer solution obtained by solving the RMP, the reconstruction process will stop, and a destroy operator will be invoked to determine a new large neighbourhood of the integer solution. Otherwise, the column generation process will stop if the value of (4) cannot be improved in the last I_{max}^{CG} iterations of solving the RMP, where I_{max}^{CG} represents the maximum number of column generation iterations and is a parameter to be set.

4.5 Branching Strategy

When the solution obtained by RMP in the last iteration is not integer, a heuristic branching strategy is applied to derive an integer solution. The branching strategy is to simply fix the decision variable with the largest fractional value at 1, and solve the linear problem again. This process is repeated until the solution is integer or no feasible solution can be found.

If any new integer solution can be obtained by applying the branching strategy, no matter whether it is better or worse than the initial solution at the beginning of the current iteration of large neighbourhood search, the solution will be used for the next iteration. This can help to diversify the search. If there is no feasible integer solution, the initial solution at the beginning of the current iteration of large neighbourhood search is used again in the next iteration.

5. Experiments with Real Traffic Data

The VRP algorithm is tested in this section with real traffic data for a road network located in London.

5.1 Data Description

The data set used includes the locations of 60 stores and a depot. The depot and the stores are located in the southeast of London and the locations are based on those used by a well-known supermarket company. A map showing the locations is provided in Figure 1. Customer nodes 51 to 60 are located in the London congestion charge area. There are 208,488 nodes in the London network, which are linked by 219,880 bidirectional road segments and 37,651 unidirectional road segments. Each bidirectional road segment is replaced by a pair of unidirectional arcs, one in each direction. Hence, the network has 208,488 nodes and 477,411 unidirectional arcs. The distribution of arc lengths has a mean of 92.2m with a maximum of 2848.7m. About 70% of the arcs are shorter than 100m, and only about 9% of arcs are longer than 200m.



Figure 1: Customer and depot locations

Traffic information was supplied from ITIS Holdings. From information obtained by tracking fleets of vehicles in the area, observations of speeds were obtained for all road segments. The 24-hour period during a weekday was divided into 15 different time slots. Within each time slot the observed speeds were relatively stable, but speeds could be very different in different time slots due to the way that traffic congestion builds up at different times of the day. For each road segment, the mean speed in each time slot was taken to be the maximum speed that could apply to any vehicle starting to travel over the road segment within that time slot.

The stores are divided into five sets named A, B, C, D and E, and each set contains 25 stores or customers. Specifically, set A contains customers 1-25, set B contains customers 36-60 and set C contains customers 11-35. In set D, the customers are clustered and set E contains a randomly selected customer set. Both set B and set D include all of the customers located in the congestion charge area. The demands of customers are between 4

and 8 cages. The average service time for unloading one cage is two minutes. For some runs, no particular customer time windows were applied. In these cases, any store may have a delivery between 7am and 5pm which corresponds to the drivers' shift time. For runs where customer time windows are applied, the earliest start times are randomly generated by following a uniform distribution with associated range between 7am and 2pm. The time window intervals are uniformly distributed between 1 and 5 hours. Instances with no time windows are labeled by the set of customers with the addition of “_0”, and the instances with time windows are labeled with the addition of “_1”. The details of the customer demands and time windows are given in Appendix A. Due to London night time regulations for freight deliveries, the starting time from depot is set to be after 7am. The driver shift time is 10 hours, so the finishing time at the depot must be by 5pm. Waiting time at each customer node or the depot is initially set to be a maximum of 5 minutes. A fleet of homogenous vehicles is used in the case study, which are rigid Diesel HGVs >32 t subject to the EURO V emission standard [7]. The corresponding speed-fuel curve is shown in Figure 2. The valid speed range is between 6km/h and 90km/h.

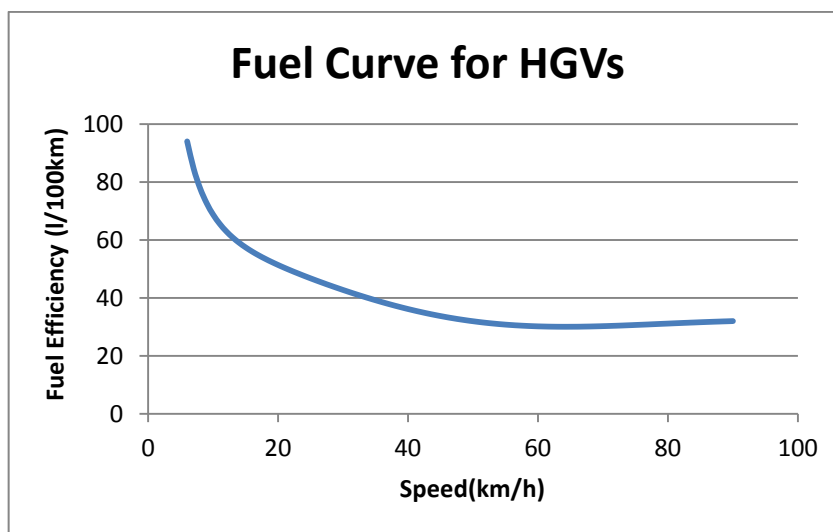


Figure 2: Speed-Fuel Consumption Curve for Euro V Diesel Rigids HGVs >32 Tonnes

Using this fuel consumption curve is an approximation as it does not take into account effects like acceleration, deceleration and the gradients of the roads. However, in the case of road gradients, vehicles depart from and return to the same depot, so there is no overall change in altitude for a vehicle for each route. This means that there will be some cancellation of the effects of uphill gradients requiring more fuel, while downhill

gradients require less fuel. Also no allowance has been made for the changing weight of the load carried. The fuel used in litres is converted to the equivalent greenhouse gas effect of CO₂ in kg by multiplying by a factor of 3.1787. This was the factor proposed by the UK Government in 2010 in its guidance for reporting emissions [6].

5.2 Results Analysis

The approach described in this paper was applied to the instances described and a summary of the results is provided in Appendix B. This shows the sequence of customers on each route, the total CO₂ emissions, the distance travelled, the time required and the total waiting time for each instance. There are two sets of results: one set for instances without customer time windows and one set for instances with time windows.

In the computational experiments, the parameter values used are $I_{\max}^{\text{tabu}} = 100$, $I_{\max}^{\text{CG}} = 30$, and the length of the tabu list $tb^{\max} = 4$. Details concerning the parameter values can be found in Qian [25]. All the results are recorded as the better of two runs. All computations were carried out on the Lancaster high performance computer cluster with quad core Intel Nehalem 2.26 GHz CPU and 24G of memory.

Each vehicle is only allowed to undertake one trip. The results for the instances without customer time windows show that the average trip takes between 2 and 3 hours for each instance. If more than one trip per day were allowed, then the number of vehicles could be reduced, otherwise the vehicles would be available for other duties following the completion of their routes.

The results show that the set of instances with customer time windows require about 20% longer computational running time to produce the solutions compared to the set of instances with no customer time windows. Overall, the solutions for the set of instances with customer time windows produce just over 1% more emissions compared to the set of instances with no time windows, even though a longer waiting time is allowed for the set of instances with customer time windows to ensure feasible solutions.

In the following subsections, various aspects of the problem and solution method are investigated. The first subsection investigates the impact of the speed adjustment and path selection process, waiting time and starting time based on the tests on the five customer

sets without time windows. The speed adjustment and path selection process refers to the path chosen and speeds adopted on the road network in travelling between specific customers. Then, the original method is compared against a simpler method, which generates customer sequences based on static information. Finally, the effect of the time window constraint is examined.

5.2.1 Speed Adjustment & Path Selection

NHA, as described in [26], finds the best path between two customers and the best speeds for the vehicle when leaving the first customer at a particular time. NHA thus has a path selection (PS) process and a speed adjustment (SA) process. The contributions from these different processes are examined in this section. The solutions from three methods are presented in the following table. The column labelled ‘With PS & SA’ shows the solutions obtained by the model proposed in the first part of this paper using NHA; the ‘PS Only’ column shows the solutions obtained by the model without the speed adjustment process; and the last column shows results from the model with neither the path selection process nor the speed adjustment process being used. Without the speed adjustment process, the vehicle is assumed to travel at the speed allowed by the current congestion up to a specified maximum speed, corresponding to the optimal speed of the vehicle; while without the path selection process, the fastest path is used. However, the sequences of customers in all the methods are determined by the approximated fuel emissions matrices.

Scenario Instance	With PS & SA	PS Only	Without PS & SA
A_0	329.5 kg	335.8 kg	337.4 kg
B_0	379.7 kg	382.7 kg	394.8 kg
C_0	344.4 kg	348.5 kg	352.5 kg
D_0	330.0 kg	332.9kg	340.2 kg
E_0	359.7 kg	363.6 kg	371.5 kg

Table 1 Impact on CO2e Emissions of the PS and SA Process

The path selection and speed adjustment processes reveal some contribution to the CO2e emissions reduction. On the average, about 10 kg CO2e emissions can be saved by applying both the path selection and speed adjustment processes, equivalent to a reduction of about 2-3%. In particular, the reduction for B_0 is up to 15 kg, which is 4%

less than 'Without PS & SA'. Compared to the speed adjustment process, the path selection process plays a more significant role in reducing fuel emissions. Around 4 kg extra CO₂e is emitted on average by choosing a promising path and travelling with the fastest speeds allowed by the traffic conditions up to an optimal speed; while in the worst case B_0, another 12 kg emissions is produced by following the fastest paths at the fastest allowed speeds.

One reason that the performance of the speed adjustment process does not have a greater effect could be the shape of the fuel curves used in the experiments. As illustrated in Figure 1, the maximum valid speed is 90km/h, whose fuel efficiency is close to that of the optimal speed 65 km/h; and the fuel efficiency reduces quickly with the decrease of the speed. Therefore, it would not be wise to slow the speed any more than necessary. If the fuel emissions curve is very sensitive to the speeds, or increasing the speed makes the fuel curve steep, then the speed adjustment would be worth applying; otherwise, just carefully selecting a path, then travelling as fast as traffic conditions allow up to an optimal speed still provides a good solution.

5.2.2 Starting Time & Waiting Time

All experiments so far are based on the condition that all the deliveries should start at 7 am, and waiting time at each customer node as well as the depot cannot exceed 5 minutes. Note that 7 am is the start of a peak period, when the road traffic is getting busy. Vehicles may be caught in traffic congestion, which results in extra fuel emissions. The fuel emissions may be improved by allowing longer waiting, or starting after the rush hour. Two scenarios are tested: one is starting at 7 am, but allowing waiting up to 4 hours; and the other is starting at 10 am, and with maximum waiting time being 5 minutes. They are compared to the original solutions with the starting time being 7 am and maximum waiting time being 5 minutes. Results are summarized in Tables 2 and 3.

Starting Instance	7 am (5 min)	7 am (4 hour)	10 am (5 min)
A_0	329.5 kg	322.3 kg	325.0 kg
B_0	379.7 kg	371.8 kg	380.2 kg
C_0	344.4 kg	332.4 kg	338.4 kg
D_0	330.0 kg	324.8 kg	336.7 kg
E_0	359.7 kg	348.4 kg	356.8 kg

Table 2 Results of Different Starting Times & Maximum Waiting Times

Starting Instance	7 am (5 min)	7 am (4 hour)
A_0	1.0h	22.3h
B_0	0.4h	17.6h
C_0	0.7h	30.3h
D_0	0.6h	16.7h
E_0	1.1h	25.7h

Table 3 Total Waiting Time (hours) when starting at 7am

For each instance, 7 vehicle routes are required to serve all customers. By allowing more waiting time, about 2-3 % CO₂e can be saved. However, it is at the expense of 16 to 30 hours' total waiting time. In instance C₀, for example, one complete route starts from the depot, then serves customers 2, 7, 1, 17 and returns back to the depot. By allowing a longer waiting time, the vehicle and driver wait at customer 2 for about 1.8 hours and waits at customer 1 for about 3 hours to avoid peak time congestion. As a result, the total CO₂e for the route is reduced from 85kg to 81kg. In practice, the effect on costs of the extra waiting time will depend on how the drivers are being paid and whether any of the additional waiting time coincides with their breaks. For instance for B₀ and D₀, the improvement by waiting is smaller compared with others, but the total waiting time is also shorter, which is about 17 hours while others are more than 20 hours.

When the starting time is changed to 10 am and the morning peak is avoided, results from instances A₀, C₀ and E₀ show that about 1-2% CO₂e can be saved. However the results for B₀ and D₀ are slightly worse than those for a starting time of 7 am. This may be because both instances contain 10 customers located in the congestion charge area,

and the traffic flows in the city centre keep busy during the day time. It may also be because more travel is moved towards the evening peak when traffic is more congested.

In practice, the starting time is restricted by many factors, such as local authority regulations and the driver shift times. Attempts may be made to provide night time deliveries to avoid daytime congestion, but other problems like noise, safety and out-of-hours deliveries may mean that night time deliveries are not feasible.

5.2.3 Distance Based Approach

In the original method, the customer sequence is determined by approximate fuel emission matrices, the values of which are updated from one iteration to the next. This requires repeated application of NHA within the method and results in high run times as shown in Appendix B. In order to reduce the computational burden, a simpler method is proposed for comparison that only considers static information when determining the vehicle assignment and the customer sequence. The VRP is first solved by minimizing the total distance, i.e. using the distance matrices within the column generation based tabu search algorithm. Then, NHA is applied to decide on speeds and paths for the pre-determined complete routes. In other words, the time-varying speeds are not considered until the sequences of customers have been decided. A comparison between the original method and the distance based approach (DBA) in terms of solution quality as well as computational time is shown in Table 3.

Instances		Original Method	DBA
A_0	Objective Value	329.5 kg	331.4 kg
	Running Time	19h 58m 04s	5m 18s
B_0	Objective Value	379.7 kg	385.8 kg
	Running Time	22h 35m 49s	6m 32s
C_0	Objective Value	344.4 kg	346.6 kg
	Running Time	21h 55m 01s	5m 57s
D_0	Objective Value	330.0 kg	332.7 kg
	Running Time	20h 46m 33s	5m 26s
E_0	Objective Value	359.7 kg	360.8 kg
	Running Time	21h 17m 48s	5m 54s

Table 4 Results of the Original Method and DBA on Instances without Time Window Constraints

As shown in the table, DBA is able to obtain a solution close to the original method for instances A_0, C_0, D_0 and E_0. The performance of DBA for instance B_0 is not as good as for other instances, but it is only 1.6 % worse than the solution obtained by the original method. In general, minimizing distance seems to be a good criterion for obtaining a good sequence of customers, before using NHA to find the detailed paths and speeds to be used when travelling between customers.

Meanwhile, the running time for DBA is much less than the original method. The running time of DBA is about 5-6 minutes, while that of the original method is more than 20 hours. This is because NHA is only used at the end of the process, after the customers have been allocated to routes and their sequences have been determined. DBA is able to get solutions that are very close to those obtained by the original method, but it needs less than 0.5% of the running time of the original method.

For the instances without time window constraints, DBA provides good solutions in terms of running time as well as solution quality. However, DBA has the potential risk of missing time windows by using static travel time data in scheduling the customer sequence. The impact of the time window constraint will be discussed in the next section.

5.2.4 Time Window Constraints

All the above discussion is based on the instances without time window constraints, but only with 7 am as the earliest starting time and 5pm as the latest finish time for the drivers' working shift. In this section, the instances with time windows are tested. In order to guarantee a feasible solution for each instance, the maximum waiting time is set at 4 hours. The following table shows results obtained by the original method as well as the simpler method.

Instances		Original Method	DBA
A_1	Objective Value	342.5 kg	370.2 kg
	Running Time	24h 12m 05s	7m 34s
B_1	Objective Value	375.4 kg	No feasible solution
	Running Time	28h 46m 21s	8m 28s
C_1	Objective Value	354.3 kg	357.9 kg
	Running Time	25h 41m 21s	8m 14s
D_1	Objective Value	333.3 kg	334.0 kg
	Running Time	24h 20m 17s	7m 20s
E_1	Objective Value	360.0 kg	366.7 kg
	Running Time	24h 55m 29s	7m 52s

Table 5 Results of the Original Method and DBA on Instances with Time Window Constraints

We note that DBA has again found solutions close to those obtained by the original method for instances C_1, D_1 and E_1 within a relatively short running time. However, the performance of DBA on A_1 is about 8% worse; and it cannot even find a feasible solution for instance B_1. DBA does not consider the time-varying speeds when scheduling the vehicle assignment and the customer sequence, and the travel time may be underestimated. Consequently, there is a risk that a vehicle has to speed up and follow a higher-emission strategy in order to satisfy the time window constraints or even cannot meet the time window constraints, when the time-varying speeds are taken into account.

6. Conclusions

In this paper, a column generation based tabu search algorithm is proposed to solve a vehicle routing problem where there are time-varying speeds and the objective is to minimize fuel emissions. The main findings can be summarized as follows:

- The proposed algorithm can produce sets of routes for an urban distribution problem with a reduction in GHG emissions of about 3% compared with an approach where the objective is to minimize total time.
- Within NHA, the path selection process plays a more significant role in reducing fuel emissions, when compared with the speed adjustment process. When the speeds are allowed to be as fast as the traffic allows up to a maximum speed, the best fuel

emission solution is only slightly worse than the one obtained using NHA, where the speeds are decision variables.

- By allowing a long waiting time at customer nodes, vehicles can avoid being caught in congestion, and the fuel emissions can be reduced. However, the costs of such waiting, such as drivers' salaries, have to be considered. An alternative way to attempt to avoid congestion is to depart later to avoid the morning peak period, but this may not significantly reduce the fuel emissions.
- Minimizing distance can be an effective way to obtain a good sequence of customers. DBA, which determines the customer sequences by minimizing distance, has a good performance on instances without time windows, but there is a risk that it could fail to find a feasible solution, when it is applied with time-varying speeds. In this case, some simple repair strategy could be used to obtain a good feasible solution.

Acknowledgement

The first author was sponsored by the LANCS Initiative (EPSRC); Grant number: EP/F033613/1.

References

- [1] Bektaş, T., & Laporte, G. (2011). The pollution-routing problem. *Transportation Research B*, 45: 1232–1250.
- [2] 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.
- [3] Demir, E., Bektaş, T., & Laporte, G. (2011). A comparative analysis of several vehicle emission models for road freight transportation. *Transportation Research D*, 16: 347–357.
- [4] Demir, E., Bektaş, T., & Laporte, G. (2012). An adaptive large neighborhood search heuristic for the Pollution-Routing Problem. *European Journal of Operational Research*, 223: 346–359.
- [5] Demir, E., Bektaş, T., & Laporte, G. (2014). The bi-objective Pollution-Routing Problem. *European Journal of Operational Research*, 232: 464–478.

- [6] Department for Environment Food and Rural Affairs, Guidelines to Defra / DECC's greenhouse gas conversion factors for company reporting, [online], 2010, Available at: <http://archive.defra.gov.uk/environment/business/reporting/conversion-factors.htm> (Accessed 28 October 2014).
- [7] Department for Transport, Road vehicle emission factors 2009: regulated (online) (2009). Available at: <https://www.gov.uk/government/publications/road-vehicle-emission-factors-2009> (Accessed 28 October 2014).
- [8] Dijkstra, E.W. (1959). A note on two problems in connexion with graphs. *Numerische Mathematik*, 1.
- [9] Eglese, R., & Bektaş, T. (2014). Green Vehicle Routing. In P. Toth, & D. Vigo (Eds.), *Vehicle Routing: Problems, Methods, and Applications (2nd edition)* (pp. 437-458). Philadelphia : MOS/SIAM Ser. Optim. 18., SIAM.
- [10] Eglese, R.W., Maden, W., & Slater, A. (2006). A Road TimetableTM to aid vehicle routing and scheduling. *Computers & Operations Research*, 33: 3508–3519.
- [11] Erdoğan, S., & Miller-Hooks, E. (2012). A green vehicle routing problem. *Transportation Research Part E: Logistics and Transportation Review*, 48: 100–114.
- [12] Fagerholt, K. Laporte, G., & Norstad, I. (2010). Reducing fuel emissions by optimizing speed on shipping routes. *Journal of the Operational Research Society*, 61: 523–529.
- [13] Figliozzi, M.A. (2010). Vehicle routing problem for emissions minimization. *Transportation Research Record: Journal of the Transportation Research Board*, 2197: 1–7.
- [14] Figliozzi, M.A. (2011). The impacts of congestion on time-definitive urban freight distribution networks CO₂ emission levels: Results from a case study in Portland, Oregon. *Transportation Research Part C: Emerging Technologies*, 19: 766–778.
- [15] Franceschetti, A., Van Woensel, T., Honhon, D.B.L.P., Bektaş, T., & Laporte, G. (2012). The Time-Dependent Pollution Routing Problem. *Technical Report*, School of Industrial Engineering, Eindhoven University of Technology, Eindhoven, The Netherlands.

- [16] Harwood, K., Mumford, C., & Eglese, R. (2013). Investigating the use of metaheuristics for solving single vehicle routing problems with time-varying traversal costs. *Journal of the Operational Research Society*, 64: 34-47.
- [17] Hickman, J., Hassel, D., Jourard, R., Samaras, Z., & Sorenson, S. (1999). MEET methodology for calculating transport emissions and energy consumption, *Technical Report, European Commission, DG VII, Luxembourg*. Available at: <http://www.transport-research.info/Upload/Documents/200310/meet.pdf> (accessed on August 5, 2013).
- [18] Hvattum, L.M., Norstad, I., Fagerholt, K., & Laporte, G. (2013). Analysis of an exact algorithm for the vessel speed optimization problem. *Networks*, 62: 132–135.
- [19] Jabali, O., van Woensel, T., & de Kok, A.G. (2012). Analysis of travel times and CO₂ emissions in time-dependent vehicle routing. *Production and Operations Management*, 21:1060–1074.
- [20] Jemai, J., Zekri, M., & Mellouli, K. (2012). An NSGA-II algorithm for the green vehicle routing problem. In J.K. Hao and N. Middendorf (Eds.) *Evolutionary Computation in Combinatorial Optimization* (pp. 37–48). Berlin: Vol. 7245 of Lecture Notes in Computer Science, Springer-Verlag.
- [21] Maden, W., Eglese, R.W., & Black, D. (2010). Vehicle routing and scheduling with time-varying data: A case study. *Journal of the Operational Research Society*, 61: 515–522.
- [22] Norstad, I. Fagerholt, K., & Laporte, G. (2011). Tramp ship routing and scheduling with speed optimization, *Transportation Research C*, 19: 853–865.
- [23] Palmer, A. (2007). *The Development of an Integrated Routing and Carbon Dioxide Emissions Model for Goods Vehicles*, PhD thesis, Cranfield University, School of Management, Bedford, UK.
- [24] Prescott-Gagnon, E., Desaulniers, G., & Rousseau, L. M. (2009). A branch-and-price-based large neighborhood search algorithm for the vehicle routing problem with time windows. *Networks*, 54(4): 190-204.
- [25] Qian, J. (2012). *Fuel Emission Optimization in Vehicle Routing Problems with Time-Varying Speeds*. PhD Dissertation, Lancaster University.
- [26] Qian, J. & Eglese, R. (2014). Finding least fuel emission paths in a network with time-varying speeds. *Networks*, 63:96–106.

[27] Scora, G. & Barth, M. (2006) Comprehensive modal emission model (CMEM), version 3.01, user's guide, Technical Report, Center for Environmental Research and Technology, University of California, Riverside. Available at:

http://www.cert.ucr.edu/cmем/docs/CMEM_User_Guide_v3.01d.pdf (accessed on August 5, 2013).

[28] Suzuki, Y. (2011). A new truck-routing approach for reducing fuel consumption and pollutants emission. *Transportation Research D*, 16: 73–77.

[29] Ubeda, S., Arcelus, F.J., & Faulin, J. (2011). Green logistics at Eroski: A case study. *International Journal of Production Economics*, 131: 44–51.

[30] van Woensel, T., Creten, R., & Vandaele, N. (2001). Managing the environmental externalities of traffic logistics: The issue of emissions. *Production and Operations Management*, 10: 207–223.

Appendix A

Customer Data

The demands (cages), service times (minutes), and time windows (hours) of the 60 customers as well as one depot are shown in the following table.

Nodes	Demands	Service Times	Time Windows	
			Earliest	Latest
0(Depot)	0	0	7	17
1	8	16	11.6	12.6
2	7	14	7.1	11.2
3	6	12	10.8	15.1
4	8	16	8.5	10.2
5	7	14	12.1	14.9
6	6	12	9.6	12.5
7	6	12	14	17
8	4	8	9.9	13.6
9	7	14	14	16.7
10	6	12	7.3	8.5
11	6	12	8.6	9.8
12	4	8	9.2	13.8
13	8	16	11.9	15.3
14	6	12	9.6	11.1
15	5	10	9.2	11.1
16	4	8	12.5	14.5
17	5	10	9.1	12.3
18	5	10	13.8	16.5
19	4	8	13.4	16.6
20	6	12	13.4	14.9
21	5	10	13.1	16.9
22	5	10	8	10
23	6	12	7.4	10.7
24	7	14	8.4	10.8
25	4	8	11.3	12.9
26	8	16	14	17
27	6	12	7.1	11.3
28	7	14	11.8	13.4
29	6	12	9.3	11.2
30	6	12	10.4	12
31	6	12	8.4	11.2
32	5	10	12.5	16.7
33	8	16	9.8	11.1
34	5	10	8.4	13.3
35	4	8	13.8	17
36	5	10	10.4	13.8
37	7	14	7.8	10.2

38	7	14	8	10
39	4	8	9.8	10.9
40	6	12	12.2	14
41	7	14	11.2	13.2
42	6	12	9.5	13.1
43	6	12	11.4	13.7
44	7	14	13.8	16.5
45	7	14	10.5	15.2
46	4	8	10.7	15.7
47	7	14	7.2	11
48	5	10	8.6	11.1
49	8	16	12.3	15.8
50	4	8	10.5	13.2
51	6	12	12.9	15.7
52	6	12	13.5	16.5
53	4	8	8	11.2
54	5	10	13.6	17
55	5	10	9.4	11.6
56	8	16	13.9	16.2
57	6	12	9.8	12.9
58	5	10	13.3	15.3
59	6	12	8.7	10
60	8	16	7.3	12

For five customers set, A contains customer 1-25; B contains customer 36-60; C contains customer 11-35; D contains customer 1,2,13,18,21,24,25,30,31,34,35,38,43,44,48,51-60; and E contains customer 2,6,7,13-16,18,19,23,27-30,35,38,41,42,47,48,50,51,54-56.

Appendix B

Results of Experiments

The results of the instances A_0, B_0, C_0, D_0 and E_0 are shown as follows. The starting time is 7 am, and the maximum waiting time is 5 minutes.

Instance Name	A_0	B_0	C_0
Routes	0-4-0	0-3-13-2-1-0	0-2-7-1-17-0
	0-8-12-17-11-15-0	0-9-8-7-4-0	0-8-24-20-25-0
	0-21-6-5-20-0	0-16-5-19-12-0	0-9-6-5-22-4-0
	0-2-1-7-0	0-17-21-14-0	0-11-19-18-10-0
	0-18-25-24-13-0	0-18-24-25-20-0	0-13-21-12-15-0
	0-9-22-23-3-0	0-22-23-0	0-14-23-3-0
	0-19-16-10-14-0	0-6-10-11-15-0	0-16-0
Total CO2e	329.5 kg	379.7 kg	344.4 kg
Total Distance	231.8 km	244.2 km	240.8 km
Total Time	15.1 h	16.7 h	15.5 h
Total Waiting Time	1.0 h	0.4 h	0.7 h
Running Time	19h58m04s	22h35m49s	21h55m01s

Instance Name	D_0	E_0
Routes	0-5-4-0	0-13-9-12-2-0
	0-3-6-12-0	0-19-23-14-15-0
	0-15-7-1-2-0	0-20-3-18-1-0
	0-11-8-17-22-0	0-7-11-6-5-0
	0-10-16-21-23-0	0-8-17-10-0
	0-19-14-9-13-0	0-21-4-16-0
	0-18-24-25-20-0	0-25-24-22-0
Total CO2e	330.0 kg	359.7 kg
Total Distance	213.9 km	241.7 km
Total Time	15.8 h	16.8 h
Total Waiting Time	0.6 h	1.1 h
Running Time	20h46m33s	21h17m48s

The results of the instances A_1, B_1, C_1, D_1 and E_1 are shown as follows. The starting time is 7 am, and the maximum waiting time is 4 hours.

Instance Name	A_1	B_1	C_1
Routes	0-4-0	0-3-13-2-1-0	0-4-5-6-9-0
	0-10-15-16-19-21-0	0-4-7-8-9-0	0-21-13-12-15-0
	0-6-5-20-18-0	0-12-5-14-0	0-17-1-7-2-0
	0-2-1-3-0	0-15-17-16-21-0	0-19-11-3-0
	0-11-17-12-8-25-0	0-18-24-25-20-0	0-18-10-16-0
	0-22-23-7-9-0	0-22-23-0	0-14-23-22-0
	0-14-24-13-0	0-6-10-11-19-0	0-24-20-25-8-0
Total CO2e	342.5 kg	375.4 kg	354.3 kg
Total Distance	242.3 km	243.9 km	248.5 km
Total Time	52.3 h	50.9 h	49.4 h
Total Waiting Time	37.9 h	34.7 h	35.1 h
Running Time	24h12m05s	28h46m21s	25h41m21s

Instance Name	D_1	E_1
Routes	0-5-4-0	0-13-5-7-9-0
	0-9-13-14-19-0	0-19-23-22-15-0
	0-15-2-1-7-0	0-20-10-17-3-0
	0-12-6-3-0	0-1-18-11-6-0
	0-22-23-21-11-0	0-12-2-21-0
	0-8-10-16-17-0	0-16-4-0
	0-18-24-25-20-0	0-14-24-25-8-0
Total CO2e	333.3 kg	360.0 kg
Total Distance	215.5 km	245.5 km
Total Time	52.2 h	57.5 h
Total Waiting Time	37.2 h	43.3 h
Running Time	24h20m17s	24h55m29s