TRANSPORT PROBLEMS 2022 Volume 17 Issue 1
PROBLEMY TRANSPORTU DOI: 10.20858/tp.2022.17.1.14

Keywords: public transport; driver scheduling; evaluation framework; working time estimation; local public bus passenger transport

János BALOGH[1], Balázs DÁVID[2], Mikós KRÉSZ[3]*, Attila TÓTH[4], László TÓTH[5]

A GENERAL FRAMEWORK FOR EVALUATING DRIVER SCHEDULES IN PUBLIC TRANSPORT

Summary. A unified approach is presented as the principle for a potential framework for evaluating driver schedules in public transport. There are two reasons for the need for such a framework. On the one hand, constructing optimal driver schedules is a challenging problem, and practical solutions cannot be compared to the optimum, which brings about the need to analyse the results with respect to an appropriate estimated value. On the other hand, the specific constraints and rules in different countries and for different companies make it difficult to model solutions in a unified way. Our new approach provides a solution to the above problems by employing an efficient general methodology from both the data modelling and process modelling point of view. By introducing the concept of reducible working time, our approach gives a realistic evaluation framework with an efficient solution method. The applicability of our approach is demonstrated through real-world cases.

1. INTRODUCTION

Public transport is a globally relevant concept responsible for transporting passengers between geographical locations based on a timetable of pre-established routes. This service is provided through a complex system of different activities that involve balancing a wide variety of constraints and considering highly varied optimisation objectives at the same time. Because of complex regulations and high operational costs, the management of this system is an important task for companies, cities and countries. Problems arising in public transport can usually be divided into three categories [1]. Strategic planning is responsible for the design of the transportation network, resulting in fixed routes along which the vehicle will travel. The frequency of these journeys is decided by tactical planning, which provides the timetable for the system. The results of the above phases are then used during operational planning to create schedules for vehicles and drivers and to give daily rosters to be executed.

While the stages of strategic and tactical planning both have a dedicated problem to solve, the three interconnected subproblems of operational planning make this stage even more challenging. The vehicle scheduling problem assigns the timetabled trips of a single day to the fleet of the company. Several characteristics, such as vehicle type and starting depot, can act as constraints for these assignments. Dead runs, during which there are no passengers on a vehicle, are also defined from the stop point of

[1] University of Szeged; Árpád tér 2, H-6720 Szeged, Hungary; email: baloghj@inf.u-szeged.hu; orcid.org/0000-0002-4611-616X
[2] University of Primorska; Glagoljaška 8, SI-6000 Koper, Slovenia; email: balazs.david@famnit.upr.si; orcid.org/0000-0003-4414-4797
[3] InnoRenew CoE; Livade 6, SI-6310 Izola, Slovenia; email: miklos.kresz@innorenew.eu; orcid.org/0000-0002-7547-1128
[4] University of Szeged; Boldogasszony sgt 6, H-6725 Szeged, Hungary; email: toth.attila.02@szte.hu; orcid.org/0000-0003-1014-7965
[5] University of Szeged; Boldogasszony sgt 6, H-6725 Szeged, Hungary; email: toth.laszlo.03@szte.hu; orcid.org/0000-0003-1185-3845
* Corresponding author. E-mail: miklos.kresz@innorenew.eu

one trip to the start point of the next trip if it is different. Vehicle blocks are the solution to this problem as a block corresponding to the work of one vehicle on the given day. All the timetabled trips must be covered by exactly one vehicle block. Driver scheduling is responsible for the assignment of drivers to the timetabled trips considering the regulations for employee work (e.g. working time, breaks). Driver duties are defined (i.e. shifts) to cover the daily trips, such that one duty can be performed by one employee. Since the duties are different on different dates, driver rostering defines the duties that a driver shall perform for each day of a longer planning period (typically, several weeks or months), which also contains rest days. Here, the relevant regulations also have to be maintained (e.g. rest time between duties, number of free days).

The above tasks are hard optimisation problems even when isolated, and their combined solution would usually be too complex to tackle. One solution to this issue is to solve the tasks sequentially, using the output of one task as the input for the next. The most important models and methods of vehicle scheduling have been reviewed by Bunte and Kliewer [2], while Ernst et al. [3] presented an overview of the field of driver scheduling and rostering. Though this sequential solution technique significantly decreases the complexity of developing vehicle and driver schedules and rosters, the efficiency of the solution for the global problem might be inadequate. Moreover, it might even be infeasible if the solution produced in one phase cannot be used to produce a feasible solution in the next. Therefore, integrated methods are also used by which some of the above sub-problems are solved simultaneously. This way, a better global solution can be found. The drawback of this approach is the increased complexity of the integrated problems, which usually lead to models or methods that can only deal with small problem cases. The most common approach in this case is the integration of vehicle and crew scheduling (e.g. [4-6]), but driver scheduling and rostering can also be integrated (e.g. [7]). Another possibility is to solve the problems sequentially while considering the needs of the subsequent phase at every step. For an example, see the approach developed by Árgilán et al. [8], where "driver-friendly" vehicle schedules were constructed that contain possible gaps for driver breaks in the vehicle blocks.

Since the problems of operational planning are complex, heuristic methods are often applied to achieve high-quality results. While finding an optimal (or close-to-optimal) solution is desired, it is also important for the results to be easily implementable in real life. Many studies limit themselves to considering only the most basic constraints of the problems; however, there is usually a massive number of additional regulations that should also be considered if these solutions are to be comparable with the schedules of real companies. These regulations can change from company to company, and different countries can have specific laws (especially labour laws) that affect the organisation of public transport. For the same input instance, two companies would probably create schedules based on different sets of regulations, which would result in solutions that are hard to compare from an efficiency point of view. Over the years, public transport companies and responsible organisations have developed their own definitions, terminologies, optimisation systems, and evaluation methods, which has created a miscellaneous state in the international public transport area. This situation, in turn, has made cooperation difficult and complicated among the actors of this field. Therefore, general and universal terminologies and methods need to be developed to interconnect the different application areas. Moreover, there is a growing need for efficient ways to present the effectiveness of public transport systems [9]. Assessing the actual quality of public transport solutions can be challenging due to the multitude of regulations and constraints used to produce them, and there are no general methods or frameworks that provide a common point of comparison for them. Moreover, it is often hard to determine the efficiency of suboptimal heuristic or approximate solutions, as there might not be an optimum value to use for comparisons, either due to the complexity of the optimal problem or the difficulty of describing company requirements as an objective mathematical function.

This paper presents a universal framework that can provide this common point of comparison for driver schedules. As most of the driver scheduling costs come from working time or arising constraints [10], a method will be presented for calculating a lower bound for working time that considers all the activities in vehicle or driver schedules that can contribute to this value. This lower bound will be general enough that it can be extended to other stages of operational planning with appropriate modifications and provide a bound for integrated approaches. This framework can serve as a universal tool and reference point for evaluating and comparing different solutions for the same datasets.

The public bus passenger transport problem will be used as an example in this paper to demonstrate the usefulness of the above method. However, the method can be adapted to any mode of transport where a set of vehicle journeys is given (e.g. timetabled trips) that have to be performed by drivers. Obviously, the routes, regulations, objectives, and so on may differ, but the concept of the presented evaluation framework remains the same. Therefore, the intention of this work is not to present a specific transport mode or regulation system but simply to provide demonstrative examples to ease comprehension. Concerning terminology, a well-known general standard is used, which was defined by Transmodel [11]. This terminology provides a consistent language that covers many aspects of public transport. It has been developed and enhanced over several decades by public transport experts from many different countries in Europe.

The above goal will be achieved through the following steps. First, the driver scheduling problem will be introduced in detail, and its components that contribute to the working time of drivers will be identified. Then, a method will be proposed for determining a lower bound for these work schedules. After that, the advantage of using the proposed lower bound to evaluate driver schedules and the solution method is expressed based on a comparison of the results of an efficient heuristic algorithm from the literature with the schedules of a public transport company for several real-life input instances.

2. DRIVER SCHEDULING

In the driver scheduling problem, there is a set of input tasks (i.e. the daily tasks that have to be performed by employees). These input tasks are usually vehicle journeys (e.g. timetabled trips or dead runs) but can include other vehicle tasks (e.g. refuelling) if the driver schedule is built around already existing vehicle schedules. These tasks are partitioned into categories called depots. A depot defines the properties that a driver has to possess in order to perform its corresponding tasks. Depots can be physical (e.g. geographical places) or logical (e.g. based on driving licenses or other qualifications). A depot may also have a capacity that cannot be exceeded; for example, there can be a limit on the number of drivers with a specific driving license. Based on driver scheduling, driver duties (i.e. shifts) are defined. All input tasks are included in exactly one duty, and every duty is performed by exactly one driver. These duties have to meet the predefined regulations, a good example of which has been given by Nurmi et al. [12]. The qualifications of employees can also be given dedicated attention, as shown by the review of De Bruecker et al. [13]. The basic problem can be formulated as a set-covering or set-partitioning problem, both of which are well known NP-hard problems [14].

Considering the real-world application of the problem, the regulations of a duty can be defined by different entities (usually the EU, the government, or the transportation company). Some of these regulations are general, but others can be significantly different based on the mode or area of transport. The main goal is to minimise the costs that arise, which can also be defined in different ways. Based on the above, three main features of the driver scheduling that define a specific problem instance are the set of tasks, the regulations, and the cost calculation method. A driver's duties comprise driving tasks, as well as a series of activities that must be performed at given points in time. For example, a duty starts with a check-in, followed by a block of driving tasks (performing trips), a break, additional driving, administration, and, finally, a check-out. Idle time periods can also occur when the driver has nothing to do. Some of these activities are obligatory, while others are potential. For example, the duty has to start with a check-in, but a specific administration has to be done only if the driver leaves a vehicle and starts to work using another one. Some activities count as working time (e.g. driving), but there are non-paid activities too (e.g. longer resting periods when the driver can leave the workplace). Since the distribution of the vehicle tasks throughout a day is not uniform and different activities and rules have to be applied in different cases, the structure of a driver's duties can also vary.

There are usually many requirements that have to be satisfied as a driver carries out their duties. Some of these rules are common (e.g. each vehicle journey has to be included in exactly one duty, and the tasks of a duty must not overlap in time). Other regulations are defined by the local government, work union, or the EU. These are usually connected to driver work (e.g. the maximum working time in a day, minimum resting time, and the frequency and length of breaks). Finally, the public transport company itself may have their own set of rules that are specific to the local environment – for example, different kinds of administrative tasks can be defined. Some rules are hard constraints that have to be satisfied, while others can be soft constraints that can be violated for a cost penalty [15]. For instance, a soft constraint can be that a duty should finish at the same place where it started.

Cost functions for driver scheduling used in the literature are varied and are usually easy to calculate. Most commonly, they are the linear combination of terms such as the number of duties, the total length of the duties, or the penalties of the soft constraint violations. While these functions are suitable for optimisation and provide an estimated cost, they are not appropriate in real life. The cost of the driver schedules of a public transportation company mostly come from employee payments, which is usually the most significant component of the total operational cost [10]. The employees are paid based on their contracts, which define the average working hours per workday in one month. Drivers receive their salary in any case (even if their working hours are below the level prescribed by the contract) and gain extra payment for overtime. Since the lengths of the duties are not equal every day, drivers usually work more than the expected daily working time on some days and less on others. Therefore, in the driver scheduling phase that is executed for one day, costs can only be estimated. The exact cost can be calculated after the rostering phase, which is defined for a relatively long period (usually the contract period). That is why it is important to consider the properties of scheduling for cost calculation that ultimately cause high global costs.

Since employing an extra driver also brings additional fixed costs, many papers have presented solutions that reduce the number of duties; these solutions lead to a lower number of drivers and, consequently, a lower total cost. However, decreasing the number of duties principally means longer daily duties, resulting in extra overtime, the cost of which is usually higher than the basic employment cost. Therefore, reducing the number of duties is not only an appropriate way of minimising the operational cost, but methods should concentrate on decreasing the total working time in the duties.

We have mentioned that duties can contain activities that do not contribute to the paid working time (usually caused by longer idle periods). Furthermore, there are activities that only appear in certain cases depending on the structure of the duty. Typical examples of these potential activities are a driver's travel from the end place of a vehicle journey to a different starting place of the next one or a specific administrative task that has to be executed when changing vehicles. Therefore, in order to minimise cost, the working time has to be reduced by decreasing the number of potential paid activities.

The high variation in regulations and cost calculation techniques usually result in solution methods that might work perfectly for the use-case or application field for which they were designed, but they can be hard to adapt to other real-world problems. A common terminology that allows clear communication and general frameworks in which the differences of the application fields can be handled is needed to overcome this barrier. One such initiative potentially leading to a common terminology is the international standard Transmodel [11], which provides a general model for driver management that defines the actors, the duty parts and a general way to manage driver scheduling in a real-life application.

Frameworks for driver scheduling can also be found in the scientific literature, such as the general framework of Tóth and Krész [16]. This framework defines activity categories and a two-step solution that presents an effective way to develop general solution methods and apply them to different real-life problems. Another framework that considers general employee scheduling problems was presented by Kletzander and Musliu [17]. They proposed a unified method for handling constraints between different application areas of employee scheduling. Solution and simulation frameworks for public transport optimisation also exist, such as the academic algorithm and dataset library LinTim [18].

In addition to the unified terminology and solution methods, the general evaluation of the schedules is also important. Since cost calculations differ noticeably in the literature and in real-life applications, a general method is necessary to measure how good a driver scheduling solution is from a practical point of view.

3. EVALUATION FRAMEWORK FOR DRIVER SCHEDULING

As discussed in the previous section, driver scheduling costs can be calculated in many different ways, and it is hard to compare these methods or to check the effectiveness of a method from a specific environment in another problem case. Solution methods should be evaluated in a unified way to overcome this difficulty. As described in Section 2, the cost of the driver scheduling from a practical point of view depends on the working time in the schedules; thus, the best common metric would be to check the total working time generated by a solution method. Since the structure of a duty may include otherwise non-obligatory activities that count as paid working time, the solution should contain as few of these activities as possible. For example, if a driver changes vehicles during their duties, additional administrative activities are performed every time the driver leaves a vehicle and gets on another one. These activities will not occur if the driver uses the same vehicle for the duration of their schedule. The time requirement of such activities will be referred to as *reducible working time*.

Most of the activities are fixed and have to be included in any solution. Good examples of these are the vehicle tasks that have to be performed by one of the drivers or the obligatory activities for the drivers in each duty, such as checking in and out. The time requirement of such activities will be referred to as *fixed working time*. Therefore, identifying the working time that cannot be reduced in a problem instance is important. Calculating this value can help evaluate solutions more realistically. Comparing the reduction of the working time by a solution method to the total working time is not a correct evaluation. A more realistic and reliable way is to compare only the reducible working time to the total amount (see Fig. 1).

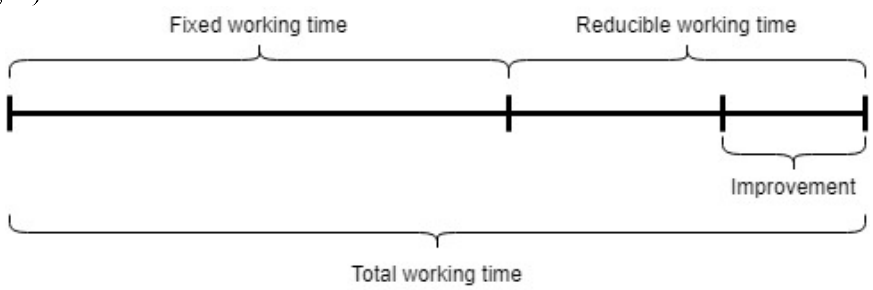

Fig. 1. The reduction of working time compared to the total and the reducible working time

Instead of solving the original driver scheduling problem optimally to get the minimum working time, it is also possible to give a lower bound estimation on it. This lower bound can be calculated by examining all activities that contribute to the working time. These activities can be divided into two classes – namely, *inner activities* and *duty activities*. The first class represents activities that are connected to other activities inside the duty; their time value can be determined based on the corresponding activities. Two examples of this type are the passengers getting on and off the vehicle before and after a timetabled trip or a driver changing vehicles between two vehicle journeys. The properties of these tasks depend on the structure of the duty and the involved vehicle activities. Therefore, the structure of the duties has to be examined to estimate the working time of these activities. The activities of the second class do not depend on other activities and will appear in a duty independently of any other part of the duty. Typical examples of this type are the obligatory checking in and checking out tasks at the beginning and end of the duty or the general daily administration. Since the contributed working times of these activities do not depend on the structure of the duties, the estimation of their total working time can be realised simply by estimating the number of duties. The sets of these activities are application-dependent. Some activities appear in one transport mode while missing from another and vice versa. Alternatively, an activity can be an inner activity in some cases and a duty activity in another. For example, sometimes administrative tasks (filling in documents) have to be done at the beginning (and/or at the end) of the duty immediately after the check-in, while, in other cases, it can be done at any time during the duty.

Since the driver tasks depend on the vehicle activities that must be performed by drivers, the set of input vehicle activities has a strong effect on the working time. When the driver scheduling is solved together with the vehicle scheduling, the working time estimation will be based on the timetabled trips only. This lower bound is general and independent of the vehicle schedules. However, if vehicle scheduling is executed before driver scheduling, extra vehicle tasks are added to the obligatory activities of the drivers (e.g. dead runs, fueling). In this case, the lower bound can be tighter since the extra vehicle activities definitely appear in the final driver schedules; however, this lower bound is specified to the fixed vehicle schedules

3.1. Estimation of inner activity working time

As shown in the previous section, the working time required by inner activities is either directly connected to other activities in the duty (e.g. passengers getting on and off to the timetabled trip) or depend on a sequence of the other activities (e.g. driver changes between two vehicle activities). When estimating the total working time, it is first necessary to construct theoretical duties from the obligatory activities and check the types of connected inner activities that they require. Either the original driver scheduling problem has to be solved or some relaxation is needed to calculate the working time in reasonable time to generate these duties.

A minimum-cost flow relaxation of the driver scheduling problem can be given if only the constraint that forbids overlapping between the activities is considered. If the separate depots in this network are merged into a single depot, the resulting single commodity network flow problem can be solved in polynomial time [19]. Obviously, the solution of the relaxed problem may not be feasible for the original problem, as it will violate some of the original rules that were ignored, but the resulting working time can be used as a lower bound for the original working time.

This flow will be constructed as a connection-based network, a widely used representation of public transport scheduling problems (e.g. [20, 21]), where the nodes represent the possible activities, and edges denote compatibility between them. Thus, the network explicitly represents all possible connections between activities. As the inner activities of duties depend on vehicle tasks and their sequentially, the nodes of this network will represent the input vehicle tasks, while the edges will denote their possible connection (i.e. one vehicle task can directly follow the other in the same duty). Two tasks are connected by an edge only if they are compatible in the original problem (i.e. they can be served from the same depot, and a driver can perform both of them subsequently in the same duty).

The source and sink nodes of the network represent the starting and ending points of the duties. A limit can be set on the maximum number of duties by providing an upper bound on the circulation edge (or the sum of outgoing edges from the source). The working time can be represented as cost on the edges of this network. Since every vehicle task has to be included in a unique duty of the final crew schedule, these activities always provide the same constant added working time. Therefore, only the working time of other activities is taken into consideration as an edge weight. The weight of an edge (i,j) comes from the travelling time of the driver between the ending location of activity i and the staring location of activity j and the total working time of the inner activities that have to be performed between activities i and j. Only the inner activities that are obligatory to perform between these tasks (e.g. passenger getting on before a timetabled trip) are considered.

The working time of driver activities that are not related to vehicle tasks are not considered in this network (e.g. employee breaks that depend on the total working time of the duty). This means that the weight of an edge is a lower bound for the real working time between the two vehicle tasks if they follow each other in one duty in any driver scheduling solution. The working time associated with obligatory activities that have to be performed at the beginning and end of a duty (without a specific location) can be set as a weight on the outgoing edges of the source or the incoming edges of the sink. Administrative activities are typical examples of these.

The inner activities that have to be performed usually depend on the type of vehicle tasks. Therefore, the working time weight on an edge is calculated based on the type of its end-nodes. For example, time has to be allocated before and after a timetabled trip to allow the passengers to get off and on. As a result, getting on and off activities will contribute to their working time as an added edge weight between two timetabled trip nodes. However, no such activity is considered between two dead run nodes (journey without passengers). The inner activities also depend on the time available between the two vehicle tasks. For example, with a short time gap, the crew have to wait on the vehicle, which is measured as work – if the time is long enough, though, the crew may leave the vehicle and rest, which is not working time.

Based on the above concept, a universal weight table can be defined that gives the estimated working time of non-travel activities for the edges depending on the types of their end-nodes and the time required for the corresponding connection. For example, see the Szeged local public bus transport in Table 1, for which there are two vehicle activity types (timetabled trip and dead run) and the time is scaled in minutes. In this example, rows give the time scale and columns the end-nodes of the connections, where *tt* means timetabled trip and *dr* means deadhead.

Obviously, the weight definition method can be different or more sophisticated if the application area requires it to be. The formal definition of the above network is as follows.

Let J be the set of vehicle tasks (timetabled trips, dead runs, and other vehicle activities) in our problem. D_0 represents the starting depot node, and D_1 represent the ending depot node. The flow network $G = (V, E)$ constitutes of nodes $V = J \cup \{D_o, D_1\}$ and edges $E = \{ \{(i,j) \mid i \neq j \in J$ are compatible $\} \cup \{ (D_o, j) \mid j \in J\} \cup \{ (i, D_1) \mid j \in J\} \cup \{ (D_1, D_o)\}$. While the nodes represent every vehicle task and the starting and ending locations of the duties, the edges provide all possible connections between these events (i.e. two nodes will be connected if they can be executed by the same driver within the same duty).

Table 1

An example for estimated working time (in minutes) of edges in Szeged local public bus transportation

	tt – tt	*tt – dr*	*dr – tt*	*dr – dr*
<=5	5	5	5	5
6	6	6	6	0
7	7	7	7	0
8	8	2	2	0
9	9	2	2	0
10	4	2	2	0
>10	4	2	2	0

Based on this network, we can construct an integer programming model for the estimation of the lower bounds of working time. As the edges of G represent all possible connections between the tasks, they provide the x_e, $e \in E$ variables in our model. The corresponding c_e cost of edge e:(i,j) denotes the necessary working time of the tasks that have to be performed between events i and j. For example, if i and j are both timetabled trips, then c_e will account for both the working time of the dead runs between the ending station of i and the beginning station of j. This variable will also account for the working time of any other additional activity needed between the two trips as seen in Table 1. The model is formalised in the following way:

$$minimize \sum_{(i,j)\in E} c_{ij}\, x_{ij}$$

subject to

$$\sum_{(i,j)\in E} x_{ij} = 1, \quad \forall j \in J \tag{1}$$

$$\sum_{i:(i,j)\in E} x_{ij} - \sum_{i:(j,i)\in E} x_{ji} = 0, \quad \forall j \in V \tag{2}$$

$$\sum_{j:(D_0,j)\in E} x_{D_0 j} \le k_u \tag{3}$$

$$x_{ij} \in \{0,1\}, \forall (i,j) \in E \backslash \{(D_1, D_0)\} \tag{4}$$

$$x_{D_1 D_0} \ge k_l \text{ and integer} \tag{5}$$

Constraint (1) of the above model guarantees that every vehicle task is executed exactly once. Constraint (2) ensures flow conservation for all nodes in the network. Constraint (3) provides an upper bound for the number of duties in the solution of the model, while constraints (4) and (5) are binary and integrality constraints for task edges and the circulation edge, respectively. Constraint (5) also sets a lower bound for the circulation edge, which has to be at least 0 to guarantee non-negativity.

3.2. Estimation of the minimum number of duties

The method in Section 3.1 ignores all regulations about the limitation of the working time of the employees. Therefore, it could be that the resulting duties are too long and dense to be allowed for drivers. Such a solution would usually result in an unrealistically low number of duties. However, the total working time comes from not only the inner activities but also the duty activities, and the number of duties is an important factor. The duty activities depend on the duty of interest and not on any other activities. These types of activities may not have an exact place or time at which the driver performs them, but they must be performed once (or as many times as prescribed) within each duty. Typical activities of this type are administrative tasks.

The number of these activities depends on the number of duties. Therefore, to estimate the total working time, it is necessary to calculate the minimum number of duties needed to cover the vehicle tasks. Two factors affect the duty number: the minimal number of duties in any time window of the day and the maximal time length of the duty defined by the regulations.

Since the activities in any duty cannot overlap (e.g. a driver cannot drive more than one vehicle at the same time), the maximal number of overlapping vehicle tasks in each minute gives a lower bound for the number of duties. However, some inner activities connected to the vehicle tasks require the driver to be present. For example, the driver is not allowed to leave the vehicle after a timetabled trip until the passengers get off. Therefore, the time of these activities extends the starting and ending times of the vehicle tasks. Furthermore, if a vehicle journey ends in a different place than the next journey starts, the driver needs time to travel from the previous place to the next one. If the gap between two vehicle journeys is not sufficient, they cannot be part of the same duty. Considering the overlapping of extended vehicle tasks by the attached inner activities and the travelling time, the minimal number of duties can be calculated in any time window in a day. This problem can be solved in polynomial time.

This problem also can be formulised as a minimal cost network flow problem. Here, the nodes are the vehicle tasks in the corresponding time window (any part of the vehicle task intersects the time window), and their starting and ending times are extended by the time required to complete the associated inner activities. Furthermore, edges are added to the network graph only when there is enough time to travel between the places of the vehicle tasks. The weights of the edges from the source node are 1, while the weights of any other edges are 0. Solving this problem by minimising the number of outgoing source edges gives the minimum number of duties needed in the given time window.

Since the regulations for employees define the maximal working time and minimal resting time in a day, the duties have a maximal length limit called the *service time*. Usually, the vehicle tasks on a day consume more time than the maximal duty length. Therefore, the duties started in the early morning have to be finished before the end of the day, and new duties have to start to cover the evening tasks.

The following method is used to calculate the minimum number of duties that cover all the vehicle tasks in a day. Define time interval I (the length of which is equal to the maximal duty length). Interval I will contain all vehicle tasks on a given day that start within its boundaries. As was discussed before, such an interval will probably not cover all the vehicle tasks to be completed that day. Let the time interval B contain all the tasks starting before I on the same day, and let interval A contain the tasks starting after I in a similar way. Calculate the minimal number of duties in each interval and label them as n_I, n_B, n_A correspondingly. While n_I, n_B and n_A are lower bounds for the minimum number of drivers, a stronger bound can be given. If a driver works in interval B, they cannot work in A at the same time because of the maximal duty length rule. As a result, the minimal number of drivers will be $n = max(n_I, n_B+n_A)$. If the starting point of I is iterated through each minute of the day, the minimal number of drivers in iteration i is n_i. Then, the minimal number of drivers required on that day is $\max_i(n_i)$.

3.3. Calculating a lower bound for the total working time

Using the methodology described in Sections 3.1 and 3.2, we give the method for estimating a lower bound of the total working time of driver schedules in this section. The outline of this approach can be seen in Fig. 2.

The following information is required as the input for the calculation method:

(1) The set of vehicle tasks (with starting and ending times and places, depots, and so on). These vehicle tasks can either be only timetabled trips derived directly from the timetable of the input, or they can contain other vehicle activities (e.g. dead runs). In the latter case, our set of vehicle tasks is extracted from an existing vehicle schedule or assignment.
(2) The set of travelling times between any two geographical places where vehicle tasks take place.
(3) The estimated working time of inner activities between the vehicle task types, see Section 3.1 and Table 1.
(4) The sum of the working time of the duty activities performed in each duty.
(5) The upper capacity limit for the number of duties (i.e. drivers)
(6) The maximum duration of the duty.

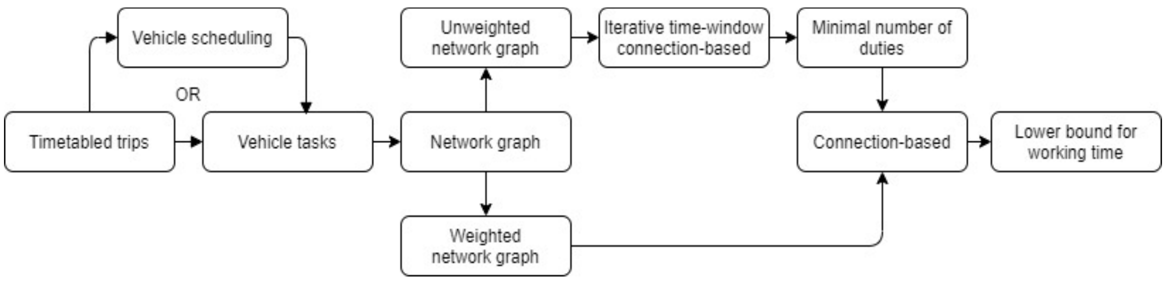

Fig. 2. The method for calculating the lower bound of working time

The first step in calculating the lower bound for the working time is to determine the minimal number of required drivers for the day using the method in Section 3.2. Then, a network graph is built from the input vehicle activities (1) described in Section 3.1. The weights of the edges come from the travelling time values (2) and the inner activity working timetable (3). The weights of the source edges are derived from the duty activity working times (4). The upper capacity of the circulation edge is then set to the input value (5), and the lower capacity bound is set to the minimum number of required drivers according to the previous step. Finally, the network flow problem is solved. The resulting flow value is the lower bound for the total working time of the given problem.

4. NUMERICAL EXPERIMENTS

The advantage of using the introduced lower bound to evaluate solutions is shown by the efficient driver scheduling method Cut and Join (CAJ), published by Tóth and Krész [22]. This method was tested on real-life data from Szeged (a middle-sized city in Hungary). The data was taken from a local bus passenger transport system; the routes (length and topology), the timetable, the frequency of the trips and the regulations for the driver work are typical of a European city. The results were compared to the schedules of the local public bus passenger transport company for 11 distinct inputs (corresponding to 11 different day types, such as weekdays and holidays).

The size of the input data (i.e. the number of vehicle activities and their total working time), the calculated lower bound, the working time of the schedules of the bus company, and the working time of the schedules of the CAJ method are presented in Table 2. The lower bounds are shown both for the timetable trips and for the vehicle activities of the previously generated optimal vehicle scheduling using a time-space network model [23].

The CAJ method can be evaluated using the results of Table 3. It can be seen that the improvements in working time of the CAJ method compared to the total working time of the company (columns Imp Comp) are always less than 3%. This might not seem like a significant improvement. However, the reducible working times (columns RWT) are less than 10% of the company's scheduling, which gives little room for improvement to any optimisation method. When only the reducible parts (columns Imp RWT) are considered when comparing the working times of CAJ compared to the company working times, the average improvements are 20.2% based on the timetable's lower bounds and 32.6% based on the vehicle scheduling lower bounds. This is a more correct and realistic evaluation that shows the efficiency of the CAJ method much better than simply comparing the objective values.

Table 2

The number of vehicle tasks (NumTr and NumTa), the total task time (TrT and TaT), the lower bounds (LB) for working time, the working time of the company (Comp), and the working time of an optimised driver scheduling system (CAJ) for 11 day types (DT); times are given in minutes

	Timetable			Vehicle scheduling				
DT	NumTr	TrT	LB	NumTa	TaT	LB	Comp	CAJ
Day_01	1646	33922	44346	1933	34952	45726	48879	47955
Day_02	1765	36283	47351	2072	37419	48936	52124	51650
Day_03	1981	41219	54159	2325	42438	55803	59063	58584
Day_04	1981	41235	54175	2323	42494	55848	59007	58837
Day_05	2687	56257	75228	3247	58489	78128	82565	80505
Day_06	2717	56794	75976	3283	59061	78889	83559	81908
Day_07	2720	56850	76044	3284	59118	78988	83570	81790
Day_08	2720	56860	76056	3294	59193	79069	83607	81577
Day_09	2721	56880	76080	3304	59170	79053	83604	81972
Day_10	2721	56882	76082	3283	59091	79047	83627	81408
Day_11	2721	56882	76080	3285	59142	79038	83591	81558

Table 3

Comparison (in percentages) for timetable and the vehicle scheduling cases considering the reducible working time of the company scheduling (RWT), the improvement of CAJ compared to the company working time (Imp Comp) and to the reducible working time (Imp RWT)

	Timetable			Vehicle scheduling		
DT	RWT	Imp Comp	Imp RWT	RWT	Imp Comp	Imp RWT
Day_01	9,27%	1,89%	20,38%	6,45%	1,89%	29,31%
Day_02	9,16%	0,91%	9,93%	6,12%	0,91%	14,87%
Day_03	8,30%	0,81%	9,77%	5,52%	0,81%	14,69%
Day_04	8,19%	0,29%	3,52%	5,35%	0,29%	5,38%
Day_05	8,89%	2,50%	28,08%	5,37%	2,50%	46,43%
Day_06	9,08%	1,98%	21,77%	5,59%	1,98%	35,35%
Day_07	9,01%	2,13%	23,65%	5,48%	2,13%	38,85%
Day_08	9,03%	2,43%	26,88%	5,43%	2,43%	44,73%
Day_09	9,00%	1,95%	21,69%	5,44%	1,95%	35,86%
Day_10	9,02%	2,65%	29,41%	5,48%	2,65%	48,45%
Day_11	8,99%	2,43%	27,07%	5,45%	2,43%	44,65%

5. CONCLUSION

We have introduced a general evaluation framework for driver schedules in public transport. Our unified approach is based on a lower bound technique for the estimated required minimal working time using network flow modelling. Since the structure of the input determines a fixed working time, our approach focuses on the new concept of reducible working time to obtain a more realistic evaluation of driver schedules. We have developed an efficient modelling and solution methodology for calculating this value. The framework is flexible; its data model does not depend on special national or company rules, but special rules and activities related to driver duties can be easily integrated. To present the advantage of our methodology, we have demonstrated the effective use of our approach by testing on real-life data. While our approach is fundamental for the evaluation framework, there is room for further improvement. One potential approach is to extend the lower bound technique for shorter timeslots (even if only by minutes) by constructing a temporal lower bound curve and providing an analysis framework for evaluating the solutions with their dynamics.

Acknowledgements

János Balogh was supported by the projects "Extending the activities of the HU-MATHS-IN Hungarian Industrial and Innovation Mathematical Service Network" EFOP-3.6.2-16-2017-00015.

Balázs Dávid was supported in part by the University of Primorska postdoc grant No. 2991-5/2021.

Balázs Dávid and Miklós Krész gratefully acknowledge the European Commission for funding the InnoRenew CoE project (Grant Agreement #739574) under the Horizon2020 Widespread-Teaming program, as well as the Republic of Slovenia (Investment funding of the Republic of Slovenia and the European Union of the European Regional Development Fund). They are also grateful for the support of the Slovenian Research Agency (ARRS) through grants N1-0093 and BI-AT/20-21-014.

László Tóth was supported by the project "Integrated program for training new generation of scientists in the fields of computer science", no. EFOP-3.6.3-VEKOP-16-2017-0002 (supported by the European Union and co-funded by the European Social Fund)

The authors acknowledge the work of Szabolcs Dávid, who was supported by "Integrated program for training new generation of scientists in the fields of computer science", no. EFOP-3.6.3-VEKOP-16-2017-0002 (supported by the European Union and co-funded by the European Social Fund).

The Authors also acknowledge the support of the Dél-alföldi Közlekedési Központ.

References

1. Ibarra-Rojas, O.J. & Delgado, F. & Giesen, R. & Muñoz, J.C. Planning, operation, and control of bus transport systems: A literature review. *Transportation Research Part B: Methodological*. 2015. Vol. 77. P. 38-75.
2. Bunte, S. & Kliewer, N. An overview on vehicle scheduling models, *Journal of Public Transport.* 2009. Vol. 1. No. 4. P. 299-317.
3. Ernst, A.T. & Jiang, H. & Krishnamoorthy, M. & Sier, D. Staff scheduling and rostering: A review of applications, methods and models. *European Journal of Operational Research*. 2004. Vol. 153. No. 1. P. 3-27.
4. Mesquita, M. & Paias, A. & Respício, A. Branching approaches for integrated vehicle and crew scheduling. *Public Transport.* 2009. Vol. 1. P. 21-37.
5. Steinzen, I. & Gintner, V. & Suhl, L. & Kliewer, N. A time-space network approach for the integrated vehicle-and crew-scheduling problem with multiple depots. *Transportation Science*. 2010. Vol. 44. No. 3. P. 367-382.
6. Békési, J. & Nagy, A. Combined vehicle and driver scheduling with fuel consumption and parking constraints: a case study. *Acta Polytechnica Hungarica*. 2020. Vol. 17. No. 7. P. 45-65.
7. Borndörfer, R. & Schulz, C. & Seidl, S. & Weider, S. Integration of duty scheduling and rostering to increase driver satisfaction. *Public Transport.* 2017. Vol. 9. P. 177-191.
8. Árgilán, V. & Balogh, J. & Békési, J. & Dávid, B. & Krész, M. & Tóth, A. Driver scheduling based on "driver-friendly" vehicle schedules. *Operations Research Proceedings.* 2011. P. 323-328.
9. Vdovychenko, V. Formation of methodological levels of assessing city public passenger transport efficiency. *Eastern-European Journal of Enterprise Technologies*. 2016. No. 3(81). P. 44-51.
10. Asvin, G. Vehicle scheduling and routing with drivers' working hours. *Transportation Science*. 2009. Vol. 43. No. 1. 2009. P. 17-26.
11. CEN- EN12896-1:2016. *Public transport - Reference data model - Part 1: Common concepts*. Comité Européen de Normalisation. 132 p.
12. Nurmi, K. & Kyngäs, J & Post, G. Driver rostering for bus transit companies. *Engineering Letters*. 2011. Vol. 19. No. 2. P. 125-132.

13. De Bruecker, P. & Van den Bergh, J. & Beliën, J. & Demeulemeester, E. Workforce planning incorporating skills: State of the art. *European Journal of Operational Research*, 2015. Vol. 243. No. 1. P. 1-16.
14. Garey, M.R. & Johnson, D.S. *Computers and Interactability: A Guide to the Theory of NP-Completness*. Freeman, San Fransisco. 1979.
15. Demirović, E. & Musliu, N. & Winter, F. Modeling and solving staff scheduling with partial weighted maxSAT. *Annals of Operations Research*. 2019. Vol. 275. P. 79-99.
16. Tóth, A. & Krész, M. A flexible optimization framework for driver scheduling, *Proceedings of the 11th International Symposium on Operational Research SOR '11*. 2011. P. 341-346.
17. Kletzander, L. & Musliu, N. Solving the general employee scheduling problem. *Computers & Operations Research*. 2020. Vol. 113. P. 1-13.
18. Schiewe, A. & Albert, S. & Schiewe, P. & Schöbel, A. & Spühler, F. LinTim: An integrated environment for mathematical public transport optimization. Documentation for version 2020.12. TU Kaiserslautern, 2020. Available at: https://nbn-resolving.org/urn:nbn:de:hbz:386-kluedo-62025.
19. Ahuja, R.K. & Magnanti, T.L. & Orlin, J.B. *Network Flows: Theory, Algorithms, and Applications*. Prentice-Hall, Inc. 1993.
20. Bertossi, A.A. & Carraresi, P. & Gallo, G. On some matching problems arising in vehicle scheduling models. *Networks*. 1987. Vol. 17. P. 271-281.
21. Löbel, A. Vehicle scheduling in public transit and lagrangean pricing. *Management Science*. 1998. Vol. 44. No. 12. P. 1637-1650.
22. Tóth, A. & Krész, M. An efficient solution approach for real-world driver scheduling problems in urban bus transportation. *Central European Journal of Operations Research.* 2013. Vol. 21. No. 1. P. 75-94.
23. Kliewer, N. & Mellouli, T. A time–space network based exact optimization model for multi-depot bus scheduling. *European Journal of Operational Research*. 2006. Vol. 175. P. 1616-1627.

Received 16.01.2021; accepted in revised form 14.03.2022