

The Bus Driver Scheduling Problem

About this chapter

This chapter introduces the Bus Driver Scheduling Problem with Complex Break Constraints. This is a challenging scheduling problem originating from the general Transportation Planning System.

1.1 Background

We can distinguish three goals of the **Bus Driver Scheduling Problem (BDSP)**. The *first* goal is to define daily shifts that cover predetermined bus tours and meet the transportation system's demand.

Fatigue is a major safety concern for drivers, and they must have enough time to rest and recharge to avoid accidents and maintain a high level of safety for passengers. This is why the *second* goal of the **BDSP** is to respect all the requirements from the collective agreement and legal regulations. For example, we must ensure that the drivers have sufficient driving breaks during their shifts.

The *third* goal of the **BDSP** is to reduce cost and maximise effectiveness. This means not only saving money for the bus company, but also minimising the dissatisfaction of the drivers. For example, breaks that last more than three hours are usually poorly regarded by the drivers (they are unpaid), and we consider this in the objective function.

This study considers regulations from two documents: The first one is the *European Regulation (EC) No 561/2006* [?] that describes rules on driving times, breaks and rest periods for vehicle drivers of public transport. The second document is the *Austrian*

Collective Agreement for employees in private omnibus providers [?] using the rules for regional lines up to 50 km. This is particularly strict regarding the break regulations, which makes the **BDSP** extremely constrained and challenging to solve.

Although we could rename this variant the Bus Driver Scheduling Problem with Austrian Regulations (or Bus Driver Scheduling Problem with Complex Constraints), for clarity we simply refer to it as the *Bus Driver Scheduling Problem*.

1.2 Problem Description

This section provides a formal specification of the **BDSP**. This section is partially taken from previous work in the literature [?, ?].

1.2.1 Problem Input

The input of the problem consists of three pieces of data:

- **Positions and Distance Matrix:** A finite set $P \subseteq \mathbb{N}$ of *positions* or bus stops. A time distance matrix $D = (d_{ij}) \in \mathbb{N}^{|P| \times |P|}$ where d_{ij} represents the time needed for a driver to go from position i to j when not actively driving a bus. If no transfer is possible, then we set $d_{ij} = +\infty$ ¹. When $i \neq j$, d_{ij} is referred to as *passive ride time*. In contrast, d_{ii} represents the time required to switch tour at the same position, but is not considered passive ride time.
- **Start and End Work:** for every position $p \in P$, two integer values $startWork_p$ and $endWork_p$ respectively represent the time required to start or end a shift at that position.
- **Bus Legs:** A set L of **Bus Legs**, where each leg $\ell \in L$ is defined as a 5-tuple:

$$\ell = (tour_\ell, startPos_\ell, endPos_\ell, start_\ell, end_\ell),$$

representing the trip of a vehicle between two stops at a certain time:

- $tour_\ell \in \mathbb{N}$ is the ID of the vehicle.
- $startPos_\ell, endPos_\ell \in P$ the starting and the ending positions of the leg.
- $start_\ell \in \mathbb{N}$ is the time at which the vehicle departs from position $startPos_\ell$.
- $end_\ell \in \mathbb{N}$ is the time at which the vehicle arrives to position $endPos_\ell$.

A bus leg ℓ is uniquely defined by its pair of start time $start_\ell$ and bus tour $tour_\ell$.

The driving time of a bus leg is $length_\ell = end_\ell - start_\ell$. **Bus Legs** belonging to the same **Bus Tour** t do not overlap, which means that the intervals $(start_\ell, end_\ell)$ for ℓ with $tour_\ell = t$ are disjoint. Note that all times are expressed in minutes, and time the horizon spans 24 hours.

¹I apologise to any mathematician reading this. In fact, in such cases, we set $d_{ij} = 999\,999 = 10^6 - 1$.

Table 1.1 shows a small example of a set of six legs. The first vehicle starts at time 360 (6:00 am) at position 0 and travels between positions 1 and 2 with waiting times in between. Finally, it returns to position 0.

Table 1.1: A toy example with two bus tours from [?]

$tour_\ell$	$startPos_\ell$	$endPos_\ell$	$start_\ell$	end_ℓ
1	0	1	360	395
1	1	2	410	455
1	2	1	460	502
1	1	0	508	540
2	0	3	360	400
2	3	0	415	500

We can think of the set of bus legs L as a totally ordered set (Lemma 1).

Lemma 1. *Let \preceq be the relation on L defined as follows: for any $\ell_1, \ell_2 \in L$, write $\ell_1 \preceq \ell_2$ if either*

- $start_{\ell_1} < start_{\ell_2}$, or
- $start_{\ell_1} = start_{\ell_2}$ and $tour_{\ell_1} \leq tour_{\ell_2}$

Then \preceq is an order relation, and (L, \preceq) is a totally ordered set.

Proof.

- *Reflexivity:* For each $\ell \in L$ we have $start_\ell = start_\ell$ and $tour_\ell = tour_\ell$. So $\ell \preceq \ell$.
- *Antisymmetry* Let $\ell_1, \ell_2 \in L$ be such that $\ell_1 \preceq \ell_2$ and $\ell_2 \preceq \ell_1$. This implies that $start_{\ell_1} = start_{\ell_2}$, $tour_{\ell_1} \leq tour_{\ell_2}$, and $tour_{\ell_2} \leq tour_{\ell_1}$. So $\ell_1 = \ell_2$.
- *Transitivity* Let $\ell_1, \ell_2, \ell_3 \in L$ be such that $\ell_1 \preceq \ell_2$, and $\ell_2 \preceq \ell_3$. Then there are four cases:
 1. $start_{\ell_1} < start_{\ell_2} < start_{\ell_3}$. Then $start_{\ell_1} < start_{\ell_3}$ and therefore $\ell_1 \preceq \ell_3$.
 2. $start_{\ell_1} = start_{\ell_2} < start_{\ell_3}$ so $\ell_1 \preceq \ell_3$.
 3. $start_{\ell_1} < start_{\ell_2} = start_{\ell_3}$ so $\ell_1 \preceq \ell_3$.
 4. $start_{\ell_1} = start_{\ell_2} = start_{\ell_3}$ and $tour_{\ell_1} \leq tour_{\ell_2} \leq tour_{\ell_3}$. This implies $\ell_1 \preceq \ell_3$.
- *Totally ordered:* For any $\ell_1, \ell_2 \in L$, exactly one of the following holds:
 - $start_{\ell_1} < start_{\ell_2}$,

- $start_{\ell_1} > start_{\ell_2}$, or
- $start_{\ell_1} = start_{\ell_2}$.

In the first two cases, $\ell_1 \preceq \ell_2$ or $\ell_2 \preceq \ell_1$ follows directly. If $start_{\ell_1} = start_{\ell_2}$, then $tour_{\ell_1} \leq tour_{\ell_2}$ or $tour_{\ell_1} \geq tour_{\ell_2}$. Thus, $\ell_1 \preceq \ell_2$ or $\ell_2 \preceq \ell_1$. Hence, (L, \preceq) is totally ordered. \square

1.2.2 Solution

A *solution* S to the **BDSP** is an assignment of drivers to bus legs. Formally, it is represented as a *set partition* of L , denoted as $S = \{s_1, s_2, \dots, s_n\}$, where each block $s_i \in S$ is denoted as a *shift*. Each shift s_i represents a subset of bus legs that are assigned to a single driver. A priori, the number of shifts n is not given. Nevertheless, we can imagine to set it as large as we need in order to get a feasible solution. The largest possible partition occurs when each shift contains only one leg, implying that $|S| \leq |L|$. A lower bound for $|S|$ will be discussed in **Lemma 2**.

Equivalently, it can be useful to think about *shifts* as *the work scheduled to be performed by a driver in one day* [?]. For convenience, we may sometimes refer to the shift s as the *driver* s .

Note that, since the set L is totally ordered (**Lemma 1**), the notion of *consecutive* legs in a shift s is well defined. Moreover, a solution is also totally ordered by the order induced by the bus legs.

A solution is feasible if it satisfies the following criteria:

- For every shift s in the solution, if $i, j \in s$ are consecutive bus legs and $tour_i \neq tour_j$ or $endPos_i \neq startPos_j$, then the driver must have enough time to cover the distance between $endPos_i$, and $startPos_j$. This rule is expressed by the constraint

$$start_j \geq end_i + d_{endPos_i, startPos_j}.$$

This implicitly guarantees no overlapping bus legs within a single shift, since $d_{p,q} > 0$ for every $p, q \in P$.

- Each shift must satisfy all hard constraints depending on the laws specified in the next sections.

1.2.3 Constraints

This section describes the constraints of our **BDSP** variant, derived from the Austrian collective agreement.

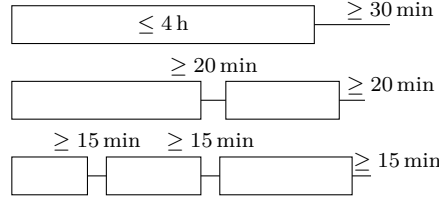


Figure 1.1: Driving time constraints and required break options [?].

Driving time

$$D_s = \sum_{i \in s} \text{length}_i = \sum_{i \in e} (end_i - start_i) \quad \forall s \in S \quad (1.1)$$

$$D_s \leq D_{\max} = 540 \quad \forall s \in S \quad (1.2)$$

Equation (1.1) defines the *driving time* D_s of a shift e . Constraints Equation (1.2) set the upper bound of D_s of nine hours. The driving time is subject to additional rules regarding driving breaks. The length of a *driving break* between two consecutive bus legs i and j is

$$\text{diff}_{ij} = start_j - end_i.$$

The driving break can be split in multiple parts, all of which must be completed *before* the cumulative driving time without a break reaches 4 h:

- One driving break of at least 30 min.
- Two driving breaks of at least 20 min each.
- Three driving breaks of at least 15 min each.

These driving rules are shown in Figure 1.1. Once we reach all required breaks, a new block of at most 4 h begins.

Total Time

$$T_s = end_\ell + endWork_\ell - (start_f - startWork_f) \quad \forall s \in S \quad (1.3)$$

$$T_s \leq T_{\max} = 840 \quad \forall s \in S \quad (1.4)$$

Equation (1.3) defined the *total time* of a shift s : it is the span from the start of work until the end of work, where f is the first leg and ℓ is the last leg in the shift s . Equation (1.4) sets the upper bound of T_s : no driver can work more than fourteen hours.

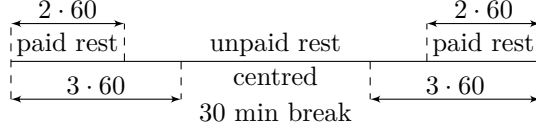


Figure 1.2: Rest break positioning [?]

Shift Splits.

An important concept is the one of **shift splits**. We say that the shift e contains a shift split if there exists a pair of consecutive legs $i, j \in e$ such that the break between them satisfies

$$start_j - end_i - r_{endPos_i, startPos_j} \geq 180.$$

Here, $r_{p,q} = d_{p,q}$ denotes the time required for a passive ride between position p and q , where $r_{p,q} = d_{p,q}$ if $p \neq q$ and $r_{p,p} = 0$. Hence, shift splits refer to breaks longer than three hours. As these breaks are unpaid, they are generally poorly regarded by bus drivers. This plays a role in designing the objective function.

Denote by $split_s$ the number of shift splits in shift s and by $splitTime_s$ the total duration of these shift splits. A shift split resets the driving time (i.e., it counts as a *driving* break). A shift contains up to two shift splits.

Rest Break

Let i, j be two consecutive bus legs in a shift. The break between these two legs can be represented as the time interval $[end_i, start_j]$, and its length as $diff_{ij} = start_j - end_i$. We denote with $diff'_{ij}$ the length of a *rest break*:

$$diff'_{ij} = \begin{cases} diff_{ij} - r_{endPos_i, startPos_j} & \text{if } 15 \leq diff_{ij} - r_{endPos_i, startPos_j} \leq 180 \\ 0 & \text{otherwise} \end{cases}$$

If the duration of a (partial) rest break is at least at least 15 min long, the break is considered unpaid if it does not fall within the first 2 or last 2 hours of the shift. The maximum amount of unpaid rest breaks ($upmax_s$) is limited, as shown in [Figure 1.2](#):

- If 30 consecutive minutes of rest break do not intersect the first 3 h or the last 3 h of the shift, at most 1.5 h of unpaid rest are allowed and therefore we set $upmax_s = 75$.
- Otherwise, at most 1 h of unpaid rest is allowed, and therefore we set $upmax_s = 60$.

In formulas,

$$upmax_s = \begin{cases} 90 & \text{if a 30-minute consecutive rest break does not intersect} \\ & \text{the first 3 h or the last 3 h of the shift,} \\ 60 & \text{if there is a 30-minute break centered between the first} \\ & \text{3 h and the last 3 h of the shift,} \\ 0 & \text{otherwise.} \end{cases}$$

Rest breaks beyond this limit are paid. We denote by $unpaid_s$ the sum of the length of unpaid rest breaks.

Working time

$$W_s = T_s - splitTime_s - \min\{unpaid_s, upmax_s\} \quad \forall s \in S \quad (1.5)$$

$$W_s \leq W_{\max} = 600 \quad \forall s \in S \quad (1.6)$$

Equation (1.5) defines the *working time* of a shift s . Equation (1.6) sets the upper bound of W_s as ten hours.

The working time is subject to additional rules regarding rest breaks. The minimum rest break depends on the working time:

- $W_s < 300$: no rest break required.
- $300 \leq W_s \leq 540$: at least one 30-minute break.
- $W_s > 540$: at least one 45-minute break.

Rest breaks may be split into smaller parts. If a break is split, one part must be at least 30 min in duration, and any additional part must be at least 15 min. The first part of the break must occur within the first six hours of working time.

1.2.4 Objective function

Let S be a solution. The objective function, that we want to minimise, is defined as follows [?, ?]:

$$z(S) = \sum_{s \in S} (2W'_s + T_s + ride_s + 30change_s + 180split_s), \quad (1.7)$$

where, for every shift s :

- $W'_s = \max\{W_s, 390\}$, where W_s is the working time defined by Equation (1.5). This objective ensures that drivers are paid for at least 6.5 hours (390 minutes) of work, even if they work less.
- T_s is the total time of the shift as defined in Equation (1.3).
- $ride_s$ is the sum of passive ride times between consecutive legs.
- $change_s$ is the number of *tour changes* that is, the number of occurrences of consecutive bus legs $i, j \in s$ with $tour_i \neq tour_j$.
- $split_s$ is the number of shift splits.

The coefficients of the linear combination were determined by a previous work [?] based on preferences agreed upon by different stakeholders at Austrian bus companies and employee scheduling experts.

As argued by Kletzander and Musliu [?], practical schedules must not only consider operating costs, but also the well-being of the drivers. For this reason, *change*, *split* and total time T_s are included in the objective function.

In Figure 1.3 we picture an example of a shift with three bus legs.

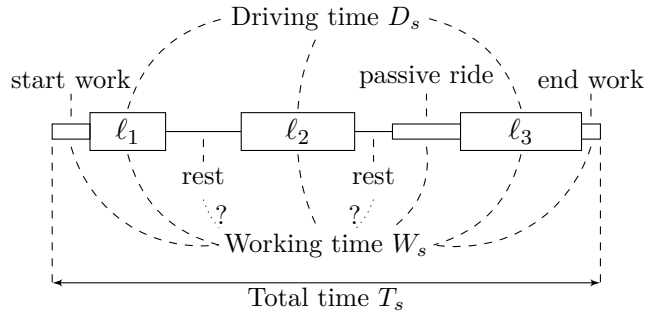


Figure 1.3: Example shift $s = \{\ell_1, \ell_2, \ell_3\}$ [?]

Upper bound of the objective function

Consider an instance of the **BDSP**. A *trivial* solution consists of assigning one shift to each leg. Let $L = \{\ell_1, \ell_2, \dots, \ell_n\}$ be the set of bus legs and let S be the solution defined as $S = \{\{\ell_1\}, \{\ell_2\}, \dots, \{\ell_n\}\}$.

The solution S is feasible and the objective function [eq. \(1.7\)](#) is

$$\begin{aligned} z(S) &= \sum_{s \in S} (T_s + 390) \\ &= 780 \cdot |L| + \sum_{\ell \in L} (\text{startWork}(\text{startPos}(\ell)) + \text{endWork}(\text{endPos}(\ell)) + \text{length}_\ell). \end{aligned}$$

This gives us an **upper bound** for the optimal solution S^* :

$$z(S^*) \leq 780 \cdot |L| + \sum_{\ell \in L} (\text{startWork}(\text{startPos}(\ell)) + \text{endWork}(\text{endPos}(\ell)) + \text{length}_\ell).$$

We are now ready to evaluate a lower bound for the number of shifts of a solution $|S|$. This proof is adapted and expanded from previous work [?].

Lemma 2. Consider one instance of the **BDSP** and define

$$\lambda_1 := \left\lceil \frac{1}{D_{\max}} \sum_{\ell \in L} \text{length}_\ell \right\rceil \quad \text{and} \quad \lambda_2 := \max \{a(t_1) + a(t_2) : |t_1 - t_2| > T_{\max}\},$$

where $a(t) = |\{\ell \in L : \text{start}_\ell \leq t \leq \text{end}_\ell\}|$ is the number of bus legs at time t that are active, i.e., their correspondent bus tour (vehicle) is currently driving.

Then, for every feasible solution S , it holds that $\max\{\lambda_1, \lambda_2\} \leq |S|$.

Proof. let S be a feasible solution. Summing the length of every bus leg in L , we have

$$\sum_{\ell \in L} \text{length}_\ell = \sum_{s \in S} D_s \leq |S| \cdot D_{\max}. \quad (1.8)$$

where D_s and D_{\max} have been defined in [Equation \(1.1\)](#) and [Equation \(1.2\)](#). This implies that

$$|S| \geq \left\lceil \frac{1}{D_{\max}} \sum_{\ell \in L} \text{length}_\ell \right\rceil =: \lambda_1.$$

A second lower bound takes into account the number of active bus legs $a(t)$ at time t . Clearly, $|S| \geq a(t)$ for any given time t , because each active leg must have a driver assigned to it. However, this lower bound can be improved using the fact that for every time t , no active employee at time t can be active at time $t + T_{\max}$ or $t - T_{\max}$, because of [Equation \(1.4\)](#). Therefore,

$$|S| \geq \max \{a(t_1) + a(t_2) : |t_1 - t_2| > T_{\max}\} =: \lambda_2.$$

Hence, $|S| \geq \max(\lambda_1, \lambda_2)$. □

Note that two optimal solutions can have different numbers of shifts, as shown by the following example.

Two optimal solutions with different number of shifts

Consider the instance with

- one position p , and $d_{pp} = 10$.
- $startWork_p = endWork_p = 0$.
- two bus legs $L = \{\ell_1, \ell_2\}$:

$$\ell_1 = (1, 100, 200, p, p),$$

$$\ell_2 = (1, 800, 900, p, p).$$

Consider the two solutions $S_1 = \{\ell_1, \ell_2\}$ and $S_2 = \{\{\ell_1\}, \{\ell_2\}\}$. Using Equation (1.7), we can compute

$$z(S_1) = 800 + 2 \cdot 390 + 180 = 1760$$

$$z(S_2) = 2 \cdot 390 + |\ell_1| + 2 \cdot 390 + |\ell_2| = 1560 + 200 = 1760$$

The two solutions S_1 and S_2 have the **same** objective function value, but **distinct** number of shifts.

1.3 Benchmark Instances

The original instances cannot be publicly shared, due to agreements with the companies. That is why an instance generator was developed, able to generate real-world like instances that follow a similar distribution as the original ones. For our work, we use a set of 50 real-world like instances [?]. There 50 instances are divided in 10 sizes, ranging from around 10 tours to around 100 bus tours. The instance name is `realistic_xx_y` where `xx` is the size of the instance and `y` is the instance number. All instances are publicly available: <https://cdlab-artis.dbai.tuwien.ac.at/papers/sa-bds/>.

In addition to the 50 instances in the literature, we generated 15 new test instances based on real-world-like distributions that range in size from 148 tours (more than 1300 bus legs) to 250 tours (about 2300 bus legs). Instances of this size occur in practice when larger (Austrian) cities are considered in their whole, which presents more potential savings compared to only dealing with subsections of the city.