The Australian 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

January 11, 2012

Abstract

Transport companies seek to maximise vehicle utilisation and minimise labour costs. Both goals can be achieved if the time required to fulfil a sequence of transportation tasks is minimised. However, if schedule durations are too short drivers may not have enough time for recuperation and road safety is impaired. In Australia transport companies must ensure that truck drivers can comply with Australian Heavy Vehicle Driver Fatigue Law and schedules must give enough time for drivers to take the amount of rest required by the regulation. This paper shows how transport companies can minimise the duration of truck driver schedules complying with Australian Heavy Vehicle Driver Fatigue Law. A mixed integer programming formulation is presented and effective cuts are given which provide significant reduction in computational effort.

Keywords: Australian Heavy Vehicle Driver Fatigue Law, 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. A survey among Australian truck drivers (Williamson et al. (2001)) revealed that many drivers feel that fatigue is a substantial problem for the industry and feel that their companies should ease unreasonably tight schedules allowing more time for breaks and rests during their trips. Fatigue is felt as a contributing factor in every fifth accident and one out of three drivers reported breaking road rules on at least half of their trips. In their efforts to increase road safety the Australian transport ministers adopted new regulations for managing heavy vehicle driver fatigue. Under these new regulations, everyone in the supply chain, not just the driver, will have responsibilities to prevent driver fatigue and ensure drivers are able to comply with Australian Heavy Vehicle Driver Fatigue Law. If actions, inactions or demands of any person or entity cause or contribute to road safety breaches then that person or entity can be held legally accountable. Authorities can investigate along the supply chain and up and down the corporate chain of command. Consequently, road transport companies must 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 (2011) 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 (2011). 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 (2010a) 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. 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 in the United States and the European Union, 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 (2012b) presents a mixed integer programming formulation and a dynamic programming approach for minimum duration truck driver scheduling problems with multiple time windows. Among the regulations supported by the formulation presented by Goel (2012b) are the hours of service regulations in the United States and the European Union. For Canadian regulations, Goel (2012a) presents a mixed integer programming formulation and an iterative dynamic programming approach which can be used to minimise schedule durations.

Australian regulations enforce both daily rests and breaks, but the type of constraints imposed by the regulation differs significantly from the type of constraints imposed by the regulations in the United States, Canada, and the European Union. Thus, the above mentioned approaches cannot be used for Australian regulations. First heuristic approaches for determining whether a schedule complying with Australian regulations exists in which each customer must be visited within a given time window are presented by Goel (2010b). Goel et al. (2012) present the first exact method for this problem.

This paper presents a mixed integer programming formulation for a variant of the Australian 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. Effective cuts are given which help reducing the computational effort significantly. Computational experiments are presented which demonstrate that the duration of schedules is significantly smaller compared to the duration of schedules obtained when simply checking for feasibility.

The remainder of this paper is organised as follows. In Section 2 Australian Heavy Vehicle Driver Fatigue Law is described. Section 3 presents a mixed integer programming formulation of the Australian minimum duration truck driver scheduling problem. Section 4 discusses solution approaches for the problem and gives valid inequalities which strengthen the mixed integer programming formulation. In Section 5 computational experiments are presented demonstrating the effectiveness of these inequalities. Section 6 concludes the paper.

2 Australian Heavy Vehicle Driver Fatigue Law

In Australia new regulations for managing heavy vehicle driver fatigue entered into force on September 29, 2008. The new regulations comprise three different sets of rules. Operators accredited in the National Heavy Vehicle Accreditation Scheme may operate according to the Basic Fatigue Management Standard (see National Transport Commission (2008c)) or the Advanced Fatigue Management Standard (see National Transport Commission (2008b)). One condition for being accredited is that operators must plan schedules and rosters to ensure that they comply with the respective operating limits. Furthermore, operators must have a system for identifying non-compliance with the regulations. Without accreditation operators must comply with the Standard Hours option (see National Transport Commission (2008a)).

Standard Hours

The Standard Hours option of the Australian Heavy Vehicle Driver Fatigue Law imposes the following constraints on drivers' schedules:

1. In any period of $5\frac{1}{2}$ hours a driver must not work for more than $5\frac{1}{4}$ hours and must have at least 15 continuous minutes of rest time.

2. In any period of 8 hours a driver must not work for more than $7\frac{1}{2}$ hours and must have at least 30 minutes rest time in blocks of not less than 15 continuous minutes.

3. In any period of 11 hours a driver must not work for more than 10 hours and must have at least 60 minutes rest time in blocks of not less than 15 continuous minutes.

4. In any period of 24 hours a driver must not work for more than 12 hours and must have at least 7 continuous hours of stationary rest time.

5. In any period of 7 days a driver must not work for more than 72 hours and must have at least 24 continuous hours of stationary rest time.

6. In any period of 14 days a driver must not work for more than 144 hours and must have at least 4 night rest breaks (2 of which must be taken on consecutive days); the term *night rest break* refers to a rest break consisting of (a) 7 continuous hours of stationary rest time taken between 10.00 PM and 8.00 AM on the following day; or (b) 24 continuous hours of stationary rest time.

When calculating whether a truck driver schedule complies with these provisions the duration of each work period is rounded up to the nearest multiple of 15 minutes and the duration of each rest period is rounded down to the nearest multiple of 15 minutes.

Basic Fatigue Management

In order to use the Basic Fatigue Management (BFM) option of the Australian Heavy Vehicle Driver Fatigue Law an operator must be accredited in the National Heavy Vehicle Accreditation Scheme (NHVAS). The requirements for accreditation are that the operator complies with the six BFM standards described in National Transport Commission (2008c) comprising scheduling and rostering, fitness for duty, fatigue knowledge and awareness, responsibilities, internal review, and records and documentation. According to these standards the operator must plan schedules and rosters to ensure that they comply with the following constraints:

1. In any period of $6\frac{1}{4}$ hours a driver must not work for more than 6 hours and must have at least 15 continuous minutes of rest time.

2. In any period of 9 hours a driver must not work for more than $8\frac{1}{2}$ hours and must have at least 30 minutes rest time in blocks of not less than 15 continuous minutes.

3. In any period of 12 hours a driver must not work for more than 11 hours and must have at least 60 minutes rest time in blocks of not less than 15 continuous minutes.

4. In any period of 24 hours a driver must not work for more than 14 hours and must have at least 7 continuous hours of stationary rest time.

5. In any period of 7 days a driver must not accumulate more than 36 hours of long/night work time; the term *long/night work time* refers to any work time in excess of 12 hours in a 24 hour period plus any work time between midnight and 6.00 AM.

6. In any period of 14 days a driver must not work for more than 144 hours and must have at least 4 night rest breaks (2 of them must be taken on consecutive days and 2 of them must be stationary rest times of at least 24 hours); after accumulating 84 hours of work time a driver must have a stationary rest time of at least 24 hours

The duration of work and rest periods is rounded in the same way as in the Standard Hours options.

Advanced Fatigue Management

In order to use the Advanced Fatigue Management (AFM) option of the Australian Heavy Vehicle Driver Fatigue Law an operator must be NHVAS AFM accredited and comply with ten AFM standards. These standards are described in National Transport Commission (2008b) and comprise scheduling and rostering, operating limits, readiness for duty, health, management practises, workplace conditions, fatigue knowledge and awareness, responsibilities, records and documentation and internal review.

An operator using the Advanced Fatigue Management option must propose normal operating limits considering the maximum amount of work and the minimum amount of rest required within certain time frames. Planners must comply with these normal operating limits when generating schedules and rosters. In exceptional circumstances, i.e. in the case of unforeseen long delays, a driver is allowed to work between the normal operating limits and the outer limit specified in National Transport Commission (2008b). As normal operating limits are set on a case-by-case basis and as they depend on the individual circumstances, we will not consider the Advanced Fatigue Management option in the remainder of this paper.

Notation and remarks

Table 1 summarises the parameters imposed by the Standard Hours and BFM option for a planning horizon of one week. Note, that the condition of Provision 4, which requires that a rest of at least 7 hours is scheduled within each period of 24 hours, is equivalent to the condition that a new rest period of at least 7 hours begins at most 17 hours after the end of the last rest period of 7 hours or more. Furthermore, the condition concerning the maximum amount of work within a period of 24 hours implicitly defines the minimum amount of rest that must be accumulated within 24 hours.

Notation	Std.	BFM	Description
$t^{\text{rest-15m}}$	15	15	Minimum amount of rest required by Provision 1
$t^{\text{work-15m}}$	315	360	Maximum amount of work allowed without accumulating $t^{\text{rest-15m}}$ minutes of rest
$t^{\text{rest-30m}}$	30	30	Minimum amount of accumulated rest required by Provision 2
$t^{\text{work-30m}}$	450	510	Maximum amount of work allowed without accumulating $t^{\text{rest-30m}}$ minutes of rest
$t^{\text{rest-60m}}$	60	60	Minimum amount of accumulated rest required by Provision 3
$t^{\text{work-60m}}$	600	660	Maximum amount of work allowed without accumulating $t^{\text{rest-60m}}$ minutes of rest
$t^{\text{rest-7h}}$	420	420	Minimum amount of continuous rest required by Provision 4
$t^{\text{norest-7h}}$	1020	1020	Maximum amount of time without a rest period of duration $t^{\text{rest-7h}}$ or more
$t^{\text{rest-24h}}$	720	600	Minimum amount of accumulated rest required by Provision 4
$t^{\text{work-24h}}$	720	840	Maximum amount of work allowed without accumulating $t^{\text{rest-24h}}$ minutes of rest

Table 1: Parameters imposed by the regulation

Like Goel et al. (2012) we will assume in the remainder of this paper that drivers do not work on Saturdays and Sundays and only consider a planning horizon starting on Monday 4.00 AM or later and ending on Friday 11.59 PM or earlier. We furthermore assume that no more than 72 hours of work are assigned to a driver within the planning horizon. Under these assumptions a driver can execute the same schedule on a weekly basis and the requirements of Provision 5 of the standard rule and Provision 6 of the standard and BFM rule are satisfied. Note that a driver can only accumulate 10 long hours (work time in excess of 12 hours) and at most 26 night hours (work time between midnight and 6.00 AM) from Monday 4.00 AM to Friday 11.59 PM. Thus, Provision 5 of the BFM rule is also satisfied for the planning horizon considered.

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 $d_{i,i+1}$. The time horizon shall be denoted by t^{horizon}. We assume that all parameters representing time values are a multiple of 15 minutes and 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 (2012b) dummy 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 Australian minimum duration truck driver scheduling problem (AUS-MDTDSP) is the problem of determining a schedule complying with Australian regulations in which all work activities begin within one of the corresponding time windows and which has the minimal duration.

Let $C = \{$15m, 30m, 60m, 24h $\}$ denote the set of constraints concerning the amount of rest that must be accumulated within the time frames imposed by the regulation. For each constraint $c \in C$ let $t^{\text{work-}c}$ and $t^{\text{rest-}c}$ denote the maximum amount of work and the minimum amount of rest that must be accumulated within a time frame of $t^{\text{work-}c} + t^{\text{rest-}c}$.

We can now provide a mixed integer programming formulation of the AUS-MDTDSP. For each location $1 \le i \le n$ the formulation comprises variables $x_i := (x_i^{\text{arrival}}, x_i^{\text{start}}, x_i^{\text{end}})$ representing the arrival time, the start time, and the end time of the work at location i. For each location $1 \le i \le n$ the formulation comprises variables $y_i = (y_{i,\tau})_{1 \le \tau \le T_n}$ where $y_{i,\tau}$ is a binary variable indicating whether the τth time window of location i is used ($y_{i,\tau} = 1$) or not ($y_{i,\tau} = 0$). Furthermore, the formulation comprises for each location $1 \le i \le k \le n$ the variables $z_{i,k} = (z_{i,k}^c)_{c \in C}$ where $z_{i,k}^c$ is a binary variable indicating whether an amount of $t^{\text{rest-}c}$ or more rest time must be accumulated between start of work at location i and the end of the trip leaving from location k to the next location ($z_{i,k}^c = 1$) or not ($z_{i,k}^c = 0$). Eventually, the formulation comprises for each location $1 \le i \le n$ the binary variable r_i indicating whether the amount of off-duty time taken at location i can be regarded as a rest period of at least $t^{\text{rest-7h}}$ duration ($r_i = 1$) or not ($r_i = 0$).

The AUS-MDTDSP is

minimise

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

subject to

$$x_i^{\text{arrival}} \le x_i^{\text{start}} \text{ for all } 1 \le i \le n \tag{2}$$

$$x_i^{\text{start}} + w_i = x_i^{\text{end}} \text{ for all } 1 \le i \le n \tag{3}$$

$$x_i^{\text{end}} + d_{i,i+1} = x_{i+1}^{\text{arrival}} \text{ for all } 1 \le i < n \tag{4}$$

$$\sum_{\tau=1}^{\tau \le T_i} y_{i,\tau} = 1 \text{ for all } 1 \le i \le n \tag{5}$$

$$y_{i,\tau} t_{i,\tau}^{\text{min}} \le x_i^{\text{start}} \text{ for all } 1 \le i \le n, 1 \le \tau \le T_i \tag{6}$$

$$x_i^{\text{start}} \le t^{\text{horizon}} - y_{i,\tau}(t^{\text{horizon}} - t_{i,\tau}^{\text{max}}) \text{ for all } 1 \le i \le n, 1 \le \tau \le T_i \tag{7}$$

$$\sum_{j=i}^{j \le k} (w_j + d_{j,j+1}) \le t^{\text{work-}c} + t^{\text{horizon}} \cdot z_{i,k}^c \text{ for all } 1 \le i \le k \le n, c \in C \tag{8}$$

$$\sum_{j=i+1}^{j \le k} (x_j^{\text{start}} - x_j^{\text{arrival}}) + t^{\text{horizon}} \cdot (1 - z_{i,k}^c) \ge t^{\text{rest-}c} \text{ for all } 1 \le i \le k \le n, c \in C \tag{9}$$

$$x_i^{\text{start}} - x_i^{\text{arrival}} + t^{\text{horizon}} \cdot (1 - r_i) \geq t^{\text{rest-7h}} \text{ for all } 1 \leq i \leq n, \tag{10}$$

$$x_k^{\text{end}} - x_i^{\text{start}} \leq t^{\text{norest-7h}} + t^{\text{horizon}} \cdot \sum_{j=i+1}^{j \leq k} r_j \text{ for all } 1 \leq i \leq k \leq n \tag{11}$$

$$x_k^{\text{arrival}} - x_i^{\text{start}} \leq t^{\text{norest-7h}} + t^{\text{horizon}} \cdot \sum_{j=i+1}^{j < k} r_j \text{ for all } 1 \leq i < k \leq n \tag{12}$$

$$x_i \in [0, t^{\text{horizon}}]^3 \tag{13}$$

$$y_i \in \{0,1\}^{T_i}, \text{ for all } 1 \leq i \leq n \tag{14}$$

$$z_{i,k} \in \{0,1\}^{|C|} \text{ for all } 1 \leq i \leq k \leq n \tag{15}$$

$$r_i \in \{0,1\} \text{ for all } 1 \leq i \leq n \tag{16}$$

The objective function (1) is to minimise the duration between the start of the first work and the end of the last work. Constraint (2) demands that the driver arrives at a location before she or he may start to work at the location. Constraint (3) demands that the work at any location i ends w_i minutes after it starts. Constraint (4) demands that the arrival at a location equals the end time of the previous location plus the required driving time. Constraint (5) demands that at any location exactly one of the time windows is used. Constraints (6) and (7) are the time windows constraints. Constraint (8) demands that $z_{i,k}^c = 1$ if the accumulated amount of working and driving between the start of work at location i and the end of the trip leaving location k exceeds $t^{\text{work-}c}$. Constraint (9) demands that the accumulated amount of rest time between the start of work at location i and the end of the trip leaving location k is at least $t^{\text{rest-}c}$ if $z_{i,k}^c = 1$. Note that for the ease of notation we assume that $d_{n,n+1} = 0$. Constraint (10) requires that the amount of off-duty time at a location is at least $t^{\text{rest-7h}}$ if $r_i = 1$. Constraint (11) demands that the time elapsed between the start of work at a location i and end of work at a location $k \geq i$ may only exceed $t^{\text{norest-7h}}$ if $r_j = 1$ for some $i < j \leq k$. Constraint (12) demands that the time elapsed between the start of work at a location i and the arrival at a location $k > i$ may only exceed $t^{\text{norest-7h}}$ if $r_j = 1$ for some $i < j < k$. The variable domains are given by (13) to (16).

4 Solution Approaches

The AUS-MDTDSP defined by (1) to (16) can be solved using any mixed integer programming solver. However, as we will see the computational effort for solving the problem is unnecessarily high. With dynamic programming we can quickly find truck driver schedules complying with Australian regulations using the approach presented by Goel et al. (2012). This approach can be modified in a way that it can also be used for the problem variant studied in this paper. If a location has multiple time windows, the approach must be modified in a way that a different schedules is generated for each of the time windows. In order to minimise the computational effort, extensive usage of the dominance criteria presented by Goel et al. (2012) has to be made. In particular a partial schedule should be discarded if another partial schedule is found with a completion time which begins at least $t^{\text{rest-24h}}$ minutes earlier. To ensure that rest periods are only taken at customer locations or parking lots, the dynamic programming approach must 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. As the dynamic programming approach only focuses on determining whether a feasible solution exists or not, the duration of schedules found by this approach is unnecessarily high.

Goel (2012b) present a fast dynamic programming approach for minimising the duration of truck driver schedules in the United States and the European Union. Due to the different structure of Australian regulations, however, applying similar ideas to the approach by Goel et al. (2012) would significantly weaken the criteria for dominance and the modified approach would not be competitive in terms of computational effort.

Goel (2012a) presents an iterative dynamic programming approach for minimising the duration of truck driver schedules using a dynamic programming approach for determining whether a feasible schedule exists or not. The iterative dynamic programming approach is presented for Canadian regulations, but can also be used for Australian regulations. The approach begins with solving the problem with the original parameters using the dynamic programming approach for determining whether a feasible schedule exists or not. If a feasible schedule is found, the approach updates the best solution value found so far. Then, the iterative approach cuts of the first 15 minutes from the set of feasible start times at the first location and resolves the problem with the modified parameters. The method continues with the next iteration until all possible start times are enumerated. If in any iteration no feasible schedule is found, the iterative approach terminates with the solution of the minimum duration truck driver scheduling problem. The iterative dynamic programming approach is significantly faster than CPLEX 12 when solving the mixed integer programming formulation of the Canadian minimum duration truck driver scheduling problem. For Australian regulations, however, we will see that the iterative dynamic programming approach is not competitive.

The Australian minimum duration truck driver scheduling problem can be solved faster if the mixed integer programming formulation is strengthened. Let us consider two locations i and k with $1 \leq i < k \leq n$ and $w_k + d_{k,k+1} > 0$. We can determine a lower bound on the accumulated amount of rest to be taken between the ith and the kth work by

$$\begin{aligned}
l^{\text{rest}}_{i,k} := \quad & t^{\text{rest-24h}} \cdot \Big\lfloor \Big(\sum_{j=i}^{j<k} (w_j + d_{j,j+1}) \Big) / t^{\text{work-24h}} \Big\rfloor \\
+ \quad & t^{\text{rest-60m}} \cdot \Big\lfloor \Big(\sum_{j=i}^{j<k} (w_j + d_{j,j+1}) \bmod t^{\text{work-24h}} \Big) / t^{\text{work-60m}} \Big\rfloor \\
+ \quad & t^{\text{rest-30m}} \cdot \Big\lfloor \Big(\Big(\sum_{j=i}^{j<k} (w_j + d_{j,j+1}) \bmod t^{\text{work-24h}} \Big) \bmod t^{\text{work-60m}} \Big) / t^{\text{work-30m}} \Big\rfloor \\
+ \quad & t^{\text{rest-15m}} \cdot \Big\lfloor \Big(\Big(\Big(\sum_{j=i}^{j<k} (w_j + d_{j,j+1}) \bmod t^{\text{work-24h}} \Big) \bmod t^{\text{work-60m}} \Big) \bmod t^{\text{work-30m}} \Big) / t^{\text{work-15m}} \Big\rfloor
\end{aligned}$$

The lower bound $l^{\text{rest}}_{i,k}$ includes the amount of rest required by all constraints $c \in C$. We can add the following inequality to strengthen the problem formulation.

$$x^{\text{start}}_k \geq x^{\text{start}}_i + \sum_{j=i}^{j<k} (w_j + d_{j,j+1}) + l^{\text{rest}}_{i,k} \text{ for all } 1 \leq i < k \leq n \text{ with } w_k + d_{k,k+1} > 0 \qquad (17)$$

The inequality states that for any $1 \leq i < k \leq n$ with $w_k + d_{k,k+1} > 0$ the difference between the start time of the kth and the ith work must be large enough to give enough time for the work plus the lower bound on the amount of rest. As we will see in the next section, the computational effort for solving the problem defined by (1) to (17) using CPLEX 12 is significantly smaller than the effort required by the iterative dynamic programming approach.

5 Computational Experiments

This section reports on computational experiments conducted on the benchmark sets presented by Goel (2012b). 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.

The problems defined by constraints (1) to (16) and by constraints (1) to (17) are compared with results obtained with an adaptation of the dynamic programming approach presented by Goel et al. (2012) and the iterative dynamic programming approach presented by Goel (2012a).

				DP (feasibility check)		Iterative DP	MIP (1)-(16)	MIP (1)-(17)	
Rules	Time Windows	Instances	Feasible	Avg. CPU (in ms)	Avg. Duration (in min)	Avg. CPU (in ms)	Avg. CPU (in ms)	Avg. CPU (in ms)	Avg. Duration (in min)
Std.	1 day: 6-20	816	262	29.43	6081	1613.94	2161.31	585.41	5699
Std.	1 day: 6-12, 14-20	816	262	82.64	6095	2180.80	2052.62	611.90	5702
Std.	2 days: 6-20	816	350	52.75	6138	3014.69	4273.02	711.92	5696
Std.	2 days: 6-12, 14-20	816	350	154.17	6154	4195.78	4208.75	760.23	5697
BFM	1 day: 6-20	816	499	12.81	6112	1080.15	2266.71	781.44	5620
BFM	1 day: 6-12, 14-20	816	485	79.03	6128	2458.65	2178.95	823.93	5629
BFM	2 days: 6-20	816	647	18.24	6159	2376.74	4509.91	1145.90	5429
BFM	2 days: 6-12, 14-20	816	639	123.11	6173	5868.14	4493.15	1256.30	5423

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. 816 of these instances had an accumulated amount of working of not more than 72 hours and were used for the experiments. It can be seen that the minimal schedule duration is significantly smaller than the duration of schedules obtained by the dynamic approach (which only seeks to determine whether a feasible schedule exists or not). 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. While the iterative dynamic programming approach appears to be faster on average compared with the CPU time required by CPLEX 12 for solving the AUS-MDTDSP defined by constraints (1) to (16), it is not faster for all sets of instances. The computational effort required CPLEX 12 for solving the AUS-MDTDSP defined by constraints (1) to (17) is significantly smaller for all sets of instances. This shows that inequality (17) is particularly useful. Furthermore, with inequality (17) the computation time required does not grow significantly with an increased number of time windows.

6 Conclusions

This paper presents a mixed integer programming formulation for the Australian minimum duration truck driver scheduling problem (AUS-MDTDSP). The AUS-MDTDSP is the problem of determining a schedule complying with Australian regulations with minimal duration in which all work activities begin within one of multiple time windows. Valid inequalities are presented which strengthen the formulation and significantly reduce the computational effort when solving the problem. 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. Furthermore, the experiments also show that using the additional inequalities helps in reducing and stabilising the computational effort required.

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.

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, 2010a. doi: 10.1287/trsc.1100.0330.

A. Goel. Truck Driver Scheduling and Australian Heavy Vehicle Driver Fatigue Law. In B. McCollum, E. Burke, and G. White, editors, *Proceedings of the 8th International Conference on the Practice and Theory of Automated Timetabling (PATAT2010)*, pages 201–210, 2010b. ISBN 08-538-9973-3.

A. Goel. The Canadian Minimum Duration Truck Driver Scheduling Problem. *SSRN eLibrary - Working Paper Series*, 2012a. URL `http://ssrn.com/paper=1852201`.

A. Goel. The Minimum Duration Truck Driver Scheduling Problem. *SSRN eLibrary - Working Paper Series*, 2012b. URL `http://ssrn.com/paper=1798569`.

A. Goel and L. Kok. Truck Driver Scheduling in the United States. *Transportation Science*, (to appear), 2011. doi: 10.1287/trsc.1110.0382.

A. Goel, C. Archetti, and M. Savelsbergh. Truck Driver Scheduling in Australia. *Computers & Operations Research*, 39(5):1122–113, 2012.

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.

National Transport Commission. Heavy Vehicle Driver Fatique Reform – Standard Hours explained. Information Bulletin, 2008a.

National Transport Commission. Heavy Vehicle Driver Fatique Reform – Advanced Fatigue Management explained. Information Bulletin, 2008b.

National Transport Commission. Heavy Vehicle Driver Fatique Reform – Basic Fatigue Management explained. Information Bulletin, 2008c.

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`.

A. Williamson, S. Sadural, A. Feyer, and R. Friswell. Driver Fatigue: A Survey of Long Distance Heavy Vehicle Drivers in Australia. Information Paper CR 198, National Road Transport Commission, 2001.

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