A Dynamic Programming Heuristic for the Vehicle Routing Problem with Time Windows and the European Community Social Legislation

A. Leendert Kok

Operational Methods for Production and Logistics, University of Twente,

P.O. Box 217, 7500AE Enschede, Netherlands, a.l.kok@utwente.nl,

http://www.mb.utwente.nl/ompl

C. Manuel Meyer, Herbert Kopfer

Chair of Logistics, University of Bremen, Wilhelm-Herbst-Strasse 5, 28359 Bremen,

Germany, cmmeyer@uni-bremen.de, kopfer@uni-bremen.de

http://www.logistik.uni-bremen.de

J. Marco J. Schutten

Operational Methods for Production and Logistics, University of Twente,

P.O. Box 217, 7500AE Enschede, Netherlands, m.schutten@utwente.nl,

http://www.mb.utwente.nl/ompl

Abstract

In practice, apart from the problem of vehicle routing, schedulers also face the problem of finding feasible driver schedules complying with complex restrictions on drivers' driving and working hours. To address this complex interdependent problem of vehicle routing and break scheduling, we propose a restricted dynamic programming heuristic for the vehicle routing problem with time windows and the full European social legislation on drivers' driving and working hours. The problem we consider includes all rules in this legislation, whereas in the literature only a basic set of rules has been addressed. In addition to this basic set of rules, the legislation contains a set of modifications that allow for more flexibility. To include the legislation in the restricted dynamic programming heuristic, we propose a break scheduling heuristic. Computational results show that our method finds solutions to benchmark instances - which only consider the basic set of rules - with 18% less vehicles and 5% less travel distance than state of the art approaches. Moreover, our results are obtained with significant less computational effort. Furthermore, the results show that including a set of rules on drivers' working hours - which has been generally ignored in the literature - has a significant impact on the resulting vehicle schedules: 3.9% more vehicle routes and 1.0% more travel distance are needed. Finally, using the modified rules of the legislation leads to an additional reduction of 4% in the number of vehicles and of 1.5% regarding the travel distance. Therefore, the modified rules should be exploited in practice.

Keywords: Vehicle Routing and Scheduling; Restricted Dynamic Programming; Break Scheduling; EC Regulations on Driving Time; Drivers' Working Hours

1 Introduction

In all member countries of the European Union and in many other countries, legislation on driving and working hours of persons engaged in road transportation is effective. In the European Union, driving hours are restricted by European Community (EC) Regulation No 561/2006. Moreover, Directive 2002/15/EC restricting drivers' working hours has been implemented into national laws in most member countries of the European Union. These legal acts have to be taken into account by schedulers when establishing vehicle tours. As their negligence can be fined severely, these acts have an enormous impact on the design of vehicle tours in practice. The problem which arises here is a problem of combined vehicle routing and break scheduling. In the literature, however, only a few studies on vehicle routing including breaks and rest periods can be found. In all of these studies, only parts of the mandatory legislation are included, resulting in vehicle schedules which do not comply with the legal requirements.

Gietz (1994) investigates a vehicle routing problem (VRP) with breaks modeled as fictitious customers. Rochat and Semet (1994) use a similar approach. Stumpf (1998) includes driving time restrictions specified by the former Regulation (EEC) No 3820/85 into a tabu search metaheuristic, a great deluge algorithm, and a threshold accepting algorithm. Savelsberg and Sol (1998) include breaks and daily rest periods into a branch and price algorithm for a pickup and delivery problem. Cordeau et al. (2002) suggest the use of a multi-stage network for the inclusion of breaks in a VRP. Xu et al. (2003) present a column generation algorithm and some heuristics to solve a pickup and delivery problem which includes restrictions on driving times specified by the US Department of Transportation. Campbell and Savelsberg (2004) modify an insertion heuristic in such a way that it considers maximum shift times for drivers. Goel and Gruhn (2006) introduce a large neighborhood search algorithm for a VRP which takes into account maximum driving times according to the former Regulation (EEC) No 3820/85. Goel (2009) considers parts of the current Regulation (EC) No 561/2006 in a large neighborhood search algorithm. He presents computational results based on modified problem instances of the Solomon (1987) test instances for the vehicle routing problem with time windows (VRPTW). However, Goel (2009) concentrates on a set of basic rules and does not take into account the whole set of rules of Regulation (EC) No 561/2006. Additionally, he ignores the restrictions on working times set by Directive 2002/15/EC. Zäpfel and Bögl (2008) present a mixed-integer model for a combined vehicle routing and crew pairing problem which considers breaks after 4.5 hours. To solve the model they apply a tabu search metaheuristic and a genetic algorithm. Bartodziej et al. (2009) use a column generation approach and some local search based metaheuristics for solving a combined vehicle and crew scheduling problem which incorporates rest periods for drivers. Kopfer and Meyer (2009) present an integer programming model for a traveling salesman problem (TSP) which considers all relevant rules of Regulation (EC) No 561/2006 for a weekly period.

For the VRP, none of the above algorithms considers the entire set of rules laid down in Regulation (EC) No 561/2006 and none of it includes Directive 2002/15/EC. The extension to the complete legal act implies some additional restrictions. However, exploiting the entire set of rules may also allow for considerable improvements of the resulting vehicle schedules since some specific rules are modifications of the basic rules that increase the flexibility of the planning. We propose a restricted dynamic programming (DP) heuristic which considers all legal rules for a weekly planning period. Moreover, we show how this solution method can be extended to longer time horizons and to a rolling horizon framework.

In line with previous studies on break scheduling algorithms for the EC social legislation, we propose a heuristic for scheduling breaks and rest periods. This break scheduling method does not guarantee to find a feasible schedule, if one exists. However, it does consider the break scheduling problem including all legal rules of the legislation, in contrary to previous studies. The results generated with our restricted DP heuristic comply with the rules of the EC social legislation for drivers. Furthermore, computational experiments on the modified Solomon benchmark instances for the VRPTW show that our approach of using a constructive solution heuristic results both in reduced computational effort and in strongly improved results compared with recently published state of the art metaheuristics.

The contributions of this paper are the following. First, to the best of our knowledge this is the first paper which proposes a solution method for the VRPTW which respects all restrictions on drivers' driving and working hours laid down in Regulation (EC) No 561/2006 and in Directive 2002/15/EC by the European Union. Second, it is shown that exploiting the modified rules of both legal acts results in significantly improved vehicle schedules in terms of number of used vehicles and distance traveled. Third, the proposed algorithm significantly improves results of state of the art metaheuristics on the modified Solomon instances of Goel (2009). Fourth, this paper demonstrates that restricted DP forms a general framework for incorporating complex timing restrictions.

Our paper is organized as follows. Section 2 presents all restrictions of the EC social legislation which have an impact on vehicle routing and scheduling. Section 3 describes our solution approach for the VRPTW with the EC social legislation. Section 4 shows the performance of our algorithm for the VRPTW using the modified Solomon test instances presented by Goel (2009), and Section 5 summarizes our main contributions.

2 EC Legislation on Driving and Working Hours

The EC social legislation on drivers' driving and working hours mainly comprises two legislative acts which we describe in Sections 2.1 and 2.2. Regulation (EC) No 561/2006 restricts driving hours of persons engaged in road

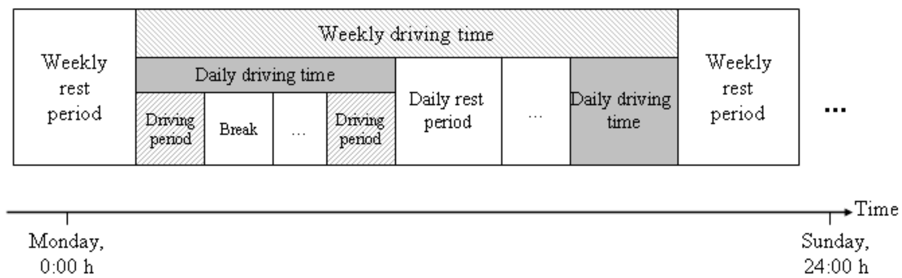

Figure 1: Relation of the different time horizons (Kopfer et al., 2007)

transportation and Directive 2002/15/EC gives restrictions on drivers' working hours.

2.1 Regulation (EC) No 561/2006 on Driving Hours

Regulation (EC) No 561/2006 lays down rules for maximum driving hours and for the required breaks and rest periods. It postulates that transport undertakings have to organize the work of their drivers in such a way that the drivers are able to adhere to the restrictions set by this regulation. For infringements of the regulation committed by the driver his employer is held responsible, too. Furthermore, the regulation demands that every party involved in the transportation process, i.e. the transport undertakings, consignors, forwarders, tour operators, principal contractors, subcontractors, and even driver employment agencies ensure that the schedules of the drivers comply with the legal requirements. Therefore, the regulation's impact on vehicle routing and scheduling in real life applications is enormous.

Regulation (EC) No 561/2006 covers different but interconnected time horizons: single driving periods, daily driving times, and weekly driving times. These time horizons are ended by breaks, daily rest periods, and weekly rest periods. Figure 1 depicts their relationship. We describe the rules on the durations of these time horizons and breaks in detail below. Some modifications of these rules have been introduced to allow for more flexibility for the drivers. We will indicate these cases by the distinction between basic and modified rules.

***Driving periods*:** A driving period may contain at most 4.5 hours of accumulated driving time.

***Breaks to end driving periods*:** A break of at least 45 minutes ends a driving period (basic rule). The duration may be reduced to 30 minutes if an additional break of 15 minutes has been taken anywhere during the same driving period (modified rule). Since the total break time of 45 minutes is now divided into two parts, we refer to this modified rule as splitting breaks. This may be beneficial, for example, if waiting time at a customer site allows a 15 minute break, but not a 45 minute break. If in such a case a 15 minute break is scheduled during the waiting time, then only a 30 minute break is required when the 4.5 hour driving limit is reached.

***Daily driving times*:** The total daily driving time may not exceed 9 hours (basic rule). Twice a week, i.e. twice between Monday 0:00 am and Sunday 23:59 pm, the daily driving time can be extended to 10 hours (modified rule). We refer to this modified rule as extending driving times. A daily driving time ends when a daily or weekly rest period starts.

***Daily rest periods*:** The duration of a daily rest period is at least 11 hours (basic rule). Any daily rest period may be reduced to 9 hours if an additional rest of 3 hours has been taken anywhere after the end of the previous daily rest period (modified rule). We refer to this modified rule as splitting rests. Moreover, drivers are allowed to reduce their daily rest periods to 9 hours without an additional rest of 3 hours up to three times between two weekly rest periods (modified rule). We refer to this modified rule as reducing rest periods. Within 24 hours after the end of a daily rest period, a new daily rest period must have been taken, allowing a nonrest period (a period between two daily rest periods) to last for at most 13 hours (15 hours in case the nonrest period is ended with a reduced rest period).

***Weekly driving times*:** The total driving time during a week, i.e. from Monday 0:00 am until Sunday 23:59 pm, may not exceed 56 hours. The accumulated driving time in any two consecutive weeks must not exceed 90 hours.

***Weekly rest periods*:** The duration of a weekly rest period is at least 45 hours (basic rule). Drivers are allowed to reduce one weekly rest period to 24 hours in any two consecutive weeks (modified rule). This reduction has to be compensated by an equal extension of another rest period before the end of the third week following the week considered. Within 144 hours (6 days) after the end of a weekly rest period, drivers have to start a new weekly rest period.

2.2 Directive 2002/15/EC on Working Hours

Directive 2002/15/EC supplements the restrictions on driving times laid down by Regulation (EC) No 561/2006. As driving times are part of the total working time, these legal acts are interdependent and therefore both have to be considered in vehicle routing and scheduling. Besides driving times, also times for loading and unloading, time to assist passengers while boarding and disembarking from the vehicle, cleaning and maintenance times, and other times in which a driver cannot freely dispose of his time, such as unforeseen waiting times, are included in the working time. Since in the remainder we will address a deterministic vehicle routing problem, only driving and service times are taken into account as working times. Waiting times need not be considered as working times, since in deterministic problems all waiting times are known in advance.

The directive comprises the following restrictions on working periods and weekly working times:

***Working period*:** A working period may contain at most 6 hours of accumulated driving time.

***Breaks to end working periods*:** A break of at least 30 minutes ends a working period (basic rule). If the total working time between two daily rest periods exceeds 9 hours, the total break time in this period has to be extended to at least 45 minutes (basic rule). The total break time can be divided into parts of at least 15 minutes each (modified rule).

***Weekly working time*:** The total working time during a week may not exceed 60 hours. The average weekly working time must not exceed 48 hours over a period of four months.

In order to observe the law, both legal acts have to be respected by drivers. Therefore, each basic rule or its modification must be respected and both are considered of equal importance in practice. In the literature, however, the modified rules have been neglected so far, and Directive 2002/15/EC on working hours has been completely neglected.

3 Restricted Dynamic Programming Heuristic including the EC Social Legislation

We propose a solution method for the VRPTW including the EC social legislation for the planning horizon of one weekly driving period. However, this solution method can be extended to longer time horizons and a rolling horizon framework (see Section 3.3). For the development of an efficient solution method, we use the restricted DP framework proposed by Gromicho et al. (2008). Within this framework, customers are sequentially added to the end of partial vehicle routes. Feasibility of such additions, for example checking whether the added customer is visited within its time window, is controlled by extra state dimensions. Checking compliance with the EC social legislation can also be done by adding state dimensions. For this purpose, we propose a break scheduling method which schedules breaks at or on the travel to the customer to be added. Before we describe this break scheduling method in detail, we provide a short explanation of the restricted DP heuristic of Gromicho et al. (2008).

The restricted DP heuristic for the VRP is based on the exact DP algorithm for the TSP of Held and Karp (1962) and Bellman (1962). This DP algorithm defines states $(S, j), j \in S, S \subseteq V \backslash 0$, which represent a minimum-length tour with cost $C(S, j)$ and in which V represents the entire set of nodes to be visited. This tour starts at node 0 and visits all nodes in S, which is a proper subset of V, and it ends in node $j \in S$. The costs of the states in the first stage are calculated by $C(\{j\}, j) = c_{0j}, \forall j \in V \backslash 0$, in which c_{ij} is the cost of traveling directly from node i to node j. Next, the costs of the states in all subsequent stages are calculated by the recurrence relation $C(S, j) = \min_{i \in S \backslash j} \{C(S \backslash j, i) + c_{ij}\}$.

The DP algorithm for the TSP is applied to the VRP through the giant-tour representation of vehicle routing solutions introduced by Funke et al. (2005). In this representation, the vehicles are ordered and for each vehicle k a unique origin node o_k and destination node d_k are introduced. Next, the destination node of each vehicle is connected to the origin node of its successive vehicle, as well as the destination node of the last vehicle with the origin node of the first vehicle, creating a giant-tour. The DP algorithm is applied to the extended node set with the vehicle origin and destination nodes, where each node addition now requires a feasibility check.

The feasibility checks ensure that an origin node of a vehicle o_k can be added to a partial route represented by a state if and only if the last visited node is d_{k-1}. Furthermore, these checks only allow d_k to be added if o_k is already in the visited node set S. To account for other restrictions, such as capacity restrictions or time windows, state dimensions are added. For example, in case of capacity restrictions a state dimension c is added which keeps track of the accumulated demand of the active vehicle k. With active vehicle we refer to the last vehicle for which o_k has been added to the set of visited nodes. Each time a vehicle origin node o_k is added to a state, c is reset to zero. Furthermore, a customer addition is only allowed if the accumulated demand c together with the customer demand does not exceed the capacity of the active vehicle. Many other restrictions such as time windows, sequencing restrictions (pickup and delivery), multiple depots, and heterogeneous vehicle fleets can be incorporated by adding state dimensions or control via the input, allowing for a general framework for solving VRPs.

Since the (unrestricted) DP algorithm does not run in practically acceptable computation times for problem instances of realistic sizes, the state space is restricted by parameters H and E. The value of H specifies the maximum number of states to be taken to the next iteration, where the smallest cost states are maintained, as proposed by Malandraki and Dial (1996). Since states in the same stage represent partial tours of the same length, states with smaller costs are more likely to lead to good overall solutions.

The value of E restricts the number of state expansions of a single state: only the E nearest, unvisited neighbors allowing feasible state expansions are considered. Since in good VRP solutions successive nodes are in general near neighbors of each other, this restriction cuts off less promising parts of the state space.

These restrictions on the number of state expansions results in the following running time complexity of the restricted DP heuristic. In each stage, at most H states are expanded to at most nH states. Since we have to select the H best states for the next stage, each stage requires $O(nHlog(H))$ time. The total number of stages equals the number of nodes in the network, which is $O(n)$. Therefore, the running time complexity of the restricted DP heuristic is $O(n^2Hlog(H))$.

We incorporate the EC social legislation in the DP framework by adding state dimensions. For this purpose, we propose a break scheduling algorithm, which decides locally, i.e., at or on the travel to the customer to be added, when and where breaks have to be scheduled. There are two main reasons to use a local view for scheduling breaks and rest periods.

First, it allows us to schedule breaks in constant time. Therefore, the running time complexity of the restricted DP heuristic does not increase. This even holds when complex modified rules, which are generally ignored in the literature, are incorporated.

Second, the rules we introduce for scheduling the breaks are intuitive and, therefore, they are both easy to implement, as well as easily acceptable by planners and operations managers in practice. If a global scheduling algorithm is used, then breaks and rests may be scheduled and extended prematurely, such that the benefits are less clear. For example, it may turn out that a state expansion results in an early arrival at a customer j, such that the active nonrest period reaches its maximum of 13 hours before the time window at customer j opens, requiring a rest period before serving customer j. If, due to this rest period the time window at customer j cannot be met, then this state expansion is infeasible. However, if there is waiting time at some predecessing customer i, then it might be possible to schedule an early rest period partially during this waiting time without violating any of the time windows between customers i and j. This may advance the start of the active nonrest period when arriving at customer j, allowing to serve customer j before having to schedule a rest period. This global view in which also breaks and rests at predecessing customers are considered requires at least linear time, which increases the running time complexity of the restricted DP heuristic. Note that this also implies that the local view for scheduling breaks does not guarantee to find a feasible schedule, if one exists.

We propose two break scheduling methods: a basic method and an extended method. The basic method is an extension of the naive label setting method proposed by Goel (2009), which is improved by allowing for more local flexibility of customer additions. This is done by first minimizing the start service time of the added customer. Next, for this minimum start time the accumulated time since the last rest, and the accumulated driving and working time since the last break are minimized by trying to schedule rests or breaks in waiting time caused by hard time windows. The extended method extends the basic method by incorporating the modified rules of the legislation. The same methodology of optimizing local flexibility at the last visited customer is applied. We now describe the break scheduling methods in detail.

3.1 Basic Break Scheduling Method

For the basic approach, we make the simplification that after no more than 6 hours of working time, we schedule a break of 45 minutes (instead of 30 minutes). This ensures that the second requirement of Directive 2002/15/EC on the break length between working periods, which states that the total break time on a day should be at least 45 minutes if that day contains more than 9 hours of working time, is also satisfied. On top of that, it fulfills the requirements of Regulation (EC) No 561/2006 on the break length between two driving periods, such that also a new driving period is initiated.

To include the legislation on driving and working hours into our restricted DP heuristic, we have to ensure that the partial route represented by each state is feasible with respect to these restrictions. For this purpose, we introduce six state dimensions: nonbreak working time, nonbreak driving time, nonrest time, daily driving time, weekly working time, and weekly driving time.

t_{nbw}**:** accumulated nonbreak working time. This variable denotes the total amount of working time since the last break of at least 45 minutes.

t_{nbd}**:** accumulated nonbreak driving time. This variable denotes the total amount of driving time since the last break of at least 45 minutes.

t_{nr}**:** accumulated nonrest time. This variable denotes the total amount of time passed by since the last rest period of at least 11 hours.

t_{dd}**:** accumulated daily driving time. This variable denotes the total amount of driving time since the last rest period of at least 11 hours.

t_{ww}**:** accumulated weekly working time. This variable denotes the total amount of working time since the last rest period of at least 45 hours.

t_{wd}**:** accumulated weekly driving time. This variable denotes the total amount of driving time since the last rest period of at least 45 hours.

For our planning purposes, we first consider one week, i.e., the time between Monday, 0:00 am, and Sunday, 23:59 pm. Furthermore, we first assume that the planning starts right after a weekly rest period has been taken by all drivers. This results in all state dimensions t_{nbw}, t_{nbd}, t_{nr}, t_{dd}, t_{ww}, and t_{wd} being zero for all vehicles at the start of the planning period. Section

3.3 discusses extensions to longer time horizons and state dimensions not being zero at the start of the planning period, allowing for a rolling horizon framework.

When we start a new vehicle, we check for the first customer to be visited whether it can be reached from the depot. This might not be the case if a vehicle starts from the depot at time zero and requires a break or rest period before starting service, since this might violate the time window. If the customer cannot be served by a vehicle leaving the depot at time zero, we delay the departure time of the vehicle such that the vehicle arrives at the customer node exactly at the start of the time window.

Within our basic approach we do not consider the modified rules of the legislation. Whenever we want to expand a state (S, i) with a customer j, then we first determine the arrival time a_j at this customer, considering possible breaks and rest periods that have to be scheduled along the travel from i to j. For this purpose, we first set a_j to the departure time (service completion time) from customer i, and we introduce a variable δ_{ij}, denoting the remaining driving time to customer j. This variable is initially set to the total driving time d_{ij} from customer i to customer j. We define $\Delta = min(\delta_{ij}, 6 - t_{nbw}, 4.5 - t_{nbd}, 13 - t_{nr}, 9 - t_{dd})$, which represents the minimum driving time until a break or rest period must be scheduled, or the next customer is reached. Next, we recursively check whether $\delta_{ij} = \Delta$ holds.

If $\delta_{ij} \neq \Delta$, then we are forced to schedule either a break or a rest period along the route. We check whether Δ equals $13 - t_{nr}$ or $9 - t_{dd}$. If so, we schedule an 11 hour rest period and set the values of t_{nbw}, t_{nbd}, t_{nr}, and t_{dd} to zero. Otherwise, either $6 - t_{nbw}$ or $4.5 - t_{nbd}$ equals Δ. Thus, we have to schedule a 45 minute break and we set the values of t_{nbw} and t_{nbd} to zero. However, if we are forced to schedule a break, we check whether this fits within the remaining available nonrest time. Otherwise we schedule a rest period instead of a break.

After scheduling a rest or break, we update our remaining driving time δ_{ij}, and in case of a break also the values of t_{nr} and t_{dd}, as follows:

$$\delta_{ij} := \delta_{ij} - \Delta \tag{1}$$

$$t_{dd} := t_{dd} + \Delta \tag{2}$$

$$t_{nr} := t_{nr} + \Delta + 0.75 \tag{3}$$

The remaining driving time to the next customer is reduced by Δ (1). In case a break is scheduled, the accumulated daily driving time is increased with Δ (2), and the accumulated nonrest time is increased with Δ and with the break time (3).

After determining a_j, we check whether the accumulated nonbreak working time and the accumulated nonrest time allow to serve the customer without scheduling another break or rest period at customer j. To check this, we need the service time s_j of customer j and the time window $\{e_j, l_j\}$ in which service must start at customer j. If $t_{nbw} + s_j > 6$, then we schedule a break and update a_j, t_{nr}, t_{nbw}, and t_{nbd}. Next, if $t_{nr} + \max\{0, e_j - a_j\} + s_j > 13$, then we schedule a rest period. However, if both inequalities hold, then we extend the 45 minute break forced by the nonbreak working time to an 11 hour rest period to avoid scheduling a 45 minute break and an 11 hour rest period directly after each other. Finally, if $a_j \leq l_j$, then we can arrive in time to add customer j to the partial route.

To decide whether the addition of customer j is feasible with respect to all rules of the social legislation, we still need to check whether the vehicle can return to the depot, without violating the restrictions on the weekly driving and working times. We forbid the expansion if after visiting the customer a return to the depot would be infeasible in order to avoid including infeasible states. Consequently, we only allow an expansion if (4) and (5) are satisfied.

$$d_{ij} + d_{j0} \leq 56 - t_{wd} \quad (4)$$

$$d_{ij} + s_j + d_{j0} \leq 60 - t_{ww} \quad (5)$$

To improve this scheduling procedure by increasing the local flexibility at customer j, we introduce a number of scheduling features that reduce the values of t_{nbw}, t_{nbd}, t_{nr}, and t_{dd}, without delaying the start service at customer j. We give the highest priority to reducing the accumulated nonrest time, since in VRPTWs, large waiting times often cause this to be the tightest restriction. Therefore, in a first attempt we try to schedule a daily rest period whenever waiting times allow us to do so without postponing the start of service at a customer node. This means that we schedule a rest period before serving a customer node whenever the waiting time until the ready time of the customer's time window is more than 11 hours. In this case we can reset all values t_{nr}, t_{dd}, t_{nbw}, and t_{nbd} to zero. If after taking the rest period there is still waiting time left, we extend the rest period until the ready time of the customer, such that t_{nr} is not increased before starting service.

If it is not possible to schedule a rest during waiting time, but there is a rest scheduled along the route to customer j, then we extend this rest by the waiting time at customer j (if any). This reduces the value of t_{nr} at the start of service at customer j without affecting the other variables. This feature might even reduce the start of service time, if otherwise the additional waiting time would make the value of t_{nr} to force another rest period before starting service. This additional rest period might postpone the start of service after e_j, or even after l_j making the expansion infeasible.

If the first two cases do not apply, but there is waiting time at the customer, then we check whether we can schedule a 45 minute break in order to reset t_{nbw} and t_{nbd} to zero. This increases the flexibility of adding customers afterwards. We give a detailed description of the basic break scheduling method in pseudo-code in Appendix A.

3.2 Extended Break Scheduling Method

To make the above presented algorithm more suitable for realistic planning purposes and to allow for an enlargement of the solution space, we incorporate all the modified rules of Regulation (EC) No 561/2006 as described in Section 2.1. Furthermore, we take into consideration the modified rules of Directive 2002/15/EC. In line with the restricted DP approach, the extended set of rules is only exploited if they allow for a local improvement of the current partial solution. In the following, we describe the implementation of the modified rules.

3.2.1 Extending Driving Times

Regulation (EC) No 561/2006 allows drivers to extend their daily driving time up to 10 hours twice a week while the basic rule restricts the daily driving time to no more than 9 hours. Driving 9 hours can be accomplished with only one break if the driver takes this break exactly after 4.5 hours and afterwards continues driving for another 4.5 hours. However, the extension to more than 9 hours forces the driver to take at least two breaks as the daily driving time exceeds the maximum length of two driving periods. Therefore, this extension might cause a delayed arrival at a customer due to the additional break. On the other hand, a driving time extension might allow drivers to arrive earlier at a customer, because of not having to schedule a rest period.

In our algorithm we schedule a driving time extension if it reduces the start time of the service at the customer to be added. Besides, if the driving time extension increases the waiting time at this customer making it possible to schedule a rest period during this waiting time, we also include the extension. To calculate the arrival time at the customer in case of extending the driving time we use a similar procedure as described in the basic method. However, we set the maximum daily driving time to 10 hours. We compare this arrival time with the arrival time calculated in the traditional way and we decide whether a driving time extension is profitable.

Since we can extend driving times up to two times a week, we need to account for the number of driving time extensions used. For this purpose, we introduce a new state dimension:

n_{dte}: number of driving time extensions taken by the active vehicle.

The state dimension n_{dte} is initialized to zero and each time a driving time extension is scheduled it is increased by one. Moreover, it is restricted to two and when the current node is the depot, n_{dte} is updated to zero since a new vehicle is used.

3.2.2 Reducing Rest Periods

Reducing rests can be beneficial in two ways. First, it might allow an earlier start of the next nonrest period. Second, it might extend the current nonrest period with at most 2 hours. The latter case appears, since this rest must have been taken within 24 hours after the end of the previous rest period, while this rest is reduced by at most 2 hours. When a rest period must be taken during a travel, then we check whether it is beneficial to schedule a reduced rest period. We do this by calculating the arrival time at the customer to be added in case we reduce the rest period. If this arrival time reduces the start service time or increases the waiting time allowing for another (reduced) rest period, then we schedule a reduced rest period. This procedure is similar to the procedure applied for checking the profitability of driving time extensions.

Since we may also choose to extend driving times besides reducing rest periods, there are four different scenarios to consider when a rest has to be scheduled during a travel. Therefore, we calculate the arrival times for each of these scenarios. Next, we check whether some of the arrival times allow for a (reduced) rest period during waiting time. If this is the case, we select the one with the least number of modified rules. In case of having to choose between extending driving times and reducing rest periods we proceed as follows. Since there is a limited number of times we can use each modified rule and rest reductions increase the available time for all working activities, we give priority to using driving time extensions such that more rest reductions remain.

If none of the scenarios allows to schedule a rest during waiting time, we select the scenario which minimizes the start service time. Again, if different scenarios result in this minimal start service time, then we choose the one with the least number of modified rules.

Since a driver is only allowed to reduce daily rest periods three times between two weekly rest periods we need to keep track of the number of reduced rest periods left. For this purpose, we introduce a new state dimension n_{rr} indicating the number of rest reductions taken by the active vehicle.

n_{rr}: number of rest reductions taken by the active vehicle.

Whenever a rest reduction is scheduled, n_{rr} is increased by one and if the current node visited is the depot then n_{rr} is reset to zero.

Upon arrival at a customer, we also check whether it is beneficial to reduce the next rest period. This is the case if a nonrest time of 13 hours does not allow to serve the customer before taking a rest period, while a nonrest time of 15 hours does allow this. Consequently, we reduce the next rest period, thereby allowing to serve the customer without having to schedule a rest before service and reducing the start service time at this customer.

3.2.3 Splitting Breaks

Both Regulation (EC) No 561/2006 and Directive 2002/15/EC allow drivers to split their breaks. The regulation on driving times allows to split breaks of at least 45 minutes into two parts. The first part has to last for at least 15 and the second part for at least 30 minutes. Besides, the directive on working hours allows to split the total time required for breaks into parts of at least 15 minutes each.

In our algorithm, the modified rule of splitting breaks is applied whenever there is waiting time of at least 15 but less than 45 minutes before serving a customer. This waiting time is not sufficient to schedule a regular break as required by Regulation (EC) No 561/2006. Therefore, a 15 minute break is scheduled and extended until the ready time of the customer. If the break lasts for at least 30 minutes, it counts as a full break for the nonbreak working time and t_{nbw} is set to zero. If it is less than 30 minutes, then it counts as a 15 minute break and we require another break of 15 minutes to be taken when t_{nbw} reaches its maximum value of 6 hours.

If a break of at least 15 minutes is taken (but less than 45 minutes), either during waiting time or forced by the accumulated nonbreak working time, then we can count this as a 15 minute break for the nonbreak driving time. Therefore, when in this case the nonbreak driving time t_{nbd} reaches its maximum of 4.5 hours we require only a break of 30 minutes to be scheduled.

Note that we do not schedule a 45 minute break anymore when the accumulated nonbreak working time reaches its maximum value. This is, because a 30 minute break now also counts as a 15 minute break for the nonbreak driving time. Therefore, if later on a break is forced by the nonbreak driving time then it benefits from this 30 minute break, as opposed to the case where we ignore the modified rules.

Directive 2002/15/EC also requires that if the working time on a day exceeds 9 hours, the total break time on that day should be at least 45 minutes, instead of 30 minutes if the working time is between 6 and 9 hours. To account for this rule, we introduce the state variable t_{dw}, which indicates the daily working time:

t_{dw}: daily working time of the active vehicle.

Whenever this state dimension reaches its maximum of 9 hours another break of at least 15 minutes is introduced if the total break time of this day does not add up to at least 45 minutes already. In the latter case namely, the total duration of breaks satisfies the working time directive and since only breaks of at least 15 minutes are scheduled also the required structure of the breaks is satisfied.

3.2.4 Splitting Rest Periods

The modified rule on rest periods allows to split regular rest periods into two parts of which the first must last for at least 3 hours and the second for at least 9 hours. It has to be noticed that the total time required for split rest periods equals 12 hours instead of 11 hours as required for a regular rest period. Therefore, in order to avoid an increased time required for rests we only consider scheduling the 3 hours part of a reduced rest period if the waiting time before serving a customer lies between 3 and (9) 11 hours such that no (reduced) rest period can be taken during waiting time. To schedule a 3 hour rest period in this case is beneficial, since it allows an extension of the nonrest period to 15 hours.

The 3 hour part of a split rest period is only scheduled if no such part has been scheduled already and it is extended until the ready time of the customer. As the rest time of 3 hours lies above 45 minutes we can reset the state dimensions t_{nbw} and t_{nbd} to zero when the service starts.

When the next rest period is required by t_{nr} or t_{dd} then only the second part of the split rest period of 9 hours is scheduled. Furthermore, the maximum nonrest period is extended to 15 hours until this next rest is scheduled. After taking the second part of the split rest the state dimensions t_{nr}, t_{dd}, t_{nbw}, t_{nbd}, and t_{dw} are set to zero.

There is one other case where a split rest may be beneficial. This is, when there is less than 3 hours of waiting time at a customer, but the accumulated nonrest period would exceed 13 hours if there is no rest scheduled before serving this customer, while it would not exceed 15 hours. If in this situation the maximum number of reduced rest periods are already taken, while a split rest of 3 hours together with the customer service time still fits within the 15 hour nonrest period, then a split rest of 3 hours is scheduled.

Table 1 summarizes all implementations of the modified rules into the break scheduling method. We give an outline of the changes of the extended break scheduling method with respect to the basic break scheduling method in Appendix B.

Table 1: Implementation of the modified rules into the break scheduling method

Modified Rule	Implementation
Extend driving time	Apply it if it reduces start service time; apply it if it increases the waiting time, allowing for a rest period before service
Reduce rest period	Apply it if it reduces start service time; apply it if it increases the waiting time, allowing for a rest period before service

Split breaks	Schedule a 15 minute break if there is enough waiting time
Split rest periods	Schedule a 3 hour rest if there is enough waiting time; schedule a 3 hour rest if this allows a service without taking a daily rest before and no rest reductions are left

3.3 Extensions to Other Time Horizons

So far, we considered a weekly planning horizon and a situation in which all drivers have just completed a weekly rest period. However, our algorithm can be extended to longer time horizons and to a rolling horizon framework.

In case a longer planning horizon is considered, a state dimension (t_{nwr}) must be added to account for the maximum period of 144 hours between two weekly rest periods. As soon as one of the state dimensions t_{ww}, t_{wd}, or t_{nwr} reaches its maximum value, a weekly rest period must be scheduled. Afterwards, all state dimensions are reset to 0. To use the modified rules on the weekly rest period, the weekly driving time, and the weekly working time, we can follow a similar methodology as described with the extended break scheduling method.

In case drivers have *not* just completed a weekly rest period at the start of the planning period - which is typically the case in a rolling horizon framework - then we can set the initial state dimensions for each vehicle to the initial conditions of the corresponding drivers. Therefore, the algorithm can also be used in a rolling horizon framework.

4 Computational Experiments

We test the restricted DP heuristic including the break scheduling methods on the modified Solomon instances proposed by Goel (2009). The Solomon instances consist of 6 problem sets: in the c1 and c2 instances customer locations are clustered, in the r1 and r2 instances they are random, and in the rc1 and rc2 instances they are semi-clustered; the 2-instances have a relatively longer time horizon and larger vehicle capacities than the 1-instances allowing for larger vehicle routes in terms of number of customers. Goel proposes the following modifications of the Solomon instances for the VRPTW with the EC social legislation.

He proposes to consider the depot opening times as a period of 144 hours and to scale the customer time windows accordingly. Next, he suggests a driving speed of 5 distance units per hour, and he sets all service times to 1 hour. Due to the required breaks and rest periods it may be impossible to reach certain customers before their due dates, or the vehicle may not be able to return in time to the depot after serving a customer at his ready time. Therefore, Goel suggests to adjust such time windows in such a way that the ready time equals the earliest time the vehicle can reach the customer, and the due date is such that starting service at this due date and directly returning to the depot results in a return time at the depot's due date, respectively.

We implemented the restricted DP heuristic in Delphi 7 and we ran our experiments on a Pentium M, 2.00 GHz CPU and 1.0 GB of RAM. We first report the results of our basic method, in which the modified rules are not used. We compare our results with the best results found by Goel (2009). Since the method proposed by Goel does not consider Directive 2002/15/EC on the drivers' working hours, we relax our break scheduling method by setting the maximum nonbreak working period to 13 hours, i.e., the maximum period between two rests in the basic method. Next, we present computational results on the impact of Directive 2002/15/EC. Finally, we present computational results on the impact of the modified rules by applying the extended method.

As described in Gromicho et al. (2008), the value of H, which restricts the stage width after each iteration of the restricted DP heuristic, has a large impact on computation time and solution quality. We set the value of H to $10,000$ since this gives an average computation time of 65 seconds (with a maximum of 107 seconds) per instance, which is practically acceptable. Furthermore, we do not restrict the number of state expansions of a single state (we set the maximum number of state expansions E of a single state to n, the number of customers). As in Goel (2009), we minimize the number of vehicles as primary objective and the total travel distance as secondary objective. In order to obtain this objective hierarchy we add a large cost M to a state each time a vehicle returns to the depot.

Table 2 presents the results of our basic method with the relaxation of

Directive 2002/15/EC and the best results found by Goel (2009). Note that in Goel (2009) significantly larger computation times are allowed: Goel's results are the best out of five runs of half an hour each per problem instance.

Table 2 clearly shows that our method outperforms the large neighborhood search algorithm proposed by Goel (2009). Only one problem instance (r103) requires one more vehicle with our method, while for 47 other problem instances a smaller number of vehicles is found. On average over all problem instances, our method finds solutions requiring 18.26% less vehicles.

Also the results on the travel distances show significant improvements by our solution method. Only for the r1 problem instances no improvement is found, on average. In total, our method reduces the travel distances of 37 problem instances with an average reduction over all problem instances of 5.41%.

Table 2: Results basic method without Directive 2002/15/EC

	Our method			Best in Goel (2009)		Change	
Problem	vehicles	distance	cpu(s)	vehicles	distance	vehicles	distance
c101	11	923.66	43	13	1,143.32	-15.38%	-19.21%
c102	11	1,097.97	53	13	1,198.82	-15.38%	-8.41%
c103	10	1,080.04	72	11	971.11	-9.09%	11.22%
c104	10	1,053.27	89	10	1,101.42	0.00%	-4.37%
c105	10	839.99	42	11	908.29	-9.09%	-7.52%
c106	11	900.10	42	11	1,079.24	0.00%	-16.60%
c107	10	874.03	47	10	1,023.77	0.00%	-14.63%
c108	10	892.71	51	10	975.20	0.00%	-8.46%
c109	10	1027.19	57	11	1,088.87	-9.09%	-5.66%
c1	10.33	965.44	55.04	11.11	1,054.45	-7.00%	-8.44%
c201	6	941.60	55	9	1,064.57	-33.33%	-11.55%
c202	5	866.09	71	9	990.03	-44.44%	-12.52%
c203	5	810.74	80	9	982.49	-44.44%	-17.48%
c204	4	768.19	107	8	873.22	-50.00%	-12.03%
c205	5	711.96	62	8	973.53	-37.50%	-26.87%
c206	5	677.79	66	7	838.91	-28.57%	-19.21%

Table 2 (Cont'd.)

	Our method			Best in Goel (2009)		Change	
Problem	vehicles	distance	cpu(s)	vehicles	distance	vehicles	distance
c207	5	709.36	64	9	966.19	-44.44%	-26.58%
c208	5	677.62	68	8	948.21	-37.50%	-28.54%
c2	5.00	770.42	71.60	8.38	954.64	-40.30%	-19.30%
r101	13	1,483.95	39	15	1,413.43	-13.33%	4.99%
r102	13	1,398.59	44	13	1,296.16	0.00%	7.90%
r103	11	1,256.53	54	10	1,251.81	10.00%	0.38%
r104	8	1,023.47	74	10	1,024.13	-20.00%	-0.06%
r105	11	1,207.87	47	12	1,276.23	-8.33%	-5.36%
r106	9	1,162.18	53	11	1,150.95	-18.18%	0.98%
r107	9	1,068.90	65	10	1,098.62	-10.00%	-2.71%
r108	8	1,011.90	77	9	1,047.53	-11.11%	-3.40%
r109	9	1,094.14	60	11	1,058.01	-18.18%	3.42%
r110	8	1,061.92	77	10	1,062.43	-20.00%	-0.05%
r111	9	1,085.39	64	10	1,008.31	-10.00%	7.64%
r112	8	973.86	96	10	1,043.10	-20.00%	-6.64%
r1	9.67	1,152.39	62.51	10.92	1,144.23	-11.45%	0.71%
r201	10	1,337.07	41	13	1,335.17	-23.08%	0.14%
r202	10	1,258.97	46	12	1,215.88	-16.67%	3.54%
r203	9	1,130.86	62	10	1,122.58	-10.00%	0.74%
r204	6	913.46	91	9	1,013.70	-33.33%	-9.89%
r205	8	1,136.25	53	12	1,183.14	-33.33%	-3.96%
r206	7	1,084.71	62	9	1,068.91	-22.22%	1.48%
r207	7	1,024.53	76	11	1,064.22	-36.36%	-3.73%
r208	6	918.88	92	8	1,088.12	-25.00%	-15.55%
r209	7	1,104.62	64	10	1,067.09	-30.00%	3.52%
r210	7	1,185.38	59	10	1,076.23	-30.00%	10.14%
r211	6	1014.32	77	9	943.45	-33.33%	7.51%
r2	7.55	1,100.83	65.69	10.27	1,107.14	-26.55%	-0.57%
rc101	12	1,454.01	49	13	1,599.01	-7.69%	-9.07%
rc102	11	1,403.06	57	11	1,434.52	0.00%	-2.19%

Table 2 (Cont'd.)

	Our method			Best in Goel (2009)		Change	
Problem	vehicles	distance	cpu(s)	vehicles	distance	vehicles	distance
rc103	10	1,278.33	73	11	1,268.81	-9.09%	0.75%
rc104	9	1,188.22	96	9	1,263.25	0.00%	-5.94%
rc105	12	1,426.29	59	12	1,405.72	0.00%	1.46%
rc106	10	1,253.11	67	12	1,297.67	-16.67%	-3.43%
rc107	9	1,189.06	81	11	1,243.08	-18.18%	-4.35%
rc108	9	1,212.69	89	10	1,269.90	-10.00%	-4.50%
rc1	10.25	1,300.60	71.27	11.13	1,347.75	-7.87%	-3.50%
rc201	10	1,554.93	46	11	1,510.67	-9.09%	2.93%
rc202	9	1,356.14	60	10	1,415.67	-10.00%	-4.21%
rc203	8	1,295.72	72	10	1,274.45	-20.00%	1.67%
rc204	6	975.56	104	9	1,264.73	-33.33%	-22.86%
rc205	9	1,437.07	56	11	1,521.10	-18.18%	-5.52%
rc206	8	1,220.06	59	11	1,418.40	-27.27%	-13.98%
rc207	8	1,234.27	66	10	1,171.94	-20.00%	5.32%
rc208	7	1.059.39	87	8	1,201.13	-12.50%	-11.80%
rc2	8.13	1,266.64	68.68	10.00	1,347.26	-18.75%	-5.98%

The main reason for this remarkably large improvement with respect to the solutions found by the large neighborhood search algorithm proposed by Goel (2009) is presumably the following. Determining the feasibility of neighborhood solutions which respect Regulation (EC) No 561/2006 requires significantly larger computation times than when this regulation is ignored. Therefore, the number of neighborhood solutions which can be evaluated significantly reduces when respecting this regulation. In contrast, the running time complexity of our restricted DP heuristic does not increase with respecting Regulation (EC) No 561/2006. Therefore, the number of states that can be investigated during a fixed amount of computation time does not significantly decrease when this regulation is respected. This benefit of the restricted DP heuristic with respect to solution approaches based on local search does not hold for the classical VRPTW, since no difficult (but realistic) timing restrictions are included in this problem type. This is demonstrated by applying the restricted DP heuristic to the classical Solomon instances, which results in 1.2 more vehicles and 8.8% more travel distance than the best known solutions for these problem instances, on average.

If a practical application allows more computation time, then this would be beneficial for our method. For example, if H is set to $100,000$ then the average computation time increases to 11 minutes (which is still much smaller than the computation times allowed in Goel, 2009), but with an average additional reduction of the number of vehicles and travel distance of 1.46% and 1.90%, respectively.

Restrictions on drivers' working hours imposed by Directive 2002/15/EC are generally ignored in the literature. However, they do reduce the solution space and, therefore, may have a significant impact on the solution quality. We tested this impact by solving the benchmarks of Goel with our basic method. For the six problem sets Table 3 presents the average results of our basic method including Directive 2002/15/EC. Columns four and five present the objective changes caused by including Directive 2002/15/EC. As can be observed, these changes are significant (3.89% on average for the number of vehicle routes and 0.96% on average for the distance traveled). Therefore, Directive 2002/15/EC has a significant impact on the resulting vehicle schedules.

Table 3: Results basic method including Directive 2002/15/EC

	Incl. working hours			Change [a]	
Problem	vehicles	distance	cpu(s)	vehicles	distance
c1	10.33	949.31	52.80	0.00%	-1.67%
c2	5.75	834.47	72.18	15.00%	8.31%
r1	9.67	1155.89	59.22	0.00%	0.30%
r2	7.91	1097.26	62.82	4.82%	-0.32%
rc1	10.25	1300.14	67.57	0.00%	-0.04%
rc2	8.50	1264.52	67.56	4.62%	-0.17%

[a]Change with respect to the results without Directive 2002/15/EC

Finally, we tested the impact of the modified rules on the quality of vehicle routing solutions. These modified rules have been ignored in the literature, since they are hard to incorporate in existing solution methods for the VRPTW. However, in practice they are usually considered.

Table 4 reports the average objective values for the six problem sets using our solution approach with the extended break scheduling method. In columns four and five we compare the results with the results of ignoring the modified rules (see Table 3). These columns indicate the profitability of using the modified rules.

Table 4: Results extended method

	Incl. modified rules			Change [a]	
Problem	vehicles	distance	cpu(s)	vehicles	distance
c1	10.11	937.08	85.90	-2.15%	-1.29%
c2	5.25	773.80	138.25	-8.70%	-7.27%
r1	9.33	1142.62	102.15	-3.45%	-1.15%
r2	7.36	1084.70	105.20	-6.90%	-1.15%
rc1	10.00	1322.41	122.69	-2.44%	1.71%
rc2	8.13	1247.37	120.51	-4.41%	-1.36%

[a] Change with respect to the results with the basic break scheduling method

The average results for all problem sets are improved. There is a significant reduction in the number of vehicles used (4.28% on average) and in the total distance traveled (1.54% on average). Therefore, the benefits of using the modified rules are significant and these rules should be accounted for when constructing vehicle routes.

The computation times are larger than with the basic break scheduling method (111 seconds on average versus 65 seconds, and 241 seconds maximum versus 107 seconds). This can be explained by the additional checks in the extended break scheduling method to exploit the modified rules. Moreover, we check whether a return to the depot is possible after adding a customer by invoking the extended break scheduling method.

5 Conclusion

We proposed a solution method for the VRPTW including the full EC social legislation. The method satisfies both the European legislation on drivers' driving hours and on drivers' working hours, formalized in Regulation (EC) No 561/2006 and Directive 2002/15/EC, respectively. It also considers all modified rules in these laws. To the best of our knowledge, this is the first paper considering both Regulation (EC) No 561/2006 and Directive 2002/15/EC, as well as the modified rules in there.

We proposed a basic break scheduling method without the modified rules which is embedded in a restricted dynamic programming framework to construct the vehicle routes. The methodology applied to scheduling the breaks is to maximize local flexibility at the last visited customer of the partial routes. This is done by minimizing the start service time and by maximizing the available driving and working time after service without having to schedule a rest period or a break. This methodology both fits well in the restricted dynamic programming framework as well as in practice. The basic break scheduling method is extended with the modified rules, in which this methodology is maintained such that local flexibility is increased even further.

The computational results show that the basic method outperforms state of the art heuristics for the VRPTW with the EC social legislation. The average number of vehicle routes is reduced by more than 18% and the average travel distance by more than 5%. On top of that, the computational effort of our approach is much smaller than for these state of the art methods. The reason for this remarkable performance is that complex timing restictions can be incorporated in the restricted dynamic programming framework without increasing its running time complexity. In contrary, the running time complexity of evaluating moves in local search methods does generally increase with such timing restrictions, since these moves can only be evaluated by considering the whole routes involved. The running time complexity of the restricted DP heuristic is maintained, since customers are sequentially added to the end of partial vehicle routes and the quality of such partial routes is estimated locally.

The results also show that Directive 2002/15/EC on drivers' working hours has a significant impact on the VRPTW solutions and, therefore, cannot be ignored when constructing the vehicle routes. Finally, the results show that the modified rules allow significant cost reductions by reducing the number of vehicles by more than 4%, on average, and the total travel distance by more than 1.5%, on average. Therefore, it is highly recommended that these modified rules are exploited in practice and are incorporated in solution methods for the VRPTW.

Appendix A

Algorithm 1 Basic break scheduling method

Input: state dimensions t_{nbw}, t_{nbd}, t_{nr}, t_{dd}, t_{ww}, and t_{wd}, driving time d_{ij}, and service completion time c_i at customer i

Output: state dimensions t_{nbw}, t_{nbd}, t_{nr}, t_{dd}, t_{ww}, and t_{wd}, and service completion time c_j at customer j

$a_j \Leftarrow c_i$ // initialize arrival time
$\delta_{ij} \Leftarrow d_{ij}$ // initialize remaining driving time
$\Delta \Leftarrow \min\{\delta_{ij}, 6 - t_{nbw}, 4.5 - t_{nbd}, 13 - t_{nr}, 9 - t_{dd}\}$ // initialize Δ
$c_j \Leftarrow \infty$ // initialize completion time
if $d_{ij} + d_{j0} > 56 - t_{wd}$ **then** // not enough weekly driving time
  STOP // adding customer j to the partial route is not feasible
end if
if $d_{ij} + s_j + d_{j0} > 60 - t_{ww}$ **then** // not enough weekly working time
  STOP
end if
while $\delta_{ij} > \Delta$ **do** // a break or rest must be scheduled during travel
  if $\Delta = 9 - t_{dd}$ **or** $\Delta + 0.75 \geq 13 - t_{nr}$ **then** // a rest must be scheduled (because of the driving limit or the non-rest limit. The latter limit may also be reached when a break must be scheduled.)
    $a_j \Leftarrow a_j + \Delta + 11 + RestExtension$ // Procedure 1
    $t_{dd} \Leftarrow 0$, $t_{nr} \Leftarrow 0$
  else // a break must be scheduled
    $a_j \Leftarrow a_j + \Delta + 0.75$
    $t_{dd} \Leftarrow t_{dd} + \Delta$
    $t_{nr} \Leftarrow t_{nr} + \Delta + 0.75$
  end if
  $t_{nbw} \Leftarrow 0$, $t_{nbd} \Leftarrow 0$
  $\delta_{ij} \Leftarrow \delta_{ij} - \Delta$
  $\Delta \Leftarrow \min(\delta_{ij}, 6 - t_{nbw}, 4.5 - t_{nbd}, 13 - t_{nr}, 9 - t_{dd})$
end while

continued on next page

continued from previous page

if $13 - t_{nr} < \max\{0, e_j - a_j\} + s_j$ **or** $(6 - t_{nbw} < s_j$ **and** $13 - (t_{nr} + 0.75 < \max\{0, e_j - (a_j + 0.75)\} + s_j)$ **or** $a_j + 11 \leq e_j$ **then** // a rest is required before service, or a break is required before service causing a rest to be required before service, or a rest can be taken during waiting time
    $t_{dd} \Leftarrow 0$, $t_{nr} \Leftarrow 0$, $t_{nbw} \Leftarrow 0$, $t_{nbd} \Leftarrow 0$
    $a_j \Leftarrow \max\{a_j + 11, e_j\}$
end if
if $6 - t_{nbw} < s_j$ **or** $a_j + 0.75 \leq e_j$ **then** // a break is required, or can be taken during waiting time
    $t_{nr} \Leftarrow t_{nr} + \max\{0.75, e_j - a_j\}$
    $t_{nbw} \Leftarrow 0$, $t_{nbd} \Leftarrow 0$
    $a_j \Leftarrow \max\{a_j + 0.75, e_j\}$
end if
if $a_j > l_j$ **then** // arrival time exceeds due date
    STOP
end if

Procedure 1 Rest extension

Output: rest extension *ext*

$ext \Leftarrow 0$
if $\delta_{ij} > 9$ **then** // remaining driving time requires another rest
    STOP // only the last rest before arrival is possibly extended
end if
$a_j \Leftarrow a_j + \delta_{ij} + \lfloor \delta_{ij}/4.5 \rfloor * 0.75$
if $a_j + 11 > e_j$ **then** // no rest possible in waiting time
    $ext \Leftarrow \max\{0, e_j - a_j\}$
end if

Appendix B

Algorithm 2 Extended break scheduling method: outline changes with respect to Algorithm 1

1: To account for t_{dw}, replace Line 3 and 22 by
$\Delta \Leftarrow \min\{\delta_{ij}, 9 - t_{dw}, 6 - t_{nbw}, 4.5 - t_{nbd}, 13 - t_{nr}, 9 - t_{dd}\}$

2: In line 12, 16, and 18, the break duration of 0.75 becomes 0.25, 0.5, or 0.75, as described in the following cases:
a) when break forced by t_{nbd} *and no split break has been scheduled: 0.75,*
b) when break forced by t_{nbd} *and a split break has been scheduled: 0.5,*
c) when break forced by t_{nbw} *and no split break has been scheduled: 0.5,*
d) when break forced by t_{nbw} *and a split break has been scheduled: 0.25,*
e) when break forced by t_{dw}*: 0.25.*

3: In line 3, 12, and 22, the nonrest duration of 13 becomes 13 or 15, as described in the following cases:
a) no split rest has been scheduled: 13
b) a split rest has been scheduled: 15

4: In Procedure 1, first calculate arrival times and state dimensions at customer j for the following strategies:
a) use no modified rules,
b) use only modified rule on extending driving times,
c) use only modified rule on reducing rest periods,
d) use modified rules on extending driving times and reducing rest periods.
Next, select best strategy α using the following hierarchical criteria:
1) service completion time is minimal,
2) a rest is possible during waiting time,
3) highest rank in the list above.
Finally, determine rest extension similarly as in Procedure 1, but using the modified rules in strategy α

5: Upon arrival at customer j: Procedure 2 describes the cases in which a rest or break is scheduled before starting service.

Procedure 2 Rest or break duration upon arrival at customer j, outline cases (extended break scheduling method)

Upon arrival, check/schedule hierarchically:

1) if at least 11 hours waiting time, schedule a rest,

2) if at least 9 hours waiting time and a split rest has been scheduled, schedule a reduced rest,

3) if at least 9 hours waiting time and $n_{rr} < 3$*, schedule a reduced rest,*

4) if rest required before service and a split rest has been scheduled, schedule a reduced rest,

5) if rest required before service and $n_{rr} < 3$*, schedule a reduced rest,*

6) if rest required before service, schedule a rest,

7) if at least 3 hours waiting time, schedule a split rest,

8) if at least 0.75 hours waiting time, schedule a 0.75 hour break,

9) if at least 0.5 hours waiting time, schedule a 0.5 hour break,

10) if a break is required before service, forced by t_{nbw}*, and no split break has been scheduled, schedule a 0.5 hour break,*

11) if a break is required before service, schedule a 0.25 hour break.

Acknowledgment

This work was financially supported by Stichting Transumo through the project ketensynchronisatie and by the German Research Foundation (DFG) as part of the Collaborative Research Centre 637 “Autonomous Cooperating Logistics Processes - A Paradigm Shift and its Limitations” (subproject B9).

References

Bartodziej P, Derigs U, Malcherek D, Vogel U (2009) Models and algorithms for solving combined vehicle and crew scheduling problems with rest constraints: an application to road feeder service planning in air cargo transportation. OR Spectrum 31(2):405–429

Bellman R (1962) Dynamic programming treatment of the travelling salesman problem. Journal of the Association for Computing Machinery 9(1):61

Campbell AM, Savelsberg M (2004) Efficient insertion heuristics for vehicle routing and scheduling problems. Transportation Science 38:369–378

Cordeau JF, Desaulniers G, Desrosiers J, Solomon MM, Soumis F (2002) VRP with time windows. In: Toth P, Vigo D (eds) The Vehicle Routing Problem, SIAM, Philadelphia, pp 157–193

European Union (1985) Regulation (EEC) No 3820/85 of 20 December 1985 on the harmonization of certain social legislation relating to road transport. Official Journal of the European Union L 370, 31.12.1985

European Union (2002) Directive 2002/15/EC of the European Parliament and of the Council of 11 March 2002 on the organisation of the working time of persons performing mobile road transport activities. Official Journal of the European Union L 080, 23.03.2002

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

Funke B, Grünert T, Irnich S (2005) Local search for vehicle routing and scheduling problems: Review and conceptual integration. Journal of Heuristics 11(4):267–306

Gietz M (1994) Computergestützte Tourenplanung mit zeitkritischen Restriktionen. Physica-Verlag, Heidelberg

Goel A (2009) Vehicle scheduling and routing with drivers' working hours. Transportation Science 43(1):17–26

Goel A, Gruhn V (2006) Solving a dynamic real-life vehicle routing problem. In: Haasis HD, Kopfer H, Schönberger J (eds) Operations Research Proceedings 2005, Springer, pp 367–372

Gromicho J, van Hoorn J, Kok AL, Schutten JMJ (2008) The flexibility of restricted dynamic programming for the VRP. Beta working paper series 266

Held M, Karp RM (1962) A dynamic programming approach to sequencing problems. Journal of the Society for Industrial and Applied Mathematics 10(1):196

Kopfer H, Meyer CM (2009) A model for the traveling salesman problem including the EC regulations on driving hours. In: Tuma A, Fleischmann B, Borgwardt K, Klein R (eds) Operations Research Proceedings 2009, Springer, Berlin Heidelberg New York

Kopfer H, Meyer CM, Wagenknecht A (2007) Die EU-Sozialvorschriften und ihr Einfluss auf die Tourenplanung. Logistik Management 9(2):32–47

Malandraki C, Dial RB (1996) A restricted dynamic programming heuristic algorithm for the time dependent traveling salesman problem. European Journal of Operational Research 90(1):45–55

Rochat Y, Semet F (1994) A tabu search approach for delivering pet food and flour in switzerland. Journal of the Operational Research Society 45:1233–1246

Savelsberg M, Sol M (1998) Drive: Dynamic routing of independent vehicles. Operations Research 46:474–490

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

Stumpf P (1998) Tourenplanung im speditionellen Güterverkehr. GVB, Nürnberg

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

Zäpfel G, Bögl M (2008) Multi-period vehicle routing and crew scheduling with outsourcing options. International Journal of Production Economics 113(2):980–996