Optimizing departure times in vehicle routes

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. However, in practice, temporal traffic congestions make that such solutions are not optimal with respect to minimizing the total duty time. Furthermore, VRPTW solutions do not account for complex driving hours regulations, which severely restrict the daily travel time available for a truck driver. To deal with these problems, we consider the vehicle departure time optimization (VDO) problem as a post-processing step of solving a VRPTW. We propose an ILP-formulation that minimizes the total duty time. The obtained solutions are feasible with respect to driving hours regulations and they account for temporal traffic congestions by modeling time-dependent travel times. For the latter, we assume a piecewise constant speed function. Computational experiments show that problem instances of realistic sizes can be solved to optimality within practical computation times. Furthermore, duty time reductions of 8 percent can be achieved. Finally, the results show that ignoring time-dependent travel times and driving hours regulations during the development of vehicle routes leads to many infeasible vehicle routes. Therefore, vehicle routing methods should account for these real-life restrictions.

Keywords: Time-dependent travel times; Driving hours regulations; Duty times; Vehicle scheduling; ILP-formulation

*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

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 (Toth and Vigo (2002) present an extensive overview of the VRP and solution methods). However, two real-life restrictions have hardly been discussed, namely temporary traffic congestions 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 congestions form a major problem for businesses such as logistical service providers and distribution firms. Due to temporary traffic congestions, vehicles arrive too late at customers and driving hours regulations are violated. In practice, travel times do not only depend on distance, but also on the time of departure. For this purpose, Malandraki and Daskin (1992) introduces the time dependent vehicle routing problem (TDVRP). Furthermore, Hill and Benton (1992), Ichoua et al. (2003), Fleischmann et al. (2004), Haghani and Jung (2005) 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) and Archetti and Savelsbergh (2007). Archetti and Savelsbergh develop a polynomial time algorithm for the problem of finding a feasible driver schedule, after it has been decided which customers the driver has to serve and in which order. However, they do not account for time-dependent travel times. Furthermore, their algorithm is capable of handling driving hours regulations concerning night's rest, but they do not account for complex driving hours regulations regarding the scheduling of (lunch-)breaks during the day.

Since travel times 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 logistical service providers and distribution firms. The only paper we are aware of that considers minimizing route duration as objective is of Savelsbergh (1992).

To the best of our knowledge, this is the first paper which addresses the vehicle departure time optimization problem (VDO). Since a change of departure time at one customer results in different departure times at its succeeding customers, it is computationally expensive to incorporate departure time optimization within sophisticated solution methods for the VRP, like local search methods. Therefore, we approach the VDO as a post-processing step of a VRPTW. Consequently, the input of the VDO is a vehicle route in which a set of customers has to be visited in the given order. In practice, the VDO is solved as a post-processing step of a VRP. The Dutch company ORTEC, a key-player in the vehicle routing systems market, also suggested us to approach the VDO as such.

This paper is organized as follows. In Section 2, we formally introduce the VDO. Next, in Section 3, we propose 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. In Section 5, we show 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 a 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. In Section 5, we show that our ILP-formulation can easily be extended 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 (2006) are more restrictive than the North-American ones (Hours-Of-Service Regulations; 2005), 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 to consider service time at customers 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 and/or 2.

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

In practice, breaks are usually scheduled at customers. However, there are exceptions, especially in long distance (international) transports, where breaks are also scheduled at parking lots along the routes. For simplicity, we assume that breaks can only be taken at customers. In Section 5, we show how our ILP-formulation can be extended to the case where breaks can also be scheduled at parking lots.

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^s_i and X^d_i 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^s_i

and W_i^d to indicate the waiting time of the vehicle 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:

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

$$X_i^s = X_{i-1}^d + c_{i-1}(X_{i-1}^d) + W_i^s \qquad (\forall i = 1, ..., n+1) \tag{2}$$

$$X_i^d = X_i^s + s_i + W_i^d \qquad (\forall i = 0, ..., n) \tag{3}$$

$$X_i^s \geq e_i \qquad (\forall i = 0, ..., n+1) \tag{4}$$

$$X_i^s \leq l_i \qquad (\forall i = 0, ..., n+1) \tag{5}$$

$$W_i^p \geq b_{min}^l Y_i^{p,l} \qquad (\forall i = 1, ..., n, l = 1, 2, p = s, d) \tag{6}$$

$$B_i^{p,l} \leq M Y_i^{p,l} \qquad (\forall i = 1, ..., n, l = 1, 2, p = s, d) \tag{7}$$

$$B_i^{p,l} \leq W_i^p \qquad (\forall i = 1, ..., n, l = 1, 2, p = s, d) \tag{8}$$

$$\sum_{k=0}^{j} c_k(X_k^d) \leq b_{cp} + M \sum_{k=1}^{j} V_{0k} \qquad (\forall j = 1, ..., n) \tag{9}$$

$$\sum_{k=i}^{j} c_k(X_k^d) \leq b_{cp} + M \left(\sum_{k=i+1}^{j} V_{ik} + 1 - \sum_{k=0}^{i-1} V_{ki} \right) \tag{10}$$

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

$$\sum_{j=1}^{n} V_{0j} \leq 1 \tag{11}$$

$$\sum_{j=i+1}^{n} V_{ij} \leq \sum_{k=0}^{i-1} V_{ki} \quad \forall i = 1, ..., n-1 \tag{12}$$

$$B_j^{s,1} + B_j^{d,1} \geq b_{min}^1 V_{ij} \quad (\forall i = 0, ..., n-1, j = i+1, ..., n) \tag{13}$$

$$\sum_{k=i+1}^{j} \left(B_k^{s,2} + B_k^{d,2} \right) \geq b_{total} V_{ij} \quad (\forall i = 0, ..., n-1, j = i+1, ..., n) \tag{14}$$

$$\sum_{k=0}^{n} c_k(X_k^d) \leq t_{max} \tag{15}$$

$$\text{All variables } \geq 0 \tag{16}$$

$$Y_i^{p,l} \in \{0, 1\} \quad (\forall i = 1, ..., n, l = 1, 2, p = s, d) \tag{17}$$

$$V_{ij} \in \{0, 1\} \quad (\forall i = 0, ..., n-1, j = i+1, ..., n) \tag{18}$$

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 will never exceed 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 function that depends on the time of departure. However, the ILP-formulation is only valid if this travel time function is linear. In Section 3.1, we model the time-dependent travel times as a linear travel time function, and 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 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 piecewise linear travel time function. Since two vehicles traversing the same link will drive 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.

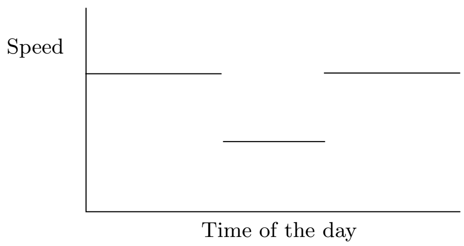

Figure 1: Speed function

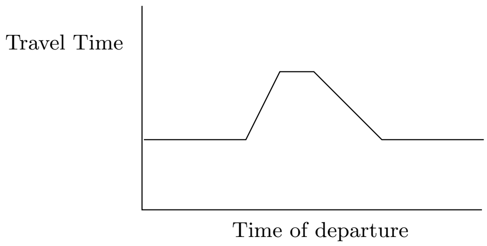

Figure 2: Travel time function

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 $g_{i,r}$, $r = 1, ..., m_i$, indicates 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:

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

$$g_{i,r}U_{i,r} \leq X_{i,r}^d \qquad \forall i = 0, ..., n, r = 1, ..., m_i \tag{20}$$

$$g_{i,r+1}U_{i,r} \geq X_{i,r}^d \qquad \forall i = 0, ..., n, r = 1, ..., m_i \tag{21}$$

$$\sum_{r=1}^{m_i} X_{i,r}^d = X_i^d \qquad \forall i = 0, ..., n \tag{22}$$

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

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

4 Computational Experiments

We test 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 can be obtained from the literature. The routes obtained from these solutions form the problem instances for the VDO. We choose to test the VDO on these problem instances, because these vehicle routes are widely considered as 'good' vehicle routes in the VRP-literature. 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.

The Solomon problem instances are categorized in C-instances, where customer locations are clustered, R-instances, where customers are uniformly randomly located in a square, and 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 one vehicle route defines the input size of the VDO, we make a distinction between 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 (< 21 customers), medium ($21-35$ customers), and large (> 35 customers) problem instances.

The travel times in the Solomon instances equal the euclidean distance between the customer locations, i.e., the travel speed in the network equals one. Since this travel speed is time-independent, we develop speed patterns, such that the average travel speed remains one. This methodology is similar to the methodology proposed in 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 make a distinction between light, medium, and heavy congestion. These three types of congestion cause speed drop downs of 33, 50, and 75 percent, respectively. Table 1 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 1. Furthermore, we set $b_{min} = 0.25$, $b_{total} = 0.75$, $b_{cp} = 4.5$,

Table 1: Speed Patterns

Type of Congestion	6-7:00	7-9:00	9-17:00	17-19:00	19-20:00
Light	1.10	0.74	1.10	0.74	1.10
Medium	1.17	0.58	1.17	0.58	1.17
Heavy	1.27	0.32	1.27	0.32	1.27

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 first investigate if the developed routes allow feasible VDO-solutions. Next, since the objective in the original Solomon instances is to minimize travel distance, we cannot compare the VDO objective values with the original objective values. However, the solutions to the Solomon instances are developed from the earliest feasible departure time. To get an impression on the duty time reductions that can be achieved by optimizing the departure times, we solve the VDO a second time with setting the departure time at the depot at zero. With this approach, the return time at the depot is minimized, given a departure time of zero at the depot, and respecting driving hours regulations and time-dependent travel times. The resulting objective values are lower bounds on the truck driver's duty times if they depart at the suggested earliest feasible departure time. Table 2 presents results on computation times, percentage of infeasible VRP routes, and duty time reductions achieved by optimizing the departure times. These duty time reductions are with respect to the lower bounds on the duty times if the vehicles depart at time zero. Averages are presented over the vehicle routes that allow feasible solutions.

The results on computation time show that even large problem instances can be solved within practical computation times. The maximum computation time over all instances is 2.016 seconds. Therefore, it is not necessary to, e.g., add valid inequalities to the ILP-formulation to speed up computation times.

The duty time reductions imply significant cost savings for hiring truck drivers and significant reductions of the total times the vehicles are in use. Therefore, the method is a valuable tool in gaining high quality vehicle routing solutions. These results also stress the importance of evaluating intermediate VRP-solutions with respect to the duty times. For example, in local

Table 2: Results

Problem Size	# Instances	Congestion Type	CPU (s)	VRP route Infeasible	Improvement [a]
Small [b]	164	Light	0.056	67.07 %	6.92 %
		Medium	0.053	74.39 %	7.09 %
		Heavy	0.039	81.10 %	8.69 %
Medium [c]	20	Light	0.211	30.00 %	10.81 %
		Medium	0.248	50.00 %	12.52 %
		Heavy	0.217	70.00 %	14.15 %
Large [d]	5	Light	0.675	100.00 %	-
		Medium	0.975	80.00 %	0.37 %
		Heavy	0.928	100.00 %	-
Average	189	Light	0.089	64.02 %	7.72 %
		Medium	0.098	71.96 %	7.99 %
		Heavy	0.081	80.42 %	9.58 %

[a] The average duty time reduction with respect to departing at time zero for the routes that allow feasible VDO-solutions

[b] 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)

[c] All routes in the best known solutions of instances RC201, RC204, RC205 and RC206, and two of the routes in the best known solutions of instances RC202, RC203 and RC207 (Czech and Czarnas; 2002)

[d] All routes in the best known solution of instance R211 (Rochat and Taillard; 1995), and 1 of the routes in the best known solutions of instances RC202, RC203 and RC207 (Czech and Czarnas; 2002)

search methods many times the best neighborhood solution needs to be selected. Since departure time optimization yields duty time reductions of 8 % on average, departure time optimization is a necessary step to evaluate neighborhood solutions.

The duty time reductions are even larger if we compare the optimal VDO solutions with a strategy which is often used in practice: departure ASAP at each customer. With this strategy, breaks are postponed as much as possible, such that the departure time is locally minimized at each customer. This approach leads to 10 % more infeasible vehicle routes on average compared to the ILP-solutions, and duty times are on average 10 % larger than the optimal VDO solutions.

The solution methods for the original VRP-instances do not account for time-dependent travel times and driving hours regulations, and as a consequence the obtained routes are often too tight with respect to the time windows to schedule mandatory breaks. Therefore, many of the developed vehicle routes cannot be used in practice. This problem is clearly caused by the methods that develop the vehicle routes; it does not affect the applicability of the VDO. As we shall argue, it is not straightforward to overcome this problem.

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 seems to be needed, 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. Another drawback is that in almost all problem instances too much slack is added. An example is when the total driving time in a vehicle route is less than 4.5 hours: slack travel time is introduced, while no break needs to be scheduled. Therefore, in case of light congestion slack travel time seems an appropriate way to obtain many feasible vehicle routes, however, the quality of the VRP-solution with respect to the overall objective (small number of vehicles, short duty times) reduces.

Second, one could use 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 slightly (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 increase, the total number of customers in all feasible vehicle routes decrease (from 845 to 796). Therefore, this approach seems to be inappropriate to solve this problem.

Finally, new vehicle routing methods could be developed that account for time-dependent travel times and driving hours regulations. This seems to be the most appropriate choice, because with such methods we can develop both feasible and high quality vehicle routes with respect to the overall objective.

5 Model Extensions

The ILP-formulation proposed in Section 3 assumes a one-day planning, and assumes 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.

If a multi-day planning is concerned, 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 a 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:

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

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

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

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:

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

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

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

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

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

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 also 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 underlined when writing this paper the European driving hours regulations changed. We could quickly adapt the ILP-formulation to the new driving hours regulations.

The computational experiments show that the VDO can be solved to optimality within practical computation times. Furthermore, duty time reductions of 8 % can be achieved by optimizing the departure times. Such duty time reductions imply significant cost savings for logistical 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.

References

Archetti, C. and Savelsbergh, M. (2007). The trip scheduling problem, *Working paper* . http://www2.isye.gatech.edu/~mwps/publications/dotfeas11.pdf.

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.

European driving hours regulations (2006). *Official Journal of the European Union* pp. 1–13. http://eur-lex.europa.eu/LexUriServ/site/en/oj/2006/l_102/l_10220060411en00010013.pdf.

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

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. (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. (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. (1992). Time dependent vehicle routing problems: Formulations, properties and heuristic algorithms, *Transportation science* **26**(3): 185–200.

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

Savelsbergh, M. (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.

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