The Canadian Minimum Duration Truck Driver Scheduling Problem

Asvin Goel[1,2]

[1] Zaragoza Logistics Center, Spain
agoel@zlc.edu.es

[2] Applied Telematics/e-Business Group,
Department of Computer Science, University of Leipzig, Germany
asvin.goel@uni-leipzig.de

May 30, 2011

Abstract

In Canada transport companies must ensure that truck drivers can comply with Canadian Commercial Vehicle Drivers Hours of Service Regulations. Canadian regulations comprise the provisions found in U.S. hours of service regulations as well as additional constraints on the maximum amount of driving and the minimum amount of off-duty time on each day. This paper presents a mixed integer programming formulation and an iterative dynamic programming approach for minimising the duration of truck driver schedules complying with Canadian hours of service regulations. Computational experiments show that schedule durations can be significantly reduced compared with a previously presented approach which only focusses on feasibility.

Keywords: Canadian Hours of Service Regulations, Multiple Time Windows, Vehicle Scheduling

1 Introduction

Economic pressure forces transport companies to maximise vehicle utilisation and minimise labour costs. An excessive focus on these economic objectives can lead to unreasonable tight schedules which do not give drivers enough time for recuperation. In a study on sleeping patterns of truck drivers by Mitler et al. (1997), Canadian truck drivers regularly drove up to 13 hours between rest periods of 8 consecutive hours. The study revealed that on average drivers sleep less than 5 hours per day which is 2 hours less than the average ideal reported by the drivers. More than half of the drivers had at least one six-minute interval of drowsiness while driving within the five-day study. This indicates that driver fatigue contributes to reduced road safety. An important key in preventing fatigue related accident risk is to explicitly consider driving and working hour regulations when generating truck driver schedules

The first research known to the author explicitly considering government regulations concerning working hours of truck drivers is the work by Xu et al. (2003) who study a rich pickup and delivery problem with multiple time windows and restrictions on drivers' working hours as imposed by the U.S. Department of Transportation. Xu et al. (2003) conjecture that the problem of finding a feasible schedule complying with U.S. Hours of Service regulations is NP-hard in the presence of multiple time windows. Archetti and Savelsbergh (2009) show that if each location must be visited within a single time window, schedules complying with U.S. Hours of Service regulations can be determined in polynomial time. Goel and Kok (2010) show that schedules complying with U.S. Hours of Service regulations can also be determined in polynomial time in the case of multiple time windows, if the gap between subsequent time windows at the same location is at least 10 hours. This situation occurs, for example, if, because of opening hours of docks, handling operations can only be performed between 8.00 AM and 10.00 PM. Rancourt et al. (2010) present a tabu search heuristic for a combined vehicle routing and truck driver scheduling problem using a modified version of the approach by Goel and Kok (2010).

Heuristics for combined vehicle routing and truck driver scheduling in Europe are presented by Goel (2009), Kok et al. (2010), and Prescott-Gagnon et al. (2010). The work by Goel (2010) presents the first method capable of finding a feasible schedule complying with European regulations if such a schedule exists. European Union regulations are more complex than U.S. hours of service regulation, because they require that in addition to rest periods, in which drivers can sleep, shorter breaks for recuperation must be scheduled after four and a half hours of driving. An exact approach and various heuristics for truck driver scheduling in Australia are presented by Goel et al. (2011). Due to the specific structure of Australian regulations these approaches are completely different to the approaches for truck driver scheduling in the Unites States and the European Union.

Kok et al. (2011) present a mixed integer programming formulation for the minimum duration truck driver scheduling problem in the European Union focusing on a planning horizon of at most 13 hours, i.e. a working day. Unlike the other approaches for truck driver scheduling, the approach by Kok et al. (2011) assumes that truck drivers may only take rest periods at customer locations and at suitable parking lots. For longer planning horizons of multiple days Goel (2011) presents a mixed integer programming formulation for minimum duration truck driver scheduling problems with multiple time windows where truck drivers may only take rest periods at customer locations and at suitable parking lots. The formulation can be used for the regulations in the United States and the European Union.

Canadian Commercial Vehicle Drivers Hours of Service Regulations comprise regulations similar to those in the United States and additional constraints limiting the maximum amount of driving time on a day and the minimum amount of off-duty and break time on a day. Goel and Rousseau (2011) introduce the Canadian truck driver scheduling problem which is the problem of determining whether it is possible to schedule driving and working hours of truck drivers in such a way that a sequence of locations can be visited within given time windows and that Canadian hours of service regulations are complied with. Goel and Rousseau (2011) present a dynamic programming approach for the Canadian truck driver scheduling problem and show that the additional constraints which differentiate Canadian regulations from U.S. regulations significantly complicate the problem.

This paper presents a mixed integer programming formulation for a variant of the Canadian truck driver scheduling problem in which each customer must be visited within one of multiple time windows, in which rest periods must only be taken at customer locations or at suitable parking lots, and in which the objective is to minimise the total duration of the schedule. An iterative dynamic programming approach is presented which minimises schedule durations using an adaptation of a dynamic programming approach presented by Goel and Rousseau (2011). Computational experiments demonstrate that the duration of schedules is significantly smaller compared to the duration of schedules obtained by the adaptation of the dynamic programming approach. Furthermore, computational experiments show that the minimum duration Canadian truck driver scheduling problem can be solved faster with the iterative dynamic programming approach than solving the mixed integer program with CPLEX 12.

The remainder of this paper is organised as follows. In Section 2 Canadian hours of service regulations are described. Section 3 presents the mixed integer programming formulation. Section 4 presents the iterative dynamic programming approach. In Section 5 computational experiments are presented demonstrating the effectiveness of the iterative dynamic programming approach. Section 6 concludes the paper.

2 Canadian Commercial Vehicle Drivers Hours of Service Regulations

Canadian regulations concerning driving and working hours of commercial vehicles are described in Transport Canada (2005) and interpreted in Canadian Council of Motor Transport Administrators (2007). In Canada two sets of regulations exist, one of which applies to driving conducted south of latitude 60° N and one to driving north of latitude 60° N. In the remainder of this paper we focus on the subset of regulations applicable for driving south of latitude 60° N because this is the area of major economic concern. The regulation defines *on-duty time* as the period that begins when a driver begins work and ends when the driver stops work and includes any time during which the driver is driving or conducting any other work. *Off-duty time* is defined by any period other than on-duty time. The regulation imposes restrictions on the maximum amount of on-duty time and the minimum amount of off-duty time during a *day*. According to the regulation a *day* means a 24-hour period that begins at some time designated by the motor carrier. For simplicity and w.l.o.g. let us assume in the remainder that this time is midnight. The regulations imposes the following constraints on truck driver schedules:

1. The driver must not drive after accumulating 13 hours of driving time since the end of the last period of at last 8 consecutive hours of off-duty time.

2. The driver must not drive after accumulating 14 hours of on-duty time since the end of the last period of at last 8 consecutive hours of off-duty time.

3. The driver must not drive after 16 hours of time have elapsed since the end of the last period of at least 8 consecutive hours of off-duty time.

4. The driver must not drive for more than 13 hours in a day.

5. The driver must accumulate at least 10 hours of off-duty time in a day.

6. At least 2 hours of off-duty time must be taken on a day which are not a part of a period of 8 consecutive hours of off-duty time as required by provisions 1 to 3. If a period of more than 8 consecutive hours of off-duty time is scheduled, the amount exceeding the 8th hour may contribute to these 2 hours.

7. Periods of less than 30 minutes, in which the driver is neither driving nor working, do not count toward the minimum off-duty time requirements required by provisions 5 and 6, even though they are considered as off-duty time by the definition.

Table 1 summarises the parameters imposed by the regulation and the notation used in this paper.

3 A Mixed Integer Programming Model

Let us consider a sequence of n locations which shall be visited by a truck driver. At each location $1 \leq i \leq n$ some stationary work of duration w_i shall be conducted. This work shall begin within one of multiple disjunct time windows. The number of time windows at location $1 \leq i \leq n$ shall be denoted by T_i. For each $1 \leq \tau \leq T_i$ the τth time window at location i shall be denoted by the interval $[t^{\text{min}}_{i,\tau}, t^{\text{max}}_{i,\tau}]$. The driving time required for moving from location i to $i+1$ shall be denoted by $\delta_{i,i+1}$. The time horizon shall be denoted by t^{horizon}. We assume that drivers may only take rest periods after arrival at a location and before starting the work activity at the location. The model presented in this section can also be used if drivers may take rest periods after completing the work or at suitable parking lots on the trip from one location to another. As indicated by Goel (2011) dummy

Notation	Value	Description
t^{rest}	8 h	The minimum duration of a rest period
t^{drive}	13 h	The maximum accumulated driving time between two consecutive rest periods and the maximum accumulated driving time on a day
$t^{\text{on-duty}}$	14 h	The amount of accumulated on-duty time since the last rest period after which a driver may only continue to drive if a new rest period is taken
t^{elapsed}	16 h	The amount of time elapsed since the last rest period after which a driver may only continue to drive if a new rest period is taken
t^{day}	24 h	The duration of a day
$t^{\text{off-duty}}$	10 h	The minimum amount of off-duty time on a day
t^{break}	2 h	The minimum amount of off-duty time on a day which is not part of a rest period
t^{idle}	$\frac{1}{2}$ h	The minimum length of an off-duty period period to be counted

Table 1: Parameters imposed by the regulation

locations with zero working time can be inserted in the tour in order to allow drivers to take take rest periods after completing the work or at suitable parking lots.

The Canadian minimum duration truck driver scheduling problem (CAN-MDTDSP) is the problem of determining a schedule complying with Canadian regulations in which all work activities begin within one of the corresponding time windows and which has the minimal duration.

Let us begin with presenting a mixed integer programming formulation of a relaxation of the CAN-MDTDSP in which only provisions 1 to 3 of the regulation must be complied with. These provisions are similarly structured as the provisions in the United States. Thus we can provide a mixed integer programming formulation of the relaxed CAN-MDTDSP similar to the formulation presented by Goel (2011).

For each location $1 \leq i \leq n$ the formulation comprises variables x_i^{arrival}, $x_i^{\text{restbegin}}$, x_i^{restend}, x_i^{start}, x_i^{end} representing the arrival time, the start and end time of a period of at least 8 consecutive hours of off-duty time, and the start and end time of the work at location i. For each location $1 \leq i \leq n$ and each time window $1 \leq \tau \leq T_i$ the formulation comprises a binary variable $y_{i,\tau}$ indicating whether the τth time window of location i is used ($y_{i,\tau} = 1$) or not ($y_{i,\tau} = 0$). Furthermore, for each location

$1 \leq i \leq n$ the formulation comprises the binary variable z_i^{rest} indicating whether a period of at least 8 consecutive hours of off-duty time is taken at location i ($z_i^{\text{rest}} = 1$) or not ($z_i^{\text{rest}} = 0$).

The relaxed CAN-MDTDSP is

minimise

$$x_n^{\text{end}} - x_1^{\text{start}} \tag{1}$$

subject to

$$x_i^{\text{arrival}} \leq x_i^{\text{restbegin}} \text{ for all } 1 \leq i \leq n \tag{2.1}$$

$$x_i^{\text{restbegin}} + z_i^{\text{rest}} \cdot t^{\text{rest}} \leq x_i^{\text{restend}} \text{ for all } 1 \leq i \leq n \tag{2.2}$$

$$x_i^{\text{restend}} \leq x_i^{\text{start}} \text{ for all } 1 \leq i \leq n \tag{2.3}$$

$$x_i^{\text{start}} + w_i = x_i^{\text{end}} \text{ for all } 1 \leq i \leq n \tag{2.4}$$

$$x_i^{\text{end}} + \delta_{i,i+1} = x_{i+1}^{\text{arrival}} \text{ for all } 1 \leq i < n \tag{2.5}$$

$$\sum_{\tau=1}^{\tau \leq T_i} y_{i,\tau} = 1 \text{ for all } 1 \leq i \leq n \tag{3.1}$$

$$y_{i,\tau} t_{i,\tau}^{\text{min}} \leq x_i^{\text{start}} \text{ for all } 1 \leq i \leq n, 1 \leq \tau \leq T_i \tag{3.2}$$

$$x_i^{\text{start}} \leq t^{\text{horizon}} - y_{i,\tau}(t^{\text{horizon}} - t_{i,\tau}^{\text{max}}) \text{ for all } 1 \leq i \leq n, 1 \leq \tau \leq T_i \tag{3.3}$$

$$x_k^{\text{arrival}} - x_i^{\text{start}} \leq t^{\text{elapsed}} + t^{\text{horizon}} \sum_{j=i+1}^{j<k} z_j^{\text{rest}} \text{ for all } 1 \leq i < k \leq n : \delta_{k-1,k} > 0 \tag{4}$$

$$\sum_{j=i}^{j<k} \delta_{j,j+1} \leq t^{\text{drive}} + t^{\text{horizon}} \sum_{j=i+1}^{j<k} z_j^{\text{rest}} \text{ for all } 1 \leq i < k \leq n : \delta_{k-1,k} > 0 \tag{5}$$

$$\sum_{j=i}^{j<k} \delta_{j,j+1} + \sum_{j=i}^{j<k} w_j \leq t^{\text{on-duty}} + t^{\text{horizon}} \sum_{j=i+1}^{j<k} z_j^{\text{rest}} \text{ for all } 1 \leq i < k \leq n : \delta_{k-1,k} > 0 \tag{6}$$

$$x_i^{\text{arrival}} \in [0, t^{\text{horizon}}] \text{ for all } 1 \leq i \leq n \tag{7.1}$$

$$x_i^{\text{restbegin}} \in [0, t^{\text{horizon}}] \text{ for all } 1 \leq i \leq n \tag{7.2}$$

$$x_i^{\text{restend}} \in [0, t^{\text{horizon}}] \text{ for all } 1 \leq i \leq n \tag{7.3}$$

$$x_i^{\text{start}} \in [0, t^{\text{horizon}}] \text{ for all } 1 \leq i \leq n \tag{7.4}$$

$$x_i^{\text{end}} \in [0, t^{\text{horizon}}] \text{ for all } 1 \leq i \leq n \tag{7.5}$$

$$y_{i,\tau} \in \{0, 1\} \text{ for all } 1 \leq i \leq n, \tau \in T_i \tag{7.6}$$

$$z_i^{\text{rest}} \in \{0, 1\} \text{ for all } 1 \leq i \leq n \tag{7.7}$$

The objective function (1) is to minimise the duration between the start of the first work and the end of the last work. Constraints (2.1) to (2.5) demand that all time values coincide with the course of events. If $z_i^{\text{rest}} = 1$ for some$1 \leq i \leq n$ then constraint (2.2) demands that the end of the rest period is at least t^{rest} higher than the begin of the rest period. Otherwise, $x_i^{\text{restbegin}}$ and x_i^{restend} are without meaning and can be placed anywhere between x_i^{arrival} and x_i^{start}. Constraint (3.1) demands that at any location exactly one of the time windows is used. Constraints (3.2) and (3.3) are the time windows constraints. Constraint (4) demands that no driving is conducted after t^{elapsed} has elapsed since the last rest period of at least t^{rest} duration. Constraint (5) demands that the accumulated amount of driving without a rest period of at least t^{rest} duration does not exceed t^{drive}. Constraint (6) demands that no driving is conducted after $t^{\text{on-duty}}$ of driving and work have been accumulated since the last rest period of at least t^{rest} duration. The variable domains are given by (7.1) to (7.7).

The relaxed CAN-MDTDSP defined by (1) to (7) does not consider the constraints of the regulation regarding the maximum amount of driving time and the minimum amount of off-duty and break time on each day. In order to consider these constraints we need to add variables to the problem formulation which indicate the day of the arrival time at a location as well as the day of the start and end of the work conducted at a location. For each location $1 \leq i \leq n$ we add the integer variables d_i^{arrival}, d_i^{start}, and d_i^{end} and we add the following constraints.

$$(d_i^{\text{arrival}} - 1) \cdot t^{\text{day}} \leq x_i^{\text{arrival}} < d_i^{\text{arrival}} \cdot t^{\text{day}} \text{ for all } 1 \leq i \leq n \tag{8.1}$$

$$(d_i^{\text{start}} - 1) \cdot t^{\text{day}} \leq x_i^{\text{start}} < d_i^{\text{start}} \cdot t^{\text{day}} \text{ for all } 1 \leq i \leq n \tag{8.2}$$

$$(d_i^{\text{end}} - 1) \cdot t^{\text{day}} \leq x_i^{\text{end}} < d_i^{\text{end}} \cdot t^{\text{day}} \text{ for all } 1 \leq i \leq n \tag{8.3}$$

$$d_i^{\text{arrival}} \in \{1, \ldots, \lfloor t^{\text{horizon}}/t^{\text{day}} \rfloor + 1\} \text{ for all } 1 \leq i \leq n \tag{8.4}$$

$$d_i^{\text{start}} \in \{1, \ldots, \lfloor t^{\text{horizon}}/t^{\text{day}} \rfloor + 1\} \text{ for all } 1 \leq i \leq n \tag{8.5}$$

$$d_i^{\text{end}} \in \{1, \ldots, \lfloor t^{\text{horizon}}/t^{\text{day}} \rfloor + 1\} \text{ for all } 1 \leq i \leq n \tag{8.6}$$

We assume that, if an optimal solution exists, the optimal solution satisfies

$$d_i^{\text{end}} \leq d_i^{\text{arrival}} + 1 \text{ for all } 1 \leq i \leq n.$$

Note that this condition is only violated if $x_i^{\text{start}} - x_i^{\text{arrival}}$ is very large for any $1 \leq i \leq n$, i.e. there is a long period of off-duty time. If there is the possibility that this condition cannot be satisfied for all locations, we can add dummy locations to the tour with zero driving time, zero working time and unlimited time window. By this the long period of off-duty time can be split into several parts and the condition of the assumption can be satisfied.

The constraints on the maximum amount of driving time on a day can be considered by introducing additional variables r_i^{drive}, p_i^{drive}, and q_i^{drive}. r_i^{drive} represents the accumulated amount of driving until arrival at location i. p_i^{drive} and q_i^{drive} represent the accumulated amount of driving until the end of the last day preceding d_i^{arrival} and d_i^{end}. We add the following constraints to the formulation.

$$r_i^{\text{drive}} = \sum_{j=1}^{j<i} \delta_{j,j+1} \text{ for all } 1 \leq i \leq n \tag{9.1}$$

$$p_1^{\text{drive}} = 0 \tag{9.2}$$

$$r_n^{\text{drive}} \leq q_n^{\text{drive}} + t^{\text{drive}} \tag{9.3}$$

Constraint (9.1) determines the value of r_i^{drive}. Constraint (9.2) initialises p_1^{drive} and constraint (9.3) demands that the accumulated amount of driving until the end of the schedule does not exceed the accumulated amount of driving until the end of the day preceding d_n^{end} plus t^{drive}.

If $d_i^{\text{arrival}} = d_i^{\text{end}}$ for some $1 \leq i \leq n$ then the value of q_i^{drive} must have the same value as p_i^{drive}. This is achieved by adding the following constraints to the formulation.

$$p_i^{\text{drive}} \leq q_i^{\text{drive}} + (d_i^{\text{end}} - d_i^{\text{arrival}}) \cdot t^{\text{horizon}} \text{ for all } 1 \leq i \leq n \tag{9.4}$$

$$q_i^{\text{drive}} \leq p_i^{\text{drive}} + (d_i^{\text{end}} - d_i^{\text{arrival}}) \cdot t^{\text{horizon}} \text{ for all } 1 \leq i \leq n \tag{9.5}$$

If $d_i^{\text{arrival}} < d_i^{\text{end}}$ for some $1 \leq i \leq n$ then the time between x_i^{arrival} and x_i^{end} includes midnight. As no driving is conducted in this time q_i^{drive} must have the same value as r_i^{drive} and we add the following constraints to the formulation.

$$q_i^{\text{drive}} \leq r_i^{\text{drive}} + (d_i^{\text{arrival}} + 1 - d_i^{\text{end}}) \cdot t^{\text{horizon}} \text{ for all } 1 \leq i \leq n \tag{9.6}$$

$$r_i^{\text{drive}} \leq q_i^{\text{drive}} + (d_i^{\text{arrival}} + 1 - d_i^{\text{end}}) \cdot t^{\text{horizon}} \text{ for all } 1 \leq i \leq n \tag{9.7}$$

Furthermore, we need the make sure that the accumulated amount of driving on day d_i^{arrival} does not exceed the limit. This is achieved by adding the following constraints to the formulation.

$$q_i^{\text{drive}} \leq p_i^{\text{drive}} + t^{\text{drive}} + (d_i^{\text{arrival}} + 1 - d_i^{\text{end}}) \cdot t^{\text{horizon}} \text{ for all } 1 \leq i \leq n \tag{9.8}$$

If $d_{i-1}^{\text{end}} = d_i^{\text{arrival}}$ for some $1 < i \leq n$ then the value of p_i^{drive} must have the same value as q_{i-1}^{drive}. This is achieved by adding the following constraints to the formulation.

$$p_i^{\text{drive}} \leq q_{i-1}^{\text{drive}} + (d_i^{\text{arrival}} - d_{i-1}^{\text{end}}) \cdot t^{\text{horizon}} \text{ for all } 1 < i \leq n \tag{9.9}$$

$$q_{i-1}^{\text{drive}} \leq p_i^{\text{drive}} + (d_i^{\text{arrival}} - d_{i-1}^{\text{end}}) \cdot t^{\text{horizon}} \text{ for all } 1 < i \leq n \tag{9.10}$$

If $d_{i-1}^{\text{end}} < d_i^{\text{arrival}}$ for some $1 \leq i \leq n$ then the time between x_{i-1}^{end} and x_i^{arrival} includes midnight. The amount of driving conducted on day d_{i-1}^{end} is determined by subtracting the amount of driving between midnight and x_i^{arrival} from r_i^{drive}. This is achieved by adding the following constraints to the formulation.

$$p_i^{\text{drive}} \leq r_i^{\text{drive}} - (x_i^{\text{arrival}} - d_{i-1}^{\text{end}} \cdot t^{\text{day}}) + (d_{i-1}^{\text{end}} + 1 - d_i^{\text{arrival}}) \cdot t^{\text{horizon}} \text{ for all } 1 < i \leq n \tag{9.11}$$

$$r_i^{\text{drive}} - (x_i^{\text{arrival}} - d_{i-1}^{\text{end}} \cdot t^{\text{day}}) \leq p_i^{\text{drive}} + (d_{i-1}^{\text{end}} + 1 - d_i^{\text{arrival}}) \cdot t^{\text{horizon}} \text{ for all } 1 < i \leq n \tag{9.12}$$

We need the make sure that the accumulated amount of driving on day d_i^{arrival} does not exceed the limit. This is achieved by adding the following constraints to the formulation.

$$p_i^{\text{drive}} \leq q_{i-1}^{\text{drive}} + t^{\text{drive}} + (d_{i-1}^{\text{end}} + 1 - d_i^{\text{arrival}}) \cdot t^{\text{horizon}} \text{ for all } 1 < i \leq n \tag{9.13}$$

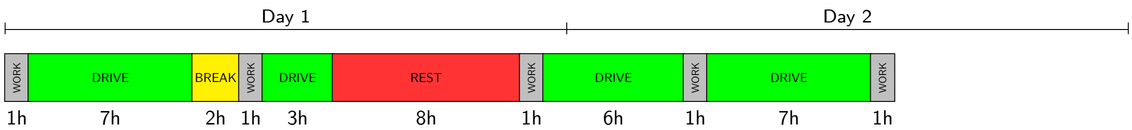
Eventually, we add the variable domains to the formulation.

$$p_i^{\text{drive}} \in [0, t^{\text{horizon}}] \text{ for all } 1 \leq i \leq n \tag{9.14}$$

$$q_i^{\text{drive}} \in [0, t^{\text{horizon}}] \text{ for all } 1 \leq i \leq n \tag{9.15}$$

$$r_i^{\text{drive}} \in [0, t^{\text{horizon}}] \text{ for all } 1 \leq i \leq n \tag{9.16}$$

The example illustrated in Figure 1 shows the values of r_i^{drive}, p_i^{drive}, and q_i^{drive} resulting from constraints (9.1) to (9.16).



$$\begin{array}{lllllllll}
r_1^{\text{drive}} = 0, & p_1^{\text{drive}} = 0, & q_1^{\text{drive}} = 0, & r_1^{\text{offduty}} = 0, & p_1^{\text{offduty}} = 0, & q_1^{\text{offduty}} = 0, & r_1^{\text{break}} = 0, & p_1^{\text{break}} = 0, & q_1^{\text{break}} = 0 \\
r_2^{\text{drive}} = 7, & p_2^{\text{drive}} = 0, & q_2^{\text{drive}} = 0, & r_2^{\text{offduty}} = 2, & p_2^{\text{offduty}} = 0, & q_2^{\text{offduty}} = 0, & r_2^{\text{break}} = 2, & p_2^{\text{break}} = 0, & q_2^{\text{break}} = 0 \\
r_3^{\text{drive}} = 10, & p_3^{\text{drive}} = 0, & q_3^{\text{drive}} = 0, & r_3^{\text{offduty}} = 10, & p_3^{\text{offduty}} = 0, & q_3^{\text{offduty}} = 0, & r_3^{\text{break}} = 2, & p_3^{\text{break}} = 0, & q_3^{\text{break}} = 0 \\
r_4^{\text{drive}} = 16, & p_4^{\text{drive}} = 11, & q_4^{\text{drive}} = 11, & r_4^{\text{offduty}} = 10, & p_4^{\text{offduty}} = 10, & q_4^{\text{offduty}} = 10, & r_4^{\text{break}} = 2, & p_4^{\text{break}} = 2, & q_4^{\text{break}} = 2 \\
r_5^{\text{drive}} = 23, & p_5^{\text{drive}} = 11, & q_5^{\text{drive}} = 11, & r_5^{\text{offduty}} = 10, & p_5^{\text{offduty}} = 10, & q_5^{\text{offduty}} = 10, & r_5^{\text{break}} = 2, & p_5^{\text{break}} = 2, & q_5^{\text{break}} = 2
\end{array}$$

Figure 1: Example

In order to consider the constraints on the minimum amount of off-duty and break time on a day we must distinguish between off-duty periods of less than t^{idle} and off-duty periods of at least t^{idle} duration. For this we add a binary variable z_i^{idle} which indicates whether $x_i^{\text{start}} - x_i^{\text{arrival}} < t^{\text{idle}}$ ($z_i^{\text{idle}} = 1$) or not ($z_i^{\text{idle}} = 0$). We add the following constraints to the formulation.

$$z_1^{\text{idle}} = 0 \tag{10.1}$$

$$x_i^{\text{start}} - x_i^{\text{arrival}} + z_i^{\text{idle}} \cdot t^{\text{horizon}} \geq t^{\text{idle}} \text{ for all } 1 < i \leq n \tag{10.2}$$

$$x_i^{\text{start}} - x_i^{\text{arrival}} < t^{\text{idle}} + (1 - z_i^{\text{idle}}) \cdot t^{\text{horizon}} \text{ for all } 1 < i \leq n \tag{10.3}$$

$$z_i^{\text{idle}} \in \{0, 1\} \text{ for all } 1 \leq i \leq n \tag{10.4}$$

The constraints on the minimum amount of off-duty time on a day can be considered by introducing additional variables r_i^{offduty}, p_i^{offduty}, and q_i^{offduty}. r_i^{offduty} represents the accumulated amount of off-duty time until the start of the work at location i. p_i^{offduty} and q_i^{offduty} represent the accumulated amount of off-duty time until the end of the last day preceding d_i^{arrival} and d_i^{start}. We add the following constraints to the formulation.

$$r_1^{\text{offduty}} = x_1^{\text{start}} \tag{11.1}$$

$$r_{i-1}^{\text{offduty}} \leq r_i^{\text{offduty}} \leq r_{i-1}^{\text{offduty}} + x_i^{\text{start}} - x_i^{\text{arrival}} \text{ for all } 1 < i \leq n \tag{11.2}$$

$$r_i^{\text{offduty}} \leq r_{i-1}^{\text{offduty}} + (1 - z_i^{\text{idle}}) \cdot t^{\text{horizon}} \text{ for all } 1 < i \leq n \tag{11.3}$$

$$r_{i-1}^{\text{offduty}} + x_i^{\text{start}} - x_i^{\text{arrival}} \leq r_i^{\text{offduty}} + z_i^{\text{idle}} \cdot t^{\text{horizon}} \text{ for all } 1 < i \leq n \tag{11.4}$$

$$p_1^{\text{offduty}} = 0 \tag{11.5}$$

$$r_n^{\text{offduty}} + d_n^{\text{end}} \cdot t^{\text{day}} - x_n^{\text{end}} \geq q_n^{\text{offduty}} + t^{\text{offduty}} \tag{11.6}$$

Constraints (11.1) to (11.4) determine the value of r_i^{offduty} considering that off-duty periods of less than t^{idle} are not counted. Constraint (11.5) initialises p_1^{offduty} and constraint (11.6) demands that the accumulated amount of off-duty time until the end of the schedule plus the remaining time until the end of the last day is at least q_n^{offduty} plus t^{offduty}.

If $d_i^{\text{arrival}} = d_i^{\text{start}}$ for some $1 \leq i \leq n$ then the value of q_i^{offduty} must have the same value as p_i^{offduty}. This is achieved by adding the following constraints to the formulation.

$$p_i^{\text{offduty}} \leq q_i^{\text{offduty}} + (d_i^{\text{start}} - d_i^{\text{arrival}}) \cdot t^{\text{horizon}} \text{ for all } 1 \leq i \leq n \tag{11.7}$$

$$q_i^{\text{offduty}} \leq p_i^{\text{offduty}} + (d_i^{\text{start}} - d_i^{\text{arrival}}) \cdot t^{\text{horizon}} \text{ for all } 1 \leq i \leq n \tag{11.8}$$

If $d_i^{\text{arrival}} < d_i^{\text{start}}$ for some $1 \leq i \leq n$ then the time between x_i^{arrival} and x_i^{start} includes midnight. If $z_i^{\text{idle}} = 0$ then the amount of off-duty time on day d_i^{arrival} is determined by subtracting the amount of off-duty between midnight and x_i^{start} from r_i^{offduty}. If $z_i^{\text{idle}} = 1$ then the amount of off-duty time on day d_i^{arrival} is r_i^{offduty}. We add the following constraints to the formulation.

$$q_i^{\text{offduty}} \leq r_i^{\text{offduty}} + (d_i^{\text{arrival}} + 1 - d_i^{\text{start}}) \cdot t^{\text{horizon}} \text{ for all } 1 \leq i \leq n \tag{11.9}$$

$$r_i^{\text{offduty}} - (x_i^{\text{start}} - d_i^{\text{arrival}} \cdot t^{\text{day}}) \leq q_i^{\text{offduty}} + (d_i^{\text{arrival}} + 1 - d_i^{\text{start}}) \cdot t^{\text{horizon}} \text{ for all } 1 \leq i \leq n \tag{11.10}$$

$$r_i^{\text{offduty}} \leq q_i^{\text{offduty}} + (1 - z_i^{\text{idle}}) \cdot t^{\text{horizon}} + (d_i^{\text{arrival}} + 1 - d_i^{\text{start}}) \cdot t^{\text{horizon}} \text{ for all } 1 \leq i \leq n \tag{11.11}$$

$$q_i^{\text{offduty}} \leq r_i^{\text{offduty}} - (x_i^{\text{start}} - d_i^{\text{arrival}} \cdot t^{\text{day}}) + z_i^{\text{idle}} \cdot t^{\text{horizon}} + (d_i^{\text{arrival}} + 1 - d_i^{\text{start}}) \cdot t^{\text{horizon}} \text{ for all } 1 \leq i \leq n \tag{11.12}$$

We need the make sure that the accumulated amount of off-duty time on day d_i^{arrival} is at least t^{offduty}. This is achieved by adding the following constraints to the formulation.

$$p_i^{\text{offduty}} + t^{\text{offduty}} \leq q_i^{\text{offduty}} + (d_i^{\text{arrival}} + 1 - d_i^{\text{start}}) \cdot t^{\text{horizon}} \text{ for all } 1 \leq i \leq n \tag{11.13}$$

If $d_i^{\text{arrival}} = d_{i-1}^{\text{start}}$ for some $1 < i \leq n$ then the value of p_i^{offduty} must have the same value as q_{i-1}^{offduty}. This is achieved by adding the following constraints to the formulation.

$$p_i^{\text{offduty}} \leq q_{i-1}^{\text{offduty}} + (d_i^{\text{arrival}} - d_{i-1}^{\text{start}}) \cdot t^{\text{horizon}} \text{ for all } 1 < i \leq n \tag{11.14}$$

$$q_{i-1}^{\text{offduty}} \leq p_i^{\text{offduty}} + (d_i^{\text{arrival}} - d_{i-1}^{\text{start}}) \cdot t^{\text{horizon}} \text{ for all } 1 < i \leq n \tag{11.15}$$

If $d_{i-1}^{\text{start}} < d_i^{\text{arrival}}$ for some $1 < i \leq n$ then the time between x_{i-1}^{start} and x_i^{arrival} includes midnight. As no off-duty time is taken in this period p_i^{offduty} must have the same value as r_{i-1}^{offduty} and we add the following constraints to the formulation.

$$p_i^{\text{offduty}} \leq r_{i-1}^{\text{offduty}} + (d_{i-1}^{\text{start}} + 1 - d_i^{\text{arrival}}) \cdot t^{\text{horizon}} \text{ for all } 1 < i \leq n \tag{11.16}$$

$$r_{i-1}^{\text{offduty}} \leq p_i^{\text{offduty}} + (d_{i-1}^{\text{start}} + 1 - d_i^{\text{arrival}}) \cdot t^{\text{horizon}} \text{ for all } 1 < i \leq n \tag{11.17}$$

We need the make sure that the accumulated amount of off-duty time on day preceding d_i^{arrival} is at least t^{offduty}. This is achieved by adding the following constraints to the formulation.

$$q_{i-1}^{\text{offduty}} + t^{\text{offduty}} \leq p_i^{\text{offduty}} + (d_{i-1}^{\text{start}} + 1 - d_i^{\text{arrival}}) \cdot t^{\text{horizon}} \text{ for all } 1 < i \leq n \tag{11.18}$$

Eventually, we add the variable domains to the formulation.

$$p_i^{\text{offduty}} \in [0, t^{\text{horizon}}] \text{ for all } 1 \leq i \leq n \tag{11.19}$$

$$q_i^{\text{offduty}} \in [0, t^{\text{horizon}}] \text{ for all } 1 \leq i \leq n \tag{11.20}$$

$$r_i^{\text{offduty}} \in [0, t^{\text{horizon}}] \text{ for all } 1 \leq i \leq n \tag{11.21}$$

The example illustrated in Figure 1 shows the values of r_i^{offduty}, p_i^{offduty}, and q_i^{offduty} resulting from constraints (11.1) to (11.21).

The constraints on the minimum amount of break time on a day can be considered similarly to the constraints on the minimum amount of off-duty time on a day. Break time can be scheduled between x_i^{arrival} and $x_i^{\text{restbegin}}$ and between x_i^{restend} and x_i^{start}. We can demand that if $d_i^{\text{arrival}} < d_i^{\text{start}}$ then $x_i^{\text{restbegin}} \leq d_i^{\text{arrival}} \cdot t^{\text{day}}$ and $x_i^{\text{restend}} \geq d_i^{\text{arrival}} \cdot t^{\text{day}}$. Otherwise, we can move some of the break time scheduled before $x_i^{\text{restbegin}}$ to after x_i^{restend} or vice versa without violating any other constraint. Thus, we add the following constraints to the formulation.

$$x_i^{\text{restbegin}} \leq d_i^{\text{arrival}} \cdot t^{\text{day}} + (d_i^{\text{arrival}} + 1 - d_i^{\text{start}}) \cdot t^{\text{horizon}} \text{ for all } 1 \leq i \leq n \tag{12.1}$$

$$d_i^{\text{arrival}} \cdot t^{\text{day}} \leq x_i^{\text{restend}} + (d_i^{\text{arrival}} + 1 - d_i^{\text{start}}) \cdot t^{\text{horizon}} \text{ for all } 1 \leq i \leq n \tag{12.2}$$

In order to consider the constraints on the minimum amount of break time we introduce additional variables r_i^{break}, p_i^{break}, and q_i^{break}. r_i^{break} represents the accumulated amount of break time until the start of the work at location i. p_i^{break} and q_i^{break} represent the accumulated amount of break time until the end of the last day preceding d_i^{arrival} and d_i^{start}. We add the following constraints to the formulation.

$$r_1^{\text{break}} = x_1^{\text{start}} \tag{13.1}$$

$$r_{i-1}^{\text{break}} \leq r_i^{\text{break}} \leq r_{i-1}^{\text{break}} + x_i^{\text{restbegin}} - x_i^{\text{arrival}} + x_i^{\text{start}} - x_i^{\text{restend}} \text{ for all } 1 < i \leq n \tag{13.2}$$

$$r_i^{\text{break}} \leq r_{i-1}^{\text{break}} + (1 - z_i^{\text{idle}}) \cdot t^{\text{horizon}} \text{ for all } 1 < i \leq n \tag{13.3}$$

$$r_{i-1}^{\text{break}} + x_i^{\text{restbegin}} - x_i^{\text{arrival}} + x_i^{\text{start}} - x_i^{\text{restend}} \leq r_i^{\text{break}} + z_i^{\text{idle}} \cdot t^{\text{horizon}} \text{ for all } 1 < i \leq n \tag{13.4}$$

$$p_1^{\text{break}} = 0 \tag{13.5}$$

$$r_n^{\text{break}} + d_n^{\text{end}} \cdot t^{\text{day}} - x_n^{\text{end}} \geq q_n^{\text{break}} + t^{\text{break}} \tag{13.6}$$

Constraints (13.1) to (13.4) determine the value of r_i^{break} considering that off-duty periods of less than t^{idle} are not counted. Constraint (13.5) initialises p_1^{break} and constraint (13.6) demands that the accumulated amount of break time until the end of the schedule plus the remaining time until the end of the last day is at least q_n^{break} plus t^{break}.

If $d_i^{\text{arrival}} = d_i^{\text{start}}$ for some $1 \leq i \leq n$ then the value of q_i^{break} must have the same value as p_i^{break}. This is achieved by adding the following constraints to the formulation.

$$p_i^{\text{break}} \leq q_i^{\text{break}} + (d_i^{\text{start}} - d_i^{\text{arrival}}) \cdot t^{\text{horizon}} \text{ for all } 1 \leq i \leq n \tag{13.7}$$

$$q_i^{\text{break}} \leq p_i^{\text{break}} + (d_i^{\text{start}} - d_i^{\text{arrival}}) \cdot t^{\text{horizon}} \text{ for all } 1 \leq i \leq n \tag{13.8}$$

If $d_i^{\text{arrival}} < d_i^{\text{start}}$ for some $1 \leq i \leq n$ then the time between x_i^{arrival} and x_i^{start} includes midnight. If $z_i^{\text{idle}} = 0$ then the amount of break on day d_i^{arrival} is determined by subtracting the amount of break between x_i^{restend} and x_i^{start} from r_i^{break}. If $z_i^{\text{idle}} = 1$ then the amount of off-duty time on day d_i^{arrival} is r_i^{break}. We add the following constraints to the formulation.

$$q_i^{\text{break}} \leq r_i^{\text{break}} + (d_i^{\text{arrival}} + 1 - d_i^{\text{start}}) \cdot t^{\text{horizon}} \text{ for all } 1 \leq i \leq n \tag{13.9}$$

$$r_i^{\text{break}} - (x_i^{\text{start}} - x_i^{\text{restend}}) \leq q_i^{\text{break}} + (d_i^{\text{arrival}} + 1 - d_i^{\text{start}}) \cdot t^{\text{horizon}} \text{ for all } 1 \leq i \leq n \tag{13.10}$$

$$r_i^{\text{break}} \leq q_i^{\text{break}} + (1 - z_i^{\text{idle}}) \cdot t^{\text{horizon}} + (d_i^{\text{arrival}} + 1 - d_i^{\text{start}}) \cdot t^{\text{horizon}} \text{ for all } 1 \leq i \leq n \tag{13.11}$$

$$q_i^{\text{break}} \leq r_i^{\text{break}} - (x_i^{\text{start}} - x_i^{\text{restend}}) + z_i^{\text{idle}} \cdot t^{\text{horizon}} + (d_i^{\text{arrival}} + 1 - d_i^{\text{start}}) \cdot t^{\text{horizon}} \text{ for all } 1 \leq i \leq n \tag{13.12}$$

We need the make sure that the accumulated amount of break time on day d_i^{arrival} is at least t^{break}. This is achieved by adding the following constraints to the formulation.

$$p_i^{\text{break}} + t^{\text{break}} \leq q_i^{\text{break}} + (d_i^{\text{arrival}} + 1 - d_i^{\text{start}}) \cdot t^{\text{horizon}} \text{ for all } 1 \leq i \leq n \tag{13.13}$$

If $d_i^{\text{arrival}} = d_{i-1}^{\text{start}}$ for some $1 < i \leq n$ then the value of p_i^{break} must have the same value as q_{i-1}^{break}. This is achieved by adding the following constraints to the formulation.

$$p_i^{\text{break}} \leq q_{i-1}^{\text{break}} + (d_i^{\text{arrival}} - d_{i-1}^{\text{start}}) \cdot t^{\text{horizon}} \text{ for all } 1 < i \leq n \tag{13.14}$$

$$q_{i-1}^{\text{break}} \leq p_i^{\text{break}} + (d_i^{\text{arrival}} - d_{i-1}^{\text{start}}) \cdot t^{\text{horizon}} \text{ for all } 1 < i \leq n \tag{13.15}$$

If $d_{i-1}^{\text{start}} < d_i^{\text{arrival}}$ for some $1 < i \leq n$ then the time between x_{i-1}^{start} and x_i^{arrival} includes midnight. As no break time is taken in this period p_i^{break} must have the same value as r_{i-1}^{break} and we add the following constraints to the formulation.

$$p_i^{\text{break}} \leq r_{i-1}^{\text{break}} + (d_{i-1}^{\text{start}} + 1 - d_i^{\text{arrival}}) \cdot t^{\text{horizon}} \text{ for all } 1 < i \leq n \tag{13.16}$$

$$r_{i-1}^{\text{break}} \leq p_i^{\text{break}} + (d_{i-1}^{\text{start}} + 1 - d_i^{\text{arrival}}) \cdot t^{\text{horizon}} \text{ for all } 1 < i \leq n \tag{13.17}$$

We need the make sure that the accumulated amount of break time on day preceding d_i^{arrival} is at least t^{break}. This is achieved by adding the following constraints to the formulation.

$$q_{i-1}^{\text{break}} + t^{\text{break}} \leq p_i^{\text{break}} + (d_{i-1}^{\text{start}} + 1 - d_i^{\text{arrival}}) \cdot t^{\text{horizon}} \text{ for all } 1 < i \leq n \tag{13.18}$$

Eventually, we add the variable domains to the formulation.

$$p_i^{\text{break}} \in [0, t^{\text{horizon}}] \text{ for all } 1 \leq i \leq n \tag{13.19}$$

$$q_i^{\text{break}} \in [0, t^{\text{horizon}}] \text{ for all } 1 \leq i \leq n \tag{13.20}$$

$$r_i^{\text{break}} \in [0, t^{\text{horizon}}] \text{ for all } 1 \leq i \leq n \tag{13.21}$$

The example illustrated in Figure 1 shows the values of r_i^{break}, p_i^{break}, and q_i^{break} resulting from constraints (13.1) to (13.21). With (1) to (13) we have a mixed integer programming formulation of the Canadian minimum duration truck driver scheduling problem.

4 Dynamic Programming

The CAN-MDTDSP defined by (1) to (13) can be solved using any mixed integer programming solver. However, as Section 5 will reveal, the computational effort is extensive. Instead of solving the CAN-MDTDSP using a mixed integer solver we can using dynamic programming. Goel and Rousseau (2011) present a dynamic programming approach for truck driver scheduling in Canada. In the Canadian truck driver scheduling problem studied by Goel and Rousseau (2011) each location is associated a single time window and truck drivers may take rest periods everywhere and not only at customer locations and parking lots. The duration of schedules is not considered at all. Instead the goal is to determine whether a sequence of customer locations can be visited within their time windows without violating Canadian driver regulations. In fact the dynamic programming appproach by Goel and Rousseau (2011) imposes stricter constraints than the regulation. An additional constraint is imposed and exploited which demands that between two rest periods of 8 consecutive hours, the accumulated amount of on-duty time and off-duty time of less than 30 minutes does not exceed $t^{\text{day}} - t^{\text{offduty}}$. Furthermore, it is assumed that all parameters representing time values are a multiple of 15 minutes.

The dynamic programming approach is fairly complicated and a detailed description would go beyond the scope of this paper. The fundamental idea of the dynamic programming approach is to take an initially empty schedule and use different operators to transform this schedule into a schedule satisfying the constraints of the problem. Three types of operators are used. The first operator appends a driver activity with one of the following five types `DRIVE`, `WORK`, `REST`, `BREAK`, and `IDLE`. Activities of type `DRIVE` represent driving time, activities of type `WORK` represent time in which the driver is conducting stationary work, activities of type `REST` represent periods of at least 8 consecutive hours of off-duty time, activities of type `BREAK` represent other off-duty periods of at least 30 minutes duration, and activities of type `IDLE` represent off-duty periods of less than 30 minutes duration. The second and third operator increase the duration of the last period of type `REST` or `BREAK`. With these operators the dynamic programming approach generates a large set of alternative schedules in order to guarantee that a feasible schedule is found if one exists. To reduce the computational effort the approach uses dominance criteria to discard as many schedules as possible.

In order to be able to adopt this dynamic programming approach for the problem variant studied in this paper we need to be able to consider multiple time windows. The approach can be modified in a way that for each of the time window a different schedule is generated. To ensure that rest periods are only taken at customer locations or parking lots, the dynamic programming approach can be modified in such a way that if a driving period can not be fully scheduled due to rest requirements, all rest required is scheduled before the driving period. With these modifications we can determine whether a feasible solution exists for the problem variant studied in this paper. Note that we assume that rest periods may be taken immediately after completion of the work at a customer location. For hours of service regulations in the United States and the European Union, Goel (2011) shows in detail how the specific problem characteristics of this paper can be considered within a dynamic programming approach for truck driver scheduling.

As we will see, the schedules generated using the dynamic programming approach have a poor quality and the total duration of the schedule is significantly larger than necessary. However, we can also use this approach to minimise schedule durations. Each schedule generated by the approach has different characteristics which determine the sequence of operator moves that can be applied. A core feature of the approach is to discard schedules which are dominated, i.e. a schedule is discarded if another schedule is found which has better characteristics. Unfortunately, the schedule duration does not belong to the characteristics considered within the procedure for checking whether a schedule is dominated or not. Modifying the dominance check in such a way that the schedule duration is also considered, however, would significantly weaken the criteria for dominance and the modified approach would not be competitive in terms of computational effort. The completion time of the schedule, however, is one of the characteristics considered within the dominance check and the schedule with the smallest completion time is never discarded.

Suppose we have a schedule with minimal duration. Then the completion time of all other feasible schedules with the same start time is at least as high as for the schedule with minimal duration. If we modify the problem instance in such a way that the first work must not start before the begin of the schedule with minimal duration, then the dynamic programming approach generates this (or another) feasible schedule with minimal duration. Figure 2 illustrates an iterative dynamic programming approach for minimising schedule durations using the adaptation of the dynamic programming approach presented by Goel and Rousseau (2011). The iterative dynamic programming approach begins with solving the problems with the original parameters. If a feasible schedule is found the approach updates the best solution value found so far. After cutting of the first 15 minutes from the set of feasible start time at location 1, it continuous with solving the problem with the modified parameters. The method terminates if no feasible schedule is found or if the set of feasible start times is empty.

---

1. `best_duration` $= \infty$
2. while $T_1 \neq \emptyset$
   (a) solve problem using the dynamic programming approach
   (b) if no feasible schedule is found then stop
   (c) let `duration` denote the minimal duration of the feasible schedules found
   (d) `best_duration` $= \min\{$`best_duration`, `duration`$\}$
   (e) cut off first 15 minutes of T_1

Figure 2: The iterative dynamic programming approach

Note that the iterative dynamic programming approach does not depend on the regulation considered. Thus, it can be used to determine truck driver schedules with minimal duration whenever a dynamic programming approach is available which finds a feasible schedule with minimal completion time. If the set of start times to be enumerated is relatively small and the search for feasible schedules is fast, the iterative dynamic programming approach is a competitive alternative to solving the problem using the mixed integer program.

5 Computational Experiments

This section reports on computational experiments conducted on the benchmark sets presented by Goel (2011). All benchmark sets have a planning horizon starting on Monday 6.00 AM and ending on Friday 8.00 PM. In all benchmark sets one hour of work time shall be conducted at each work location in the tour and the driving time between two subsequent work locations is randomly set to a value between 1 and 10 hours. Assuming an average speed of 75 km/h, this implies that the distance between two subsequent locations ranges from 75 km to 750 km. Drivers may take rest periods before and after the work at any customer location. Furthermore, they may take rest periods at parking lots which were randomly distributed on the trip from one work location to another. The minimum driving time between parking lots is 15 minutes, the maximum driving time is 2 hours. In the first benchmark set all locations have a single time window starting at some day in the planning horizon at 6.00 AM and ending at 8.00 PM. In the second benchmark set all locations have two time windows: the first starts at some day in the planning horizon at 6.00 AM and ends at 12.00 PM and the second starts at 2.00 PM and ends at 8.00 PM. In the third and fourth benchmark set the time windows in the first two benchmark sets are repeated on two days.

			Dynamic Programming		Iterative Dynamic Programming		Mixed Integer Programming	
Time Windows	Instances	Feasible	Avg. CPU (in ms)	Avg. Duration (in min)	Avg. CPU (in ms)	Avg. Duration (in min)	Avg. CPU (in ms)	Avg. Duration (in min)
1 day: 6-20	781	499	2.69	6417	232.90	5547	11569.90	5542
1 day: 6-12, 14-20	781	493	105.69	6441	6146.76	5564	8168.29	5560
2 days: 6-20	781	647	3.24	6456	584.07	5288	45282.72	5284
2 days: 6-12, 14-20	781	645	223.61	6480	23805.81	5288	36642.62	5285

Table 2: Results

Table 2 shows the results of computational experiments conducted on a personal computer with an Intel 1.66 GHz CPU. Each benchmark set includes 1000 instances. 781 of these instances had an accumulated amount of working of not more than 70 hours and were used for the experiments. It can be seen that the duration of the schedules obtained using the iterative dynamic programming approach and mixed integer programming formulation is significantly smaller than the duration of schedules obtained by the dynamic programming approach. This shows that explicitly considering the objective of minimising the duration of schedules can bring advantages in terms of vehicle utilisation and labour costs. Exploiting this advantage comes at the cost of larger computational efforts. For the first and third benchmark set the iterative dynamic programming approach stays well below one second per instance and requires much smaller computation time than CPLEX 12 which was used for the solving the mixed integer program. For the second and fourth benchmark set the computational effort is still smaller compared to CPLEX 12, however, the difference is not as large. The lunch break included in the second and fourth benchmark set increases the search space explored by the dynamic programming approach significantly. The average duration using the iterative dynamic programming approach is slightly higher than the average duration obtained using the mixed integer programming formulation. The reason for the higher values is that an additional restriction is considered within the dynamic programming approach. The additional constraint which demands that between two rest periods of 8 consecutive hours, the accumulated amount of on-duty time and off-duty time of less than 30 minutes does not exceed $t^{\text{day}} - t^{\text{offduty}}$, appears to have very little impact on solution quality. The slight increase in average duration can likely be justified by the significantly smaller computation times.

6 Conclusions

This paper presents a mixed integer programming formulation for the Canadian minimum duration truck driver scheduling problem which is the problem of determining a schedule complying with Canadian hours of service regulations with minimal duration in which all work activities begin within one of multiple time windows. Computational experiments demonstrate that the duration of schedules can be significantly reduced compared to the duration of schedules obtained when simply searching for feasible schedules. An iterative dynamic programming approach is presented which - at the cost of a slight increase in schedule durations - requires significantly smaller computation time than using CPLEX 12 for solving the mixed integer program.

References

C. Archetti and M. W. P. Savelsbergh. The trip scheduling problem. *Transportation Science*, 43(4): 417–431, 2009. doi: 10.1287/trsc.1090.0278.

Canadian Council of Motor Transport Administrators. Commercial Vehicle Drivers Hours of Service Regulations - Application Guide. ISBN 0-921795-82-3, 2007.

A. Goel. Vehicle scheduling and routing with drivers' working hours. *Transportation Science*, 43(1): 17–26, 2009. doi: 10.1287/trsc.1070.0226.

A. Goel. Truck Driver Scheduling in the European Union. *Transportation Science*, 44(4):429–441, 2010. doi: 10.1287/trsc.1100.0330.

A. Goel. The Minimum Duration Truck Driver Scheduling Problem. *SSRN eLibrary - Working Paper Series*, 2011. URL http://ssrn.com/paper=1798569.

A. Goel and L. Kok. Truck Driver Scheduling in the United States. *SSRN eLibrary - Working Paper Series*, 2010. URL http://ssrn.com/paper=1675604.

A. Goel and L. M. Rousseau. Truck Driver Scheduling in Canada. Technical Report CIRRELT-2011-20, CIRRELT, 2011. URL https://www.cirrelt.ca/DocumentsTravail/CIRRELT-2011-20.pdf.

A. Goel, C. Archetti, and M. Savelsbergh. Truck Driver Scheduling in Australia. *Computers & Operations Research*, (to appear), 2011.

A. L. Kok, E. W. Hans, and J. M. J. Schutten. Optimizing departure times in vehicle routes. *European Journal of Operational Research*, 210(3):579 – 587, 2011. doi: 10.1016/j.ejor.2010.10.017.

L. Kok, C. M. Meyer, H. Kopfer, and J. M. J. Schutten. A Dynamic Programming Heuristic for the Vehicle Routing Problem with Time Windows and European Community Social Legislation. *Transportation Science*, 44(4):442–454, 2010. doi: 10.1287/trsc.1100.0331.

M. M. Mitler, J. C. Miller, J. J. Lipsitz, J. K. Walsh, and C. D. Wylie. The sleep of long-haul truck drivers. *New England Journal of Medicine*, 337(11):755–762, 1997. doi: 10.1056/NEJM199709113371106.

E. Prescott-Gagnon, G. Desaulniers, M. Drexl, and L. M. Rousseau. European driver rules in vehicle routing with time windows. *Transportation Science*, 44(4):455–473, 2010. doi: 10.1287/trsc.1100.0328.

M. E. Rancourt, J. F. Cordeau, and G. Laporte. Long-haul vehicle routing and scheduling with working hour rules. Technical Report CIRRELT-2010-46, CIRRELT, 2010. URL https://www.cirrelt.ca/DocumentsTravail/CIRRELT-2010-46.pdf.

Transport Canada. Commercial Vehicle Drivers Hours of Service Regulations. SOR/2005-313, 2005. URL http://laws.justice.gc.ca/en/SOR-2005-313/.

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