

Research Article

Time-Dependent Vehicle Routing of Free Pickup and Delivery Service in Flight Ticket Sales Companies Based on Carbon Emissions

Jingmei Guo¹ and Chao Liu²

¹*Institute of Systems Engineering, Northeastern University, Shenyang 110004, China*

²*Institute of Systems Science, Northeastern University, Shenyang 110004, China*

Correspondence should be addressed to Jingmei Guo; guojingmei.007@163.com

Received 29 June 2016; Accepted 16 August 2016; Published 10 January 2017

Academic Editor: Jose E. Naranjo

Copyright © 2017 J. Guo and C. Liu. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The time-dependent pollution-routing problem of free pickup and delivery of passengers to the airport service (TDFPDS) is an extension of the time-dependent pollution-routing problems, arising in flight ticket sales companies for the service of free pickup and delivery of airline passengers to the airport. The problem consists of routing a fleet of vehicles in order to deliver a set of customers to the airport considering the traffic congestion, time window constraints, and maximum ride time constraints. The cost function includes fuel consumption and driver costs. We provide an analytical characterization of the optimal solutions for a fixed route and propose a novel heuristic for a given route based on the analysis of the illustrative examples. The heuristic algorithm is embedded into a set-partitioning model to produce high-quality routing plans. Finally, using wide variety of random instances, we present results on the computational performance of the heuristic and also on the impact of the congestion and the maximum ride time constraints.

1. Introduction

Green Logistics has recently received closed attention from governments and business organizations. The importance of Green Logistics is motivated by the fact that current production and distribution logistics strategies are not sustainable development in the long term due to the negative impacts on the environment and the ecology. It is no doubt that transportation accounts for the major part of logistics. The trend is that more and more countries will start to adopt emission-reducing actions, build green intelligent transportation systems, and exploit other strategies to obtain Green Logistics, which enhances the importance of vehicle routing problems with environmental considerations.

The studies of routing problems concerning the change of transportation systems are mainly divided into two topics. One aspect makes use of new energy sources, named alternative fuel vehicles, such as electric cars and liquid natural gas,

to complete the transportation of the goods, which is denoted as Green Vehicle Routing Problems (GVRP). The main issue of GVRP is that it contains not only the algorithms of routing and scheduling but also the key procedures of recharge due to the limited energy capacity, as described in Felipe et al. [1] and Montoya et al. [2]. Another topic is committed to the reasonable arrangement of paths and selection of speeds to reduce carbon emissions or fuel consumption, since better utilization of vehicles and a cost effective vehicle routing solution would more directly achieve sustainable transportation schemes (Lin et al. [3]). A formulation of fuel consumption is provided in Xiao et al. [4]. They proposed a Fuel Consumption Rate (FCR) to denote the fuel consumption per unit distance as a linear function associated with the vehicle's load, which extends to Capacitated Vehicle Routing Problems (CVRP) with the objective of minimizing fuel consumption. Besides the transportation distance and the loading weight, Kuo [5] adds the transportation speed to

the fuel consumption calculation model in time-dependent Vehicle Routing Problems (VRP). Pollution-routing problem (PRP) introduced by Bektaş and Laporte [6] converts the traditional objective function of distance or duration to carbon emissions and driver costs. PRP consists of routing a number of vehicles to serve a set of customers within preset time windows and determining their speed on each arc of the route to minimize the total cost. This problem seeks optimal travel routes and speeds subject to time constraints, the combination of two intricate sets of decision variables making it more complicated than VRP with time windows (VRPTW). PRP is solved mainly by metaheuristic for medium and large cases (Demir et al. [7]; Kramer et al. [8]). Franceschetti et al. [9] integrate the PRP with time-dependent VRP (TDPRP) as a more realistic approach to deal with the minimization of emissions considering the impact of congestion. More details about research of the Green Transportation can be seen in the survey of Lin et al. [3].

Free pickup and delivery to the airport is a personalized service from flight ticket sales companies for passengers to buy tickets from the company firstly introduced by Dong et al. [10] and denoted as free pickup and delivery of passengers to the airport service (FPDS). The FPDS consists of routing a number of vehicles to serve customers with their desired delivery time so as to balance the trade-off between the company's cost and the passenger satisfaction degree. The FPDS is an extension of VRPTW and has typical characteristics such as small vehicle capacity and time windows restricted in the unified destination (airport). Flight ticket sales companies supply the service in order to compete in the market on the premise of cost savings. The pollution-routing problem (Bektaş and Laporte [6]) converts the carbon emissions to the fuel consumption, which considers the environmental impact but also the fuel cost. As a result, the fuel cost association with the driver wages cost as the ticket companies' cost is an appropriate measurement compared with the total distance traveled, especially in the congestion case.

We introduce the time-dependent routing problem of free pickup and delivery of passengers to the airport service (TDFPDS), which extends the FPDS by explicitly taking into account traffic congestion, the environmental implication, and the reasonable cost measurement. A set-partitioning model is introduced as the basic routing algorithm and the main contribution is focused on the feasibility check and schedule of a fixed route with the vehicle's departure time and speeds. We describe a mixed integer linear programming formulation for a given route where the vehicle speed of each leg of the route is optimized among a set of discrete values. Through the numerical analysis solved by the formulation, we also propose a heuristic algorithm to solve medium- and large-sized problems with near-optimal schedule for fixed route. Finally, we report computational experiments with the formulation and the heuristic algorithm on simulation instances.

The contribution of this paper can be stated as follows: (i) we introduce time-dependent travel time, which is the first time that such constraint is addressed in FPDS, to our

knowledge, (ii) we also consider the maximum ride time constraints in the TDFPDS, which are not considered in the VRPTW, as the main difference between the TDPRP and TDFPDS, (iii) we demonstrate that in the optimal schedule speeds on each leg of the route are the same, and hence an effective heuristic algorithm is proposed to schedule the route with complex maximum ride time constraints and traffic congestion.

The remainder of the paper is structured as follows. The next section presents a formal description of each composition of the TDFPDS and the general model of it. Section 3 provides departure time and speed optimization procedure on a fixed route by integer linear programming formulation, illustrative examples to motivate the problem, and a heuristic to solve the fixed route schedule in the last. Section 4 describes the set-partitioning model routing the TDFPDS. Computational results with the proposed exact and heuristic formulation are presented in Section 5. Conclusions follow in Section 6. For the sake of conciseness, proofs and the optimal departure time results of the heuristic are provided in Appendices A and B.

2. Problem Description

The TDFPDS is defined on a complete graph $G = \{N, A\}$, where N is the set of nodes, 0 is the depot, V is the airport, $N_0 = N \setminus \{0, V\}$ is the set of orders, and A is the set of arcs between each pair of nodes. The distance between two nodes i and j is denoted by d_{ij} . A homogeneous fleet of K vehicles, each with a capacity of Q customers, is available to serve all orders. Each node is visited exactly once by one vehicle. Each order $i \in N_0$ is associated with a number of passengers q_i , a desired delivery time window $[E_i, L_i]$, and a maximum ride time MRT_i which represent the service quality. Without loss of generality the service time of each order is set equal to zero as the boarding time is short enough compared with the travel time. The following sections present the way in which time-dependency and congestion are modeled and how CO_2 emissions are calculated and elaborate the complete model of the TDFPDS in the last.

2.1. Congestion and Time-Dependency. In the case of time-dependent travel time, the speed model is based on the key elements of peak hour traffic congestion. The travel time of a vehicle mainly depends on distance and the departure time of the vehicle because the speed is constrained by periods of traffic congestion. Without loss of generality we assume that the vehicles depart from the depot at time zero. Here, we model traffic congestion using a two-level speed function as in Jabali et al. [11] and Franceschetti et al. [9]. It is assumed that there is an initial period of congestion, lasting a units of time, followed by a period of free-flow. In the peak-period, the vehicle travels at a limited speed v_c whereas in the free-flow period that follows it is only constrained by the speed limits v_{\min} and v_{\max} ; that is, the vehicle drives at free-flow speed $v_f \in [v_{\min}, v_{\max}]$. The speed v_c and the time a are constants which can be extracted from

archived travel data and we assume that the same values hold between each pair of locations. Let $T(w, v_f)$ denote the travel time of a vehicle between two locations of distance d as a function of its departure time w from the first location and the chosen free-flow speed v_f . It can be computed using the following formulation proposed by Jabali et al. [11]:

$$T(w, v_f) = \begin{cases} \frac{d}{v_c}, & \text{if } w \leq \left(a - \frac{d}{v_c}\right)^+, \\ a + \frac{d - v_c(a - w)}{v_f}, & \text{if } \left(a - \frac{d}{v_c}\right)^+ < w < a, \\ \frac{d}{v_f}, & \text{if } w \geq a, \end{cases} \quad (1)$$

where $(x)^+ = \max\{x, 0\}$.

2.2. Calculating Emissions. Our modeling of emissions follows the same approach as in Bektaş and Laporte [6] and Franceschetti et al. [9], where the fuel use rate is used as a proxy to estimate the amount of CO₂ emissions and the fuel consumption is calculated to model the emissions.

Given the departure time w and the free-flow speed v_f on a road of distance d , in the time-dependent setting, the travel time $T(w, v_f)$ is split into two terms: the time spent by the vehicle in congestion, denoted as $T^c(w)$, and the time spent in the free-flow speed, denoted as $T^f(w, v_f)$. The total amount of fuel consumption $TC(w, v_f)$ can be written as

$$\begin{aligned} TC(w, v_f) &= k_1 T(w, v_f) \\ &+ k_2 \left[T^c(w) v_c^3 + T^f(w, v_f) v_f^3 \right] \\ &+ k_3 (\mu + f) d, \end{aligned} \quad (2)$$

where k_1 , k_2 , and k_3 are parameters based on fuel properties, vehicle, and the road network characteristics (Demir et al. [7]) and μ and f are the curb weight and the weight of the persons in the vehicle.

2.3. Description of the TDFPDS. In this section, we briefly describe the objective and constraints of the TDFPDS, which could help us to understand the problem better and work out an effective method to solve it. We follow the precious work of pollution routing which shows that the total travel cost function is composed of the cost of the vehicle emissions and the driver cost for each trip instead of the traditional total distance multiplying unit cost. As the CO₂ emissions cost is not easy to be quantified, we assume that the CO₂ emissions cost is equal to the fuel

cost similar to Franceschetti et al. [9], denoted as f_e . The vehicle's fuel consumption is related to the emissions [11]; thus, a conversion factor h is used to calculate the vehicle's fuel consumption. Then, the unit cost of the fuel consumption and the emissions is calculated as $f_c = f_e(1 + h)$ per liter. Thus, the cost is especially suitable for the FPDS, since it adopts fuel cost and driver wage cost which satisfy the cost saving requirements of the company but also decrease the carbon emissions for environmental considerations. The driver cost is calculated from the beginning of the trip until arriving at the airport. Let f_d denote the wage rate for the drivers.

The aim of the TDFPDS is to determine a set of routes, starting at the depot, ending at the airport, with free speeds on each road of the routes and departure times from each node to minimize the total travel cost, under the constraints of vehicle capacity, time window, and the maximum ride time.

3. Departure Time and Speed Optimization on Fixed Route

In this section, we consider a given route in which the customer nodes and sequence to be visited are fixed. Let $(0, 1, \dots, n, n + 1)$ be the fixed sequence of nodes. Node 0 is the depot and node $n + 1$ is the airport. Let d_i denote the distance on arc $(i, i + 1)$ with $0 \leq i \leq n$. The main job of this section is to determine departure time and free-flow speed of each arc in the route so as to minimize the total travel cost, denoted as DSP.

The following sections will present a mixed linear integer programming for DSP, extensive examples to illustrate the characteristics of the solutions, and a heuristic algorithm to solve the DSP.

3.1. A Mixed Linear Integer Programming for DSP. This section presents a mathematical formulation for DSP. Firstly, we simplify the time window constraints. Since customers in a route arrive at the airport at the same time, the vehicle should arrive at the airport in $[l, u]$, where $l = \max\{E_i \mid i = 1, \dots, n\}$ and $u = \min\{L_i \mid i = 1, \dots, n\}$. Obviously, if $l > u$, the route is infeasible. In order to clarify the effect of waiting time to the solution, waiting time is allowed in the model. As the free-flow speed is continuous, the objective function and the constraints are not linear. To linearize it, we discretize the free-flow speed following the approach used by Bektaş and Laporte [6] and Franceschetti et al. [9]. Here we provide a brief exposition for the sake of well understanding. Let v_1, \dots, v_k denote the different free-flow speeds where $v_{\min} = v_1 \leq \dots \leq v_k = v_{\max}$. To rewrite the travel time as a linear equation, the following definitions are given. Let $b_{i0} = 0$, $b_{i1} = (a - d_i/v_c)^+$, $b_{i2} = a$, and $b_{i3} = \infty$. Then $[b_{i,m-1}, b_{i,m})$ denotes the m th time interval of arc $(i, i + 1)$, where $m = \{1, 2, 3\}$. Let V_{mr} denote the vehicle speed in time region m given free-flow speed v_r ; that is, $V_{1,r} = v_c$, $V_{2,r} = v_c$, and $V_{3,r} = v_r$. These definitions allow us to rewrite (1) for arc $(i, i + 1)$ as

$T(w, v_r) = \theta_{mr}w + \eta_{imr}$ if $w \in [b_{i,m-1}, b_{i,m})$ with free-flow speed v_r , where

$$\theta_{mr} = \begin{cases} 0, & \text{if } m = 1, 3, \\ \frac{V_{2r} - V_{3r}}{V_{3r}}, & \text{if } m = 2, \end{cases}$$

$$\eta_{imr} = \begin{cases} \frac{d_i}{V_{1r}}, & \text{if } m = 1, \\ \frac{d_i}{V_{3r}} + \theta_{mr}a, & \text{if } m = 2, \\ \frac{d_i}{V_{3r}}, & \text{if } m = 3. \end{cases} \quad (3)$$

The model uses the following decision variables: z_{imr} is binary variable equal to 1 if the vehicle leaves node i to node $i + 1$ within time interval $[b_{i,m-1}, b_{i,m})$ with free-flow speed v_r , w_{imr} is equal to the time instant at which the vehicle leaves node i to node $i + 1$ within time interval $[b_{i,m-1}, b_{i,m})$ with free-flow speed v_r , where $i = 0, 1, \dots, n$, $m = 1, 2, 3$, and

$r = 1, \dots, k$. w_{n+1} is equal to the time instant at which the vehicle arrives at the airport.

A mixed integer linear programming formulation for the DSP is given below. The first three parts of the objective function represent the total cost of the fixed route. In particular, (4)-(5) compute the cost induced by the engine module and the speed module. Term (6) measures the cost induced by the driver wage and the weight module, where $F = \sum_{i=0}^n f_c k_3 (\mu + i\nu) d_i$ is the cost induced by the weight module, which is constant when the route is determined. Constraints (7)-(8) are used to express the temporal relationship between arrival time and departure time, with inequality constraints indicating that wait is allowed in the trip. Constraints (9) and (10) guarantee that the departure time of node i belongs to exactly one of the three intervals. The time window restriction at the airport is imposed by constraint (11). The maximum ride time restriction at each node is imposed by constraint (12). Finally, constraint (13) enforces the integrality and nonnegativity restrictions on the variables. We provide a numerical analysis of the performance of this formulation in the following section and Section 5.

$$\text{minimize} \quad f_c k_1 \sum_{i=0}^n \sum_{m=1}^3 \sum_{r=1}^k (\theta_{mr} w_{imr} + \eta_{imr} z_{imr}) + \sum_{i=0}^n \sum_{m=1}^3 \sum_{r=1}^k f_c k_2 v_{mr}^3 (\theta_{mr} w_{imr} + \eta_{imr} z_{imr}) \quad (4)$$

$$+ \sum_{i=0}^n \sum_{r=1}^k f_c k_2 v_{2r}^3 (a z_{i2r} - w_{i2r}) + \sum_{i=0}^n \sum_{r=1}^k f_c k_2 v_{3r}^3 (w_{i2r} + \theta_{2r} w_{i2r} + \eta_{i2r} z_{i2r} - a z_{i2r}) \quad (5)$$

$$+ f_d \left(w_{n+1} - \sum_{m=1}^3 \sum_{r=1}^k w_{0mr} \right) + F \quad (6)$$

$$\text{subject to} \quad \sum_{m=1}^3 \sum_{r=1}^k w_{i+1,mr} \geq \sum_{m=1}^3 \sum_{r=1}^k (w_{imr} + \theta_{mr} w_{imr} + \eta_{imr} z_{imr}), \quad i = 0, 1, \dots, n-1 \quad (7)$$

$$w_{n+1} \geq \sum_{m=1}^3 \sum_{r=1}^k (w_{nmr} + \theta_{mr} w_{nmr} + \eta_{imr} z_{imr}) \quad (8)$$

$$\sum_{m=1}^3 \sum_{r=1}^k z_{imr} = 1, \quad i = 0, 1, \dots, n \quad (9)$$

$$b_{i,m-1} z_{imr} \leq w_{imr} \leq b_{i,m} z_{imr}, \quad i = 0, 1, \dots, n, \quad m = 1, 2, 3, \quad r = 1, \dots, k \quad (10)$$

$$l \leq w_{n+1} \leq u \quad (11)$$

$$w_{n+1} - \sum_{m=1}^3 \sum_{r=1}^k w_{imr} \leq \text{MRT}_i, \quad i = 1, 2, \dots, n \quad (12)$$

$$z_{imr} \in \{0, 1\},$$

$$w_{imr} \geq 0, \quad (13)$$

$$i = 0, 1, \dots, n, \quad m = 1, 2, 3, \quad r = 1, \dots, k.$$

TABLE 1: Results in Example 1.

Arc	Dis	S_D			S_I		
		Speed (m/s)	Dep (s)	TC (£)	Speed (m/s)	Dep (s)	TC (£)
(0, 1)	6370	5.28	0	4.4925	21.91	648	1.7637
(1, 2)	580	5.28	1207	0.4095	21.91	939	0.1610
(2, 3)	2600	5.28	1317	1.8377	21.91	965	0.7238
Total			1810	6.7397		1084	2.6485

TABLE 2: Results in Example 2.

Arc	Dis	S_D			S_I		
		Speed (m/s)	Dep (s)	TC (£)	Speed (m/s)	Dep (s)	TC (£)
(0, 1)	5117	5.28	2582	3.61	21.91	1877	1.42
(1, 2)	5294	5.28, 30.56	3551	0.94	21.91	2110	1.47
(2, 3)	12653	30.56	3765	4.07	21.91	2352	3.52
Total			4179	8.62		2929	6.41

TABLE 3: Results in Example 3.

Arc	Dis	S_D			S_I		
		Speed (m/s)	Dep (s)	TC (£)	Speed (m/s)	Dep (s)	TC (£)
(0, 1)	5117	5.28, 30.56	2746	2.03	21.91	3160	1.42
(1, 2)	8440	30.56	3677	2.72	21.91	3393	2.35
(2, 3)	5104	30.56	3734	1.64	21.91	3780	1.42
Total			4179	6.39		4013	5.19

3.2. *Case Study of the Mixed Linear Integer Programming for DSP.* In this section, we give three types of examples to investigate the impact of considering traffic congestion on the routing and scheduling planning activities. We then enable a heuristic for DSP through the test results.

In both cases, we analyze a four-node network where node 0 is the depot, $\{1, 2\}$ is the set of customers, and 3 is the airport, with routing $(0, 1, 2, 3)$. The congestion speed v_c is set to 19 km/h, the minimum speed limit v_{\min} to 40 km/h, and the maximum speed limit v_{\max} to 110 km/h. The congestion period is set to 3600 seconds. We uniformly discretize the free-flow speed within v_{\min} and v_{\max} . The examples differ with respect to the time window at the airport, the distance of the nodes, and the maximum ride time to guarantee feasibility. We assume that the capacity constraint is always satisfied given that the scheduling does not depend on the customer number. The parameters used to calculate the total cost function are the same as reported in Demir et al. [7]. Let S_I and S_D denote the solution of the time-independent approach and the time-dependent approach. For the computational results, the table reports cases in the first column. Then for each solution, the table reports (i) the set of traversed arcs, (ii) the distance of the arc, (iii) the speed at which each arc is traversed (for an arc traversed during the transient region, both the congestion speed and free-flow speed are reported), (iv) the departure time from the origin node, and (v) the total cost TC. The

last row of Dep denotes the arrival time at the airport. All the mixed integer problems are solved by CPLEX 12.0 and all the time parameters given below are measured in seconds.

Example 1 ($u \leq a$). In this example, the vehicle is in the all congestion period. The parameters are as follows: $l = 1084$, $u = 2489$, $MRT_1 = 657$, and $MRT_2 = 541$. The solutions to the time-independent and time-dependent approaches are displayed in Table 1.

Example 2 ($l \leq a \leq u$). In this example, the vehicle is in the transient region; the vehicle leaving within this region traverses a distance of congestion period and then the vehicle travels the remaining (if exists) distance with the free-flow speed. The parameters are as follows: $l = 2929$, $u = 4179$, $MRT_1 = 2738$, and $MRT_2 = 2814$. The solutions are displayed in Table 2.

Example 3 ($a \leq l$). In this example, the vehicle arrives at the airport after the congestion period. The vehicle may firstly traverse a distance of congestion period and then the vehicle travels the remaining (if exists) distance with the free-flow speed. The parameters are as follows: $l = 4013$, $u = 4179$, $MRT_1 = 2738$, and $MRT_2 = 251$. The solutions are displayed in Table 3. We do not consider the case of all free-flow region as it is actually the time-independent approach.

From Tables 1–3, we can see that the time-dependent approach yields a higher cost than the time-independent approach. The best speeds of the time-independent approach all exceed the congestion speed v_c , which is not feasible in the congestion period. We conclude that congestion is likely to increase costs or even leads to an infeasible solution. Thus, the congestion should be considered in the study of the pollution routing of FPDS.

From the last column of Tables 2 and 3, we see that the vehicle arrives at the airport at the maximum of the time window in order to avoid the congestion. The vehicle will never wait even if the wait time is allowed, which is not compatible with the results of Franceschetti et al. [9]. The reason is that the time window is limited to the final destination airport instead of the pickup node and the maximum ride time of each order is constrained in our problem, which makes the optimal schedule delay the departure time and decrease the wait time so as to avoid the congestion and reduce the customers ride time.

From the tables, we can see that the free-flow speeds in the time-dependent and time-independent approaches are the same in all the arcs. As a consequence, in the next section, we assume that the free-flow speed of a vehicle is the same even if it travels the different arc and provide a heuristic for the scheduling of a fixed route.

3.3. A Heuristic for DSP. In this section, we propose a heuristic for solving DSP. From Section 3.2, we can see that the DSP can be solved by a mixed integer programming. However, solving the MIP in Section 3.1 requires considerable computational effort due to the large number of binary decision variables and the accuracy of the solution depends on the level of discretization of the free-flow speeds. To overcome these limitations, we propose a heuristic algorithm which, in our numerical experiments, has been observed to solve the problem at least as well as the MIP solutions in most cases we have considered.

From the case study in Section 3.2, we can see that the free-flow speeds do not change in the route. Accordingly, we assume that the free-flow speed is unchanged during the route $(0, 1, 2, \dots, n, n+1)$. Then the DSP converts to scheduling the departure time and free-flow speed on a single arc with two nodes, the depot and the airport. This special case of a single arc network has been studied by Franceschetti et al. [9]. Our problem differs with the case of Franceschetti et al. [9] as there are maximum ride time constraints in the TDFPDS. We introduce a new speed function as the maximum ride speed together with the speed functions considered in Franceschetti et al. [9] to solve the DSP.

In order to ensure the consistency and readability of the paper, we firstly give the speed functions used in Franceschetti et al. [9]. We convert route $(0, 1, 2, \dots, n, n+1)$ to a new single arc $(0, n+1)$, with the distance of $d = \sum_{i=0}^n d_i$ and time window $[l, u]$ denoting the time in which the vehicle should arrive at the airport.

Let v_w^t be the free-flow speed required for the driver to arrive at node $n+1$ exactly at time t when leaving the depot at time w . Then

$$v_w^t = \begin{cases} \frac{d - (a - w)^+ v_c}{t - \max\{a, w\}}, & \text{if } w \in [0, t), t > a, \\ \infty, & \text{otherwise.} \end{cases} \quad (14)$$

Then v_w^u is the free-flow speed required for the driver to arrive at node $n+1$ exactly at time u when leaving the depot at time w . In other words, v_w^u is the minimum speed required to arrive at the airport which is less than or equal to u when $u > a$. v_w^l is the free-flow speed required for the driver to arrive at node $n+1$ exactly at time l when leaving the depot at time w and meanwhile is the maximum speed required to arrive at the airport which is greater than or equal to l when $l > a$.

Let w_{\max}^u denote the time at which the vehicle needs to depart from the depot to reach the node $n+1$ at exactly time u , with free-flow speed v_{\max}^u . w_{\max}^u is an upper bound on the departure time since the free-flow speed is no greater than v_{\max}^u . If $w_{\max}^u < 0$, then the route is infeasible. Without loss of generality, we assume that $w_{\max}^u \geq 0$ hereinafter.

Lemma 4. Consider a single arc instance to minimize the emission cost and driver wage cost in the absence of time window with fixed departure time w . The optimal free-flow speed is $\bar{v} = ((f_c k_1 + f_d)/(2f_c k_2))^{1/3}$.

Using Lemma 4, we can reduce DSP to one-dimensional optimization problem; that is, we set w as the unique decision variable. Proof of Lemma 4 is given in Appendix A.

We then consider the maximum ride time constraints. For each order i ($1 \leq i \leq n$) of route $(0, 1, 2, \dots, n, n+1)$, the minimum speed without congestion to satisfy its maximum ride time constraint is $v_{\text{free}}^i = (\sum_{j=i}^n d_j)/\text{MRT}_i$. The minimum speed of the route is $v_{\text{free}} = \max_{1 \leq i \leq n} \{v_{\text{free}}^i\}$. If $v_{\text{free}} > v_{\max}$, the route is infeasible and the algorithm is stopped. If $v_{\text{free}} \leq v_c$, which makes the maximum ride time constraints always feasible, the maximum ride time constraints are trivial and the results are given in Franceschetti et al. [9]. We set that $v_{\text{free}} = v_{\min}$ if $v_{\text{free}} < v_{\min}$ in Cases 2 and 3. We integrate the congestion period into computation. For each node i , let w_{ride}^i denote the departure time from node 0 with free-flow speed v_{\max} so that the vehicle travels from order i to the airport exactly for time MRT_i , written as

$$w_{\text{ride}}^i = a - \frac{\sum_{j=i}^n d_j - \text{MRT}_i v_{\max}}{v_c - v_{\max}} - \frac{\sum_{j=0}^{i-1} d_j}{v_c}. \quad (15)$$

If $w_{\text{ride}}^i < 0$, we can see that the maximum ride time of order i is always feasible with free-flow speed v_{\max} , in which case

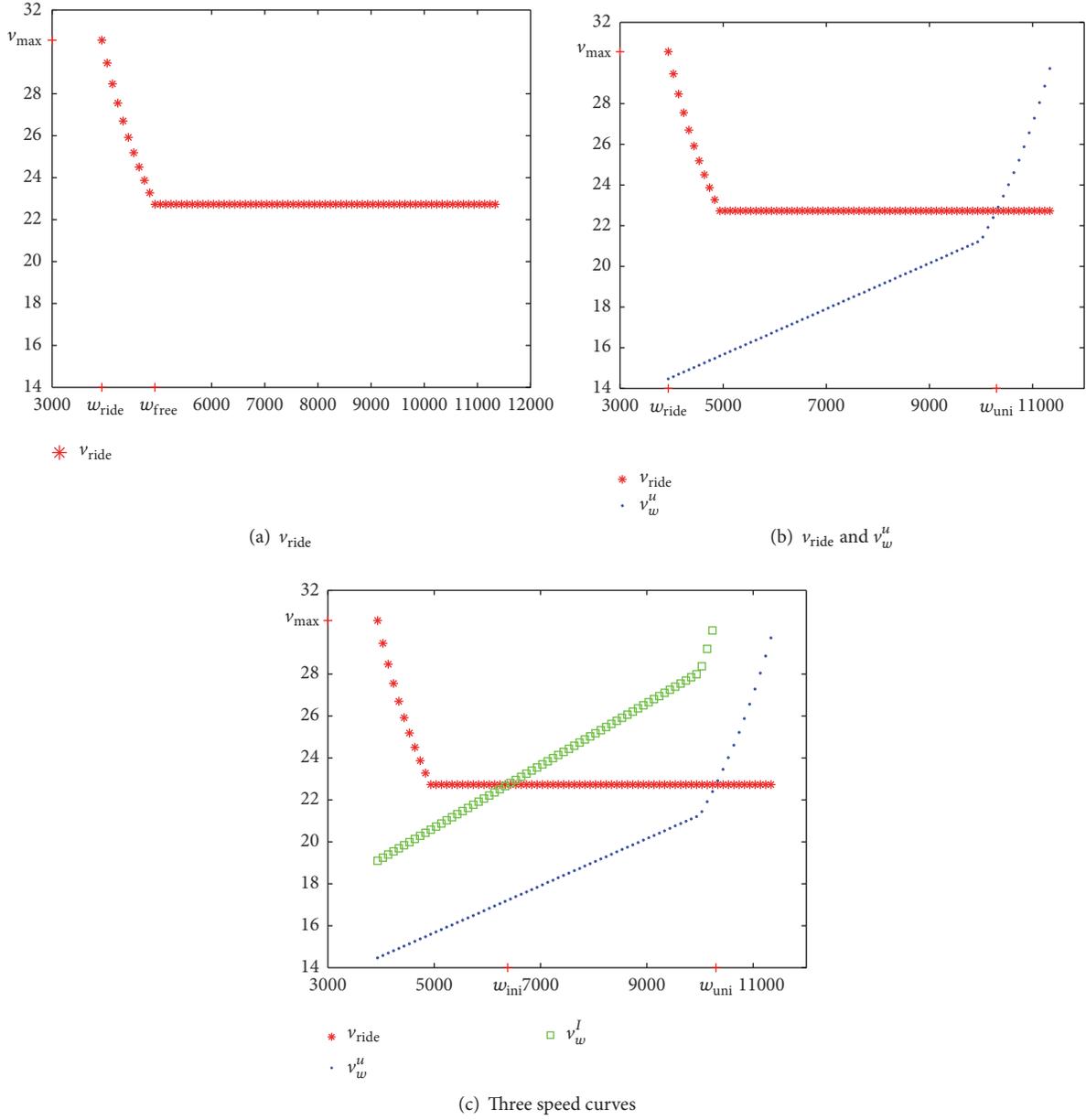


FIGURE 1: Speed functions as a function of w .

we set $w_{ride}^i = 0$. If $w_{ride}^i > w_{max}^u$, the route is infeasible. The minimum departure time to meet the maximum ride time constraints is denoted as $w_{ride} = \max_{1 \leq i \leq n} \{w_{ride}^i\}$.

Let v_{ride}^i denote the free-flow speed required for the driver to travel from node i to node $n + 1$ exactly for time MRT_i when leaving the depot at time w . In other words, v_{ride}^i is the minimum speed to meet the maximum ride time constraint of order i , which is written as follows:

$$v_{ride}^i = \frac{\sum_{j=i}^n d_j - v_c (a - w - \sum_{j=0}^{i-1} d_j / v_c)^+}{MRT_i - (a - w - \sum_{j=0}^{i-1} d_j / v_c)^+}. \quad (16)$$

Then the minimum free-flow speed to meet the maximum ride time constraints when leaving the depot at time w is denoted as $v_{ride} = \max_{1 \leq i \leq n} v_{ride}^i$. Figure 1(a) shows the minimum free-flow speed v_{ride} of a vehicle as a function of departure time w .

The calculation of v_{ride} suggests that the planning horizon can be divided into two consecutive time regions in terms of the departure time w as follows. The first one $w_{ride} \leq w \leq w_{free}$ is called the all decrease region: the vehicle leaving the depot within this region makes v_{ride} decrease all the time. The second one $w > w_{free}$ is called the stable region: the vehicle leaving the depot within this region makes v_{ride} remain a constant.

Lemma 5. *When the vehicle is leaving the depot within the all decrease region with free-flow speed v_{ride} , the emission cost and driver wage cost are monotonically decreasing with respect to the departure time w .*

Using Lemma 5, the optimal departure time in the decrease region can be easily solved. Proof of Lemma 5 is given in Appendix A.

We consider three cases: (1) $l \leq u \leq a$, (2) $l \leq a \leq u$, and (3) $a < l < u$.

Case 1 ($l \leq u \leq a$). In this case, the vehicle travels in the all congestion period. If $v_{free} > v_c$, the maximum ride time constraint is infeasible. Otherwise, the vehicle travels with speed v_c in the trip at any time w in $[(l - d/v_c)^+, w_{max}^u]$. The results of optimal departure time and the total cost are given in Appendix B.1.

Case 2 ($l \leq a \leq u$). In this case, the earliest time to arrive at the airport is less than a , which means that the vehicle travels with speed v_c when arriving at the airport exactly at l . We conclude that $w_{ride} \geq (l - d/v_c)^+$ since the vehicle arrives at the airport after a if leaving the depot at time w_{ride} . The feasible departure time belongs to $[w_{ride}, w_{max}^u]$. Figure 1(b) shows the minimum free-flow speed as a function of departure time w . Let w_{union} denote the intersection point of the speed curves v_w^u and v_{ride} . Then the calculation of the free-flow speed can be divided into two consecutive time regions in terms of the departure time w . In the first one $w_{ride} \leq w \leq w_{union}$, the feasible free-flow speed belongs to $[v_{ride}, v_{max}]$. In the second one $w_{union} \leq w \leq w_{max}^u$, the feasible free-flow speed belongs to $[v_w^u, v_{max}]$.

The results of optimal departure time and the total cost are given in Appendix B.2.

Case 3 ($a < l < u$). In this case, the earliest time to arrive at the airport is greater than a , which makes the choice of the free-flow speed more complicated. v_w^l , as the maximum speed to arrive at the airport greater than or equal to l , also needs to be considered. Let w_{max}^l denote the time at which the vehicle needs to depart from the depot to reach the node $n + 1$ at exactly time l , with free-flow speed v_{max} . If the condition $w_{max}^l \leq w_{ride}$ holds, which means that the maximum speed already reaches v_{max} , then v_w^l does not work and the feasible departure time belongs to $[w_{ride}, w_{max}^u]$ which converts to Case 2. Otherwise, let w_{ini} denote the intersection point of the speed curves v_w^l and v_{ride} . Figure 1(c) shows the minimum free-flow speed and the maximum free-flow speed as functions of departure time w . The calculation of the minimum free-flow speed can be divided into two consecutive time regions in terms of the departure time w . In the first one $w_{ini} \leq w \leq w_{union}$, the feasible free-flow speed belongs to $[v_{ride}, \min\{v_w^l, v_{max}\}]$. In the second one $w_{union} \leq w \leq w_{max}^u$, the feasible free-flow speed belongs to $[v_w^u, \min\{v_w^l, v_{max}\}]$.

The result of optimal departure time and the total cost is given in Appendix B.3.

4. A Set-Partitioning Method for Solving TDFPDS

This section presents a set-partitioning based method for solving TDFPDS following Dong et al. [10]. The set-partitioning method is a two-phase procedure. The candidate set of feasible routes is generated in the first phase and the set-partitioning model is solved to obtain the final solution in the second phase by using CPLEX 12.0 optimization software.

In phase one, we firstly sort the orders in increasing sequence by their earliest delivery time. Each order is selected as the seed order by the sorted sequence. Then all the possible routes containing combinations of the seed order and the following orders to meet the time window and capacity constraints are generated. The feasibility check and scheduling of the route is conducted using the mixed integer programming or the heuristic for DSP in Sections 3.1 and 3.3. If the feasible routes of the combinations are not single, the one with the least cost is selected to the candidate set. The candidate set of feasible routes is generated by looping all the orders. In phase two, the minimal cost set of best feasible routes is achieved as the optimal solution by solving the set-partitioning model.

5. Computational Results

This section presents the results of computational experiments using the mixed integer programming formulation and the heuristic for DSP in Sections 3.1 and 3.3, HE and MIP for short hereafter, with the whole algorithm generated in Section 4. All tests were carried out using data generated randomly with orders 30, 40, and 50. Two service areas are considered, $[-7 \text{ km}, 7 \text{ km}] \times [-4 \text{ km}, 4 \text{ km}]$ and $[-5.5 \text{ km}, 5.5 \text{ km}] \times [-4 \text{ km}, 4 \text{ km}]$, with the depot located in the center of the area and the airport in top right corner of the region. The locations of all the orders are uniformly and independently distributed in the area. The delivery time is generated in two intervals, $[0.5 \text{ h}, 4 \text{ h}]$ and $[0.5 \text{ h}, 3 \text{ h}]$, to distinguish the different order intensity degree. For each of the orders 30, 40, 50, three instances, different in the service area and the delivery time factor intensity, are generated. The maximum ride time of order i is calculated as follows: $MRT_i = \beta_1 t_1 + \beta_2 t_2$, where t_1 and t_2 are the travel time in the congestion region and the travel time with free-flow speed v_{max} from the order directly to the airport and arriving at the airport exactly at D_i . In Sections 5.1 and 5.2, set $\beta_1 = 1.1$ and $\beta_2 = 1.5$. In both instances, the congestion speed is set to 19 km/h and $v_{min} = 40$ km/h. The same parameter values as in Demir et al. [7] have been used; that is, $k_1 = 1.0176 \times 10^{-3}$, $k_2 = 1.4122 \times 10^{-7}$, $k_3 = 8.4036 \times 10^{-9}$, $f_c = 1.4 \text{ £/L}$, and $f_d = 0.0022 \text{ £/s}$. The curb weight μ is set to 1.5 ton and the average weight per person ν is set to 60 kg. The maximum free-flow speed v_{max} is set to 70 km/h and 110 km/h, respectively, to test the different cases of Appendix B. In a word, 18 different data sets are generated to test the performance of the algorithms. The whole algorithm is calculated on an Intel i5 PC computer running at 3.2 GHz with 4096 M of

TABLE 4: Comparison with HE and MIP.

Ins	TC _{HE} (£)	t_{HE} (s)	TC _{MIP} (£)	t_{MIP} (s)	Dev	Num
130	62.87	0.09	62.89	8.32	0.02	5
140	95.10	0.36	94.95	33.75	-0.15	4
150	122.22	0.41	122.34	154.74	0.09	5

TABLE 5: Effect of the congestion.

Ins	TC(N) (£)	Num(N)	$a = 3600$ s		$a = 7200$ s	
			TC (£)	Num	TC (£)	Num
130	54.57	148	62.87	139	94.48	135
140	70.94	302	95.10	276	130.18	256
150	91.95	742	123.81	555	181.97	477

RAM memory. The first phase of the algorithm is coded with Matlab software.

5.1. Performance of the Heuristic. This section compares the performance of HE with MIP. We test all cases and propose the total cost and computation time with the two methods as the fixed route optimization procedure in Table 4. Columns TC_{HE} and TC_{MIP} present the average total cost produced by the respective formulations and t_{HE} and t_{MIP} present the associated average computational times required to solve each type of instance. Column Dev presents the deviation between the two methods, computed as $Dev = 100(TC_{MIP} - TC_{HE})/TC_{MIP}$. Column Num presents the number of cases in which $TC_{HE} \leq TC_{MIP}$. Compared with MIP, HE is superior in terms of the computational time, especially in 50 node instances, with the average solution time reducing from 154.74 s to 0.41 s. HE also can solve 14 instances with better or equal results compared with MIP, in particular the 40 and 50 node instances. Even in the worst case, the deviation of the heuristic to the mix integer programming formulation is -1.2%, which is an acceptable result due to its efficient computation time.

5.2. Importance of Modeling Traffic Congestion. In this section, we compare the results of cases with and without congestion. More specifically, by using HE as the fixed route optimization procedure, we compute two time-dependent optimal solutions with respect to congestion period duration of 3600 s and 7200 s. Using MIP and fixing the congestion period to zero, we also compute a time-independent optimal solution. The results are provided in Table 5, where the total cost and the number of feasible routes (denoted by TC(N) and Num(N)) of time-independent problem and the total cost and number of feasible routes (denoted by TC and Num) using different peak hours are reported. The results show that in the presence of traffic congestion the total cost increases significantly and the number of feasible routes decreases, which imply that the congestion cannot be ignored in the actual situation. Ignoring the congestion period even obtains infeasible solutions. Furthermore the results also

TABLE 6: Impact of the maximum ride time constraints.

Ins	H		M		L		N	
	TC (£)	Num						
130	65.29	123	62.87	139	61.62	151	56.33	164
140	98.24	221	95.10	276	94.40	309	87.39	350
150	128.88	408	123.81	555	113.32	660	108.55	724

suggest that the cost is in a sharp increase and the number of feasible paths is significantly reduced with the increasing of congestion duration. It should be mentioned that the number of feasible routes does not decline substantially. This is because the maximum ride time also increases in the congestion period.

5.3. Impact of the Maximum Ride Time Constraints. In this section, we study impact of the maximum ride time constraints. Four service quality scenarios “H,” “M,” “L,” and “N” constrained by maximum ride time are considered. β_1 is set to 1.05, 1.1, and 1.2 and β_2 is set to 1.3, 1.5, and 1.8 to denote the high, middle, and low ride time service, “H,” “M,” and “L” for short. Scenario “N” denotes the case without maximum ride time constraints. The congestion period is set to 3600 s. The results are provided in Table 6. Columns TC and Num present the average total cost and number of the feasible routes produced by the respective scenarios. Results suggest that the maximum ride time constraints significantly affect the solutions of TDFPDS both in the total cost and in the number of feasible routes. The total cost decreases and the number increases with the reduction of the service quality.

6. Conclusions

In this paper, we have addressed a time-dependent pollution-routing and scheduling problem of free pickup and delivery of airline passengers to the airport faced by flight ticket sales companies. The problem is formulated under the framework of pollution-routing problem with time windows. The objective of the model is the emission cost and the driver wage cost, which proposes a reasonable calculation of the companies’ cost and the impact of the carbon emissions. Moreover, no examples could be found in the literature that simultaneously considered time-dependency and maximum ride time constraints in the VRP. A mixed integer linear programming formulation and a heuristic algorithm are then proposed for optimizing a given route.

Through the experimental results, the mixed integer linear programming formulation produces optimal schedules with same travel free-flow speed in each arc of the route and without wait time. Then the heuristic is proposed according to the analysis of the experimental results. The heuristic combines the speed functions generated by the time windows and maximum ride time constraints and displays the optimal schedule in each case. Experimental results show the efficiency of the heuristic in total cost and computation

time compared with the mixed integer linear programming formulation.

On the other hand, the key technology of the heuristic lies in the conversion to a single arc path under the consumption that the service time of each order is neglected. In the calculation model, the parameters values of the carbon emissions are used directly from Demir et al. [7], which may not be the same with the vehicle considered in our problem. Moreover, if a fleet of vehicles is composed of several different types of vehicles, the proposed value should be modified. There is an opportunity for future research to develop a more efficient optimization strategy considering the service time and heterogeneous fleet and thereby extend the results of the present research.

Appendix

A. Proof of Lemmas

To simplify the notations in the proofs below, let $B = f_c k_1$, $C = f_c k_2$, $D = f_d$, and $F = k_3(\mu + f)d$. The notations are also used in Appendix B.

A.1. Proof of Lemma 4

Proof. The total cost function $TC(w, v_f)$ for a fixed w as a function of the free-flow speed v_f can be written as

$$TC(w, v_f) = (B + Cv_c^3 + D)T^c(w) + (B + Cv_f^3 + D)T^f(w, v_f) + F. \quad (A.1)$$

We assume that the distance traveled by free-flow speed is d_f . Note that $T^c(w)$ does not change with v_f . Then, $TC(w, v_f)$ can be written as

$$TC(w, v_f) = (B + Cv_c^3 + D)T^c(w) + \frac{(B + Cv_f^3 + D)d_f}{v_f} + F. \quad (A.2)$$

Taking the derivative of function $TC(w, v_f)$ with variable v_f and setting the deviation to zero, we can see that the root is $\bar{v} = ((B + D)/(2C))^{(1/3)}$. The function $TC(w, v_f)$ is monotonically decreasing if $v_f \leq \bar{v}$ and monotonically increasing if $v_f \geq \bar{v}$. Thus, the function $TC(w, v_f)$ is minimized at \bar{v} . \square

A.2. Proof of Lemma 5

Proof. Note that when the vehicle is leaving the depot within the all decrease region, the round speed v_{ride} is a piecewise function of v_{ride}^i for some $i \in \{1, 2, \dots, n\}$. We then prove that for each order i in the decrease region, the total cost $TC(w, v_{ride}^i)$ is decreasing with respect to the departure time w . It should be pointed out that when v_{ride}^i is decreasing with respect to w , the equality $a > w + \sum_{j=0}^{i-1} d_j/v_c$ holds. The total cost $TC(w, v_{ride}^i)$ as a function of the departure

TABLE 7: Case 2.1.

Case	Conditions	(1)	(2)	Solution
2.1.1	$a \leq w_{max}^u, v_1 \leq v_2$	\searrow	\rightarrow	a
2.1.2	$a \leq w_{max}^u, v_1 > v_2$	\nearrow	\rightarrow	w_{ride}
2.1.3	$a > w_{max}^u, v_1 \leq v_2$	\searrow		w_{max}^u
2.1.4	$a > w_{max}^u, v_1 > v_2$	\nearrow		w_{ride}

time w with a decreasing free-flow speed v_{ride}^i as a function of w (equation (16)) can be rewritten as a one-dimensional function as follows:

$$TC(w, v_{ride}^i) = (B + Cv_c^3 + D)(a - w) + \left(B + C \left(\frac{\sum_{j=i}^n d_j - v_c(a - w - \sum_{j=0}^{i-1} d_j/v_c)}{MRT_i - (a - w - \sum_{j=0}^{i-1} d_j/v_c)} \right)^3 + D \right) \left(MRT_i - \left(a - w - \sum_{j=0}^{i-1} \frac{d_j}{v_c} \right) \right). \quad (A.3)$$

Take the derivative of function $TC(w, v_{ride}^i)$ with variable w and set the deviation to zero, where the root is $\bar{w} = (3av_c - MRT_i v_c - 3 \sum_{j=0}^{i-1} d_j - 2 \sum_{j=i}^n d_j)/(3v_c)$. The function $TC(w, v_{ride}^i)$ is monotonically increasing if $w \leq \bar{w}$ and monotonically decreasing if $w \geq \bar{w}$. Meanwhile, the condition that $w_{ride}^i - \bar{w} = (v_c + 2v_{max})(\sum_{j=i}^n d_j - v_c MRT_i)/3v_c(v_{max} - v_c) > 0$ holds because $MRT_i v_c < \sum_{j=i}^n d_j$. Thus the function $TC(w, v_{ride}^i)$ is monotonically decreasing with respect to the departure time w . \square

B. Results for a Fixed Route

In Tables 7–12, we use numbers such as (1) and (2), to refer to the pieces of the total cost function (TC for short) as it is a piecewise function with different free-flow speed. For each piece of TC function, we use symbols such as \rightarrow , \nearrow , \searrow , and \smile to indicate whether the TC function is, respectively, constant, nondecreasing, nonincreasing, or convex, with respect to w .

B.1. $\mathbf{l} \leq \mathbf{u} \leq \mathbf{a}$. In this case, the total cost $TC(w) = (B + Cv_c^3 + D)d/v_c + F$, if $w \in [(l - d/v_c)^+, u - d/v_c]$ is a constant and the optimal value of w belongs to $[(l - d/v_c)^+, u - d/v_c]$.

B.2. $\mathbf{l} \leq \mathbf{a} \leq \mathbf{u}$. In this case, we convert the function $TC(w, v_f)$ to one-dimensional optimization problem by using Lemma 4. We distinguish two subcases: Case 2.1: $v_{max} \leq \bar{v}$ and Case 2.2: $v_{max} > \bar{v}$. It should be pointed out that, in some cases, when the optimal solution is not single, we select the one with the earliest departure time.

TABLE 8: Case 2.2.

Case	Conditions	(1)	(2)	(3)	(4)	Solution
2.2.1	$\bar{w} > w_{\text{uni}}, a \leq \bar{w}$	↘	↘	→	↗	a
2.2.2	$\bar{w} > w_{\text{uni}}, a \leq w_{\text{max}}^u, \bar{w} \leq \underline{w} < a$	↘	↘	←	↗	\underline{w}
2.2.3	$\bar{w} > w_{\text{uni}}, \underline{w} < \bar{w} < a$	↘	↘	↗	↗	\bar{w}
2.2.4	$\bar{w} > w_{\text{uni}}, \bar{w} \leq \underline{w} \leq w_{\text{max}}^u < a$	↘	↘	←		\underline{w}
2.2.5	$\bar{w} > w_{\text{uni}}, \underline{w} < \bar{w}, w_{\text{max}}^u < a$	↘	↘	↗		\bar{w}
2.2.6	$\bar{w} > w_{\text{uni}}, \underline{w} < w_{\text{max}}^u < a$	↘	↘	↘		w_{max}^u
2.2.7	$\bar{w} \leq w_{\text{uni}}, a \leq w_{\text{uni}}, v_3 \leq v_2$	↘	↘	→	↗	a
2.2.8	$\bar{w} \leq w_{\text{uni}}, a \leq w_{\text{uni}}, v_3 > v_2$	↘	↗	→	↗	w_{free}
2.2.9	$\bar{w} \leq w_{\text{uni}}, w_{\text{free}} < w_{\text{uni}} \leq \bar{w} < a \leq w_{\text{max}}^u, v_3 \leq v_2$	↘	↘	←	↗	\underline{w}
2.2.10	$\bar{w} \leq w_{\text{uni}}, w_{\text{free}} < w_{\text{uni}} \leq \bar{w} < a \leq w_{\text{max}}^u, v_3 < v_2$	↘	↗	←	↗	\underline{w} or w_{free}
2.2.11	$\bar{w} \leq w_{\text{uni}}, w_{\text{uni}} \leq w_{\text{free}}, w_{\text{uni}} \leq \underline{w} < a \leq w_{\text{max}}^u$	↘	←	↗		\underline{w}
2.2.12	$\bar{w} \leq w_{\text{uni}}, w_{\text{free}} < w_{\text{uni}}, \underline{w} < w_{\text{uni}} < a \leq w_{\text{max}}^u, v_3 \leq v_2$	↘	↘	↗	↗	w_{uni}
2.2.13	$\bar{w} \leq w_{\text{uni}}, w_{\text{free}} < w_{\text{uni}}, \underline{w} < w_{\text{uni}} < a \leq w_{\text{max}}^u, v_3 > v_2$	↘	↗	↗	↗	w_{free}
2.2.14	$\bar{w} \leq w_{\text{uni}}, w_{\text{uni}} \leq w_{\text{free}}, \underline{w} < w_{\text{uni}} < a \leq w_{\text{max}}^u$	↘	↗	↗		w_{uni}
2.2.15	$\bar{w} \leq w_{\text{uni}}, w_{\text{free}} < w_{\text{uni}} \leq \underline{w} \leq w_{\text{max}}^u < a, v_3 \leq v_2$	↘	↘	←		\underline{w}
2.2.16	$\bar{w} \leq w_{\text{uni}}, w_{\text{free}} < w_{\text{uni}} \leq \underline{w} \leq w_{\text{max}}^u < a, v_3 > v_2$	↘	↗	←		\underline{w} or w_{free}
2.2.17	$\bar{w} \leq w_{\text{uni}}, w_{\text{uni}} \leq w_{\text{free}}, w_{\text{uni}} \leq \underline{w} \leq w_{\text{max}}^u < a$	↘	←			\underline{w}
2.2.18	$\bar{w} \leq w_{\text{uni}}, w_{\text{free}} < w_{\text{uni}}, \underline{w} < w_{\text{uni}}, w_{\text{max}}^u < a, v_3 \leq v_2$	↘	↘	↗		w_{uni}
2.2.19	$\bar{w} \leq w_{\text{uni}}, w_{\text{free}} < w_{\text{uni}}, \underline{w} < w_{\text{uni}}, w_{\text{max}}^u < a, v_3 > v_2$	↘	↗	↘		w_{free}
2.2.20	$\bar{w} \leq w_{\text{uni}}, \underline{w} < w_{\text{uni}} \leq w_{\text{free}}, w_{\text{max}}^u < a$	↘	↗			w_{uni}
2.2.21	$\bar{w} \leq w_{\text{uni}}, w_{\text{free}} < w_{\text{uni}}, \underline{w} > w_{\text{max}}^u, v_3 \leq v_2$	↘	↘	↘		w_{max}^u
2.2.22	$\bar{w} \leq w_{\text{uni}}, w_{\text{free}} < w_{\text{uni}}, \underline{w} > w_{\text{max}}^u, v_3 < v_2$	↘	↗	↘		w_{max}^u or w_{free}
2.2.23	$\bar{w} \leq w_{\text{uni}}, w_{\text{uni}} \leq w_{\text{free}}, \underline{w} > w_{\text{max}}^u$	↘	↘			w_{max}^u

TABLE 9: Case 3.1.

Case	Conditions	(1)	(2)	(3)	Solution
3.1.1	$a \leq w_{\text{ini}}$	↘	→		w_{max}^l
3.1.2	$w_{\text{ini}} \leq \underline{w}^l, a \leq w_{\text{max}}^l$	←	↘	→	w_{max}^l or \underline{w}^l
3.1.3	$\underline{w}^l < w_{\text{ini}} < a \leq w_{\text{max}}^l$	↗	↘	→	w_{max}^l or w_{ini}
3.1.4	$w_{\text{ini}} \leq \underline{w}^l \leq w_{\text{max}}^l < a \leq w_{\text{max}}^u, v_1 \leq v_2$	←	↘	→	a or \underline{w}^l
3.1.5	$w_{\text{ini}} \leq \underline{w}^l \leq w_{\text{max}}^l < a \leq w_{\text{max}}^u, v_1 > v_2$	←	↗	→	\underline{w}^l
3.1.6	$\underline{w}^l < w_{\text{ini}}, w_{\text{max}}^l < a \leq w_{\text{max}}^u, v_1 \leq v_2$	↗	↘	→	a or w_{ini}
3.1.7	$\underline{w}^l < w_{\text{ini}}, w_{\text{max}}^l < a \leq w_{\text{max}}^u, v_1 > v_2$	↗	↗	→	w_{ini}
3.1.8	$w_{\text{max}}^l < \underline{w}^l, a \leq w_{\text{max}}^u, v_1 \leq v_2$	↘	↘	→	a
3.1.9	$w_{\text{max}}^l < \underline{w}^l, a \leq w_{\text{max}}^u, v_1 > v_2$	↘	↗	→	w_{max}^l
3.1.10	$w_{\text{ini}} \leq \underline{w}^l \leq w_{\text{max}}^l < w_{\text{max}}^u < a, v_1 \leq v_2$	←	↘		w_{max}^u or \underline{w}^l
3.1.11	$w_{\text{ini}} \leq \underline{w}^l \leq w_{\text{max}}^l < w_{\text{max}}^u < a, v_1 > v_2$	←	↗		\underline{w}^l
3.1.12	$\underline{w}^l < w_{\text{ini}}, w_{\text{max}}^u < a, v_1 \leq v_2$	↗	↘		w_{max}^u or w_{ini}
3.1.13	$\underline{w}^l < w_{\text{ini}}, w_{\text{max}}^u < a, v_1 > v_2$	↗	↗		w_{ini}
3.1.14	$w_{\text{max}}^l < \underline{w}^l, w_{\text{max}}^u < a, v_1 \leq v_2$	↘	↘		w_{max}^u
3.1.15	$w_{\text{max}}^l < \underline{w}^l, w_{\text{max}}^u < a, v_1 > v_2$	↘	↗		w_{max}^l

TABLE 10: Case 3.2.1.

Case	Conditions	(1)	(2)	(3)	(4)	Solution
3.2.1.1	$a \leq w_{uni}, v_3 \leq v_2$	↘	↘	→	↗	a
3.2.1.2	$a \leq w_{uni}, v_3 > v_2$	↘	↗	→	↗	w_{free}
3.2.1.3	$w_{free} < w_{uni} \leq \bar{w} < a \leq w_{max}^u, v_3 \leq v_2$	↘	↘	←	↗	\underline{w}
3.2.1.4	$w_{free} < w_{uni} \leq \bar{w} < a \leq w_{max}^u, v_3 < v_2$	↘	↗	←	↗	\underline{w} or w_{free}
3.2.1.5	$w_{uni} \leq w_{free}, w_{uni} \leq \underline{w} < a \leq w_{max}^u$	↘	←	↗		\underline{w}
3.2.1.6	$w_{free} < w_{uni}, \underline{w} < w_{uni} < a \leq w_{max}^u, v_3 \leq v_2$	↘	↘	↗	↗	w_{uni}
3.2.1.7	$w_{free} < w_{uni}, \underline{w} < w_{uni} < a \leq w_{max}^u, v_3 > v_2$	↘	↗	↗	↗	w_{free}
3.2.1.8	$w_{uni} \leq w_{free}, \underline{w} < w_{uni} < a \leq w_{max}^u$	↘	↗	↗		w_{uni}
3.2.1.9	$w_{free} < w_{uni} \leq \underline{w} \leq w_{max}^u < a, v_3 \leq v_2$	↘	↘	←		\underline{w}
3.2.1.10	$w_{free} < w_{uni} \leq \underline{w} \leq w_{max}^u < a, v_3 > v_2$	↘	↗	←		\underline{w} or w_{free}
3.2.1.11	$w_{uni} \leq w_{free}, w_{uni} \leq \underline{w} \leq w_{max}^u < a$	↘	←			\underline{w}
3.2.1.12	$w_{free} < w_{uni}, \underline{w} < w_{uni}, w_{max}^u < a, v_3 \leq v_2$	↘	↘	↗		w_{uni}
3.2.1.13	$w_{free} < w_{uni}, \underline{w} < w_{uni}, w_{max}^u < a, v_3 > v_2$	↘	↗	↘		w_{free}
3.2.1.14	$w_{uni} \leq w_{free}, w_{max}^u < a$	↘	↗			w_{uni}
3.2.1.15	$w_{free} < w_{uni}, \underline{w} > w_{max}^u, v_3 \leq v_2$	↘	↘	↘		w_{max}^u
3.2.1.16	$w_{free} < w_{uni}, \underline{w} > w_{max}^u, v_3 < v_2$	↘	↗	↘		w_{max}^u or w_{free}
3.2.1.17	$w_{uni} \leq w_{free}, \underline{w} > w_{max}^u$	↘	↘			w_{max}^u

TABLE 11: Case 3.2.2.

Case	Conditions	(1)	(2)	(3)	(4)	Solution
3.2.2.1	$a \leq w_{ini}$	↘	→	↗		\bar{w}^l
3.2.2.2	$a \leq \bar{w}^l, w_{ini} \leq \underline{w}^l$	←	↘	→	↗	\underline{w}^l or \bar{w}^l
3.2.2.3	$\underline{w}^l < w_{ini}, a \leq \bar{w}^l$	↗	↘	→	↗	\bar{w}^l or w_{ini}
3.2.2.4	$a \leq \bar{w}, w_{ini} \leq \underline{w}^l \leq \bar{w}^l < a$	←	↘	→	↗	a or \underline{w}^l
3.2.2.5	$a \leq \bar{w}, \underline{w}^l < w_{ini}, \bar{w}^l < a$	↗	↘	→	↗	a or w_{ini}
3.2.2.6	$a \leq \bar{w}, \underline{w}^l > \bar{w}^l$	↘	↘	→	↗	a
3.2.2.7	$a \leq w_{max}^u, w_{ini} \leq \underline{w}^l \leq \bar{w}^l, \bar{w} \leq \underline{w}$	←	↘	←	↗	\underline{w}^l or \underline{w}
3.2.2.8	$a \leq w_{max}^u, \underline{w}^l < w_{ini}, \bar{w} \leq \underline{w}$	↗	↘	←	↗	w_{ini} or \underline{w}
3.2.2.9	$a \leq w_{max}^u, \underline{w}^l > \bar{w}^l, \bar{w} \leq \underline{w}$	↘	↘	←	↗	\underline{w}
3.2.2.10	$a \leq w_{max}^u, \underline{w} \leq \bar{w}, w_{ini} \leq \underline{w}^l \leq \bar{w}^l$	←	↘	↗	↗	\underline{w}^l or \bar{w}
3.2.2.11	$a \leq w_{max}^u, \underline{w} \leq \bar{w}, \underline{w}^l < w_{ini}$	↗	↘	↗	↗	w_{ini} or \bar{w}
3.2.2.12	$a \leq w_{max}^u, \underline{w} \leq \bar{w}, \underline{w}^l > \bar{w}^l$	↘	↘	↗	↗	\bar{w}
3.2.2.13	$\bar{w} \leq \underline{w} \leq w_{max}^u < a, w_{ini} \leq \underline{w}^l \leq \bar{w}^l$	←	↘	←		\underline{w}^l or \underline{w}
3.2.2.14	$\bar{w} \leq \underline{w} \leq w_{max}^u < a, \underline{w}^l < w_{ini}$	↗	↘	←		w_{ini} or \underline{w}
3.2.2.15	$\bar{w} \leq \underline{w} \leq w_{max}^u < a, \underline{w}^l > \bar{w}^l$	↘	↘	←		\underline{w}
3.2.2.16	$w_{max}^u < a, \underline{w} < \bar{w}, w_{ini} \leq \underline{w}^l \leq \bar{w}^l$	←	↘	↗		\underline{w} or \underline{w}^l
3.2.2.17	$w_{max}^u < a, \underline{w} < \bar{w}, \underline{w}^l < w_{ini}$	↗	↘	↗		w_{ini} or \underline{w}
3.2.2.18	$w_{max}^u < a, \underline{w} < \bar{w}, \underline{w}^l > \bar{w}^l$	↘	↘	↗		\underline{w}
3.2.2.19	$w_{max}^u < \underline{w}, w_{ini} \leq \underline{w}^l \leq \bar{w}^l$	←	↘	↘		w_{max}^u or \underline{w}^l
3.2.2.20	$w_{max}^u < \underline{w}, \underline{w}^l < w_{ini}$	↗	↘	↘		w_{ini} or w_{max}^u
3.2.2.21	$w_{max}^u < \underline{w}, \underline{w}^l > \bar{w}^l$	↘	↘	↘		w_{max}^u

TABLE 12: Case 3.2.3.

Case	Conditions	(1)	(2)	(3)	(4)	Solution
3.2.3.1	$a \leq \bar{w}$	↘	↘	→	↗	a
3.2.3.2	$a \leq w_{max}^u, \bar{w} \leq \underline{w}$	↘	↘	←	↗	\underline{w}
3.2.3.3	$a \leq w_{max}^u, \underline{w} < \bar{w} < a$	↘	↘	↗	↗	\bar{w}
3.2.3.4	$\bar{w} \leq \underline{w} \leq w_{max}^u < a$	↘	↘	←		\underline{w}
3.2.3.5	$\underline{w} < \bar{w}, w_{max}^u < a$	↘	↘	↗		\bar{w}
3.2.3.6	$w_{max}^u < \underline{w}$	↘	↘	↘		w_{max}^u

In Case 2.1,

$$\begin{aligned} \text{TC}(w) &= (B + Cv_c^3 + D)(a - w)^+ \\ &+ (B + Cv_{\max}^3 + D) \frac{(d - (a - w)^+ v_c)}{v_{\max}} + F, \quad (\text{B.1}) \\ &\text{if } w \in [w_{\text{ride}}, w_{\max}^u]. \end{aligned}$$

Table 7 gives the solution, where $v_1 = ((B + Cv_{\max}^3 + D)/v_{\max})^{1/2}$ and $v_2 = ((B + Cv_c^3 + D)/v_c)^{1/2}$.

In Case 2.2,

$$\text{TC}(w) = \begin{cases} (B + Cv_c^3 + D)(a - w)^+ + (B + C(\max\{v_{\text{ride}}, \bar{v}\})^3 + D) \frac{(d - (a - w)^+ v_c)}{\max\{v_{\text{ride}}, \bar{v}\}} + F, & \text{if } w \in [w_{\text{ride}}, w_{\text{uni}}], \\ (B + Cv_c^3 + D)(a - w)^+ + (B + C(\max\{v_w^u, \bar{v}\})^3 + D) \frac{(d - (a - w)^+ v_c)}{\max\{v_w^u, \bar{v}\}} + F, & \text{if } w \in [w_{\text{uni}}, w_{\max}^u]. \end{cases} \quad (\text{B.2})$$

Table 8 gives the solution, where $v_3 = ((B + Cv_{\text{free}}^3 + D)/v_{\text{free}})^{1/2}$, $\underline{v} = ((B + Cv_c^3 + D)/3C)^{1/2}$, $\underline{w} = a - (d - (u - a)\underline{v})/v_c$, and

$$\bar{w} = \begin{cases} a - \frac{d - (u - a)\bar{v}}{v_c}, & \text{if } \bar{v} \leq \frac{d}{u - a}, \\ u - \frac{d}{\bar{v}}, & \text{otherwise.} \end{cases} \quad (\text{B.3})$$

B.3. $\mathbf{a} < \mathbf{l} < \mathbf{u}$. In this case, we convert the function $\text{TC}(w, v_f)$ to one-dimensional optimization problem by using Lemma 4. We assume that $w_{\max}^l > w_{\text{ride}}$ since the case switches to Appendix B.2 if the condition does not hold. We distinguish two subcases: Case 3.1 : $v_{\max} \leq \bar{v}$ and Case 3.2 : $v_{\max} > \bar{v}$. It should be pointed out that, in some cases, when the optimal solution is not single, we select the one with the earliest departure time.

In Case 3.1,

$$\text{TC}(w) = \begin{cases} (B + Cv_c^3 + D)(a - w)^+ + (B + C(v_w^l)^3 + D) \frac{(d - (a - w)^+ v_c)}{v_w^l} + F, & \text{if } w \in [w_{\text{ini}}, w_{\max}^l], \\ (B + Cv_c^3 + D)(a - w)^+ + (B + Cv_{\max}^3 + D) \frac{(d - (a - w)^+ v_c)}{v_{\max}} + F, & \text{if } w \in [w_{\max}^l, w_{\max}^u]. \end{cases} \quad (\text{B.4})$$

Table 9 gives the solution, where $\underline{w}^l = a - (d - (l - a)\underline{v})/v_c$.

In Case 3.2, to simplify the description of the TC function, we distinguish three subcases: Case 3.2.1 : $\bar{w} \leq w_{\text{uni}}$, Case 3.2.2 : $\bar{w} > w_{\text{uni}}$ and $\bar{w}^l > w_{\text{ini}}$, and Case 3.2.3 : $\bar{w} > w_{\text{uni}}$ and $\bar{w}^l \leq w_{\text{ini}}$, where

$$\bar{w}^l = \begin{cases} a - \frac{d - (l - a)\bar{v}}{v_c}, & \text{if } \bar{v} \leq \frac{d}{l - a}, \\ l - \frac{d}{\bar{v}}, & \text{otherwise.} \end{cases} \quad (\text{B.5})$$

In Case 3.2.1,

$$\text{TC}(w) = \begin{cases} (B + Cv_c^3 + D)(a - w)^+ + (B + C(v_{\text{ride}})^3 + D) \frac{(d - (a - w)^+ v_c)}{v_{\text{ride}}} + F, & \text{if } w \in [w_{\text{ini}}, w_{\text{uni}}], \\ (B + Cv_c^3 + D)(a - w)^+ + (B + C(v_w^u)^3 + D) \frac{(d - (a - w)^+ v_c)}{v_w^u} + F, & \text{if } w \in [w_{\text{uni}}, w_{\max}^u]. \end{cases} \quad (\text{B.6})$$

Table 10 gives the solution in all possible subcases.

In Case 3.2.2,

$$TC(w) = \begin{cases} (B + Cv_c^3 + D)(a - w)^+ + (B + C(v_w^l)^3 + D) \frac{(d - (a - w)^+ v_c)}{v_w^l} + F, & \text{if } w \in [w_{ini}, \bar{w}^l], \\ (B + Cv_c^3 + D)(a - w)^+ + (B + C\bar{v}^3 + D) \frac{(d - (a - w)^+ v_c)}{\bar{v}} + F, & \text{if } w \in [\bar{w}^l, \bar{w}], \\ (B + Cv_c^3 + D)(a - w)^+ + (B + C(v_w^u)^3 + D) \frac{(d - (a - w)^+ v_c)}{v_w^u} + F, & \text{if } w \in [\bar{w}, w_{max}^u]. \end{cases} \quad (B.7)$$

Table 11 gives the solution in all possible subcases.

In Case 3.2.3,

$$TC(w) = \begin{cases} (B + Cv_c^3 + D)(a - w)^+ + (B + C(v_{ride})^3 + D) \frac{(d - (a - w)^+ v_c)}{v_{ride}} + F, & \text{if } w \in [w_{ini}, w_1], \\ (B + Cv_c^3 + D)(a - w)^+ + (B + C\bar{v}^3 + D) \frac{(d - (a - w)^+ v_c)}{\bar{v}} + F, & \text{if } w \in [w_1, \bar{w}^l], \\ (B + Cv_c^3 + D)(a - w)^+ + (B + C(v_w^u)^3 + D) \frac{(d - (a - w)^+ v_c)}{v_w^u} + F, & \text{if } w \in [\bar{w}, w_{max}^u]. \end{cases} \quad (B.8)$$

w_1 is the departure time at which the round time $v_{ride} = \bar{v}$. We can conclude that $v_{free} \leq v_{w_{ini}}^u \leq \bar{v}$; thus, v_{ride} belongs to the all decrease region when the departure time $w \in [w_{ini}, w_1]$ and $w_1 < a$.

Table 12 gives the solution in all possible subcases.

Competing Interests

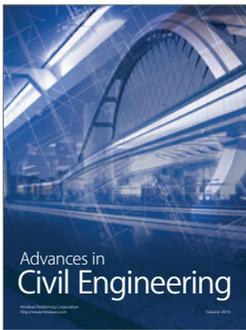
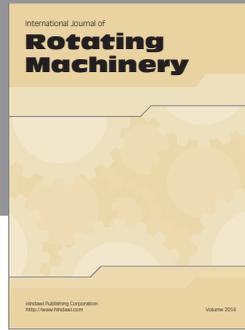
All the authors of this article declare that there is no conflict of interests regarding the publication of this article. They have no proprietary, financial, professional, or other personal interest of any nature or kind in any product, service, and/or company that could be construed as influencing the position presented in, or the review of, this article.

Acknowledgments

This work is supported by National Natural Science Foundation, Grants nos. 61104003 and 61273008; Research Program for Liaoning Excellent Talents in University, Grant no. LJQ2014027; Hebei Province Natural Science Foundation, Grant no. F2015501047; and Program of Science and Technology Research of Hebei University, Grant no. Z2015091.

References

- [1] Á. Felipe, M. T. Ortuño, G. Righini, and G. Tirado, "A heuristic approach for the green vehicle routing problem with multiple technologies and partial recharges," *Transportation Research Part E: Logistics and Transportation Review*, vol. 71, pp. 111–128, 2014.
- [2] A. Montoya, C. Guéreta, J. E. Mendoza et al., "A multi-space sampling heuristic for the green vehicle routing problem," *Transportation Research Part C: Emerging Technologies*, vol. 70, pp. 113–128, 2015.
- [3] C. Lin, K. L. Choy, G. T. S. Ho, S. H. Chung, and H. Y. Lam, "Survey of Green Vehicle Routing Problem: past and future trends," *Expert Systems with Applications*, vol. 41, no. 4, pp. 1118–1138, 2014.
- [4] Y. Y. Xiao, Q. H. Zhao, I. Kaku, and Y. C. Xu, "Development of a fuel consumption optimization model for the capacitated vehicle routing problem," *Computers & Operations Research*, vol. 39, no. 7, pp. 1419–1431, 2012.
- [5] Y. Kuo, "Using simulated annealing to minimize fuel consumption for the time-dependent vehicle routing problem," *Computers and Industrial Engineering*, vol. 59, no. 1, pp. 157–165, 2010.
- [6] T. Bektaş and G. Laporte, "The pollution-routing problem," *Transportation Research Part B: Methodological*, vol. 45, no. 8, pp. 1232–1250, 2011.
- [7] E. Demir, T. Bektaş, and G. Laporte, "An adaptive large neighborhood search heuristic for the pollution-routing problem," *European Journal of Operational Research*, vol. 223, no. 2, pp. 346–359, 2012.
- [8] R. Kramer, A. Subramanian, T. Vidal, and L. d. Cabral, "A matheuristic approach for the pollution-routing problem," *European Journal of Operational Research*, vol. 243, no. 2, pp. 523–539, 2015.
- [9] A. Franceschetti, D. Honhon, T. Van Woensel, T. Bektaş, and G. Laporte, "The time-dependent pollution-routing problem," *Transportation Research Part B: Methodological*, vol. 56, pp. 265–293, 2013.
- [10] G. Dong, J. F. Tang, K. K. Lai, and Y. Kong, "An exact algorithm for vehicle routing and scheduling problem of free pickup and delivery service in flight ticket sales companies based on set-partitioning model," *Journal of Intelligent Manufacturing*, vol. 22, no. 5, pp. 789–799, 2011.
- [11] O. Jabali, T. Van Woensel, and A. G. de Kok, "Analysis of travel times and CO2 emissions in time-dependent vehicle routing," *Production and Operations Management*, vol. 21, no. 6, pp. 1060–1074, 2012.



Hindawi

Submit your manuscripts at
<https://www.hindawi.com>

