Optimizing departure times in vehicle routes

A.L. Kok*, E.W. Hans, J.M.J. Schutten

Operational Methods for Production and Logistics, University of Twente,

P.O. Box 217, 7500AE, Enschede, Netherlands

Abstract

Most solution methods for the vehicle routing problem with time windows (VRPTW) develop routes from the earliest feasible departure time. In practice, however, temporal traffic congestions make such solutions non-optimal with respect to minimizing the total duty time. Furthermore, the VRPTW does not account for driving hours regulations, which restrict the available travel time for truck drivers. To deal with these problems, we consider the vehicle departure time optimization (VDO) problem. We propose an ILP-formulation that minimizes the total duty time. The results of a case study indicate that duty time reductions of 15% can be achieved. Furthermore, computational experiments on VRPTW benchmarks indicate that ignoring traffic congestions or driving hours regulations leads to practically infeasible solutions. Therefore, new vehicle routing methods should be developed that account for these common restrictions.

Keywords: Integer programming; Departure time scheduling; Time-dependent travel times; Driving hours regulations

1 Introduction

The VRP, which concerns the scheduling and routing of a homogeneous vehicle fleet among a set of customers, has been widely discussed in the literature

*Corresponding author. Tel.: +31 53 489 4164; fax: +31 53 489 2159

E-mail adresses: a.l.kok@utwente.nl, e.w.hans@utwente.nl, j.m.j.schutten@utwente.nl

(Toth and Vigo (2002) present an extensive overview of the VRP and solution methods). The problem arises in many applications areas such as retail distribution, mail delivery, freight operations, school bus routing, and dial-a-ride service. However, two real-life restrictions have hardly been discussed, namely temporary traffic congestion and driving hours regulations. This paper addresses a variant of the vehicle routing problem with time windows (VRPTW) in which these real-life conditions are incorporated.

Traffic congestion forms a major problem for businesses such as logistics service providers and distribution firms. Due to temporary traffic congestion, vehicles arrive late at customers and driving hours regulations are violated. Since travel times depend on both distance traveled and time of departure, Malandraki and Daskin (1992) introduce the time dependent vehicle routing problem (TDVRP). Furthermore, Hill and Benton (1992), Ichoua et al. (2003), Fleischmann et al. (2004), Haghani and Jung (2005), and Van Woensel et al. (2008) propose travel time models and algorithms for the TDVRP.

Driving hours regulations severely restrict the set of feasible vehicle routes in a VRP. These regulations impose restrictions on the total daily travel time available for a truck driver, as well as requirements on the scheduling of (lunch-)breaks during the day. The only papers we are aware of in which driving hours regulations are considered are Xu et al. (2003), Archetti and Savelsbergh (2007), and Goel (2008). Xu et al. (2003) consider a practical pickup and delivery problem in which the US hours of service regulations are considered. They conjecture that finding a feasible driver schedule after the vehicle routes are constructed is an NP-hard problem. However, Archetti and Savelsbergh (2007) develop a polynomial time algorithm for this problem. Goel (2008) considers the VRPTW with the European Legislation on driving and working hours. He proposes a labeling algorithm for determining the feasibility of vehicle routes with respect to these regulations and embeds this algorithm in a large neighborhood search algorithm. However, neither of the mentioned papers considers time-dependent travel times.

Since travel times in practice depend on the times of departure, and the amount of driving and duty time available to a truck driver is limited by driving hours regulations, the feasibility of a route depends on the chosen departure times. Furthermore, the costs of a truck driver depend on the total time the truck driver is on duty, i.e., the difference between his departure time and return time at the depot. Therefore, it is profitable to minimize a truck driver's duty time by departure time optimization. Minimizing the duty times also minimizes the total time a vehicle is in use, which is of high value for logistics service providers and distribution firms. The only paper we are aware of that considers minimizing route duration in the objective is of Savelsbergh (1992).

Since more information on historical travel speeds during each time of day is available, time-dependent travel times can now be better estimated. This information is already used by several route-planners on the Internet to provide travel time estimations depending on travel date and time of day to individual drivers. An example is the on-line route planner of the Dutch motorists' organization ANWB. This route planner provides travel time estimations based on historical information on time and location dependent travel speeds using a travel time estimator developed by the Dutch company TNO. Another example that demonstrates the positive impact of using historical travel time data to construct vehicle routes off-line is of Eglese et al. (2006). For their analysis, they use a so-called Road Timetable™ produced by the UK road networking system ITIS Floating Vehicle Data. This Road Timetable™ contains information on time-dependent journey times for a road network based on a record of past road conditions so that travel times can be related to the time of the day, the day of the week and the season of the year.

On top of these new opportunities for high quality off-line travel time estimations, compact duty times in off-line vehicle route plans have a strong positive impact on the overall quality of vehicle routing solutions. This point was stressed by the Dutch company ORTEC (Gromicho; 2008), a key-player in the vehicle routing systems market. Therefore, optimizing departure times off-line is highly profitable in practice.

To the best of our knowledge, this is the first paper which addresses the vehicle departure time optimization problem (VDO). We propose to approach the VDO as a post-processing step of solving a VRPTW. There are two main reasons for this approach.

First, our method can be directly applied in practice. As ORTEC indicated, in practice departure times are optimized after the vehicle routes are constructed (either by routing software or by hand).

Second, it is computationally expensive to incorporate departure time optimization within sophisticated solution methods for the VRP, because a change of departure time at one customer results in different departure times at its succeeding customers. Therefore, the impact of any change in departure time at a customer (caused by, e.g., inter-route customer swap or customer insertion in a route) cannot be calculated in constant time. Furthermore, we are not aware of any paper that has addressed the complex problem of both scheduling and routing vehicles under time-dependent travel times and respecting driving hours regulations. Next, departure time optimization, which has only been applied to models without time-dependent travel times or driving hours regulations is significantly harder under these real-life restrictions. Therefore, an integrated approach of vehicle routing and scheduling with time-dependent travel times, driving hours regulations, and duty time minimization is out of the scope of this paper.

The contribution of this paper is twofold. First, it proposes an exact solution method for the VDO, which is valuable for practice. This practical value is demonstrated by a case study in which departure time optimization reduces duty times by 15% on average. Second, computational experiments on VRPTW benchmarks indicate that vehicle routing models which do not account for either time-dependent travel times or driving hours regulations are in general not feasible in practice. Therefore, this paper clearly shows the need for the development of algorithms that build vehicle routes which incorporate both time-dependent travel times and driving hours regulations.

This paper is organized as follows. Section 2 formally introduces the VDO. Next, Section 3 proposes an ILP-formulation for the VDO and discuss the modeling of the time-dependent travel times in the ILP-formulation. We test the ILP-formulation in Section 4 on problem instances of realistic sizes. Section 5 shows that our approach is flexible with respect to several practical extensions and Section 6 concludes the paper.

2 Problem Description VDO

Since we approach the VDO as a post-processing step of the VRPTW, the input of the problem is a set of customers $i = 0, ..., n + 1$, which need to be serviced in this order. For simplicity, we assume that all customers have to be serviced on one day. Next, since in practice breaks are usually scheduled at customers, we assume that breaks can only be taken at customers. There are exceptions, especially in long distance (international) transports, where breaks are also scheduled at parking lots along the routes. We show in Section 5 how our ILP-formulation can be extended to the case where breaks can also be scheduled at parking lots, and we show how to extend our ILP-formulation to multi-day planning.

Each customer i has given a time window $[e_i, l_i]$ in which its service has to start. The service time of each customer is given by s_i. The travel time between two successive customers i and $i+1$ is given by $c_i(X_i^d)$, where X_i^d is the chosen departure time from customer i. The chosen departure times at the customers are restricted by driving hours regulations.

Since driving hours regulations are country dependent, it might be hard to propose a general formulation covering the driving hours regulations of each country in the world. Since the European driving hours regulations (European Union; 2006) are more restrictive than the North-American ones (Hours-Of-Service Regulations; 2005) and they are valid for all member countries of the European Union, we base our formulation on the European driving hours regulations. These regulations consist of four components:

1. A truck driver is not allowed to drive more than 9 hours (t_{max}) on a day.

2. After driving at most 4.5 hours (b_{cp}) (we call such a period a break checking period), the truck driver must take a break of at least 0.5 hours (b^1_{min}). If this break is smaller than 0.75 hours (b_{total}), then an additional break of at least 0.25 hours (b^2_{min}) must be taken, anywhere during the break checking period. Each time a break checking period ends, a new break checking period is initiated. We call a break of at least b^1_{min} (b^2_{min}) hours a break of type 1 (2). Therefore, each type 1 break is also a type 2 break.

3. The driving hours regulations do not allow service time at customers to be considered as break time. Therefore, if a truck driver takes a break at a customer, he can do that before or after servicing the customer, or both. However, each waiting period before and after servicing a customer should be checked separately whether it can be considered a break of type 1 or 2.

4. A truck driver is not allowed to be on duty for more than 13 hours (d_{max}).

These regulations apply throughout the entire European Union and they are hard constraints. There are some relaxations possible, such as an extension of the total driving time to 10 hours or an extension of the duty time to 15 hours. However, these relaxations are only allowed for a limited number of times (e.g., the extension to 10 hours of driving time is only allowed 2 times a week). We show in Section 5 how to extend our ILP-model to also handle these relaxations.

3 ILP-formulation for the VDO

Since breaks can be taken both before and after servicing a customer, we have to decide for every customer i at what time service starts and at what time the vehicle leaves the customer. Therefore, we introduce the variables X_i^s and X_i^d to indicate the start time of service at customer i and the departure time from customer i, respectively. In addition, we introduce the variables W_i^s and W_i^d to indicate the waiting time of the vehicle directly before and after servicing customer i.

There are two types of breaks, namely breaks of at least b_{min}^1 hours and breaks of at least b_{min}^2 hours. Therefore, we introduce the variables $B_i^{p,l}$, indicating the break time at customer $i = 1, ..., n$, before $(p = s)$ or after $(p = d)$ servicing the customer, and of type $l = 1, 2$. To check whether a waiting time can be considered a break, we also introduce binary variables $Y_i^{p,l}$. If a realization of W_i^p does not exceed b_{min}^l, then the corresponding variables $Y_i^{p,l}$ and $B_i^{p,l}$ are set to zero. Otherwise, the corresponding variable $B_i^{p,l}$ takes the value of W_i^p.

Finally, to ensure that enough breaks are taken during and at the end of each break checking period, we introduce binary variables V_{ij} $(j > i)$. If a break checking period starts at customer i and ends at customer j, then V_{ij} is set to 1. In that case, the break time at customer j must be at least b_{min}^1, and the total break time at customers $k \, (i < k \leq j)$ must be at least b_{total}. This results in the following ILP-formulation:

$$\text{Min } X_{n+1}^s - X_0^d \tag{1}$$

$$X_i^s = X_{i-1}^d + c_{i-1}(X_{i-1}^d) + W_i^s \quad (i = 1, ..., n+1) \tag{2}$$
$$X_i^d = X_i^s + s_i + W_i^d \quad (i = 0, ..., n) \tag{3}$$
$$X_i^s \geq e_i \quad (i = 0, ..., n+1) \tag{4}$$
$$X_i^s \leq l_i \quad (i = 0, ..., n+1) \tag{5}$$
$$W_i^p \geq b_{min}^l Y_i^{p,l} \quad (i = 1, ..., n, l = 1, 2, p = s, d) \tag{6}$$
$$B_i^{p,l} \leq M Y_i^{p,l} \quad (i = 1, ..., n, l = 1, 2, p = s, d) \tag{7}$$
$$B_i^{p,l} \leq W_i^p \quad (i = 1, ..., n, l = 1, 2, p = s, d) \tag{8}$$
$$\sum_{k=0}^{j} c_k(X_k^d) \leq b_{cp} + M \sum_{k=1}^{j} V_{0k} \quad (j = 1, ..., n) \tag{9}$$
$$\sum_{k=i}^{j} c_k(X_k^d) \leq b_{cp} + M \left(\sum_{k=i+1}^{j} V_{ik} + 1 - \sum_{k=0}^{i-1} V_{ki} \right) \tag{10}$$
$$(i = 1, ..., n-1, j = i+1, ..., n)$$

$$\sum_{j=1}^{n} V_{0j} \leq 1 \tag{11}$$
$$\sum_{j=i+1}^{n} V_{ij} \leq \sum_{k=0}^{i-1} V_{ki} \quad (i = 1, ..., n-1) \tag{12}$$
$$B_j^{s,1} + B_j^{d,1} \geq b_{min}^1 V_{ij} \quad (i = 0, ..., n-1, j = i+1, ..., n) \tag{13}$$
$$\sum_{k=i+1}^{j} \left(B_k^{s,2} + B_k^{d,2} \right) \geq b_{total} V_{ij} \quad (i = 0, ..., n-1, j = i+1, ..., n) \tag{14}$$
$$\sum_{k=0}^{n} c_k(X_k^d) \leq t_{max} \tag{15}$$
$$\text{All variables } \geq 0 \tag{16}$$
$$Y_i^{p,l} \in \{0, 1\} \quad (i = 1, ..., n, l = 1, 2, p = s, d) \tag{17}$$
$$V_{ij} \in \{0, 1\} \quad (i = 0, ..., n-1, j = i+1, ..., n) \tag{18}$$

The objective is to minimize a truck driver's duty time. Constraints (2) and (3) define the start service time at and the departure time from each customer. Constraints (4) and (5) ensure that service starts in the given time window. Constraints (6) check whether a waiting period is enough to be considered a break. If not, then $Y_i^{p,l}$ is set to zero and Constraints (7) become tight. Constraints (8) ensure that the break time never exceeds the waiting time. Constraints (9) ensure that the first break checking period does not exceed b_{cp}. If the total driving time between customers 0 and $j+1$ exceeds b_{cp} $\left(\sum_{k=0}^{j} c_k\left(X_k^d\right) > b_{cp}\right)$, then the first break checking period must end at a customer $k, 0 < k < j+1$ $\left(\sum_{k=1}^{j} V_{0k} = 1\right)$. Constraints (10) ensure that the succeeding break checking periods end in time. If a break checking period starts at customer i $\left(\sum_{k=0}^{i-1} V_{ki} = 1\right)$ and the total driving time between customers i and $j+1$ exceeds b_{cp} $\left(\sum_{k=i}^{j} c_k\left(X_k^d\right) > b_{cp}\right)$, then this break checking period must end at a customer $k, i < k < j+1$ $\left(\sum_{k=i+1}^{j} V_{ik} = 1\right)$. Constraints (11) ensure that the first break checking period ends at most once and Constraints (12) ensure that each succeeding break checking period ends at most once. Constraints (13) ensure that a break of at least b_{min}^1 hours is taken at a customer at which a break checking period ends and Constraints (14) ensure that in each break checking period the total break time is at least b_{total}. Finally, Constraint (15) ensures that the total driving time does not exceed t_{max}. Note that the parameter M used in the model does not need to be very large, $M = l_{n+1} - e_0$ is sufficient.

So far, we have modeled the travel time function as a general function that depends on the time of departure. However, in general such a function cannot be written in proper ILP-form. In Section 3.1, we model the time-dependent travel times as a continuous piecewise linear travel time function, and show how to write it in ILP-form.

3.1 Travel time modeling

Several ways of modeling the time-dependent travel times have been proposed in the literature. Malandraki and Daskin (1992) propose a travel time step function. A disadvantage of this approach is that the non-passing property is not satisfied, i.e., if vehicles A and B traverse the same link in the network, and vehicle B departs later than vehicle A, but with a smaller travel time, then vehicle B could arrive earlier than vehicle A. Haghani and Jung (2005) propose a continuous travel time function in which the slope is always greater than -1. In that case, departing later can never result in an earlier arrival. The disadvantage of an arbitrary continuous travel time function is that it does not need to be (piecewise) linear. Therefore, we choose to follow the approach of Ichoua et al. (2003), who propose a travel speed step function for each link in the network. This approach results in a continuous piecewise linear travel time function. Since two vehicles traversing the same link drive with the same speed at any moment of time, the non-passing property is satisfied. Figure 1 shows an example of a speed function; Figure 2 presents the resulting travel time function.

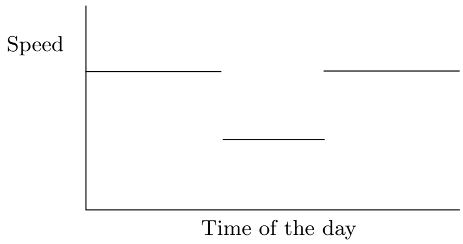

Figure 1: Speed function

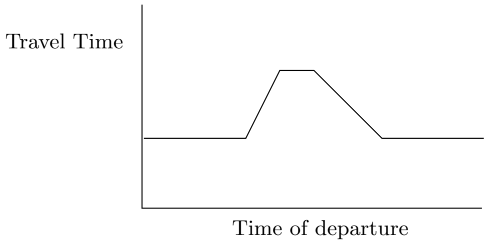

Figure 2: Travel time function

Since the travel time function is piecewise linear, we can write it as m_i different functions $a_{i,r} + b_{i,r}\left(X_i^d - g_{i,r}\right)$, where $g_{i,r}$, $r = 1, ..., m_i$ indicate the times at which the slope of the travel time function changes. Furthermore, $a_{i,r}$ is the travel time at time $g_{i,r}$ and $b_{i,r}$ is the slope of the r^{th} linear function. To determine in which interval $[g_{i,r}, g_{i,r+1}]$ the chosen departure time X_i^d falls, we introduce binary variables $U_{i,r}$ which take value one only if $g_{i,r} \leq X_i^d \leq g_{i,r+1}$. Next, we introduce variables $X_{i,r}^d$ which take the value of X_i^d if the corresponding variable $U_{i,r}$ is one, and zero otherwise. By replacing the function $c_i\left(X_i^d\right)$ by the variable C_i we derive the following ILP-formulation to determine the travel time for departure time X_i^d:

$$\sum_{r=1}^{m_i} U_{i,r} = 1 \qquad (i = 0, ..., n) \tag{19}$$

$$g_{i,r} U_{i,r} \le X^d_{i,r} \qquad (i = 0, ..., n, r = 1, ..., m_i) \tag{20}$$

$$g_{i,r+1} U_{i,r} \ge X^d_{i,r} \qquad (i = 0, ..., n, r = 1, ..., m_i) \tag{21}$$

$$\sum_{r=1}^{m_i} X^d_{i,r} = X^d_i \qquad (i = 0, ..., n) \tag{22}$$

$$C_i \ge a_{i,r} + b_{i,r} \left(X^d_i - g_{i,r} \right) + M \left(U_{i,r} - 1 \right) \quad (i = 0, ..., n, r = 1, ..., m_i) \tag{23}$$

Constraints (19) ensure that exactly one $U_{i,r}$ takes value one. The $U_{i,r}$ with value one and Constraints (20) and (21) force the corresponding variable $X^d_{i,r}$ to be in the interval $[g_{i,r}, g_{i,r+1}]$, and all other variables $X^d_{i,r}$ to be zero. Constraints (22) force the only non-zero $X^d_{i,r}$ to equal X^d_i, and therefore $U_{i,r}$ can only take value one, if $g_{i,r} \le X^d_i \le g_{i,r+1}$. Finally, Constraints (23) are only tight if $U_{i,r}$ equals one, i.e., if $g_{i,r} \le X^d_i \le g_{i,r+1}$, which result in the required travel time functions.

4 Computational Experiments

We test our solution approach for the VDO by a case study and on a set of routes obtained from best known solutions to the well-known Solomon (1987) instances for the VRPTW. The case study demonstrates the applicability and value of our solution approach in practice. The computational experiments on the Solomon routes demonstrate the necessity of constructing vehicle routes which account both for time-dependent travel times and driving hours regulations. We implemented the ILP-formulation of the VDO in Delphi 7 and solved it using CPLEX 11 on a Pentium 4, 3.40GHz CPU and 1.00 GB of RAM.

In order to test the practical impact of our solution approach for the VDO, we apply it to 12 vehicle routes provided by ORTEC. These vehicle routes are constructed for a Dutch client (of ORTEC) and contain between

12 and 36 customer visits per route (with an average of 21 visits). The routes are constructed by ORTEC's vehicle routing software SHORTREC, which contains various state of the art construction and improvement (local search) heuristics. These heuristics are adapted for practical use, implying that they account for several realistic constraints, such as time windows and driving hours regulations, and that the quality of solutions are measured in all relevant cost factors, such as number of vehicles used, total distance traveled, and total duty time. The routes do not account for time-dependent travel times, since such travel times are not integrated in the software yet. ORTEC has also implemented a greedy approach based on binary search to solve the VDO as a post-processing step of constructing the vehicle routes.

Table 1 presents the duty times of the 12 routes before solving the VDO, after solving the VDO with ORTEC's greedy approach, and with our approach. The average reduction of the duty times by the greedy approach is 75 minutes. Our solutions reduce these duty times by an additional 32 minutes, on average. This implies that departure time optimization as a post-processing of constructing the vehicle routes reduces duty times by 107 minutes, on average, which is 15% of the total duty time. In comparison with ORTEC's greedy approach, our approach reduces an additional 5.8% of the total duty time. Note that all other relevant cost factors (e.g., number of vehicles used, total distance traveled) remain the same. Therefore, the duty time reductions can be realized without introducing extra costs.

To stress the importance of incorporating time-dependent travel times and driving hours regulations in methods for constructing vehicle routes, we test our solution approach for the VDO on a selection of the 100-customer problem instances developed by Solomon (1987). We use those problem instances for which best known solutions identified by heuristics can be obtained from the literature. The routes obtained from these solutions form the problem instances for the VDO. Our preference was to test the VDO on routes obtained from good solutions to TDVRP instances, since these routes already account for time-dependent travel times. Unfortunately, the involved authors can no longer provide these routes (Ichoua et al.; 2003; Fleischmann et al.; 2004; Haghani and Jung; 2005). However, we shall demonstrate that even if one of the timing restrictions, time-dependent travel times or driving hours regulations, is neglected during the construction of the vehicle routes, then in many cases it is not possible to find feasible departure schedules. This implies that the routes are not applicable in practice.

The Solomon problem instances are categorized into 3 types of instances:

Table 1: Duty times (minutes) for vehicle routes from practical case

	VDO approach		
Route	No	Greedy	Ours
1	634	518	443
2	610	539	537
3	754	729	729
4	655	655	641
5	851	799	769
6	826	798	798
7	919	799	798
8	469	405	359
9	357	346	300
10	813	710	710
11	857	731	588
12	858	678	651

1) C-instances, where customer locations are clustered, 2) R-instances, where customers are uniformly randomly located in a square, and 3) RC-instances, where 50 percent of the customers are clustered and 50 percent are uniformly randomly located. Each customer is given a hard time window in which its service must start. The time window at the depot indicates the earliest feasible departure time and the latest feasible return time at the depot. Furthermore, some of the problem instances have a relatively large time window at the depot and vehicles with a relatively large capacity, resulting in large vehicle routes (25 up to 50 customers), while other instances have a relatively small time window at the depot, resulting in small vehicle routes (about 10 customers). Since the number of customers visited in a vehicle route defines the input size of the VDO, we discern small and large vehicle routes. This distinction allows us to investigate the impact of the input size of the VDO on the required computation time. The number of customers visited in a vehicle route ranges from 4 to 51 customers. We categorize the VDO problem instances into small ($\leq$ 20 customers) and large ($>$ 20 customers) problem instances.

The travel speed in the networks of the Solomon instances equals one. Therefore, the travel times in the Solomon instances equal the euclidean distance between the customer locations. Since the travel speed is time-independent, we develop speed patterns, such that the average travel speed

Table 2: Speed Patterns

Type of Congestion \ Time	6-7:00	7-9:00	9-17:00	17-19:00	19-20:00
Light	1.08	0.81	1.08	0.81	1.08
Medium	1.17	0.58	1.17	0.58	1.17
Heavy	1.27	0.32	1.27	0.32	1.27

remains one. This methodology is similar to the one proposed by Ichoua et al. (2003). We define the time window at the depot from 6:00 am until 8:00 pm and we assume that the morning traffic peak causes congestion from 7:00 am until 9:00 am, and the evening traffic peak from 5:00 pm until 7:00 pm. Furthermore, we distinguish between light, medium, and heavy congestion. These three types of congestion cause speed drops of 25, 50, and 75 percent, respectively. Table 2 presents the resulting speed patterns.

The VDO problem instances are composed of the vehicle routes resulting from the best known solutions to the Solomon instances and the travel speed patterns in Table 2. Furthermore, we set $b_{min} = 0.25$, $b_{total} = 0.75$, $b_{cp} = 4.5$, and $t_{max} = 9$, corresponding with the European driving hours regulations. Since the original Solomon instances do not account for driving hours regulations and time-dependent travel times, we investigate whether the developed routes allow feasible VDO-solutions. Since we are testing the impact of two different realistic factors in vehicle routing, we develop two different test scenarios: in Scenario 1 we *do not* consider driving hours regulations and in Scenario 2 we *do* consider driving hours regulations. In both scenarios, we solve the VDO for each of the three speed drops as described before, as well as the case in which there is no speed drop at all. This allows us to investigate the impact of driving hours regulations on vehicle routes in congestion free networks. Tables 3 and 4 present results on computation times and percentage of infeasible VRP routes by optimizing the departure times for Scenarios 1 and 2, respectively.

The computation times are small enough for practical use. The maximum computation time over all instances is 2.3 seconds (for Scenario 1 even 0.2 seconds) and the average is less than 0.1 seconds. Therefore, our approach to solve the VDO as a post-processing step of a VRPTW is feasible in practice.

The solution methods for the original VRP-instances do not account for time-dependent travel times and driving hours regulations, and as a conse-

Table 3: Results Scenario 1: no driving hours regulations

Problem Size	# Instances	Congestion Type	CPU (s)	VRP route Infeasible
		No	0.01	0.00%
Small[a]	164	Light	0.02	2.44%
		Medium	0.01	44.51%
		Heavy	0.01	84.76%
		No	0.07	0.00%
Large[b]	25	Light	0.07	4.00%
		Medium	0.05	44.00%
		Heavy	0.03	76.00%
		No	0.02	0.00%
Average	189	Light	0.03	2.65%
		Medium	0.02	44.44%
		Heavy	0.01	83.60%

[a]All routes in the best known solutions of instances R103 and RC106 (Li and Lim; 2003), R104, R107, R109, R111 and RC107 (Shaw; 1997), R108, R110 and RC105 (Berger and Barkaoui; 2004), and RC101, RC102, RC103, RC104 and RC108 (Czech and Czarnas; 2002)

[b]All routes in the best known solutions of instances R211 (Rochat and Taillard; 1995), and RC201, RC204, RC205,RC206, RC202, RC203 and RC207 (Czech and Czarnas; 2002)

Table 4: Results Scenario 2: with driving hours regulations

Problem Size	# Instances	Congestion Type	CPU (s)	VRP route Infeasible
		No	0.04	0.00%
Small	164	Light	0.06	24.39%
		Medium	0.02	65.24%
		Heavy	0.01	90.24%
		No	0.32	4.00%
Large	25	Light	0.32	16.00%
		Medium	0.22	52.00%
		Heavy	0.16	76.00%
		No	0.08	0.53%
Average	189	Light	0.09	23.28%
		Medium	0.05	63.49%
		Heavy	0.03	88.36%

quence the obtained routes are often too tight with respect to the time windows to schedule mandatory breaks. It generally holds that heavier traffic congestion results in fewer feasible vehicle routes. Therefore, vehicle routing methods should account for time-dependent travel times. However, this is not sufficient to obtain vehicle routes that can be used in practice under driving hours regulations. Table 5 presents the percentage of feasible vehicle routes with respect to time windows and time-dependent travel times, but that turn out to be infeasible when driving hours regulations are respected. Therefore, some 30 percent of the routes that are feasible with respect to time-dependent travel times, but that ignore driving hours regulations, fail in practice. This problem is clearly caused by the methods that build the vehicle routes; it does not affect the applicability of the VDO in practice. As we shall argue in the remainder of this section, it is not straightforward to overcome this problem.

Table 5: Relative decrease of # feasible vehicle routes when driving hours regulations have to be respected

Congestion Type	Reduction of # feasible vehicle routes
No	0.53%
Light	21.20%
Medium	34.29%
Heavy	29.03%

First, slack time could be added to the original problem instances, such that time is reserved for scheduling mandatory breaks after the vehicle routes have been developed. To keep the proposed solution methods in the VRP-literature directly applicable, this slack time should be spread out evenly over the travel times between (or service times at) the customers. We tested this approach by adding one sixth of slack travel time. At least one sixth of slack travel time is required, because the total travel time in a break checking period does not exceed 4.5 hours, while 45 minutes of break time needs to be scheduled in this period. Computational experiments show that this approach works well for light congestion (the percentage of infeasible vehicle routes reduces from 64.02% to 2.12%), however, with medium and heavy congestion the percentage of infeasible routes remains rather large (14.29% and 46.56%, respectively). A drawback of this approach is that built-up slack might be lost when truck drivers have to wait at customers before they can start service. This is one of the reasons that many routes remain infeasible in case of medium and heavy congestion.

Second, one could argue that the infeasibility problem is caused by the tightness of optimal solutions. We therefore also tested less sophisticated methods to develop the vehicle routes, resulting in worse VRP-solutions with respect to the overall objective, but with possibly less tight routes with respect to the time windows. We tested this approach with a straightforward nearest neighbor heuristic. The results show that the percentage of infeasible vehicle routes decreases (from 64.02% to 47.51%), but the number of vehicle routes increases dramatically (from 189 to 261). Although the number of feasible vehicle routes increases, the total number of customers in all feasible vehicle routes decreases (from 845 to 796). Therefore, the infeasibility is not caused by the solution method, but by the ignorance of time-dependent travel times and driving hours regulations.

Finally, new vehicle routing methods could be developed that account for time-dependent travel times and driving hours regulations. Since the first two solution approaches are not suitable, we have to develop new solution methods in order to obtain high quality vehicle routes that are feasible in practice.

5 Model Extensions

The ILP-formulation proposed in Section 3 assumes one-day planning and that breaks are only taken at customers. There are several practical cases in which it is more convenient to extend the formulation to a multi-day planning or to assume that breaks can also be taken at parking lots. We demonstrate that these extensions can easily be incorporated in our ILP-formulation.

For multi-day planning, some extra restrictions are imposed by the driving hours regulations. Both the European and North American driving hours regulations impose a maximum on the total driving time and the total working time on a day, after which a rest has to be taken. More formally, after driving at most t_{max} hours and being on duty for at most d_{max} hours, a rest of at least t_{rest} hours has to be taken. Also, a maximum is imposed on the total driving and working time in an entire week. We show how the ILP-formulation of Section 3 can be extended to one-week planning.

First, in Constraint (15), t_{max} must be replaced by the maximum driving time in a week. Next, to check whether a waiting time at a customer can be considered a rest, we introduce variables $B_i^{p,rest}, p = s, d$ and binary variables $Y_i^{p,rest}$, and we add the following constraints to the ILP-formulation:

$$W_i^p \geq t_{rest} Y_i^{p,rest} \qquad (i = 1, ..., n, p = s, d) \tag{24}$$

$$B_i^{p,rest} \leq M Y_i^{p,rest} \qquad (i = 1, ..., n, p = s, d) \tag{25}$$

$$B_i^{p,rest} \leq W_i^p \qquad (i = 1, ..., n, p = s, d) \tag{26}$$

Next, we need to check whether the driving (duty) time does not exceed the maximum driving (duty) time on each day before a night's rest is taken. Therefore, we introduce the notion of rest checking period which has the following three properties: 1) Each rest checking period ends with a night's rest, 2) in each rest checking period the driving and duty time do not exceed the maximum driving and duty time, and 3) each time a rest checking period ends, a new rest checking period is initiated. Next, we introduce binary variables V_{ij}^{rest} which are set to 1 if a rest period starts at customer i and ends at customer j. To ensure that the driving time does not exceed the maximum driving time in each rest checking period, and each rest checking period ends with a rest of at least t_{rest} hours, we add the following constraints:

$$\sum_{k=0}^{j} c_k(X_k^d) \leq t_{max} + M \sum_{k=1}^{j} V_{0k}^{rest} \qquad (j = 1, ..., n) \tag{27}$$

$$\sum_{k=i}^{j} c_k(X_k^d) \leq t_{max} + M \left(\sum_{k=i+1}^{j} V_{ik}^{rest} + 1 - \sum_{k=0}^{i-1} V_{ki}^{rest} \right) \tag{28}$$

$$(i = 1, ..., n-1, j = i+1, ..., n)$$

$$\sum_{j=1}^{n} V_{0j}^{rest} \leq 1 \tag{29}$$

$$\sum_{j=i+1}^{n} V_{ij}^{rest} \leq \sum_{k=0}^{i-1} V_{ki}^{rest} \qquad (i = 1, ..., n-1) \tag{30}$$

$$B_j^{s,rest} + B_j^{d,rest} \geq t_{rest} V_{ij}^{rest} \qquad (i = 0, ..., n-1, j = i+1, ..., n) \tag{31}$$

Ensuring that the duty time does not exceed the maximum duty time during each rest checking period can be done via similar constraints. The only difference is that waiting times and service times also add to the total duty time. Therefore, both the arrival time and the end of service time at each customer is a possible moment for exceeding the total duty time. Since there are two possible moments at each customer for starting (ending) a rest checking period, the total number of possible rest checking periods is four times the number of possible rest checking periods for the case with maximum driving time. Therefore, we need four times the number of binary variables V_{ij}^{rest} to indicate when a rest checking period starts and when it ends. Similarly, we need two times the constraints of type (27) and (30), and four times the constraints of type (28) and (31), to ensure that each rest checking period ends with a break of t_{rest}, the total duty time in the rest checking period does not exceed d_{max}, and each time a rest checking period ends, a new rest checking period is initiated.

To account for the possibility of extending the driving time twice a week, we add binary variables $E_i, i = 0, ..., n$, which take value one if a new daily driving period starts at customer i, and the total driving time of this period can be extended to 10 hours. To ensure that the total number of daily driving time extensions does not exceed two, we add the constraint $\sum_{i=0}^{n} E_i \leq 2$. Next, we ensure that $E_i, i > 0$ can only take value 1 if a new daily driving period starts at customer i by adding constraints $E_i \leq \sum_{k=0}^{i-1} V_{ki}^{rest}, i = 1, ..., n$. Finally, to allow for the driving time extensions of 1 hour, we adjust Constraints (27) and (28):

$$\sum_{k=0}^{j} c_k(X_k^d) \leq t_{max} + E_0 + M \sum_{k=1}^{j} V_{0k}^{rest} \qquad (j = 1, ..., n) \tag{32}$$

$$\sum_{k=i}^{j} c_k(X_k^d) \leq t_{max} + E_i + M \left(\sum_{k=i+1}^{j} V_{ik}^{rest} + 1 - \sum_{k=0}^{i-1} V_{ki}^{rest} \right) \tag{33}$$
$$(i = 1, ..., n-1, j = i+1, ..., n)$$

To incorporate the possibility of taking a break at parking lots along the route, we can simply model these parking lots as customers with zero service time and maximum time window (i.e., $[e_o, l_{n+1}]$).

6 Conclusions

We introduced the VDO and approached it as a post-processing step of solving a VRPTW. We proposed an ILP-formulation for the VDO which is flexible with respect to several practical extensions. This flexibility was demonstrated while writing this paper, as the European driving hours regulations changed and we were able to quickly adapt the ILP-formulation to the new regulations.

The computational experiments show that the VDO can be solved to optimality within practical computation times. Furthermore, a practical case showed a duty time reduction of 15% on average. Also a greedy approach used in practice was improved by reducing duty times by 6% on average. Such duty time reductions imply significant cost savings for logistics service providers and distribution firms.

Finally, the computational experiments show that VRP-routes will only be of practical use if driving hours regulations and time-dependent travel times are accounted for during the development of vehicle routes. We argued that the most appropriate way to solve this problem is to develop new vehicle routing methods. Since it is computationally expensive to account for time-dependent travel times, driving hours regulations, and departure time optimization within vehicle routing methods, developing such a method is a topic for further research.

Acknowledgment

This work was financially supported by Stichting Transumo through the project ketensynchronisatie. We thank ORTEC for providing the case data. We also thank the anonymous referees for their helpful comments to improve this paper.

References

Archetti, C. and Savelsbergh, M. W. P. (2007). The trip scheduling problem, *Transportation Science* . To appear.

Berger, J. and Barkaoui, M. (2004). A parallel hybrid genetic algorithm for the vehicle routing problem with time windows, *Computers & Operations Research* **31**(12): 2037–2053.

Czech, Z. J. and Czarnas, P. (2002). Parallel simulated annealing for the vehicle routing problem with time windows, *10th Euromicro Workshop on Parallel, Distributed and Network-based Processing*, pp. 376–383.

Eglese, R., Maden, W. and Slater, A. (2006). A road timetable™ to aid vehicle routing and scheduling, *Computers & Operations Research* **33**(12): 3508 – 3519.

European Union (2006). *Regulation (EC) No 561/2006 of the European Parliament and of the Council of 15 March 2006 on the harmonisation of certain social legislation relating to road transport and amending Council Regulations (EEC) No 3821/85 and (EC) No 2135/98 and repealing Council Regulation (EEC) No 3820/85.* Official Journal of the European Union L 102/1, 11.04.2006.

Fleischmann, B., Gietz, M. and Gnutzmann, S. (2004). Time-varying travel times in vehicle routing, *Transportation Science* **38**(2): 160–173.

Goel, A. (2008). Vehicle scheduling and routing with drivers' working hours, *Transportation Science* **online**: 1–10.

Gromicho, J. (2008). Personal communication, *Head of Algorithmic Development ORTEC* .

Haghani, A. and Jung, S. (2005). A dynamic vehicle routing problem with time-dependent travel times, *Computers & Operations Research* **32**(11): 2959–2986.

Hill, A. V. and Benton, W. C. (1992). Modelling intra-city time-dependent travel speeds for vehicle scheduling problems, *The Journal of the Operational Research Society* **43**(4): 343–351.

Hours-Of-Service Regulations (2005). *Federal Motor Carrier Safety Administration* . `http://www.fmcsa.dot.gov/rules-regulations/topics/hos/hos-2005.htm`.

Ichoua, S., Gendreau, M. and Potvin, J. Y. (2003). Vehicle dispatching with time-dependent travel times, *European Journal of Operational Research* **144**(2): 379–396.

Li, H. and Lim, A. (2003). Local search with annealing-like restarts to solve the VRPTW, *European Journal of Operational Research* **150**(1): 115–127.

Malandraki, C. and Daskin, M. S. (1992). Time dependent vehicle routing problems: Formulations, properties and heuristic algorithms, *Transportation Science* **26**(3): 185–200.

Rochat, Y. and Taillard, E. D. (1995). Probabilistic diversification and intensification in local search for vehicle routing, *Journal of Heuristics* **1**(1): 147–167.

Savelsbergh, M. W. P. (1992). The vehicle routing problem with time windows: minimizing route duration, *ORSA Journal on Computing* **4**(2): 146–154.

Shaw, P. (1997). A new local search algorithm providing high quality solutions to vehicle routing problems, *Working paper, University of Strathclyde, Glasgow, Scotland* .

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

Toth, P. and Vigo, D. (2002). *The vehicle routing problem*, SIAM Monographs on Discrete Mathematics and Applications, Philadelphia.

Van Woensel, T., Kerbache, L., Peremans, H. and Vandaele, N. (2008). Vehicle routing with dynamic travel times: A queueing approach, *European Journal of Operational Research* **186**(3): 990–1007.

Xu, H., Chen, Z. L., Rajagopal, S. and Arunapuram, S. (2003). Solving a practical pickup and delivery problem, *Transportation Science* **37**(3): 347–364.