ARCHIVES OF TRANSPORT *ISSN (print):* *0866-9546*
Volume 65, Issue 1, 2023 ***e-ISSN (online):*** *2300-8830*
DOI: *10.5604/01.3001.0016.2475*

DETERMINING LOWER BOUND ON NUMBER OF VEHICLE BLOCKS IN MULTI-DEPOT VEHICLE SCHEDULING PROBLEM WITH MIXED FLEET COVERING ELECTRIC BUSES

Jerzy DUDA[1], Szymon FIEREK[2], Marek KARKULA[3], Piotr KISIELEWSKI[4], Radosław PUKA[5], Adam REDMER[6], Iwona SKALNA[7]

[1, 3 ,5, 7] *AGH University of Science and Technology, Kraków, Poland*

[2, 6] *Poznan University of Technology, Poznań, Poland*

[4] *Cracow University of Technology, Kraków, Poland*

Abstract:

Scheduling buses in public transport systems consists in assigning trips to vehicle blocks. To minimize the cost of fuel and environmental impact of public transport, the number of vehicle blocks used should be as small as possible, but sufficient to cover all trips in a timetable. However, when solving real life transportation problems, it is difficult to decide whether the number of vehicle blocks obtained from an algorithm is minimal, unless the actual minimal number is already known, which is rare, or the theoretical lower bound on the number of vehicles has been determined. The lower bound on the number of vehicle blocks is even more important and useful since it can be used both as a parameter that controls the optimization process and as the minimum expected value of the respective optimization criterion. Therefore, methods for determining the lower bound in transportation optimization problems have been studied for decades. However, the existing methods for determining the lower bound on the number of vehicle blocks are very limited and do not take multiple depots or heterogeneous fleet of vehicles into account. In this research, we propose a new practical and effective method to assess the lower bound on the number of vehicle blocks in the Multi-Depot Vehicle Scheduling Problem (MDVSP) with a mixed fleet covering electric vehicles (MDVSP-EV). The considered MDVSP-EV reflects a problem of public transport planning encountered in medium-sized cities. The experimental results obtained for a real public transport system show the great potential of the proposed method in determining the fairly strong lower bound on the number of vehicle blocks. The method can generate an estimated distribution of the number of blocks during the day, which may be helpful, for example, in planning duties and crew scheduling. An important advantage of the proposed method is its low calculation time, which is very important when solving real life transportation problems.

Keywords: *vehicle scheduling, lower bound, public transport, mixed fleet, electric buses*

To cite this article:

Duda, J., Fierek, Sz., Karkula, M., Kisielewski, P., Puka, R., Redmer, A., Skalna, I., (2023). Determining lower bound on number of vehicle blocks in multi-depot vehicle scheduling problem with mixed fleet covering electric buses. Archives of Transport, 65(1), 27-38. DOI: https://doi.org/10.5604/01.3001.0016.2475

Contact:
1) [https://orcid.org/0000-0002-9225-7123]; 2) [http://orcid.org/0000-0003-1076-2722]; 3) mkarkula@zarz.agh.edu.pl [https://orcid.org/0000-0001-9468-1529] – corresponding author; 4) [https://orcid.org/0000-0003-0013-5477]; 5) [https://orcid.org/0000-0002-3201-0735]; 6) [https://orcid.org/0000-0003-2154-232X]; 7) [https://orcid.org/0000-0001-5707-7525]

1. Introduction

Vehicle scheduling problem (VSP), also called "blocking", is an important part of the public transport planning and describes the process of assigning the trips of a timetable to vehicle blocks. Given a timetable, the quality of vehicle schedules is often measured by the minimum number of vehicles required to cover all trips (Ceder, 2007), which translates directly to the environmental impact of a public transport, and to fixed and labor costs. In practice, vehicle scheduling is usually considered under different requirements, like the existence of more than one depot, a heterogeneous fleet with multiple vehicle types, the permission of trips variable departure times and further restrictions on the trips of the buses (Bunte and Kliewer, 2009).

The blocks in the basic VSP are defined as sequences of trips made by one vehicle from an initial or home depot to the same depot. The objective is to minimize the operational costs of vehicles usage. Solving this problem is of great importance for public transport operators. The most important (often depending on the company) features that define the structure of the VSP can include:

- number of depots – two basic cases are distinguished here: single depot (SD VSP) or multiple depots (MD VSP) – MD VSP has been proven to be NP-hard (Lübbecke et al. 2010);
- number of line trips – this parameter is determined by a timetable and is usually constant; in large public transport systems the number of trips may exceed 5,000;
- multiple vehicle types and their assignment to individual depots; in practice, some additional restrictions may arise, e.g., on the allocation of specific vehicle types to lines/trips;
- limitations on depots and/or charging points capacities – number of possible vehicles charged at the time;
- parameters and constraints relating to different duration of layovers between two service trips;
- minimum and maximum block length;
- preferences and restrictions for changing lines within a block;
- other parameters specific to in-company constraints.

In this paper, we consider the Multi-Depot Vehicle Scheduling Problem (MDVSP) with a mixed fleet covering electric buses. The MDVSP problem is considered to be NP-hard (Kisielewski, 2019; Olsen et al., 2022; Perumal et al., 2022; Zhang et al., 2022), therefore, computational times of exact methods can be very high, even for instances of a moderate size. One of the possible ways to improve performance of the optimization process is to find a rigorous bound (lower bound (LB) for minimization problems and upper bound (UB) for maximization problems) on the optimal solution. The computed bound can be used to guide the search for efficient solutions or to evaluate the quality of approximate solutions. Methods for determining the lower bound in transportation optimization problems have been studied for decades. However, the existing methods are very limited and do not take multiple depots or heterogeneous fleet into account, and LB estimation time for larger problems is long (strongly nonlinear wrt the size of the problem). In this paper, we propose an efficient method for computing the lower bound on the number of vehicle blocks in MDVSP with a mixed fleet covering electric buses.

The rest of the paper is organized as follows. Section 2 presents an overview of the related literature. The considered problem is defined in Section 3 and the proposed method is introduced in Section 4. Section 5 is devoted to experimental results. The paper ends with concluding remarks.

2. Literature review

The lower bound (LB) is used within Branch-and-Bound (B&B) methods to prevent exploration of suboptimal regions of the search space. The typical techniques to find LBs within B&B include Linear Programming relaxation (LP relaxation) (Caprara et al. 1999), column generation (Lübbecke et al. 2010) and Lagrangian Relaxation (LR) (Fisher, 2004). The column generation technique adds cutting planes to the dual optimization problem to improve the computed LB (Morrison, 2016).

Akker et al. (2002) find that for many NP-hard combinatorial minimization problems the strong LB on the optimal solution can be computed by formulating the problem as an integer linear program with a huge number of variables and then solving the linear programming relaxation through a column generation method. This approach has led to state-of-the-art B&B algorithms for many combinatorial optimization problems. Lagrangian relaxation, in turn, proceeds by dualizing a subset of constraints. This method results in a LB that dominates the one determined by the column generation method. However, column generation shows better convergence and, therefore, is usually faster (Akker et al., 2002).

Depending on the nature of an optimization problem, the LB has different uses. Byung-In et al. (2012) calculated the LB for the VSP (defined as an assignment problem) using the Hungarian method with a special setting of cost between two trips. These LBs can provide reference to evaluate the solution quality of the heuristic methods. Based on the experimental results, the LB provides tight bounds for the small problems. However, its tightness for large problems has not been fully evaluated yet. The gap between the LB and the solution of the heuristic methods increases as the problem size increases.

An interesting investigation was presented by Ribeiro and Soumis (1994). They used LB in Multi Depot Vehicle Scheduling (MD-VSP) problem defined as an integer programming and found the relationship between the bounds obtained in different ways, i.e., by the assignment relaxation, the shortest path relaxation, the additive technique, Lagrangian decomposition, column generation and linear programming. The conclusion from this research is that the linear relaxation of MD-VSP provides a stronger LB than does the additive bound. The additive bound technique cannot provide tighter bounds than those obtained by Lagrangian decomposition and not better than the linear programming bound in the case of the MD-VSP. The Column Generation Bound is at least as good as the additive one. The quality of the LBs provided by the column generation is very good in practice and does not deteriorate with the increase in the number of trips or depots (Ribeiro and Soumis, 1994).

Bunte and Kliewer (2009) present the different qualities of the lower bounds obtained by the different modeling approaches for the MD-VSP. They reported that lower bound obtained by the LP solution of the single-commodity model with subtour breaking constraints is smaller or equal to the single-commodity model with assignment variables (Mesquita and Paixão,1999). Both single-commodity formulations provide weaker LP bounds than the (connection-based) multi-commodity flow formulation (Mesquita and Paixão, 1999). The LP bound of the multi-commodity model and the set partitioning model have the same value.

Freling et al. (2003) utilized both column generation as well as Lagrangian relaxation to generate LB for integrated vehicle and crew scheduling problem. As they found LB is on average approximately 10% of the number of timetabled trips.

As apparent from the comparison of results MD-VSP made by Wen et al. (2016), the results of MD-VSP often do not provide a tight LB to the E-VSP (criterion function – cost). This is caused by the restricted driving range in the E-VSP that forces vehicles to recharge which increases deadheading time and decreases vehicle utilization. In their opinion, a better LB for the E-VSP can be investigated, which is not trivial due to a high complexity introduced by the partial charging.

Stern and Ceder (1983) applied LB to the minimum fleet size problem in public transport – the coverage of scheduled trips with and without technical/deadhead trips. They constructed step functions (deficit functions) for each terminal in the schedule representing the net number of trip departures fewer than arrivals up to each time. The maximum value of the overall deficit function represents a LB on the minimum fleet size. This lower bound was even further improved by Ceder (2002) by looking into artificial extensions of certain trip-arrival points without violating the generalization of requiring all possible combinations for maintaining the fleet size at its lower bound. Ceder also used the same technique in other papers (Ceder, 2007; Ceder, 2011).

3. Problem definition

Given a set of timetabled trips with fixed travel (departure and arrival) times and start and end locations as well as traveling times between all pairs of depots, the objective is to find an assignment of trips to vehicles such that:

– each trip is covered exactly once,
– each vehicle performs a feasible sequence of trips,
– the goal function is minimized.

A vehicle “block” is the schedule of travel of a vehicle for a given day, including: (1) a pull-out from the depot, (2) a sequence of trips from the timetable, (3) any dead-head trips, and (4) a pull-in back to the depot (Ceder, 2007; Perumal et al., 2022).

Scheduling of vehicle blocks is a special case of multi-depot vehicle scheduling problem (MD-VSP) where, usually, the goal is to minimize the total cost of trips including deadhead trips. The subject literature describes several different approaches to solve MD-VSP. The first group of models include single-commodity models in which a graph represents one node per trip and additional nodes represent vehicles as in Single-Commodity Model with Subtour Assignment Variables Breaking Constraints (Carpaneto, 1989) or depots as in Single-Commodity Model with Assignment Variables (Mesquita and Paixão, 1992). Another group of models are Multi-Commodity models which, usually, extend the network-flow model for SD-VSP (Bodin et al., 1983). In such models, a graph has two types of arcs: arcs denoting that the next trip can be serviced after the previous trip by the same vehicle block (relational arcs) and deadhead arcs (pull-in and pull-out). The Connection-Based Networks and Time-Space Networks models can be used as well. In the Connection-Based Networks model, the possible connections between the timetabled trips are modeled by considering all trip compatibilities explicitly (Bertossi et al., 1987; Lamatsch, 1992), whereas in the Time-Space Networks model, possible connections between groups of compatible trips are aggregated (Kliewer et al., 2002; Kliewer et al., 2006). In the Set Partitioning model, in turn, all feasible routes for the vehicles are enumerated and then only those routes that fulfill all restrictions are considered (Ribeiro and Soumis, 1994). This model is used with a column generation algorithm.

Our model, presented below, uses the currently most widely used Multi-Commodity with Connection-Based Networks approach.

In Duda et al. (2022), we presented a complex model that consists of an objective function that is a weighted combination of five components and constraints that precisely check whether there is enough state of charge (SoC) left so that the bus can perform a specific trip and taking into account the necessary recharging of the battery. In this work, we focus on the most important, from the business point of view, component of the objective function, i.e., on the minimization of the number of blocks used to cover all trips. Therefore below we present a single-criterion version of this model. Contrary to most of the models presented in the literature, the considered model takes into account, however, a heterogeneous fleet of buses (bus types) including electric ones. The nomenclature used to define the optimization model is presented in Table 1.

A mixed integer linear programming model for the considered MD-VSP with a heterogeneous fleet covering electric buses can be formulated as follows:

$$\text{minimize} \sum_{k\in K} Z_k \tag{1}$$

subject to:

$$\sum_{k\in K}\sum_{j:(i,j)\in A} X_{ijk} = 1, \qquad \forall i \in T \tag{2}$$

$$\sum_{j:(i,j)\in A} X_{ijk} - \sum_{j:(j,i)\in A} X_{jik} = 0, \qquad \forall i \in T, \forall\, k \in K \tag{3}$$

$$\sum_{\delta\in D\setminus\{\Delta(k)\}}\sum_{i\in V} X_{a^{\delta}ik} = 0,\ \ \forall\, k \in K \tag{4}$$

$$\sum_{\delta\in D\setminus\{\Delta(k)\}}\sum_{i\in V} X_{ib^{\delta}k} = 0, \qquad \forall k \in K \tag{5}$$

$$MZ_k - \sum_{(i,j)\in A} X_{ijk} \ \ \geq 0, \forall k \in K \tag{6}$$

$$\sum_{k\in K,\Delta(k)=\delta,\Pi(k)=\pi} Z_k \leq c_{\pi}^{\delta}, \quad \forall \delta \in D, \forall \pi \in P \tag{7}$$

$$X_{ijk} \in \{0,1\}, \quad \forall (i,j) \in A, k \in K \tag{8}$$

$$Z_k \in \{0,1\}, \quad \forall k \in K \tag{9}$$

The objective function (1) takes into account only the number of blocks to cover all trips, which in many cases is the most important business criterion. The constraints (2)– (9) have the following meaning:

- constraint (2) ensures that each trip is operated by only one vehicle,
- constraint (3) conserves the flow,
- constraint (4) states that each vehicle starts at a designated depot,
- constraint (5) means that each vehicle ends at a designated depot,
- constraint (6) specifies the use of the block/vehicle,
- constraint (7) limits the number of vehicles type p available at the depot δ,
- constraints (8) and (9) define the domains of the variables.

Table 1. Nomenclature used to define optimization model

Indices	
i	node representing the service trip or the starting point of a deadhead trip
(i,j)	trip connection arc that connects two consecutive service trips or service trip and a deadhead trip
k	vehicle block
δ	depot
π	bus type
t	time period (here minute)
h	service trip
w	break (waiting time in minutes) between trips
Sets	
V	set of all nodes in the trip graph (depots, trip endpoints, charging places, etc.)
T	set of trip nodes
A	set of all edges of the transport graph; there are four types of edges: (i) A_1 – connecting depots with service trips (pull-out trips), (ii) A_2 – connecting two service trips, (iii) A_3 – connecting service trips with endpoint depots (pull-in trips), (iv) A_4 – connecting service trips with charging point
D	set of depots
P	set of vehicle types
K	set of vehicle blocks
τ	set of time periods (here minutes)
H	set of service trips
Parameters	
d_{ij}	length of deadhead trip on arc (i,j) [km]
t_{ij}	duration of deadhead trip on arc (i,j) [min.]
z_i	scheduled time of departure for trip i
c_{π}^{δ}	number of available vehicles of type π in depot δ
h	minimal working time of vehicle block [min.]
M	constant, very large positive number
$\Pi(k)$	vehicle type for vehicle block k
$\Delta(k)$	home depot for vehicle block k
a^{δ}	begin node (depot)
b^{δ}	end node (depot)
W	break (waiting time) between trips
Variables	
X_{ijk}	binary decision variable, $X_{ijk} = 1$, if vehicle block k carries out the deadhead trip on arc $(i,j) \in A$ or carries out trip j after trip i
Z_k	auxiliary binary variable, $Z_k = 1$, if vehicle block k is used in the schedule

4. Method for Computing Lower Bound on Number of Vehicles

In our considerations, we assume that the vehicle blocks start their trips at the beginning of a minute and end their trips at the end of a minute. This means that if one trip starts at the minute *zm* and the other ends at that minute, then two vehicle blocks are needed to service these two "overlapping" trips.

Given the above assumption, the lower bound (LB) on the number of vehicle blocks can be determined from the formula:

$$LB_{\alpha} = max_{t=1,\dots,\tau}\left(\sum_{h\in H} x_{h,t}\right) \quad (10)$$

where τ is the number of minutes in the considered time horizon, H is the set of all service trips to be covered, and $x_{h,t}$ is a binary variable that takes the following values:

$$x_{h,t} = \begin{cases} 1 - \text{if } h\text{-th trip is serviced in minute } t \\ 0 - \text{otherwise} \end{cases}$$

The formula (10) is very simple and intuitive. However, it takes into account only those vehicles that in the given minute are during the trip, start the trip or end the trip, whereas there are also vehicles that wait for their trips, they should also be taken into account.

The LB for the number of vehicles that wait for their trips can be computed as follows:

$$LB_\beta = \max_{t=1,\ldots,\tau-1} \left(\max\left(\sum_{h\in H} s_{h,t+1} - \sum_{h\in H} f_{h,t}, 0\right)\right), \quad (11)$$

where $s_{h,t+1}$ and $f_{h,t}$ are binary variables that take, respectively, the following values:

$$s_{h,t+1} = \begin{cases} 1 - \text{if } h\text{-th trip starts in minute } t+1 \\ 0 - \text{otherwise} \end{cases}$$

$$f_{h,t} = \begin{cases} 1 - \text{if } h\text{-th trip ends in minute } t \\ 0 - \text{otherwise} \end{cases}$$

The formula (11) allows zero breaks which may not be in accordance with the requirements set by some enterprises, related to the minimum break time or which may result, for example, from the time necessary to board the vehicle by the passengers. Therefore, we proposed a modification of the formula (11) which takes into account breaks:

$$LB_{\beta,W} = max_{t=W+1,\ldots,\tau-1} \left(\max\left(\sum_{h\in H} s_{h,t+1} - \sum_{h\in H} f_{h,t}, 0\right) + \sum_{w=1}^{W} \sum_{h\in H} f_{h,t-w}\right), \quad (12)$$

where W is the length of a break (waiting time) given in minutes. The value of W can be set as a parameter of the method.

To make the value of $LB_{\beta,W}$ more precise, the formula (12) can be further extended by taking into account loops/depots where trips start or end:

$$LB_{\beta,I}^{D} = \max_{t=W+1,\ldots,\tau-1} \left(\sum_{\delta\in D}\left(\max\left(\sum_{h\in H} s_{h,t+1}^{\delta} - \sum_{h\in H} f_{h,t}^{\delta}, 0\right) + \sum_{w=1}^{W} \sum_{h\in H} f_{h,t-w}^{\delta}\right)\right), \quad (13)$$

where D is the set of loops/depots, and $s_{h,t+1}^{\delta}$ and $f_{h,t+1}^{\delta}$ are binary variables that take the following values:

$$s_{h,t+1}^{\delta} = \begin{cases} 1 - \text{if } h - \text{th trip starts in minute } m+1 \ in \frac{loop}{depot}\delta \\ 0 - \text{otherwise} \end{cases}$$

$$f_{h,t}^{\delta} = \begin{cases} 1 - \text{if } h - \text{th trip ends in minute } m+1 \ in \frac{loop}{depot}\delta \\ 0 - \text{otherwise} \end{cases}$$

If no breaks are required between trips, the formula (13) takes the form:

$$LB_{\beta}^{D} = \max_{t=1,\ldots,\tau-1} \left(\sum_{\delta\in D}\left(\max\left(\sum_{h\in H} s_{h,t+1}^{\delta} - \sum_{h\in H} f_{h,t}^{\delta}, 0\right)\right)\right). \quad (14)$$

To adapt the formula (14) to the Multi Vehicle Type problem, it is further extended by taking into account the types of vehicles (obviously, we need to know exactly which type of a vehicle will serve the trip:

$$LB_{\beta,W}^{D,P} = \max_{t=I+1,\ldots,\tau-1} \left(\sum_{\delta\in D}\sum_{p\in P}\left(\max\left(\sum_{h\in H} s_{h,t+1}^{\delta,p} - \sum_{h\in H} f_{h,t}^{\delta,p}, 0\right) + \sum_{w=1}^{W} \sum_{h\in H} f_{h,t-w}^{\delta,p}\right)\right) \quad (15)$$

where P is a set of vehicle types, and $s_{h,t+1}^{\delta,p}$ and $f_{h,t}^{\delta,p}$ are binary variables that take the following values:

$$s_{h,t+1}^{\delta,p} = \begin{cases} 1 - \text{if } h - th \text{ trip starts in minute } t+1 \text{ in } \frac{loop}{depot}\delta \\ \text{and will serviced by vehicle of type } p \\ 0 - \text{otherwise} \end{cases}$$

$$f_{h,t}^{\delta,p} = \begin{cases} 1 - \text{if } h - \text{th trip ends in minute } t \text{ in } \frac{loop}{depot}\delta \\ \text{and will serviced by vehicle of type } p \\ 0 - \text{otherwise} \end{cases}$$

If no breaks are required between trips, then the formula (15) takes the form:

$$LB_{\beta}^{D,P} = \max_{t=1,\ldots,\tau-1} \left(\sum_{\delta\in D}\sum_{p\in P}\left(\max\left(\sum_{h\in H} s_{h,t+1}^{\delta,p} - \sum_{h\in H} f_{h,t}^{\delta,p}, 0\right)\right)\right) \quad (16)$$

If vehicle types are known but start and end points are unknown, then the LB for the number of vehicle blocks that await their trips should be computed from the following formula:

$$LB_{\beta,W}^{P} = \max_{t=W+1,\ldots,\tau-1} \left(\sum_{p\in P}\left(\max\left(\sum_{h\in H} s_{h,t+1}^{p} - \sum_{h\in H} f_{h,t}^{p}, 0\right) + \sum_{w=1}^{W} \sum_{h\in H} f_{h,t-w}^{p}\right)\right) \quad (17)$$

where $s_{h,t+1}^{p}$ and $f_{h,t}^{p}$ are binary variables that take the following values:

$$s_{h,t+1}^{p} = \begin{cases} 1 - \text{if } h - \text{th trip starts in minute } t+1 \\ \text{and will serviced by vehicle of type } p \\ 0 - \text{otherwise} \end{cases}$$

$$f_{h,t}^{p} = \begin{cases} 1 - \text{if } h - \text{th trip ends in minute } t \\ \text{and will serviced by vehicle of type } p \\ 0 - \text{otherwise} \end{cases}$$

If no breaks are required between trips, then the formula (17) takes the form:

$$LB_{\beta,W}^{P} = \max_{t=W+1,\dots,\tau-1} \left(\sum_{p\in P} \left(\max\left(\sum_{h\in H} s_{h,t+1}^{\delta,p} - \sum_{h\in H} f_{h,t}^{\delta,p}, 0 \right) \right) \right) \quad (18)$$

Based on the above presented formulas, we propose the following variants (depending on the level of detail of the data held) for computing the LB on the number of vehicle blocks (assuming that the LBs α and β refer to the same minute):

$$LB_1 = LB_\alpha + LB_\beta \qquad LB_{1,I} = LB_\alpha + LB_{\beta,W}$$
$$LB_2 = LB_\alpha + LB_\beta^D \qquad LB_{2,I} = LB_\alpha + LB_{\beta,W}^D$$
$$LB_3 = LB_\alpha + LB_\beta^{D,P} \qquad LB_{3,I} = LB_\alpha + LB_{\beta,W}^{D,P}$$
$$LB_4 = LB_\alpha + LB_\beta^P \qquad LB_{4,I} = LB_\alpha + LB_{\beta,W}^P$$

The following inequalities hold between the proposed variants of the LB:

$$LB_1 \le LB_2 \le LB_3$$
$$LB_1 \le LB_4 \le LB_3$$
$$LB_{1,I} \le LB_{2,I} \le LB_{3,I}$$
$$LB_{1,I} \le LB_{4,I} \le LB_{3,I}$$

5. Computational experiments

5.1. Experiments for Real-World Problem Instances

The proposed method for determining the lower bound on the number of vehicle blocks was tested using real data coming from one of the cities in Poland. The selected city is a medium size city (with the population ranging from 100 to 200 thousand), which places it in the top 10 Polish cities in terms of the number of inhabitants.

The selected three problem instances reflect typical vehicle blocks scheduling problems in a public transport company in the analyzed city. The largest problem instance includes 1,257 service trips, the medium-sized instance has 803 service trips, and the smallest one covers 635 service trips. The given instance sizes are typical for the working week, Saturdays and Sundays, and holidays, respectively. The number of trips typical for Sunday is therefore half that of a weekday.

Experiments for real MD-VSP instances were carried out based on the previously presented model. The calculations were performed on a computer with an AMD Ryzen 4800H 2.9 GHz processor (8 cores, 16 threads) and 32GB RAM. CPLEX Solver version 20.1 was used to calculate the MILP model, and the LB heuristic was created in C# compiled for the .NET 6 framework.

The results for the MILP model are presented in Table 2. The time after which the algorithm was able to return LB was recorded (optimization time for larger instances was much longer). As can be seen, the time to obtain the LB increases strongly nonlinearly with the increase in the number of trips.

Table 2. Results of MILP model for considered three problem instances (determined value of LB and time used to compute it)

Number of service trips					
635		**803**		**1257**	
LB	**Execution time [ms]**	**LB**	**Execution time [ms]**	**LB**	**Execution time [ms]**
29	12,000	39	25,000	60	167,000

The results of the MILP model for the considered three problem instances (determined value of LB variants and time used to compute it) are presented in Table 3. For the smallest and the largest instances of the problem, the LB values differ depending on the variant of its calculation, while the values for both LB2 and LB3 are always identical. Comparing these values to the LB values calculated by CPLEX solver, they are 10–14% lower. It is worth noting that the greater the number of trips is, the smaller is the difference.

Table 3. Results obtained using proposed method for three problem instances (determined value of LB and time used to compute it)

Lower Bound formula	**Number of service trips**					
	635		**803**		**1257**	
	LB	**Execution time [ms]**	**LB**	**Execution time [ms]**	**LB**	**Execution time [ms]**
LB_1	24	0.2	34	0.3	52	0.4
LB_2	25	4.4	34	4.4	54	6.0
LB_3	25	7.5	34	15.2	54	28.6
LB_4	24	0.4	34	0.5	53	0.8

5.2. Assessment of Impact of Waiting Time Parameter on LB Values

The experiments were also carried to verify how the waiting time (W) between the trips impacts the value of LB. The results obtained are presented in Tables 4-6.

For the smallest instance (Table 4), LB increases along with the increase of waiting time, however the same values of LB are obtained for all its variants (LB1–LB4). It is worth noting that in two cases (LB2 and LB3) 1 minute waiting time does not increase the LB value, similarly there is no difference in LB values between 2 and 3 minute waiting times, as well as between 4 and 5 minute waiting times.

In the case of the medium instance (Table 5), we no longer observe any increases in LB if the break time is increased by 1 minute, and in the case of LB_1 also by 2 minutes. Contrary to the smaller instance, there is a significant two-minute difference if $W = 2$ and $W = 3$. We do not observe such a large change for increasing W by the next minutes.

For the largest instance (Table 6), in each case there is already an increase in LB, if the gap is increased. However, it is interesting that for larger gaps ($W = 4$ and $W = 5$) the difference in LB values obtained by different variants disappears, while for the lower W it was even 2 minutes.

Table 4. Results of analysis of impact of waiting time (*W*) parameter on LB value for smallest data instance (635 service trips)

	Break duration *W* [min.]					
	0	**1**	**2**	**3**	**4**	**5**
LB_1	24	25	27	27	29	29
LB_2	25	25	27	27	29	29
LB_3	25	25	27	27	29	29
LB_4	24	25	27	27	29	29

Table 5. Results of the analysis of impact of waiting time (*W*) parameter on LB value for medium-sized data instance (803 service trips)

	Break duration *W* [min.]					
	0	**1**	**2**	**3**	**4**	**5**
LB_1	34	34	34	36	37	38
LB_2	34	34	35	37	38	38
LB_3	34	34	35	37	38	38
LB_4	34	34	35	36	37	38

Table 5. Results of the analysis of impact of waiting time (*W*) parameter on LB value for largest instance (1257 service trips)

	Break duration *W* [min.]					
	0	**1**	**2**	**3**	**4**	**5**
LB_1	52	54	56	58	60	62
LB_2	54	56	58	59	60	62
LB_3	54	56	58	59	60	62
LB_4	53	55	57	58	60	62

Summing up, on the basis of the results presented in Tables 4–6, it can be concluded that the greater the number of included information, the higher the LB value. For the largest instance, the difference between the LB_1 value, which does not take into account the information on vehicle types and loops, and the LB_3 value, which takes into account both these information, was +2, which is a difference of nearly 4% of the total number of blocks. Taking into account the minimum length of breaks between consecutive trips (which can also be treated as a safety buffer for potential delays during the journey), the required number of vehicles increases, on average by 2. In the case of a minimum break between trips of at least 4 minutes, all the proposed methods of determining the LB value allowed to obtain the same result – 60 vehicles. For the 5-minute waiting time, also all variants of LB calculation obtained the same result.

The optimal number of blocks for the problem under consideration was 60. To determine it, apart from all the data used to calculate the presented LB values, it is also necessary, inter alia, to have the following information:

– travel time from the depot to the loop/end of the line,
– number of vehicles of a certain type in a given depot,
– travel time between loops/line ends.

The time to compute the LB value using mixed integer linear programming (MILP) for the largest instance was almost 3 minutes, while it took 1/3 second to compute all of the LB values presented. The difference between the optimal value and the LB value using the largest range of information (LB_3) was 10%. Taking into account the computation time as well as the linear computational complexity of the proposed method, it seems to be an efficient solution, especially for more complex problems.

An important aspect of the proposed LB determination methods is that it is possible to determine a LB value for each time period (in the case under consideration, for each minute) within the assumed time horizon. The Figures 1–3 show the LB_3 value for the analyzed problem instances.
For the smallest instance (corresponding to trips serviced on Sunday – Fig. 1), the number of blocks increases sharply from around 6:00 a.m., and then remains at a certain level (an average of 20 blocks), only to drop at the end of the day (around 11:00 p.m.) when only a few trips are serviced.
In the case of the average instance (Fig. 2), we do not observe such a flattening of the number of blocks during the day. The number of blocks needed increases successively to around 10:00 a.m.–1:00 p.m. (the average number of blocks is approximately 26– 27), and then it starts to drop slightly (with an average level of approximately 20 blocks) until 11:00 p.m.
Finally, for the largest instance (common day – Fig. 3) it is already clear that LB for blocks increases at two peak points (morning and afternoon), and in the middle of the day it remains at the average level of approximately 30 blocks. In the evening, the number of trips drops, but not as smoothly as for the smaller instances.
Thanks to the possibility of generating graphs, one can clearly see the number of necessary vehicles during the day. It also allows to observe periods of increased value in the number of trips, which correspond to peak times. The presented visualization of LB values can be a tool supporting decisions in the field of building schedules or arranging services.

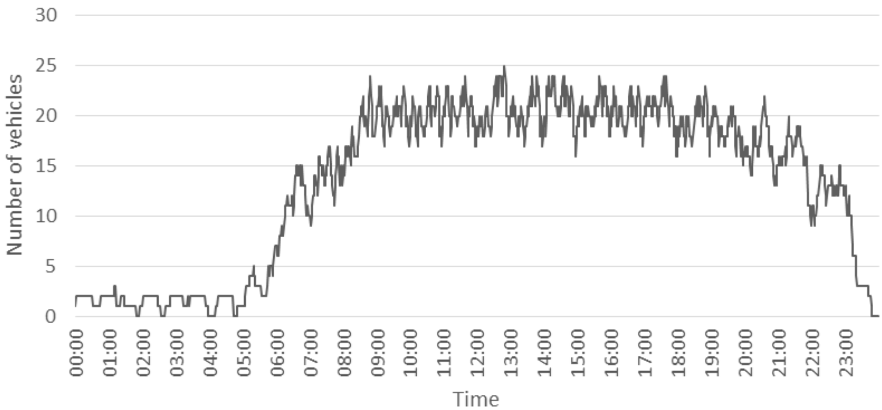

Fig. 1. LB values for smallest instance in different time periods

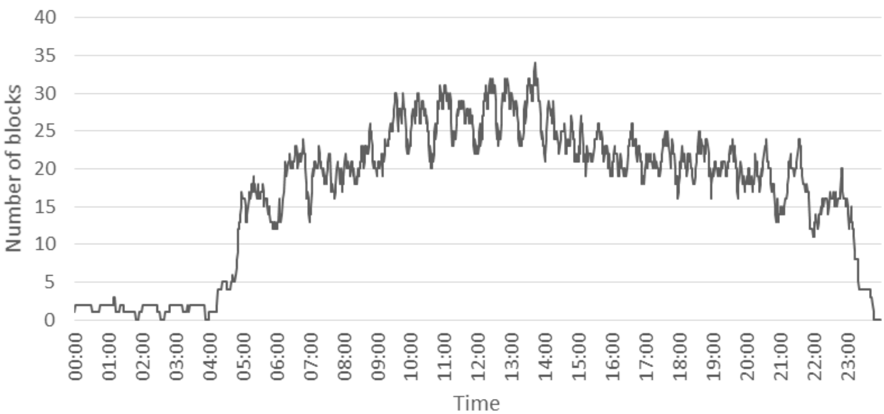

Fig. 2. LB values for smallest instance in different time periods

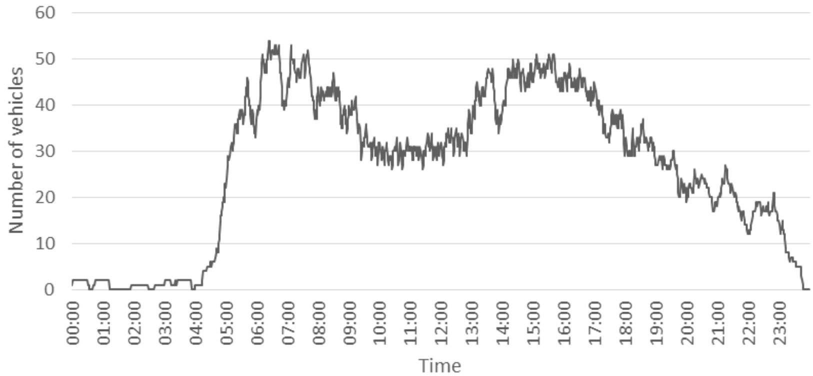

Fig. 3. Change of LB_3 during day

6. Conclusions

We presented a new method for computing the lower bound on the number of vehicle blocks in Multi-Depot Vehicle Scheduling Problem with a fleet covering electric buses. The performance of the method was tested on real data coming from the public transport company in a medium-sized city in Poland. The proposed method is able to efficiently estimate LB for the MD-VSP problem, and additionally takes into account different vehicle types including electric buses. Its execution time, even for large problem instances, is at maximum a several dozen milliseconds, while for the LB generated by OPL CPLEX MILP model one has to wait several minutes. In practice, this allows for the generation of an estimated distribution of the number of blocks during the day (the graphs), which may be helpful, for example, in planning duties and later also crew scheduling. In addition, it was shown how to use the developed LB counting method to check if the increase in the number of blocks is influenced by the setting of intervals between trips. In practice, this gives the opportunity to better plan security buffers and what follow to eliminate delays, e.g., in the case of heavy traffic. In our future work we plan to improve the proposed method by introducing additional factors, such as the charging time of electric vehicles.

Acknowledgments

The study funded by the National Centre for Research and Development (NCBR), Poland under the POIR.01.01.01-00-0514/21-00 programme – R&D project titled “Optimisation of city transport planning and dispatching in the aspect of electromobility”.

References

[1] Ceder, A., (2007). *Public Transit Planning and Operation: Theory, Modeling, and Practice*. Elsevier, Butterworth-Heinemann, 1st ed.

[2] Bunte, S., Kliewer, N. (2009). An overview on vehicle scheduling models. *Public Transport, 1,* 299–317. DOI: 10.1007/s12469-010-0018-5.

[3] Caprara, A., Fischetti, M., Toth, P., (1999). A Heuristic Method for the Set Covering Problem. *Operations Research, 47(5),* 730–743. DOI: 10.1287/opre.47.5.730.

[4] Lübbecke, M.E., Cochran, J.J., Cox L.A., Keskinocak, P., Kharoufeh, J.P., Smith, J.C., (2010). *Column generation*. John Wiley & Sons, Inc.

[5] Kisielewski, P., (2019). *Computer-Aided Planning of City Public Transport*, Publishing House of Warsaw University of Technology, Warsaw (in Polish).

[6] Olsen, N., Kliewer, N., Wolbeck, L., (2022). A study on flow decomposition methods for scheduling of electric buses in public transport based on aggregated time–space network models. *Central European Journal of Operations Research, 30*, 883–919. DOI: 10.1007/s10100-020-00705-6.

[7] Perumal, S.S.G., Lusby, R.M., Larsen, J., (2022). Electric bus planning & scheduling: A review of related problems and methodologies, *European Journal of Operational Research*, *301*(2), 395–413. DOI: 10.1016/j.ejor.2021.10.058.

[8] Zhang, A., Li, T., Zheng, Y., Li, X., Abdullah, M.G., Dong, C., (2022). Mixed electric bus

fleet scheduling problem with partial mixed-route and partial recharging. *International Journal of Sustainable Transportation*. *16*(1), 73–83. DOI: 10.1080/15568318.2021.1914791.

[9] Fisher, M.L., (2004). The Lagrangian relaxation method for solving integer programming problems. *Management Science*, *50(12)*, 1861–1871. DOI: 10.1287/mnsc.1040.0263.

[10] Morrison, D.R., Jacobson, S.H., Sauppe, J.J., Sewell, E.C., (2016). Branch-and-bound algorithms: A survey of recent advances in searching, branching, and pruning, *Discrete Optimiza*tion, *19*, 79–102. DOI: 10.1016/j.disopt.2016.01.005.

[11] Akker van den, M., Hoogeveen, H., Velde de, S., (2002). Combining Column Generation and Lagrangean Relaxation to solve a single-machine common due date problem. *INFORMS Journal on Computing*, *14*(1), 37–51. DOI: 10.1287/ijoc.14.1.37.7706.

[12] Byung-In, K., Seongbae, K., Junhyuk, P., (2012). A school bus scheduling problem, *European Journal of Operational Research*, *218*, 577–585. DOI: 10.1016/j.ejor.2011.11.035.

[13] Bunte S., Kliewer N.: An overview on vehicle scheduling models, *Public Transport*, 1, 2009, 299–317. DOI: 10.1007/s12469-010-0018-5.

[14] Mesquita M, Paias A, Respicio A (2009) Branching approaches for integrated vehicle and crew scheduling. *Public Transp* 1:21–37.

[15] Ribeiro, C., Soumis, F., (1994). A Column Generation Approach to the Multiple-Depot Vehicle Scheduling Problem. *Operations Research*, *42*(1), 41–52. DOI: 10.1287/opre.42.1.41.

[16] Freling, R., Huisman, D., Wagelmans, A.P.M., (2003). Models and Algorithms for Integration of Vehicle and Crew Scheduling, *Journal of Scheduling*, *6*, 63–85. DOI: 10.1023/A:1022287504028.

[17] Wen, M., Linde, E., Ropke, S., Mirchandani, P., Larsen, A., (2016) .An adaptive large neighbourhood search heuristic for the Electric Vehicle Scheduling Problem, *Computers & Operations Research*, *76*, 73–83. DOI: 10.1016/j.cor.2016.06.013.

[18] Stern, H.I., Ceder, A., (1983). Technical Note – An improved lower bound to the minimum fleet size problem, *Transportation Science*, *17*(4), 471–477. DOI: 10.1287/trsc.17.4.471.

[19] Ceder, A., (2011). Optimal multi-vehicle type transit timetabling and vehicle scheduling, *Procedia Social and Behavioral Sciences, 14th EWGT & 26th MEC & 1st RH*, *20*, 19–30. DOI: 10.1016/j.sbspro.2011.08.005.

[20] Ceder, A. (2002), "A Step Function for Improving Transit Operations Planning Using Fixed and Variable Scheduling", Taylor, M.A.P. (Ed.) *Transportation and Traffic Theory in the 21st Century*, Emerald Group Publishing Limited, Bingley, 1-21. DOI: 10.1108/9780585474601-001.

[21] Carpaneto, G., Dell'Amico, M., Fischetti, M., Toth, P., (1989). A branch and bound algorithm for the multiple depot vehicle scheduling problem. *Networks*, *19*, 531–548. DOI: 10.1002/net.3230190505.

[22] Mesquita, M., Paixão, J., (1992). Multiple Depot Vehicle Scheduling Problem: A New Heuristic Based on Quasi-Assignment Algorithms. In: Desrochers, M., Rousseau, JM. (eds) *Computer-Aided Transit Scheduling. Lecture Notes in Economics and Mathematical Systems*, vol. 386. Springer, Berlin, Heidelberg, 167–180. DOI: 10.1007/978-3-642-85968-7_12.

[23] Bodin, L., Golden, B., Assad, A., Ball, M., (1983). Routing and scheduling of vehicles and crews: the state of the art. *Computers & Operations Research*, *10*(2), 63–211. DOI: 10.1016/0305-0548(83)90030-8.

[24] Bertossi, A.A., Carraresi, P., Gallo, G., (1987). On some matching problems arising in vehicle scheduling models. *Networks*, *17*, 271–281. DOI: 10.1002/net.3230170303.

[25] Lamatsch, A., (1992). An Approach to Vehicle Scheduling with Depot Capacity Constraints. In: Desrochers, M., Rousseau, J.M., (eds), *Computer-Aided Transit Scheduling. Lecture Notes in Economics and Mathematical Systems*, vol. 386. Springer, Berlin, Heidelberg, 181–195. DOI: 10.1007/978-3-642-85968-7_13.

[26] Kliewer, N., Mellouli, T., Suhl, L., (2002). A new solution model for multi-depot multi-vehicle-type vehicle scheduling in (sub)urban public transport. In: *Proceedings of the 13th Mini-EURO Conference and the 9th meeting of the EURO working group on transportation*, 604–609.

[27] Kliewer, N., Mellouli, T., Suhl, L., (2006). A time-space network based exact optimization model for multi-depot bus scheduling. *European Journal of Operational Research*, *175*(3), 1616–1627. DOI: 10.1016/j.ejor.2005.02.030.

[28] Duda, J., Fierek, S., Karkula, M., Kisielewski, P., Puka, R., Redmer, A., Skalna, I., (2022). Multi-objective optimization model for a multi-depot mixed fleet electric vehicle scheduling problem with real world constraints, *Transport Problems*, 17(4), 137-149, DOI: 10.20858/tp.2022.17.4.12.