Computers & Operations Research 39 (2012) 1122–1132

Contents lists available at ScienceDirect

Computers & Operations Research

journal homepage: www.elsevier.com/locate/caor

Truck driver scheduling in Australia

Asvin Goel [a,b,*], Claudia Archetti [c], Martin Savelsbergh [d]

[a] MIT-Zaragoza International Logistics Program, Zaragoza Logistics Center, Zaragoza, Spain
[b] Applied Telematics/e-Business Group, Department of Computer Science, University of Leipzig, Leipzig, Germany
[c] Department of Quantitative Methods, University of Brescia, 25122 Brescia, Italy
[d] CSIRO Mathematics, Informatics, and Statistics, North Ryde, NSW 1670, Australia

ARTICLE INFO

Available online 31 May 2011

Keywords:
Vehicle scheduling
Hours of service regulations

ABSTRACT

In September 2008 new regulations for managing heavy vehicle driver fatigue entered into force in Australia. According to the new regulations there is a chain of responsibility ranging from drivers to dispatchers and shippers and thus, carriers must explicitly consider driving and working hour regulations when generating truck driver schedules. This paper presents and studies the Australian Truck Driver Scheduling Problem (AUS-TDSP) which is the problem of determining whether a sequence of locations can be visited within given time windows in such a way that driving and working activities of truck drivers comply with Australian Heavy Vehicle Driver Fatigue Law.

1. Introduction

According to a survey of truck drivers in Australia, fatigue is felt as a contributing factor in every fifth accident [1]. One out of five drivers reported at least one fatigue related incident on their last trip and one out of three drivers reported breaking road rules on at least half of their trips. Many drivers feel that fatigue is a substantial problem for the industry and feel that their companies should ease unreasonably tight schedules and should allow more time for breaks and rests during 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 the legal work/rest hours. 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 now ensure that truck driver schedules comply with Australian Heavy Vehicle Driver Fatigue Law. An important key in managing fatigue is to explicitly consider driving and working hour regulations when generating truck driver schedules. Planning problems considering driving and working hours of truck drivers, however, have so far attracted little interest in the vehicle routing and scheduling literature and to the best of the authors' knowledge there are currently no planning tools available that allow for truck driver scheduling considering Australian Heavy Vehicle Driver Fatigue Law.

Driver scheduling in road freight transportation differs significantly from airline crew scheduling and driver scheduling in rail transport or mass transit systems, which are covered by a comprehensive annotated bibliography by Ernst et al. [2]. While rail transport or mass transit systems operate on time tables and arrival times are fixed, arrival times in road freight transportation are typically not fixed and can be scheduled with some degree of freedom. Furthermore, in road freight transportation it is usually easy to interrupt transportation services in order to take compulsory breaks and rest periods.

The central tenet of driver regulations worldwide is to limit the number of working and driving hours without rest (work encompasses more than just driving and also includes inspecting the truck, waiting at a facility, assisting with loading and unloading, etc.).

The first research known to the authors explicitly considering driver regulations in freight transportation routing is the work by Xu et al. [3] 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. [3] 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 [4] show that if weekly rest periods do not need to be considered and each location to be visited has a single time window, schedules complying with U.S. Hours of Service regulations can be determined in polynomial time. Goel and Kok [5] show that schedules complying with U.S. Hours of Service regulations can

[*] Corresponding author at: MIT-Zaragoza International Logistics Program, Zaragoza Logistics Center, Zaragoza, Spain.

E-mail addresses: agoel@zlc.edu.es (A. Goel), archetti@eco.unibs.it (C. Archetti), Martin.Savelsbergh@csiro.au (M. Savelsbergh).

0305-0548/$ - see front matter © 2011 Elsevier Ltd. All rights reserved.
doi:10.1016/j.cor.2011.05.021

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 h. 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. [6] 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 [5]. Heuristics for combined vehicle routing and truck driver scheduling in Europe are presented by Goel [7], Kok et al. [8], and Prescott-Gagnon et al. [9]. The work by Goel [10] presents the first method capable of finding a feasible schedule complying with European regulations if such a schedule exists.

The core "daily" limits of the U.S. and European driver regulations are of the form: (1) a daily rest has to be at least X hours (10 h in the U.S. and 11 h in Europe), (2) a driver cannot accumulate more than X hours of driving between two consecutive daily rests (11 h in the U.S. and 9 h in Europe), and (3) the next daily rest has to commence at most X hours after the last daily rest (14 h in the U.S. and 13 h in Europe). This is supplemented in Europe by the introduction of mandatory breaks, or short rests, by requiring that after driving for no more than X hours a break of at least Y minutes has to occur (a 45-min break after 4.5 h of driving).

The Australian driver regulations enforce both daily rests and breaks, but do so in a different way. Most significantly, instead of requiring the next daily rest to commence X hours after the last daily rest (which, of course, also limits the working hours between consecutive rests), the Australian regulation requires that in *any* period of 24 h a driver must not work for more than X hours and must have at least Y hours of rest (at most 12 h of work and at least 7 h of rest in the Standard Hours option and at most 14 h of work and at least 7 h of rest in the Basic Fatigue Management option). This "sliding window" of 24 h in the Australian driver regulations makes determining whether a feasible driver schedule exists for visiting a given sequence of locations, each within given time window, far more difficult than for the U.S. and European driver regulations. In a sense, the Australian driver regulations introduces a "history dependence" that is not present in the U.S. and European driver regulations.

In this paper, we develop an exact method for identifying truck driver schedules complying with Australian driver regulations and derive, from the exact method, extremely effective heuristics requiring only a fraction of time of the exact method. The remainder of the paper is organized as follows. Section 2 describes the Australian Heavy Vehicle Driver Fatigue Law. Section 3 presents the Australian Truck Driver Scheduling Problem (AUS-TDSP). In Section 4 some structural properties of the AUS-TDSP are given and solution approaches are presented in Section 5. Computational experiments are reported in Section 6.

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 [11]) or the Advanced Fatigue Management Standard (see [12]). 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 [13]).

2.1. 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 h a driver must not work for more than 5 h and must have at least 15 continuous minutes of rest time.
2. In any period of 8 h a driver must not work for more than 7 h and must have at least 30 min rest time in blocks of not less than 15 continuous minutes.
3. In any period of 11 h a driver must not work for more than 10 h and must have at least 60 min rest time in blocks of not less than 15 continuous minutes.
4. In any period of 24 h a driver must not work for more than 12 h 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 h 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 h and must have at least four night rest breaks (two 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 min and the duration of each rest period is rounded down to the nearest multiple of 15 min.

2.2. 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 [11] 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 h a driver must not work for more than 6 h and must have at least 15 continuous minutes of rest time.
2. In any period of 9 h a driver must not work for more than 8 h and must have at least 30 min rest time in blocks of not less than 15 continuous minutes.
3. In any period of 12 h a driver must not work for more than 11 h and must have at least 60 min rest time in blocks of not less than 15 continuous minutes.
4. In any period of 24 h a driver must not work for more than 14 h 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 h of long/night work time; the term *long/night work time* refers to any work time in excess of 12 h in a 24 h 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 h and must have at least four night rest breaks (two of them must be taken on consecutive days and two of them must be stationary rest times of at least 24 h); after accumulating 84 h of work time a driver must have a stationary rest time of at least 24 h.

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

2.3. 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 [12] 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 [12]. 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.

2.4. Discussion

Provisions 1–4 of the regulations share the same structure for the standard and the BFM rules. However, the maximum amount that may be worked in the BFM option is considerably higher and up to 2 h more per day may be worked by a driver working according to the BFM option. Fig. 1 illustrates different feasible schedules complying with both regulations. In all schedules the driver returns from a night rest taken on two consecutive days. The only difference between the schedules is that some of the rest time is taken at an earlier or later point in time. In the first schedule the driver may continue to work for another 3 h and 45 min according to the standard rules until the maximum amount of work within a 24 h period is reached. According to BFM rules the driver may continue to work for another 5 h until 6 h of work are accumulated without a rest. In the second schedule the driver may continue to work for another 1 h and 45 min according to the standard rules until 10 h of driving are accumulated and a rest period of at least 15 min is required. According to the BFM rules, the driver may continue to work for another 2 h and 45 min until 11 h of driving are accumulated. After 2 h and 45 min of work a rest period of at least 15 min is compulsory. The remaining working time in the third schedule is the same as for the second schedule. However, after working for another 1 h and 45 min according to standard rules or 2 h and 45 min according to BFM rules, a rest period of 30 min is required before the driver may continue to work again. The next rest period of 7 h duration or more must begin no later than 7 h and 45 min after the end of the first schedule, 8 h after the end of the second schedule, and 8 h and 15 min after the end of the third schedule.

Provision 5 of the standard rules constrains the maximum amount of work within a 7 day period. Provision 5 of the BFM rules constrains the amount of work in excess of 12 h within a 24 h period and the amount of work conducted between midnight and 6.00 AM. Provisions 6 of both rules constrain the maximum amount of work within 14 days. According to standard rules a driver may not work for more than 72 h within any period of 7 days. According to BFM rules a driver may work for more than 72 h within a period of 7 days, but if the driver does so the amount of work in the next week must be smaller.

In the remainder of this paper we will assume that drivers do not work on Saturdays and Sundays and only consider a planning horizon starting on Monday 4.00 AM and ending on Friday 11.59 PM. We furthermore assume that no more than 72 h 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 h) 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. Table 1 summarizes the parameters imposed

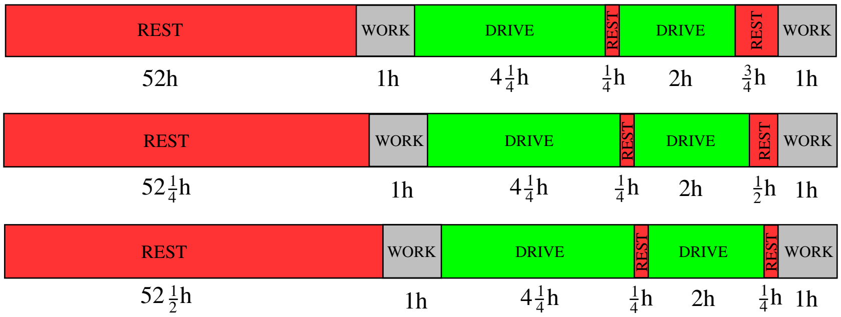

Fig. 1. Alternative schedules complying with the regulation.

Table 1
Parameters imposed by the regulation.

Notation	Standard	BFM	Description
t^{rest15}	15	15	Minimum amount of rest required by Provision 1
t^{work15}	315	360	Maximum amount of work allowed without accumulating t^{rest15} minutes of rest
t^{rest30}	30	30	Minimum amount of accumulated rest required by Provision 2
t^{work30}	450	510	Maximum amount of work allowed without accumulating t^{rest30} minutes of rest
t^{rest60}	60	60	Minimum amount of accumulated rest required by Provision 3
t^{work60}	600	660	Maximum amount of work allowed without accumulating t^{rest60} minutes of rest
t^{rest7h}	420	420	Minimum amount of continuous rest required by Provision 4
$t^{norest7h}$	1020	1020	Maximum amount of time without a rest period of duration t^{rest7h} or more
$t^{rest24h}$	720	600	Minimum amount of accumulated rest required by Provision 4
$t^{work24h}$	720	840	Maximum amount of work allowed without accumulating $t^{rest24h}$ minutes of rest

by the regulation which are relevant under the assumptions made. Note that the condition of Provision 4, which requires that a rest of at least 7 h is scheduled within each period of 24 h, is equivalent to the condition that a new rest period of at least 7 h begins at most 17 h after the end of the last rest period of 7 h or more. Furthermore, the condition concerning the maximum amount of work within a period of 24 h implicitly defines the minimum amount of rest that must be accumulated within 24 h.

3. The truck driver scheduling problem

This section describes the Australian Truck Driver Scheduling Problem for a planning horizon starting on Monday 4.00 AM and ending on Friday 11.59 PM of the same week. Let us consider a sequence of locations denoted by $n_1,n_2,\ldots,n_\lambda$ which shall be visited by a truck driver. At each location n_μ some stationary work of duration w_μ shall be conducted. This work must be a continuous period which shall begin within a time window denoted by $[t_\mu^{\min},t_\mu^{\max}]$. We assume that n_1 corresponds to the driver's current location and that the driver completes his or her work week after finishing work at location n_λ. The work to be conducted at locations n_1 and n_λ can include loading and unloading activities as well as time for getting ready or cleaning the vehicle. Note that the required duration at the locations may also be set to zero. The (positive) driving time required for moving from node n_μ to node $n_{\mu+1}$ shall be denoted by $\delta_{\mu,\mu+1}$. Let us assume that all values representing driving times, working times, and time windows are a multiple of 15 min.

In order to give a formal model of the problem, let us denote with $\texttt{DRIVE}$ any period during which the driver is driving, with $\texttt{WORK}$ any period of working time in which the driver is not driving (e.g., time in which the driver is loading or unloading the vehicle), with $\texttt{REST}$ any period in which the driver is neither working nor driving. A truck driver schedule can be specified by a sequence of activities to be performed by the drivers. Let $\mathcal{A} := \{a=(a^{\text{type}},a^{\text{length}}) | a^{\text{type}} \in \{\texttt{DRIVE},\texttt{WORK},\texttt{REST}\}, a^{\text{length}} > 0\}$ denote the set of driver activities to be scheduled. Let $\ll . \gg$ be an operator that concatenates different activities. Thus, $a_1 . a_2 . \ldots . a_k$ denotes a schedule in which for each $i \in \{1,2,\ldots,k-1\}$ activity a_{i+1} is performed immediately after activity a_i. During concatenation the operator merges consecutive driver activities of the same type. That is, for a given schedule $s := a_1 . a_2 . \ldots . a_k$ and an activity a with $a_k^{\text{type}} = a^{\text{type}}$ we have $s . a = a_1 . \ldots . a_{k-1} . (a_k^{\text{type}}, a_k^{\text{length}} + a^{\text{length}})$. For a given schedule $s := a_1 . a_2 . \ldots . a_k$ and $1 \le i \le k$ let $s_{1,i} := a_1 . a_2 . \ldots . a_i$ denote the partial schedule composed of activities a_1 to a_i. For simplicity, we will only consider schedules which begin with a rest period including the time from Saturday 0.00 AM to Monday 4.00 AM. That is, we only consider schedules $s := a_1 . a_2 . \ldots . a_k$ with $a_1^{\text{type}} = \texttt{REST}$ and $a_1^{\text{length}} \ge 3120$.

We use the following notation for determining whether a schedule complies with the regulation. For each schedule $s := a_1 . a_2 . \ldots . a_k$ we denote the completion time by

$$l_s^{\text{end}} := \sum_{1 \le i \le k} a_i^{\text{length}}.$$

Let us denote with i_s^{15}, i_s^{30}, and i_s^{60} the index of the last rest activity contributing to a cumulative rest time of at least t^{rest15}, t^{rest30}, and t^{rest60} minutes until the end of the schedule. Let us furthermore denote with $i_s^{7\text{h}}$ the index of the last rest activity of at least 7 h continuous rest and with $i_s^{24\text{h}}$ the index of the last rest activity contributing to a cumulative rest time of at least t^{rest24h} minutes until the end of the schedule. For the first schedule in Fig. 1 we have $i_s^{15} = i_s^{30} = 6$ because the sixth activity is of type $\texttt{REST}$ and has a duration of more than 30 min. Furthermore, we have $i_s^{60} = 4$, because the fourth activity is of type $\texttt{REST}$ and the accumulated duration of activities four and six is 60 min. Finally, we have $i_s^{7\text{h}} = i_s^{24\text{h}} = 1$ because the first activity is the last activity of at least 7 h duration, and because the first activity is the last activity contributing to a cumulative rest time of at least t^{rest24h} minutes until the end of the schedule. That is, we have $(i_s^{15},i_s^{30},i_s^{60},i_s^{7\text{h}},i_s^{24\text{h}}) = (6,6,4,1,1)$ for the first schedule in Fig. 1. Similarly, we have $(i_s^{15},i_s^{30},i_s^{60},i_s^{7\text{h}},i_s^{24\text{h}}) = (6,6,1,1,1)$ for the second schedule, and $(i_s^{15},i_s^{30},i_s^{60},i_s^{7\text{h}},i_s^{24\text{h}}) = (6,4,1,1,1)$ for the third schedule.

Let us denote with Δ_s^{15}, Δ_s^{30}, and Δ_s^{60} the maximum amount of work that can be appended to schedule s with respect to Provisions 1–3. Let us furthermore denote with $\Delta_s^{7\text{h}}$ the maximum remaining time until which a rest activity of at least 7 h must be scheduled and with $\Delta_s^{24\text{h}}$ the maximum amount of work that can be appended to schedule s without exceeding the maximum amount of work within 24 h. These values can be computed by

$$\Delta_s^{15} := t^{\text{work15}} - \sum_{\substack{i_s^{15} < j \le k \\ a_j^{\text{type}} \in \{\texttt{DRIVE},\texttt{WORK}\}}} a_j^{\text{length}}$$

$$\Delta_s^{30} := t^{\text{work30}} - \sum_{\substack{i_s^{30} < j \le k \\ a_j^{\text{type}} \in \{\texttt{DRIVE},\texttt{WORK}\}}} a_j^{\text{length}}$$

$$\Delta_s^{60} := t^{\text{work60}} - \sum_{\substack{i_s^{60} < j \le k \\ a_j^{\text{type}} \in \{\texttt{DRIVE},\texttt{WORK}\}}} a_j^{\text{length}}$$

$$\Delta_s^{7\text{h}} := t^{\text{norest7h}} - \sum_{i_s^{7\text{h}} < j \le k} a_j^{\text{length}}$$

$$\Delta_s^{24\text{h}} := t^{\text{work24h}} - \sum_{\substack{i_s^{24\text{h}} < j \le k \\ a_j^{\text{type}} \in \{\texttt{DRIVE},\texttt{WORK}\}}} a_j^{\text{length}}.$$

Considering the standard rules we have $(\Delta_s^{15},\Delta_s^{30},\Delta_s^{60},\Delta_s^{7\text{h}},\Delta_s^{24\text{h}}) = (255,390,420,465,225)$ for the first schedule in Fig. 1, $(\Delta_s^{15},\Delta_s^{30},\Delta_s^{60},\Delta_s^{7\text{h}},\Delta_s^{24\text{h}}) = (255,390,105,480,225)$ for the second schedule, and $(\Delta_s^{15},\Delta_s^{30},\Delta_s^{60},\Delta_s^{7\text{h}},\Delta_s^{24\text{h}}) = (255,270,105,495,225)$ for the third schedule. Considering the BFM rules we have $(\Delta_s^{15},\Delta_s^{30},\Delta_s^{60},\Delta_s^{7\text{h}},\Delta_s^{24\text{h}}) = (300,450,480,465,345)$ for the first schedule in Fig. 1, $(\Delta_s^{15},\Delta_s^{30},\Delta_s^{60},\Delta_s^{7\text{h}},\Delta_s^{24\text{h}}) = (300,450,165,480,345)$ for the second schedule, and $(\Delta_s^{15},\Delta_s^{30},\Delta_s^{60},\Delta_s^{7\text{h}},\Delta_s^{24\text{h}}) = (300,330,165,495,345)$ for the third schedule.

Suppose we have a schedule $s = a_1 . \ldots . a_k$ with $a_1^{\text{type}} = \texttt{REST}$ which complies with the regulation and an activity a. Then, schedule $s.a$ complies with the regulation if and only if $a^{\text{type}} = \texttt{REST}$ or

$$a^{\text{length}} \le \min\{\Delta_s^{15},\Delta_s^{30},\Delta_s^{60},\Delta_s^{7\text{h}},\Delta_s^{24\text{h}}\}.$$

Let us consider the schedule s which is obtained by appending 1 h and 45 min of working time to the third schedule in Fig. 1. According to standard rules we have $\Delta_s^{60} = 0$. Thus, some additional rest is required before adding additional work activities. After adding 30 min of rest we may again append further working activities, because we have $i_{s'}^{60} = 4$ and $\Delta_{s'}^{60} = t^{\text{work60}} - (120+60+105)$ for $s' := s . (\texttt{REST},30)$.

Let us now suppose we have a schedule s with $\Delta_s^{7\text{h}} = 0$, then a rest period of duration t^{rest7h} or more must be scheduled before further working activities can be appended to the schedule. By scheduling a rest period of duration t^{rest7h}, we reset Δ_s^{15}, Δ_s^{30}, Δ_s^{60}, and $\Delta_s^{7\text{h}}$, but not necessarily $\Delta_s^{24\text{h}}$. Suppose we have a schedule s with $\Delta_s^{24\text{h}} = 0$, then a rest period must be scheduled before further working activities can be appended to the schedule. The minimum duration of this rest period can be as small as 15 min or larger than t^{rest7h}, depending on the accumulated rest scheduled after activity $i_s^{24\text{h}}$.

Let us now give a formal model of the problem. For a given sequence of locations $n_1,n_2,\ldots,n_\lambda$ and a schedule $s = a_1 . a_2 . \ldots . a_k$ with $a_1^{\text{type}} = \texttt{REST}$, let us denote with $i(\mu)$ the index corresponding to the μ th stationary work period, i.e. $a_{i(\mu)}$ corresponds to the work performed at location n_μ. The Australian Truck Driver Scheduling Problem (AUS-TDSP) is the problem of determining whether a schedule $s := a_1 . a_2 . \ldots . a_k$ with $a_1^{\text{type}} = \texttt{REST}$ and $a_1^{\text{length}} \geq 3120$ exists which satisfies

$$\sum_{\substack{1 \leq j \leq k \\ a_j^{\text{type}} = \text{WORK}}} 1 = \lambda$$

$$a_{i(\mu)}^{\text{length}} = w_\mu \quad \text{for each } \mu \in \{1,2,\ldots,\lambda\}$$

$$t_\mu^{\min} \leq l_{s_{1,i(\mu)-1}}^{\text{end}} \leq t_\mu^{\max} \quad \text{for each } \mu \in \{1,2,\ldots,\lambda\}$$

$$\sum_{\substack{i(\mu) \leq j \leq i(\mu+1) \\ a_j^{\text{type}} = \text{DRIVE}}} a_j^{\text{length}} = \delta_{\mu,\mu+1} \quad \text{for each } \mu \in \{1,2,\ldots,\lambda-1\}$$

$$a_i^{\text{length}} \leq \min\{\Delta^{15}_{s_{1,i-1}},\Delta^{30}_{s_{1,i-1}},\Delta^{60}_{s_{1,i-1}},\Delta^{7\text{h}}_{s_{1,i-1}},\Delta^{24\text{h}}_{s_{1,i-1}}\}$$

for each $i \in \{1,\ldots,k\}$ with $a_i^{\text{type}} \in \{\texttt{DRIVE},\texttt{WORK}\}$.

Condition (1) demands that the number of work activities in the schedule is λ. Condition (2) demands that the duration of the μth work activity matches the specified work duration at location n_μ. Condition (3) demands that the start time of the μth work activity, i.e., the completion time $l^{\text{end}}_{s_{1,i(\mu)-1}}$ of the partial schedule $s_{1,i(\mu)-1}$ is in the time window $[t_\mu^{\min},t_\mu^{\max}]$. Condition (4) demands that the accumulated driving time between two work activities matches the driving time required to move from one location to the other. Condition (5) demands that the schedule complies with the regulation. In the remainder of this paper, we will say that a schedule $s := a_1 . a_2 . \ldots . a_k$ with $a_1^{\text{type}} = \texttt{REST}$ is *feasible* if and only if it satisfies conditions (1)–(5).

4. Structural properties

Obviously, there may be many different feasible schedules for an instance of the truck driver scheduling problem. In order to efficiently solve the truck driver scheduling problem, we restrict the search space to a smaller set of truck driver schedules that is still sufficient to solve the AUS-TDSP. Therefore, we next provide some properties of the truck driver scheduling problem which help us solving the AUS-TDSP without exploring unnecessarily many partial schedules. These properties tell us that it is sufficient to search for schedules in which any driving and working activity is scheduled as early as possible. Furthermore, they tell us under which conditions rest time can be as short as possible.

Property 1. *Let $s := a_1 . \ldots . a_k$ be a feasible schedule with $a_i^{type} = \texttt{REST}$ and $a_{i+1}^{type} \in \{\texttt{DRIVE},\texttt{WORK}\}$ for some $1 < i < k$. If the partial schedule*

$a_1 . \ldots . a_{i-1} . a_{i+1}$

complies with the regulation and all relevant time window constraints, then

$a_1 . \ldots . a_{i-1} . a_{i+1} . a_i . a_{i+2} . \ldots . a_k$

is a feasible schedule.

Proof. Let $s' := a_1 . \ldots . a_{i-1} . a_i . a_{i+1}$ and $s'' := a_1 . \ldots . a_{i-1} . a_{i+1} . a_i$. Obviously s'' complies with the regulation because $a_1 . \ldots . a_{i-1} . a_{i+1}$ complies with the regulation and $a_i^{\text{type}} = \texttt{REST}$. The only difference between schedule s' and s'' is that in s'' one rest period is moved to a later point in time. Thus, we have

$$\Delta^{15}_{s''} \geq \Delta^{15}_{s'}, \quad \Delta^{30}_{s''} \geq \Delta^{30}_{s'}, \quad \Delta^{60}_{s''} \geq \Delta^{60}_{s'}, \quad \Delta^{7\text{h}}_{s''} \geq \Delta^{7\text{h}}_{s'} \quad \text{and} \quad \Delta^{24\text{h}}_{s''} \geq \Delta^{24\text{h}}_{s'}.$$

Let us now set $s' \leftarrow s' . a_{i+2}$ and $s'' \leftarrow s'' . a_{i+2}$. If $a_{i+2}^{\text{type}} = \texttt{REST}$, schedule s'' is feasible. If $a_{i+2}^{\text{type}} \in \{\texttt{DRIVE},\texttt{WORK}\}$ we know that $a_{i+2}^{\text{length}} \leq \min\ \{\Delta^{15}_{s'},\Delta^{30}_{s'},\Delta^{60}_{s'},\Delta^{7\text{h}}_{s'},\Delta^{24\text{h}}_{s'}\}$. Thus, $a_{i+2}^{\text{length}} \leq \min\{\Delta^{15}_{s''},\Delta^{30}_{s''},\Delta^{60}_{s''},\Delta^{7\text{h}}_{s''},\Delta^{24\text{h}}_{s''}\}$ and schedule s'' is feasible. As the only difference between schedule s' and s'' is that in s'' one rest period is moved to a later point in time, condition (6) holds in both cases. We can analogously show that the same is true for $s' \leftarrow s' . a_{i+3}$ and $s'' \leftarrow s'' . a_{i+3}$ and so on. Thus, $a_1 . \ldots . a_{i-1} . a_{i+1} . a_i . a_{i+2} . \ldots . a_k$ is a feasible schedule. □

Property 2. *Let $s := a_1 . \ldots . a_k$ be a feasible schedule with $a_i^{type} = \texttt{REST}$ and $a_{i+1}^{type} = \texttt{DRIVE}$, $a_{i+1}^{length} > 15$ for some $1 \leq i < k$. If the partial schedule*

$a_1 . \ldots . a_{i-1} . (\texttt{DRIVE},15)$

complies with the regulation, then

$a_1 . \ldots . a_{i-1} . (\texttt{DRIVE},15) . a_i . (\texttt{DRIVE},a_{i+1}^{length}-15) . a_{i+2} . \ldots . a_k$

is a feasible schedule.

Proof. Analogue to previous property. □

Property 3. *Let $s := a_1 . \ldots . a_k$ be a feasible schedule with $a_i^{type} = \texttt{REST}$, $15 < a_i^{length} < t^{rest7h}$, and $a_{i+1}^{type} \in \{\texttt{DRIVE},\texttt{WORK}\}$ for some $1 \leq i < k$. If*

$a_1 . \ldots . a_{i-1} . (\texttt{REST},a_i^{length}-15) . a_{i+1}$

complies with the regulation and time window constraints, then

$a_1 . \ldots . a_{i-1} . (\texttt{REST},a_i^{length}-15) . a_{i+1} . (\texttt{REST},15) . a_{i+2} . \ldots . a_k$

is a feasible schedule.

Proof. As $a_i^{\text{length}} < t^{\text{rest7h}}$ we can split activity a_i into two parts $(\texttt{REST},a_i^{\text{length}}-15)$ and $(\texttt{REST},15)$ without violating any constraint. After splitting the rest, the property can be shown analogously to the first property. □

Property 4. *Let $s := a_1 . \ldots . a_k$ be a feasible schedule with $a_i^{type} = \texttt{REST}$, $15 < a_i^{length} < t^{rest7h}$, and $a_{i+1}^{type} = \texttt{DRIVE}$, $a_{i+1}^{length} > 15$ for some $1 \leq i < k$. If*

$a_1 . a_2 . \ldots . a_{i-1} . (\texttt{REST},a_i^{length}-15) . (\texttt{DRIVE},15)$

complies with the regulation, then

$a_1 . a_2 . \ldots . a_{i-1} . (\texttt{REST},a_i^{length}-15) . (\texttt{DRIVE},15)$
$. (\texttt{REST},15) . (\texttt{DRIVE},a_{i+1}^{length}-15) . a_{i+2} . \ldots . a_k$

is a feasible schedule.

Proof. Analogue to previous property. □

Properties 3 and 4 tell us that any period of rest which has a duration of less than 7 h can be set to the smallest possible duration allowing another driving or working period to be scheduled. Furthermore, Properties 1 and 2 tell us that rest periods only need to be scheduled when no further driving or working can be conducted. Because of these properties we know how to schedule driving, work and rest activities between two consecutive rest periods of 7 h or more. However, we do not know when rest periods of 7 h or more shall begin and how long they shall be. Assume we had an oracle telling us the best duration of each rest period of at least 7 h, we could solve the AUS-TDSP by building a search tree in which we create two branches each time a rest period is required. The first branch refers to a schedule in which the rest period has the shortest possible duration. The second branch refers to a schedule in which the rest period has a duration of 7 h or more. Unfortunately, determining the best duration of rest periods of at least 7 h is a difficult task. Obviously, no rest period should be unnecessarily long, because the vehicle is not productive during rest periods. On the other hand, a rest period of at least 7 h should not end too early, because this would reduce the time until the next rest of at least 7 h is required. Thus, it may be beneficial to schedule a rest period of more than 7 h duration.

5. Solution approaches

In this section, we will exploit the structural properties identified in the previous section to develop a scheduling method for solving the AUS-TDSP. The main idea of the scheduling method is to take a set $\mathcal{S}_\mu$ of feasible schedules for the partial tour $n_1,n_2,\dots,n_\mu$ and use these schedules to construct a set $\mathcal{S}_{\mu+1}$ of feasible schedules for partial tour $n_1,n_2,\dots,n_\mu,n_{\mu+1}$. This process is repeated until the AUS-TDSP for tour $n_1,n_2,\dots,n_\lambda$ is solved or no feasible schedule can be found.

Throughout the solution process, we will initially set the duration of each rest period of at least 7 h to exactly 7 h. If extending such a rest period can generate some benefit we will extend the rest period throughout the course of the algorithm. By extending the duration of a rest period, the start time of some subsequent work activity may be pushed to a later point in time. Thus, we need to know the maximum amount by which the duration of the rest period can be increased without increasing the arrival time at any work location to a value outside its time window. Let us denote this value by l_s^{extend}. For any schedule $s := a_1 \,.\, a_2 \,.\, \dots \,.\, a_k$, let us denote with $\mu(s)$ the index of the next work location to be visited. If $a_k^{\text{type}} = \texttt{WORK}$, the last rest period of 7 h or more can be increased by $t_{\mu(s_{1,k-1})}^{\max} - l_{s_{1,k-1}}^{\text{end}}$ without pushing the start time of a_k out of its time window. If $a_k^{\text{type}} = \texttt{REST}$ and $a^{\text{length}} \geq t^{\text{rest7h}}$, the rest period can be increased by $t_{\mu(s)}^{\max} - l_s^{\text{end}}$ without exceeding the time window of the next location to be visited. We can compute l_s^{extend} recursively by

$$l_{s_{1,1}}^{\text{extend}} := t_1^{\max} - l_{s_{1,1}}^{\text{end}}$$

and

$$l_{s.a}^{\text{extend}} := \begin{cases} t_{\mu(s.a)}^{\max} - l_{s.a}^{\text{end}} & \text{if } a^{\text{type}} = \texttt{REST} \text{ and } a^{\text{length}} \geq t^{\text{rest7h}} \\ \min\{l_s^{\text{extend}}, t_{\mu(s)}^{\max} - l_s^{\text{end}}\} & \text{else if } a^{\text{type}} = \texttt{WORK} \\ l_s^{\text{extend}} & \text{else} \end{cases}$$

For any schedule $s := a_1 \,.\, a_2 \,.\, \dots \,.\, a_k$, let us furthermore denote with l_s^{daily} the amount of driving and working after the last rest period of 7 h or more. This value can be computed by

$$l_s^{\text{daily}} := \sum_{\substack{i_s^{7\text{h}} \leq i \leq k \\ a_i^{\text{type}} \in \{\texttt{DRIVE},\texttt{WORK}\}}} a_i^{\text{length}}$$

We can now describe the scheduling method which is initialized with the sets

$$\mathcal{S}_1 := \{(\texttt{REST}, \max\{3120, t_1^{\min}\}).(\texttt{WORK}, w_1)\}$$

and

$$\mathcal{S}_\mu := \emptyset \quad \text{for all } 1 < \mu \leq \lambda.$$

The method is invoked with $\mu = 1$ to determine feasible schedules for tour $n_1,\dots,n_{\mu+1}$. The scheduling method is composed of two parts: the first part schedules driving and rest activities required on the trip from location n_μ to $n_{\mu+1}$; the second part schedules the work activity and waiting time that may be required at location $n_{\mu+1}$.

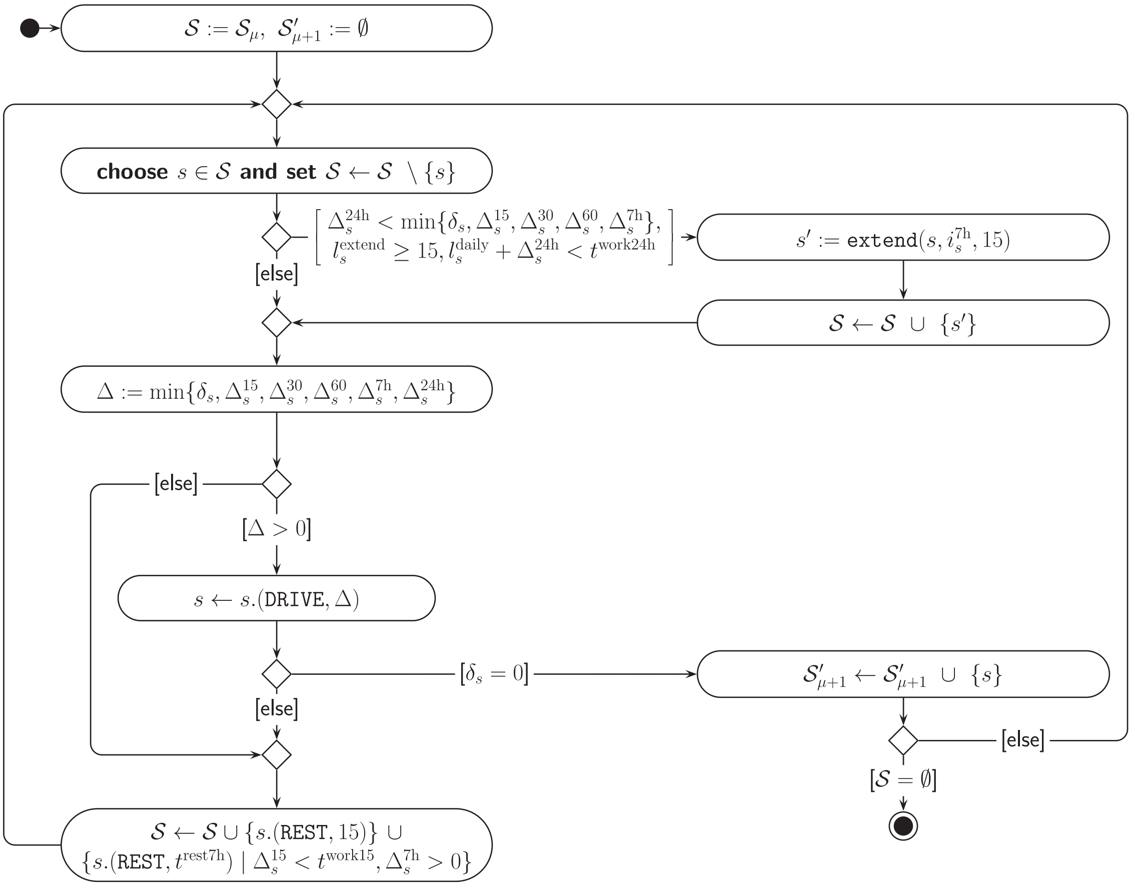
$\mathcal{S} := \mathcal{S}_\mu,\ \mathcal{S}'_{\mu+1} := \emptyset$
choose $s \in \mathcal{S}$ and set $\mathcal{S} \leftarrow \mathcal{S} \setminus \{s\}$
$\left[\begin{array}{l}\Delta_s^{24\text{h}} < \min\{\delta_s, \Delta_s^{15}, \Delta_s^{30}, \Delta_s^{60}, \Delta_s^{7\text{h}}\}, \\ l_s^{\text{extend}} \geq 15, l_s^{\text{daily}} + \Delta_s^{24\text{h}} < t^{\text{work24h}}\end{array}\right]$
$s' := \texttt{extend}(s, i_s^{7\text{h}}, 15)$
[else]
$\mathcal{S} \leftarrow \mathcal{S} \cup \{s'\}$
$\Delta := \min\{\delta_s, \Delta_s^{15}, \Delta_s^{30}, \Delta_s^{60}, \Delta_s^{7\text{h}}, \Delta_s^{24\text{h}}\}$
[else]
$[\Delta > 0]$
$s \leftarrow s.(\texttt{DRIVE}, \Delta)$
$[\delta_s = 0]$
$\mathcal{S}'_{\mu+1} \leftarrow \mathcal{S}'_{\mu+1} \cup \{s\}$
[else]
[else]
$[\mathcal{S} = \emptyset]$
$\mathcal{S} \leftarrow \mathcal{S} \cup \{s.(\texttt{REST}, 15)\} \cup$
$\{s.(\texttt{REST}, t^{\text{rest7h}}) \mid \Delta_s^{15} < t^{\text{work15}}, \Delta_s^{7\text{h}} > 0\}$

Fig. 2. Method for scheduling driving activities on the trip from location n_μ to $n_{\mu+1}$.

The first part of the scheduling method is illustrated in Fig. 2. Within the scheduling method, δ_s denotes for each partial schedule s the remaining driving time required to reach the next location $n_{\mu+1}$. The scheduling method starts by initializing the set of partial schedules $\mathcal{S}$ and the set $\mathcal{S}'_{\mu+1}$ of schedules in which no further driving is required to reach location $n_{\mu+1}$. Then, it chooses a partial schedule $s \in \mathcal{S}$ and removes it from $\mathcal{S}$. If $\Delta_s^{24\text{h}} < \min\{\delta_s, \Delta_s^{15}, \Delta_s^{30}, \Delta_s^{60}, \Delta_s^{7\text{h}}\}$, then the constraint that no more than t^{work24h} work must be scheduled within any period of 24 h limits the amount of driving that can be conducted. If furthermore $l_s^{\text{daily}} + \Delta_s^{24\text{h}} < t^{\text{work24h}}$, then the last rest period of at least 7 h is scheduled after the last rest period contributing to an accumulated amount of t^{rest24h}. Thus, extending the last rest period of at least 7 h may possibly increase the amount of driving that can be conducted. If $l_s^{\text{extend}} \geq 15$, the method creates a copy s' of the schedule in which the duration of the last rest period of at least 7 h is increased. Here, extend(s,i,Δ) denotes a function which returns the schedule s' generated by extending the duration of the ith activity in s by Δ minutes. The schedule s' is included in the set $\mathcal{S}$ of partial schedules generated so far.

The method continues by determining the maximum amount of driving that can be appended to schedule s. If this value, which is denoted by Δ, is larger than zero, a driving period of duration Δ is scheduled. If $\Delta = 0$ or $\delta_s > 0$ after scheduling the driving activity, a rest period is required before another driving activity may be scheduled.

The method includes a copy of s in which a rest period of 15 min is added to the set of partial schedules $\mathcal{S}$. Furthermore, the method includes a copy of s in which a rest period of 7 h is added to the set of partial schedules $\mathcal{S}$. This schedule, however, is only included in $\mathcal{S}$ if $\Delta_s^{15} < t^{\text{work15}}$ and $\Delta_s^{7\text{h}} > 0$. If $\Delta_s^{15} = t^{\text{work15}}$, then the last activity of the schedule s is already of type REST and a 7 h rest was already added within a previous loop. If $\Delta_s^{7\text{h}} = 0$, then a rest of at least 7 h is generated by successively adding rest periods of 15 min until the next driving period can be scheduled. Note that even though we set the duration of the rest period to exactly 7 h, the duration may be increased if this may generate a benefit later on. After scheduling rest activities the method continues with the next loop.

If $\delta_s = 0$ after scheduling a driving activity, the next location is reached and the schedule s is added to the set $\mathcal{S}'_{\mu+1}$. If $\mathcal{S} = \emptyset$, the first part of the scheduling method is terminated. Otherwise, we continue with the next loop.

Fig. 3 illustrates schedules generated by the method for scheduling driving activities according to standard rules. In the schedule in $\mathcal{S}_2$ the driver has accumulated 10 h of working time and 11 h and 30 min of rest time within the last 21 h and 30 min of the schedule. Let us assume that another 4 h of driving shall be conducted to reach the next location and that the last rest period of 7 h duration can be extended by at least 30 min without violating time window constraints. In the schedule the driver may only continue to work for another 2 h until the daily limit of 12 h working time is reached. We have $\Delta_s^{24\text{h}} < \min\{\delta_s, \Delta_s^{15}, \Delta_s^{30}, \Delta_s^{60}, \Delta_s^{7\text{h}}\}$ and the method creates a copy of the schedule in which the duration of the last rest of at least 7 h is increased by 15 min. Furthermore, it creates a schedule by appending 2 h of driving and a 15 min rest period, and another schedule by appending 2 h of driving and a 7 h rest. These three schedules are included in the set $\mathcal{S}$ and extended in the next iterations. The method

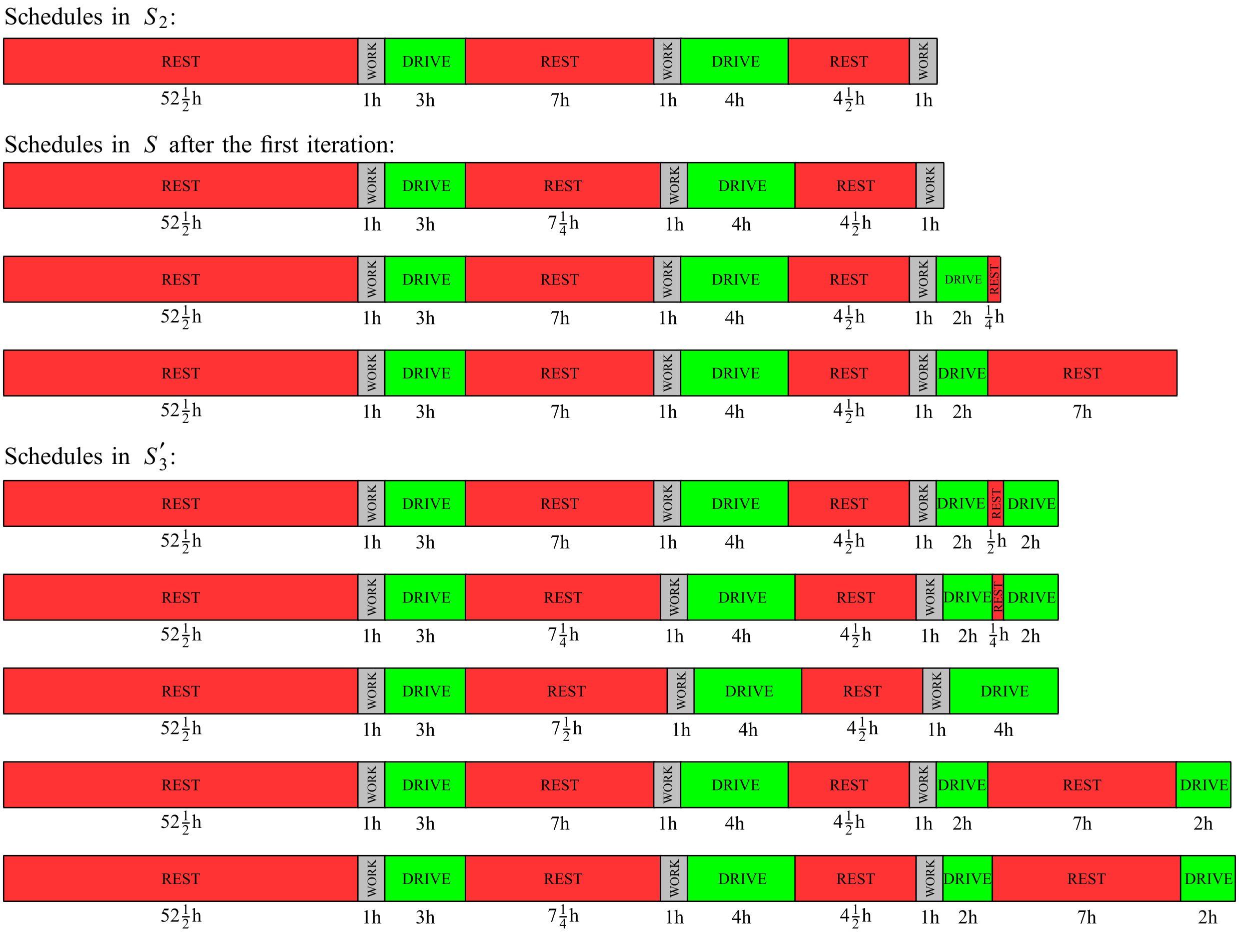
Schedules in S_2:
REST WORK DRIVE REST WORK DRIVE REST WORK
$52\frac{1}{2}$h 1h 3h 7h 1h 4h $4\frac{1}{2}$h 1h
Schedules in S after the first iteration:
REST WORK DRIVE REST WORK DRIVE REST WORK
$52\frac{1}{2}$h 1h 3h $7\frac{1}{4}$h 1h 4h $4\frac{1}{2}$h 1h
REST WORK DRIVE REST WORK DRIVE REST WORK DRIVE REST
$52\frac{1}{2}$h 1h 3h 7h 1h 4h $4\frac{1}{2}$h 1h 2h $\frac{1}{4}$h
REST WORK DRIVE REST WORK DRIVE REST WORK DRIVE REST
$52\frac{1}{2}$h 1h 3h 7h 1h 4h $4\frac{1}{2}$h 1h 2h 7h
Schedules in S'_3:
REST WORK DRIVE REST WORK DRIVE REST WORK DRIVE REST DRIVE
$52\frac{1}{2}$h 1h 3h 7h 1h 4h $4\frac{1}{2}$h 1h 2h $\frac{1}{2}$h 2h
REST WORK DRIVE REST WORK DRIVE REST WORK DRIVE REST DRIVE
$52\frac{1}{2}$h 1h 3h $7\frac{1}{4}$h 1h 4h $4\frac{1}{2}$h 1h 2h $\frac{1}{4}$h 2h
REST WORK DRIVE REST WORK DRIVE REST WORK DRIVE
$52\frac{1}{2}$h 1h 3h $7\frac{1}{2}$h 1h 4h $4\frac{1}{2}$h 1h 4h
REST WORK DRIVE REST WORK DRIVE REST WORK DRIVE REST DRIVE
$52\frac{1}{2}$h 1h 3h 7h 1h 4h $4\frac{1}{2}$h 1h 2h 7h 2h
REST WORK DRIVE REST WORK DRIVE REST WORK DRIVE REST DRIVE
$52\frac{1}{2}$h 1h 3h $7\frac{1}{4}$h 1h 4h $4\frac{1}{2}$h 1h 2h 7h 2h

Fig. 3. Schedules generated by the method for scheduling driving activities.

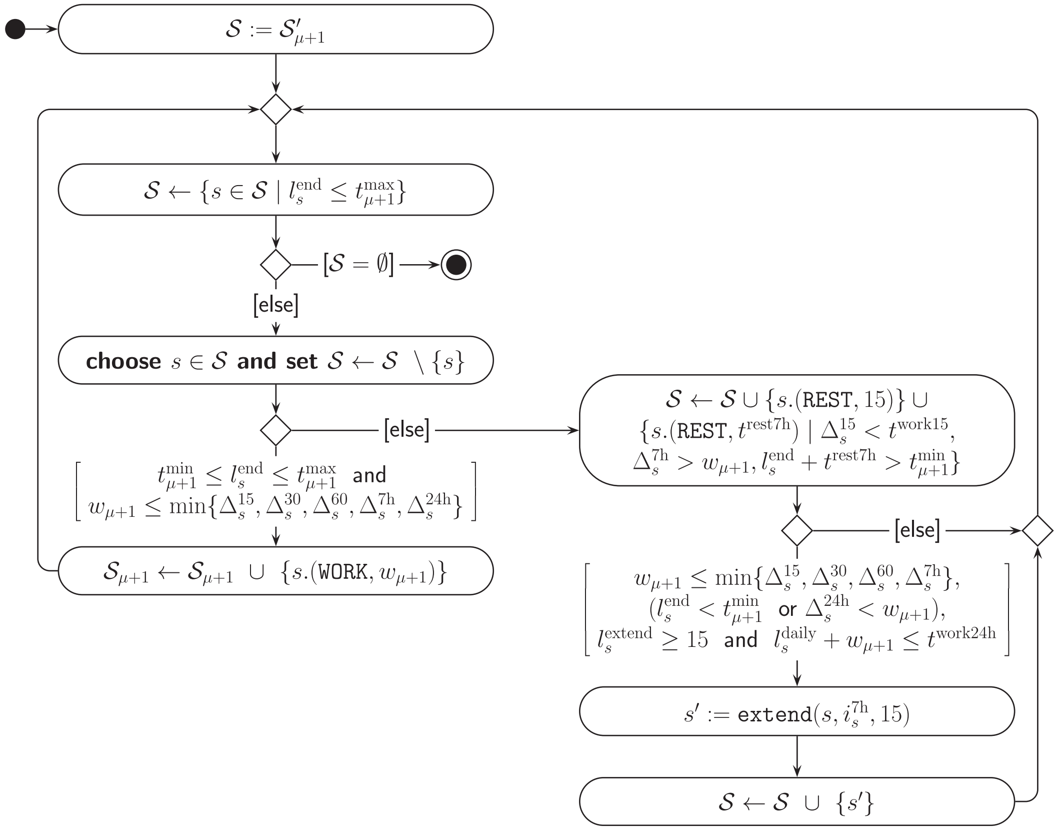
$\mathcal{S} := \mathcal{S}'_{\mu+1}$
$\mathcal{S} \leftarrow \{s \in \mathcal{S} \mid l_s^{\text{end}} \le t_{\mu+1}^{\max}\}$
$[\mathcal{S} = \emptyset]$
[else]
choose $s \in \mathcal{S}$ and set $\mathcal{S} \leftarrow \mathcal{S} \setminus \{s\}$
[else]
$\left[t_{\mu+1}^{\min} \le l_s^{\text{end}} \le t_{\mu+1}^{\max}$ and $w_{\mu+1} \le \min\{\Delta_s^{15}, \Delta_s^{30}, \Delta_s^{60}, \Delta_s^{7\text{h}}, \Delta_s^{24\text{h}}\} \right]$
$\mathcal{S}_{\mu+1} \leftarrow \mathcal{S}_{\mu+1} \cup \{s.(\texttt{WORK}, w_{\mu+1})\}$
$\mathcal{S} \leftarrow \mathcal{S} \cup \{s.(\texttt{REST}, 15)\} \cup \{s.(\texttt{REST}, t^{\text{rest7h}}) \mid \Delta_s^{15} < t^{\text{work15}}, \Delta_s^{7\text{h}} > w_{\mu+1}, l_s^{\text{end}} + t^{\text{rest7h}} > t_{\mu+1}^{\min}\}$
[else]
$\left[w_{\mu+1} \le \min\{\Delta_s^{15}, \Delta_s^{30}, \Delta_s^{60}, \Delta_s^{7\text{h}}\}, (l_s^{\text{end}} < t_{\mu+1}^{\min}$ or $\Delta_s^{24\text{h}} < w_{\mu+1}), l_s^{\text{extend}} \ge 15$ and $l_s^{\text{daily}} + w_{\mu+1} \le t^{\text{work24h}} \right]$
$s' := \texttt{extend}(s, i_s^{7\text{h}}, 15)$
$\mathcal{S} \leftarrow \mathcal{S} \cup \{s'\}$

Fig. 4. Method for scheduling waiting time at location $n_{\mu+1}$.

terminates after finding the five schedules illustrated in the figure. The third schedule in $\mathcal{S}'_3$ shows that extending the 7 h rest period by 30 min helped to avoid the need for another rest period.

The second part of the scheduling method is illustrated in Fig. 4. The second part starts by initialising the set of partial schedules $\mathcal{S}$. Then, it removes all schedules in $\mathcal{S}$ which have a completion time exceeding the time window of location $n_{\mu+1}$. If $\mathcal{S} = \emptyset$ the method terminates. Otherwise, it chooses a partial schedule $s \in \mathcal{S}$ and removes it from $\mathcal{S}$. If the completion time of schedule s is within the time window of location $n_{\mu+1}$ and if the regulation allows to schedule the entire working activity, the method adds the working activity to the schedule and includes it to the set $\mathcal{S}_{\mu+1}$. Then, the method continues with the next loop.

If the completion time of schedule s is before the opening of the time window of location $n_{\mu+1}$ or if the regulation does not allow to schedule the entire working activity, a rest activity must be scheduled. The method includes a copy of s in which a rest period of 15 min is added to the set of partial schedules $\mathcal{S}$. Furthermore, the method includes a copy of s in which a rest period of 7 h is added to the set of partial schedules $\mathcal{S}$. This schedule, however, is only included in $\mathcal{S}$ if $\Delta_s^{15} < t^{\text{work15}}$, $\Delta_s^{7\text{h}} > w_{\mu+1}$ and $l_s^{\text{end}} + t^{\text{rest7h}} > t_{\mu+1}^{\min}$. If $\Delta_s^{15} = t^{\text{work15}}$, then the last activity of the schedule s is already of type REST and a 7 h rest was already added within a previous loop. If $\Delta_s^{7\text{h}} \le w_{\mu+1}$ or $l_s^{\text{end}} + t^{\text{rest7h}} \le t_{\mu+1}^{\min}$, then a rest period of at least 7 h is generated by successively adding rest periods of 15 min until the time work period can be scheduled. Recall that even though we set the duration of the rest period to exactly 7 h, the duration may be increased if this may generate a benefit later on.

If $w_{\mu+1} \le \min\{\Delta_s^{15}, \Delta_s^{30}, \Delta_s^{60}, \Delta_s^{7\text{h}}\}$, then the only cases when the next work activity cannot be scheduled directly are the case where $l_s^{\text{end}} < t_{\mu+1}^{\min}$ and the case where $\Delta_s^{24\text{h}} < w_{\mu+1}$. In either case it is possible and potentially beneficial to extend the last rest period of at least 7 h if $l_s^{\text{extend}} \ge 15$ and $l_s^{\text{daily}} + w_{\mu+1} \le t^{\text{work24h}}$. The method creates a copy s' of the schedule in which the duration of the last rest period of at least 7 h is increased by 15 min. The schedule s' is included in the set $\mathcal{S}$ of partial schedules generated so far and the method continues with the next loop.

Fig. 5 illustrates the schedules generated by the algorithm for a small problem instance considering the standard rules. The algorithm starts with the initial schedule and adds as much driving as possible. After 5 h and 15 min work a rest period must be scheduled. The algorithm generates two schedules: one continuing with 15 min rest, the other continuing with a rest period of 7 h. Now another 2 h of driving are added to both schedules. After scheduling the second driving activity, the driver has reached the next customer location. Let us assume that the arrival time of the first schedule in $\mathcal{S}'_2$ is 45 min before the opening of the time window. Then, the algorithm generates a schedule continuing with 45 min rest. The method generates two other schedules: the first by extending the last rest of at least 7 h by 15 min and adding 30 min rest to the end of the schedule; the second by extending the last rest of at least 7 h by 30 min and adding 15 min rest to the end of the schedule. Furthermore, the method generates a schedule by adding 7 h of rest. Finally, the method adds the work period to the five partial schedules generated.

The algorithm presented finds a feasible schedule if one exists, because it enumerates all potential rest periods of 7 h or more that may be needed. However, the number of partial schedules generated can grow exponentially. Without significantly reducing the number of schedules that are considered, the solution time may be prohibitive. To reduce the computational effort, it is thus important to remove partial schedules that are not needed to solve an instance. In the following, we define criteria for dominance of a feasible schedule s' over a feasible schedule s'' which has the same amount of accumulated driving and working time as s'.

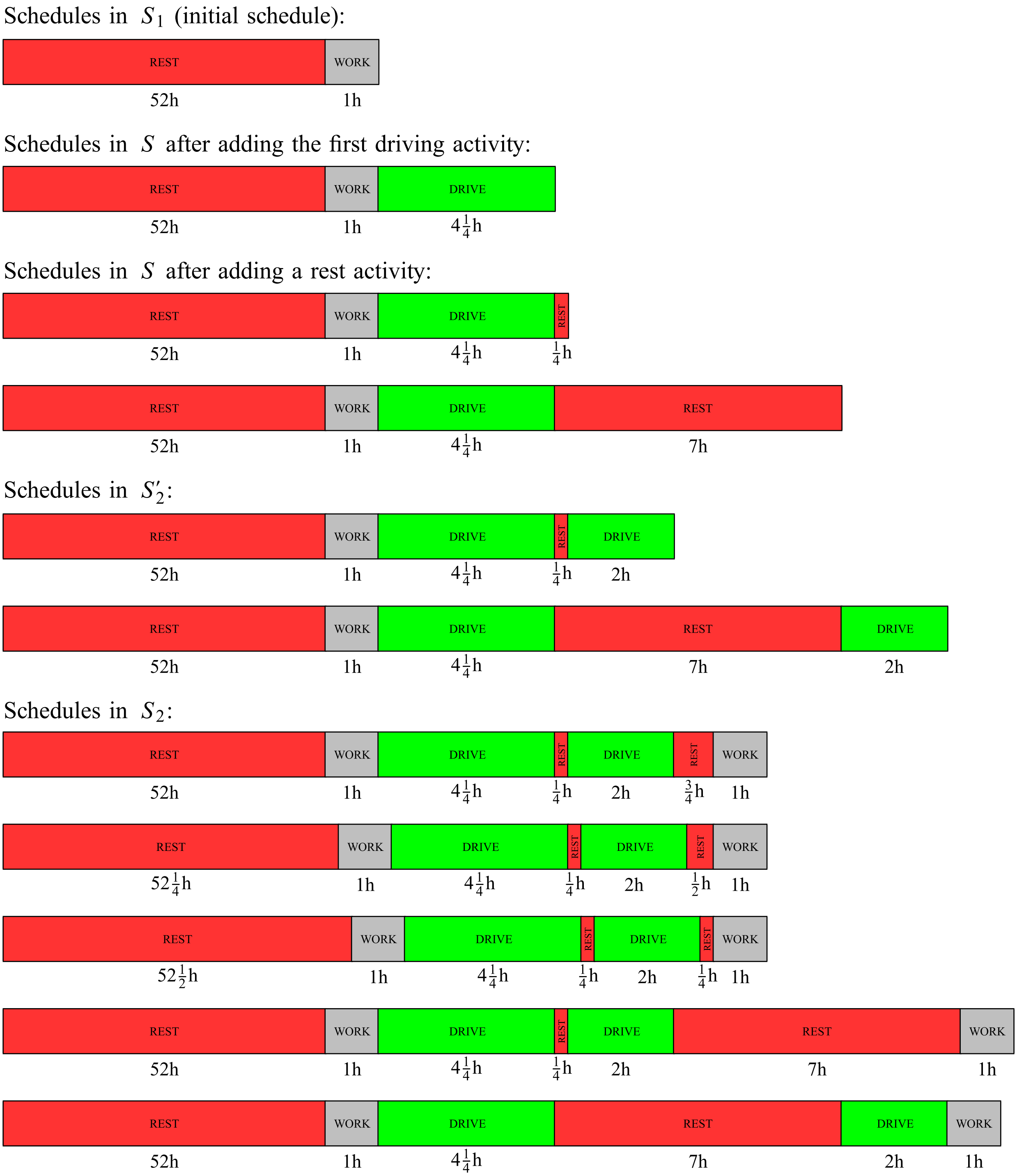
Schedules in S_1 (initial schedule):
REST WORK
52h 1h
Schedules in S after adding the first driving activity:
REST WORK DRIVE
52h 1h $4\frac{1}{4}$h
Schedules in S after adding a rest activity:
REST WORK DRIVE REST
52h 1h $4\frac{1}{4}$h $\frac{1}{4}$h
REST WORK DRIVE REST
52h 1h $4\frac{1}{4}$h 7h
Schedules in S_2':
REST WORK DRIVE REST DRIVE
52h 1h $4\frac{1}{4}$h $\frac{1}{4}$h 2h
REST WORK DRIVE REST DRIVE
52h 1h $4\frac{1}{4}$h 7h 2h
Schedules in S_2:
REST WORK DRIVE REST DRIVE REST WORK
52h 1h $4\frac{1}{4}$h $\frac{1}{4}$h 2h $\frac{3}{4}$h 1h
REST WORK DRIVE REST DRIVE REST WORK
$52\frac{1}{4}$h 1h $4\frac{1}{4}$h $\frac{1}{4}$h 2h $\frac{1}{2}$h 1h
REST WORK DRIVE REST DRIVE REST WORK
$52\frac{1}{2}$h 1h $4\frac{1}{4}$h $\frac{1}{4}$h 2h $\frac{1}{4}$h 1h
REST WORK DRIVE REST DRIVE REST WORK
52h 1h $4\frac{1}{4}$h $\frac{1}{4}$h 2h 7h 1h
REST WORK DRIVE REST DRIVE WORK
52h 1h $4\frac{1}{4}$h 7h 2h 1h

Fig. 5. Schedules generated by the algorithm.

If $l^{\text{end}}_{s'}+t^{\text{rest24h}} \le l^{\text{end}}_{s''}$ then s' dominates schedule s'', because any activity that can be appended to s'' can be appended to $\bar{s}' := s'.(\text{REST},l^{\text{end}}_{s''}-l^{\text{end}}_{s'})$ without violating any constraint of the problem.

If $l^{\text{end}}_{s'} = l^{\text{end}}_{s''}$ and $\Delta^{7\text{h}}_{s'} \ge \Delta^{7\text{h}}_{s''}$, then schedule s' dominates schedule s'' if for $i := i^{24\text{h}}_{s'}$ and all $\tau \in [0, l^{\text{end}}_{s'} - l^{\text{end}}_{s'_{1,i}}]$ the accumulated amount of rest in the last τ minutes of schedule s' is at least as high as the corresponding value for schedule s''. In this case, any activity that can be appended to s'' can be appended to s' without violating any of the constraints of the problem.

This dominance criterion can also be used if $l^{\text{end}}_{s'} < l^{\text{end}}_{s''}$ and $\Delta^{7\text{h}}_{s'} \ge \Delta^{7\text{h}}_{s''}$. For this let us use s' to generate a schedule $\bar{s}'$ with $l^{\text{end}}_{\bar{s}'} = l^{\text{end}}_{s''}$ and $\Delta^{7\text{h}}_{\bar{s}'} \ge \Delta^{7\text{h}}_{s''}$. Note that we can only append a rest period of duration $\min\{l^{\text{end}}_{s''}-l^{\text{end}}_{s'},\Delta^{7\text{h}}_{s'}-\Delta^{7\text{h}}_{s''}\}$ to s', because otherwise we would have $l^{\text{end}}_{\bar{s}'} > l^{\text{end}}_{s''}$ or $\Delta^{7\text{h}}_{\bar{s}'} < \Delta^{7\text{h}}_{s''}$. Let us thus generate $\bar{s}'$ by appending a rest period of duration $\min\{l^{\text{end}}_{s''}-l^{\text{end}}_{s'},\Delta^{7\text{h}}_{s'}-\Delta^{7\text{h}}_{s''}\}$ and by extending the last rest period of at least 7 h in s' by $l^{\text{end}}_{s''}-l^{\text{end}}_{s'}-\min\{l^{\text{end}}_{s''}-l^{\text{end}}_{s'},\Delta^{7\text{h}}_{s'}-\Delta^{7\text{h}}_{s''}\}$. The new schedule $\bar{s}'$ is feasible if $l^{\text{end}}_{s''}-l^{\text{end}}_{s'}-\min\{l^{\text{end}}_{s''}-l^{\text{end}}_{s'},\Delta^{7\text{h}}_{s'}-\Delta^{7\text{h}}_{s''}\} \le l^{\text{extend}}_{s'}$. If $\bar{s}'$ is feasible and dominates s'' then s' dominates schedule s'', because there exists a feasible schedule that can be generated by extending the last rest period of at least 7 h in s' and appending additional rest to s'. As the scheduling method explores all reasonable possibilities of extending and appending rest periods, the method will find a schedule if one exists even if s'' is discarded.

By removing dominated schedules, we can drastically reduce the computational effort required to solve the AUS-TDSP. However, if the truck driver scheduling method is integrated into a method for combined vehicle routing and truck driver scheduling, it may be desirable to solve the AUS-TDSP even faster. Below, we remove some computationally expensive steps from the method outlined above to obtain heuristics for the AUS-TDSP. All schedules generated by these heuristics are feasible, but the heuristics may sometimes fail to find a feasible schedule even though one exists.

The first heuristic, which we denote by AUS1, removes all the steps from the exact algorithm in which the last 7-h rest is extended and in which a 7-h rest is scheduled. Note that, when necessary, a rest period of 7 h or more is generated by successively scheduling rest periods of duration t^{rest15}. The second heuristic, which we denote by AUS2, uses the same method as AUS1 for scheduling driving activities on the trip from one location to the next, but the method for scheduling waiting time at a location is obtained by removing all steps from the original algorithm in which the last 7-h rest is extended. The third heuristic, which we denote by AUS3, uses the same method as AUS1 for scheduling driving activities and the method for scheduling waiting time at a location of the exact method. The fourth heuristic, which we denote by AUS4, schedules driving activities by removing the steps from the exact algorithm in which the last 7-h rest is extended and the method for scheduling waiting time at a location of the exact method.

6. Computational experiments

In order to evaluate the scheduling methods presented in this paper, we generate benchmark instances for a planning horizon starting on Monday 4.00 AM and ending on Friday 11.59 PM. The driving time between two consecutive locations is randomly set to a value between 2 and 16 h. Assuming an average speed of 75 km/h, this implies that the distance between two consecutive locations ranges from 150 to 1200 km. The length of the time window at a location is set randomly between 1 and 12 h and the start of the time window at a location is set randomly to a time between 15 min and 12 h after the earliest possible arrival time at the location when no driver regulations are considered. One hour of work has to be conducted at each location. We randomly generated 10 000 instances. Among these, 7306 instances do not exceed 72 h of accumulated work time and were used for our experiments. The number of locations to be visited in these instances ranges from 5 to 11.

Table 2 summarizes the results of our experiments. The table shows the number of instances for which each of the methods finds a feasible schedule, the time required by the method, and the maximum number of partial schedules (nodes in the search tree) generated. The table also shows the results for a version of the exact method which ignores Provisions 1–3 of the regulation, and thus only considers Provision 4. If this method, which is denoted by AUSR, does not find a feasible schedule, then we know that no feasible schedule exists. Table 2 shows the number of instances for which AUSR proves that no feasible schedule exists. Furthermore, the table shows the results for the exact method which is denoted by AUS*. For comparison, we also include results indicating how many of the instances are feasible based on the U.S. hours of service regulations (for which we used the algorithm presented in [5]) and based on the European Union regulations (for which we used the algorithm presented by [10]). As the weekly limit on the accumulated amount of driving is smaller in the United States and the European Union, 873 (US) and 5409 (EU) instances were immediately found to be infeasible.

The most noticeable result is that for more than six times as many instances a schedule complying with the rules of the BFM can be found compared to the rules of the Standard Hours. Obviously, the higher amount of work time allowed and the smaller amount of rest time required within a 24 h period strongly contributes to this difference. Furthermore, it is clear that the U.S. hours of service regulations are less restrictive than the Standard Hours, but more restrictive than the BFM. The European Union regulations are by far the most restrictive.

We can see that AUS1 finds a feasible schedule for around 81% of the instances for which a feasible schedule exists and AUS2 for around 93%. The reason for this difference is that AUS2 better handles early arrivals at customer locations. If a waiting time of less than 7 h is required at a customer, AUS2 explores two alternatives: continuing with the minimum amount of rest required, and continuing with a rest period of 7 h. Thus, waiting time may be extended to become a rest period of 7 h. AUS3 goes even further and also considers the possibility of extending rest periods so as to be able to increase the time until the next rest period of 7 h is required. As a result, AUS3 finds a feasible schedule for around 99% of the instances. However, as there are many ways to extend rest periods, the computational effort for AUS3 is significantly higher than for AUS1 and AUS2 which have running times comparable to those of the methods presented by Goel and Kok [5] and Goel [10]. AUS4, which also considers scheduling 7-h rest periods during driving activities, finds a feasible schedule for almost all the instances for which a feasible schedule exists. Not surprisingly, this extra flexibility comes at

Table 2
Results.

Algorithm	Rules	Feasible	Infeasible	Computation time (s)	Max. nodes
AUS1	Standard	299	n/a	2.4	10
AUS2	Standard	339	n/a	4.2	36
AUS3	Standard	358	n/a	111.6	1321
AUS4	Standard	366	n/a	194.5	2827
AUSR	Standard	n/a	6938	1756.5	103708
AUS*	Standard	367	6939	3509.5	117245
AUS1	BFM	1950	n/a	3.0	11
AUS2	BFM	2239	n/a	5.2	46
AUS3	BFM	2380	n/a	170.2	1323
AUS4	BFM	2398	n/a	282.9	2259
AUSR	BFM	n/a	4879	1055.4	21321
AUS*	BFM	2401	4905	1645.0	43788
Goel and Kok [5]	US	1171	–	1.4	42
Goel [10]	EU	122	–	2.3	705

the cost of another increase in computational effort. However, the computational effort of AUS4 (and thus of all heuristics) is still an order of magnitude less than the computational efforts required by AUS*, the exact method. The results also clearly show why: the exact algorithm generates many more partial feasible schedules (nodes in the search tree). The reason is that it not only considers increasing the duration of the last rest period of at least 7 h if the next work cannot be scheduled immediately, but whenever there is a potential benefit of extending a rest period of at least 7 h. The results also show that the sliding time window of 24 h, i.e., Provision 4, is greatly responsible for the high complexity of the Australian driver regulations. The AUSR method, which only considers Provision 4, may generate more than one hundred thousand non-dominated partial schedules whereas only a few hundred or less are generated when considering driver regulations in the United States or Europe for similarly sized instances.

7. Summary

This paper studies the Australian Heavy Vehicle Driver Fatigue Law which entered into force on September 29, 2008. The law incorporates three sets of alternative rules: Standard Hours, Basic Fatigue Management, and Advanced Fatigue Management. This paper formulates the Australian Truck Driver Scheduling Problem (AUS-TDSP) for the standard rules and the Basic Fatigue Management rules. Structural properties of the problem are analyzed and used to develop methods for solving the problem. An exact method for solving the AUS-TDSP is presented and dominance criteria are described which help to drastically reduce the computational effort. Four heuristics are presented that eliminate progressively most of the computationally expensive steps of the exact algorithm. Computational experiments demonstrate the efficacy of the heuristics and that the most effective heuristic can find a feasible schedule for almost all of the instances for which a feasible schedule exists; only in rare cases a feasible schedule can only be found by the exact algorithm. The computational effort required by the heuristics is only a small fraction of the computational effort required by the exact algorithm.

In order to operate according to the Basic Fatigue Management or the Advanced Fatigue Management rules an operator must be accredited in the National Heavy Vehicle Accreditation Scheme. Among the conditions to become accredited is the requirement that operators must plan schedules and rosters to ensure that they comply with the respective operating limits. The technology developed and discussed in this paper can form the foundation for a successful application for accreditation. As operating limits of the Advanced Fatigue Management rules are set on a case-by-case basis we did not explicitly consider this option. However, our technology can be adapted to accommodate specific operating limits set for an Advanced Fatigue Management option.

Acknowledgments

This research was supported by the German Research Foundation (DFG) and the National ICT Australia (NICTA).

References

[1] Williamson A, Sadural S, Feyer A, Friswell R. Driver fatigue: a survey of long distance heavy vehicle drivers in Australia. Information Paper CR 198, National Road Transport Commission, 2001.

[2] Ernst AT, Jiang H, Krishnamoorthy M, Owens B, Sier D. An annotated bibliography of personnel scheduling and rostering. Annals of Operations Research 2004;127(1):21–144, doi:10.1023/b:anor.0000019087.46656.e2.

[3] Xu H, Chen Z-L, Rajagopal S, Arunapuram S. Solving a practical pickup and delivery problem. Transportation Science 2003;37(3):347–64.

[4] Archetti C, Savelsbergh MWP. The trip scheduling problem. Transportation Science 2009;43(4):417–31, doi:10.1287/trsc.1090.0278.

[5] Goel A, Kok L. Truck driver scheduling in the United States. Transportation Science, to appear.

[6] Rancourt ME, Cordeau J-F, Laporte G. Long-haul Vehicle Routing and Scheduling with Working Hour Rules. Technical Report, CIRRELT-2010-46, Centre Interuniversitaire de Recherche sur les Réseaux d'Enterprise, la Logistique et le Transport; 2010.

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

[8] Kok AL, Meyer CM, Kopfer H, Schutten JMJ. A dynamic programming heuristic for the vehicle routing problem with time windows and European community social legislation. Transportation Science 2010;44(4):442–54.

[9] Prescott-Gagnon E, Desaulniers G, Drexl M, Rousseau L-M. European driver rules in vehicle routing with time windows. Transportation Science 2010;44(4): 455–73.

[10] Goel A. Truck driver scheduling in the European Union. Transportation Science 2010;44(4):429–41.

[11] National Transport Commission. Heavy Vehicle Driver Fatique Reform – Basic Fatigue Management Explained. Information Bulletin, 2008a.

[12] National Transport Commission. Heavy Vehicle Driver Fatique Reform – Advanced Fatigue Management Explained. Information Bulletin, 2008b.

[13] National Transport Commission. Heavy Vehicle Driver Fatique Reform – Standard Hours Explained. Information Bulletin, 2008c.