# A RELATION OF DOMINANCE FOR THE BICRITERION BUS ROUTING PROBLEM 

JACEK WIDUCH ${ }^{a}$<br>${ }^{a}$ Institute of Informatics<br>Silesian University of Technology，ul．Akademicka 16，44－100 Gliwice，Poland<br>e－mail：jacek．widuch＠polsl．pl


#### Abstract

A bicriterion bus routing（BBR）problem is described and analysed．The objective is to find a route from the start stop to the final stop minimizing the time and the cost of travel simultaneously．Additionally，the time of starting travel at the start stop is given．The BBR problem can be resolved using methods of graph theory．It comes down to resolving a bicriterion shortest path（BSP）problem in a multigraph with variable weights．In the paper，differences between the problem with constant weights and that with variable weights are described and analysed，with particular emphasis on properties satisfied only for the problem with variable weights and the description of the influence of dominated partial solutions on non－dominated final solutions．This paper proposes methods of estimation a dominated partial solution for the possibility of obtaining a non－dominated final solution from it．An algorithm for solving the BBR problem implementing these estimation methods is proposed and the results of experimental tests are presented．


Keywords：multicriteria optimization，set of non－dominated solutions，bicriterion shortest path problem，variable weights， label correcting algorithm，transportation problem．

## 1．Introduction

The shortest path（SP）problem is one of the studied issues of graph theory and one of great importance in many information systems and applications．For example， transportation problems where the goal is to determine the shortest path（the path with the minimal length，or with the minimal time or the cost of travel，etc．）between two given points can be described as a SP problem．It is solved by determining a path with minimal weight between two given vertices in the graph with a single weight function． There are well－known algorithms for finding the path with minimal weight，like the Dijkstra，Bellman－Ford， Floyd－Warshall and Johnson ones（Jungnickel，1999）．In many cases using a graph with a single weight function is insufficient because it does not describe precisely the problem considered．For example，we want to determine the path where the cost and the time of travel are considered and minimized simultaneously．Thus，a graph with $k>1$ weight functions is used and the problem is called the multicriteria shortest path（MSP）one．A special case of the MSP problem is the BSP one，where $k=2$ weight functions are considered．

The MSP and BSP problems are known to be

NP－complete by transformation from a 0－1 knapsack problem（Garey and Johnson，1990；Hansen，1980； Skriver and Andersen，2000a）．Many algorithms for solving both problems are known，and they fall into the following categories：label correcting algorithms （Brumbaugh－Smith and Shier，1989；Corley and Moon， 1985；Daellenbach and De Kluyver，1980；Skriver and Andersen，2000b），label setting algorithms（Hansen， 1980；Martins，1984；Tung and Chew，1988），$k$－th shortest path algorithms（Climaco and Martins，1982），two－phase algorithms（Mote et al．，1991）and others（Chen and Nie，2013；Dell＇Olmo et al．，2005；Machuca et al．，2009； Mandow and Pérez de la Cruz，2008；Martí et al．，2009； Raith and Ehrgott，2009）．The MSP problem is also solved using the weighed linear scalarization method，where a single－objective function is formulated and an optimal solution to a single－objective function is determined （Carraway et al．，1990）．All of the algorithms mentioned assume constant weights of the arcs of the graph，i．e．，the value of the weight function does not change for the given arc．

For the first time the BBR problem was defined and described by Widuch（2012）．The bus network is
represented by a directed multigraph with two weight functions standing for the cost and the time of travel, respectively. For a given arc the weights take variable values because they are calculated during the process of finding the paths in the multigraph. The goal of the problem is to determine a path between two given vertices minimizing the time and the cost of travel simultaneously. Additionally, the time of starting travel at the start vertex is given. In the work of Widuch (2012) an analysis of the problem and a label correcting algorithm with deleting partial solutions were presented. In the algorithm, during the process of finding the solutions only a single partial solution is stored and it represents a path from the start vertex to the given vertex $v_{i}$. The new vertices are added to the current partial solution by visiting the vertices of the multigraph representing the bus network using the depth first search and the backtracking method. Each vertex $v_{i}$ stores the list of pairs $\left(t_{i}, c_{i}\right)$, which constitute a set of non-dominated solutions, where $t_{i}$ and $c_{i}$ are equal to the time and the cost of travel of the partial solutions which have already been analysed. These values are used for estimating the partial solution after adding a new vertex if it is possible to obtain a non-dominated final solution from it. If the estimation is negative, then the partial solution is not analysed and the backtrack is performed. Otherwise, the partial solution is analysed until the final vertex has been added to it.

BBR was modified by adding the next criterion, and in the work of Widuch (2013) the multiple-criteria bus routing (MBR) problem is described, where additionally the length of the path is taken into consideration. Thus, in the MBR problem we determine the path minimizing three criteria, i.e., the time and the cost of travel and the length of the path, simultaneously. There are important differences between the properties of the paths and the methods used to solve the MBR and BBR problems. The set of non-dominated solutions contains only loopless paths. The methods of estimating the partial solutions are different and we cannot use the same methods in both the problems.

In this paper a new algorithm for solving the BBR problem is presented. The work contains theoretical analysis of the BBR problem with reference to graph theory with particular emphasis on differences between that with constant weights and the problem with variable weights. The properties satisfied only for the latter problem are described. In particular, a possibility of obtaining a non-dominated final solution from a dominated partial solution is precisely analysed. It has been proved theoretically and experimentally confirmed. There are defined necessary terms for the final solution obtained from a dominated partial solution, which are not presented in the work of Widuch (2012). All relationships between the final solutions, with one of them obtained from a dominated partial solution, are defined.

The proposed representation of a partial solution and more effective estimation of partial solutions used in the algorithm influence the number of computed and analysed partial solutions which are fewer in comparison with the method presented by Widuch (2012).

The bus routing problem has gained the attention of many researchers and have been intensively studied in the last few decades. In the work of Huang et al. (2014) the problem of optimal bus routing is studied. It is formulated as linking a series of bus stops in a certain order, aiming at minimizing the total cost, which includes user and supplier costs. Thus, a single-criterion problem is considered and ant colony optimization is used to determine an optimal solution.

In 1969 the school bus routing problem (SBRP) was formulated (Newton and Thomas, 1969). It is a problem in the management of school bus fleet and seeks to plan an efficient schedule for a fleet of school buses that pick up students from various bus stops and deliver them to the school by satisfying various constraints, such as the bus capacity, where all students are picked and each student must be assigned to a particular bus. The objective of bus route planing is to visiting all bus stops minimizing the number of school used buses and the total bus travel distance while satisfying service qualities such as student maximum riding time on a bus. The problem is widely studied and a review of papers on SPRP solutions is presented by Park and Kim (2010). The work on solving the problem has continued by the adaptation of various methods such as the branch-and-cut algorithm (Riera-Ledesma and Salazar-González, 2012), ant colony optimization (Addor et al., 2013; Arias-Rojas et al., 2012; Bronshtein and Vagapova, 2015; Yigit and Unsal, 2016), simulated annealing (Manumbu et al., 2014), the genetic algorithm (Sghaier et al., 2013), tabu search (Pacheco et al., 2013), the GRASP (greedy randomized adaptative search procedure) metaheuristic (Siqueira et al., 2016), the time saving heuristic (Worwa, 2014), the harmony search heuristic (Kim and Park, 2013), or the column-generation-based algorithm (Caceres et al., 2014). In the work of Chen et al. (2015) two algorithms for solving the SBRP are proposed: an exact method of mixed integer programming (MIP) and hybrid simulated annealing with the local search metaheuristic.

The SBRP is modified and many variants of the problem have been studied. One variant of the SBRP is the school bus routing problem with time windows (SBRPTW). It takes into account that buses must arrive to pick up students before some specific time (lower bound of the time window), and they can arrive before another specific time (the upper bound of the time window). In addition, the students were not picked up before the beginning of the time window. A hybrid column generation method (López and Romero, 2015) and a branch-and-bound algorithm (Kim et al., 2012)
are proposed to solve the problem. The next studied variant of the SBRP is the school bus routing problem with bus stop selection (SBRPBSS). Here a set of potential stops is determined first in such a way that each student lives within a given distance of at least one stop. Routes are then determined for school buses so that all students are picked up at a stop they can reach. Thus, determining the set of visited bus stops is a part of the problem. The following methods of resolving the SBRPBSS are proposed: a genetic algorithm (Díaz-Parra et al., 2012; Kang et al., 2015), a column-generation-based algorithm (Kinable et al., 2014; Riera-Ledesma and Salazar-González, 2013), a GRASP + VND (variable neighborhood descent) matheuristic (Schittekat et al., 2013), an artificial ant colony with a variable neighborhood local search algorithm (Euchi and Mraihi, 2012), continuous approximation (Ellegood et al., 2015). In the work of Chalkia et al. (2014) the SBRPBSS is modified and the safety of the bus stop (the size and location of the waiting area, the quality of the ground in the waiting area, and the visibility of the stop for approaching drivers, pedestrian crossing, etc.) is in addition considered.

The paper consists of four sections and it is organized as follows. In Section 2, the BBR problem is described. It contains the formulation of the mathematical model, the analysis of the BBR problem and the algorithm for solving it. The influence of dominated partial solutions on non-dominated final solutions is precisely analysed and the conditions, whose fulfillment makes it possible to obtain a non-dominated final solution from a dominated partial solution are presented. In Section 3 experimental test results are presented. Finally, some conclusions are drawn in Section 4

## 2. Bicriterion bus routing problem

2.1. Formulation of the problem. The BBR problem belongs to the group of problems where the goal is to choose the means of transport and to find a route of travel between two given points for a given time of starting travel. The bus network is represented by a directed weighted multigraph $G=(V, E)$. The multigraph $G$ contains $|V|=n$ vertices $v_{1}, \ldots, v_{n}$ and $|E|=m$ arcs $e_{1}, \ldots, e_{m}\left(e_{i}=\left(v_{j}, v_{k}\right) ; v_{j} \neq v_{k} ; v_{j}, v_{k} \in V\right)$. The vertices represent the bus stops, thus a vertex expression with reference to the multigraph $G$ representing the bus network determines the bus stop of the bus network. In the network buses of $M$ bus lines numbered from 1 to $M$ are run. The network is divided into zones and determines the cost of travel.

For each bus line $i(i=1, \ldots, M)$ the route is defined and consists of a sequence of stops through which the bus runs from a start stop, represented by vertex $v_{s}^{i}$, to a final stop, represented by vertex $v_{e}^{i}$, of the line. The
travel of the bus of a given bus line is directed, i.e., if it runs from $v_{a}^{i}$ to $v_{b}^{i}\left(v_{a}^{i} \neq v_{b}^{i}\right)$, this does not imply that the bus runs in the opposite direction. The bus can run in both directions but the routes can be different. Bus stops belonging to the route of the bus line are different except for the start and final stops, which can be the same. If the start and final stops are identical, then we have called a circular bus line.

Let the route of the $i$-th bus line $(i=1, \ldots, M)$ be represented by a sequence of the following vertices:

$$
\begin{equation*}
\left\langle v_{0}^{i}, v_{1}^{i}, \ldots, v_{k-1}^{i}, v_{k}^{i}\right\rangle \tag{1}
\end{equation*}
$$

where $v_{0}^{i}=v_{s}^{i}$ represents the start stop and $v_{k}^{i}=v_{e}^{i}$ represents the final stop of the line. The bus runs between stops belonging to the route represented by (1) with a given frequency. It runs from $v_{0}^{i}$ at time $T_{0}^{i}$, passes through $v_{1}^{i}, \ldots, v_{k-1}^{i}$ at times $T_{0}^{i}+\delta_{0}^{i}, \ldots, T_{0}^{i}+\delta_{k-2}^{i}$, respectively, and reaches $v_{k}^{i}$ at time $T_{0}^{i}+\delta_{k-1}^{i}$. The bus starts the next course at time $T_{1}^{i}\left(T_{1}^{i}=T_{0}^{i}+\beta_{0}^{i}\right)$, and therefore it reaches $v_{1}^{i}, \ldots, v_{k}^{i}$ at times $T_{1}^{i}+\delta_{0}^{i}, \ldots, T_{1}^{i}+\delta_{k-1}^{i}$. It executes $p^{i}$ courses and leaves $v_{0}^{i}$ at the following times: $T_{0}^{i}, \ldots, T_{p^{i}-1}^{i}\left(T_{0}^{i}<\ldots<T_{p^{i}-1}^{i}\right)$, where $T_{j}^{i}=$ $T_{0}^{i}+\beta_{j-1}^{i}\left(j=1, \ldots, p^{i}-1\right)$. The timetable of the bus of the $i$-th line $(i=1, \ldots, M)$ defines the values $T_{0}^{i}$, $\beta_{0}^{i}, \ldots, \beta_{p^{i}-2}^{i}, \delta_{0}^{i}, \ldots, \delta_{k-1}^{i}\left(0<\beta_{0}^{i}<\ldots<\beta_{p^{i}-2}^{i} ;\right.$ $0<\delta_{0}^{i}<\ldots<\delta_{k-1}^{i}$ ).

The frequency of the bus courses depends on the time of day. For example, during peak hours it is greater than in the evening. Thus the timetable defines the parameters $\beta_{0}^{i}, \ldots, \beta_{p^{i}-2}^{i}$ for the given bus line. The time of day also influences the time of travel between two given bus stops, i.e., it may be greater during peak hours than in the evening. In addition, it depends on the way of travel between the pair of stops, i.e., if the travel is directed or the travel through other stops. Therefore the parameters $\delta_{0}^{i}, \ldots, \delta_{k-1}^{i}$ are defined for each bus line. We assume a simplified model of the bus network where the times of getting on and off the bus by passengers are omitted.

The BBR problem is stated as follows. Given the bus network structure, the bus line routes and the timetable, the start stop represented by the start vertex $v_{s}$ and the final stop represented by the final vertex $v_{e}$ between which we want to travel, and the time $T_{s}$ of starting travel at $v_{s}$. The goal is to find a route from the start stop to the final stop minimizing the time and the cost of travel. The stops belonging to the route, the stops of changes, the times of departure from all stops belonging to the route, the bus lines along which the buses run between stops should be determined.

The time of travel is the sum of the time of waiting at the start stop, the times of waiting for changes and the travel times between stops belonging to the route. The travel times between stops are defined by the timetable of bus lines. The cost of travel depends on the location of the
stops in the area of zones, the number of changes in the route and the type of bus line, i.e., whether it is a regular or a fast line. Travel by a fast line is faster than by a regular line, and the cost of travel is twice as large as the cost of travel by a regular line. The cost of a single travel, i.e., travel without a bus change, by a bus of a regular line is calculated as follows. A ticket for travel within the area of a single zone equals $c_{1}\left(0<c_{1}\right)$ units, within two zones it equals $c_{2}\left(c_{1}<c_{2}\right)$ units and within the confines of more than two zones it equals $c_{3}\left(c_{2}<c_{3}\right)$ units. Therefore the cost of travel from the start stop to the final stop equals the sum of costs of travel between the stops of bus changes. In the examples the following costs of a ticket are assumed: $c_{1}=2.0, c_{2}=2.3$ and $c_{3}=2.6$ units.
2.2. Mathematical model of the BBR problem. The mathematical model of the BBR problem is formulated as follows (Table 1 shows the symbols used in the model):

$$
\begin{align*}
& \min \quad T(p)=\sum_{l=1}^{M} \sum_{i \in V} \sum_{j \in V} t_{i j l} \cdot x_{i j l},  \tag{2}\\
& \min \quad C(p)=\sum_{l=1}^{M} \sum_{i \in V} \sum_{j \in V} c_{i j l} \cdot x_{i j l}, \tag{3}
\end{align*}
$$

subject to

$$
\begin{gather*}
\sum_{l=1}^{M} \sum_{\{j \mid(i, j) \in E\}} x_{i j l}-\sum_{l=1}^{M} \sum_{\{j \mid(j, i) \in E\}} x_{j i l} \\
= \begin{cases}1 & \text { if } i=v_{s}, \\
0 & \text { if } i \neq v_{s}, v_{e} \\
-1 & \text { if } i=v_{e},\end{cases}  \tag{4}\\
\sum_{l=1}^{M} \sum_{j \in V} x_{i j l} \leq 2, \quad \forall i \in V,  \tag{5}\\
k=0,  \tag{6}\\
T_{p, a}^{k}=\left\{\begin{array}{l}
T_{s}, \quad \sum_{q=0}^{k-1} t_{a b l}, \quad a=v_{p}^{q}, b=v_{p}^{q+1} \\
l=l_{p}^{q, q+1}, \\
0<k<\operatorname{len}(p)
\end{array}\right. \\
T_{p, a}^{k} \leq T_{p, d}^{k} \wedge T_{p, d}^{k} \in D[a ; l],  \tag{7}\\
k=0, \ldots, \operatorname{len}(p)-2, \quad a=v_{p}^{k}, \quad l=l_{p}^{k, k+1}
\end{gather*}
$$

The objective functions (2) and (3) minimize the time and the cost of travel, respectively. The constraints (4) yield a directed path from the start vertex $v_{s}$ to the final vertex $v_{e}$. Constraints (5) state that each vertex is visited at most two times. The constraints (6) state the time of arrival to each vertex belonging to the path. Finally, the constraints (7) force that the time of departure from a vertex is not earlier than the time of arrival to this vertex and the time of departure is in line with the timetable.

Table 1. Symbols used in the mathematical model of the BBR problem.

| Parameters |  |
| :--- | :--- |
| $V=\{1, \ldots, n\}$ | set of vertices representing bus <br> stops |
| $E$ | set of all arcs between vertices |
| $M$ | number of bus lines |
| $D[1 \ldots, n ; 1, \ldots, M]$ | timetable (times of departures of <br> each bus line from each bus stop) |
| $t_{i j l}$ | time of travel from $v_{i}$ to $v_{j}$ by <br> bus of line $l$, it includes potential <br> time of waiting at $v_{i}$ |
| $c_{i j l}$ | cost of travel from $v_{i}$ to $v_{j}$ by <br> bus of line $l$ |
| $v_{s}, v_{e}$ | start and final vertices |
| $T_{s}$ | time of starting travel at $v_{s}$ |
| Additional symbols | $p$ path from $v_{s}$ to $v_{e}$ <br> to $p$ <br> len $(p)$ vertex in $k$-th position in path $p$ <br> $v_{p}^{k}$ line number of bus which runs <br> from vertex in $k$-th position to <br> vertex in $(k+1)$-th position in $p$ <br> $l_{p}^{k, k+1}$  <br> $T_{p, d}^{k}$ 1 if bus of line $l$ traverses arc from <br> $v_{i}$ to $v_{j}, 0$ otherwise <br> Decision variables of arrival to vertex in $k$-th  <br> position in path $p$  |
| $x_{i j l}^{k}$ | time of departure from vertex in <br> $k$-th position in path $p$ |

2.3. Analysis of the BBR problem. The BBR problem can be modeled in graph theoretical terms as follows. A directed weighted multigraph $G=(V, E)$ represents the bus network. The vertices $v_{1}, \ldots, v_{n}$ represent the bus stops. Each arc $e_{i}=\left(v_{j}, v_{k}\right),\left(v_{j} \neq v_{k} ; v_{j}, v_{k} \in V\right)$ corresponds to a specific bus line whose buses run directly from the stop represented by $v_{j}$ to the stop represented by $v_{k}$. Direct travel from $v_{j}$ to $v_{k}$ means that the route does not include other vertices. Between a pair of stops buses of many bus lines can run. Therefore the multigraph can contain parallel arcs. The arc $e_{i}$ has a single label $l\left(e_{i}\right)$ and two weights: $t\left(e_{i}\right)$ and $c\left(e_{i}\right)$.

The label $l\left(e_{i}\right)$ takes a value from the range $1, \ldots, M$ and represents the line number of the bus which runs from $v_{j}$ to $v_{k}$. During the process of finding the solutions it is used to determine if a change at the given bus stop is done, and to determine the cost of travel.

The weight $t\left(e_{i}\right)$ takes a positive value and it equals the difference between the time of arrival $T_{k}$ to $v_{k}$ and the time of arrival $T_{j}$ to $v_{j}$, i.e., $t\left(e_{i}\right)=T_{k}-T_{j}$. Thus it is a sum of the time of waiting at $v_{j}$ and the travel time from $v_{j}$ to $v_{k}$. The travel time from $v_{j}$ to $v_{k}$ is constant


Fig. 1. Route of the bus line. The dashed line denotes the border of zones.
and defined by the timetable, but the time of waiting at $v_{j}$ is variable and it depends on the time of arrival $T_{j}$ to $v_{j}$. Therefore the value of $t\left(e_{i}\right)$ is variable and determined by the time of waiting at $v_{j}$.

The weight $c\left(e_{i}\right)$ takes a non-negative value and equals $c\left(e_{i}\right)=c_{k}-c_{j}$, where $c_{k}$ is the cost of travel from $v_{s}$ to $v_{k}$ and $c_{j}$ is the cost of travel from $v_{s}$ to $v_{j}$. The weight $c\left(e_{i}\right)$ like the weight $t\left(e_{i}\right)$, is variable. The value of $c\left(e_{i}\right)$ depends on a possible change at $v_{j}$ and location the vertices $v_{j}$ and $v_{k}$ in the same zone or of in different zones. We should consider the following cases to determine the value of $c\left(e_{i}\right)$. First, travel with a change at $v_{j}$ is considered. If $v_{j}$ and $v_{k}$ are located in the same zone, then $c\left(e_{i}\right)$ equals $c_{1}$ and $c_{2}$ otherwise 1 Second, we travel without a change at the vertex $v_{j}$. If $v_{j}$ and $v_{k}$ are located in the same zone, then this does not increase the cost of travel and $c\left(e_{i}\right)=0$. If we cross a zone, then the value of $c\left(e_{i}\right)$ depends on the number of zones which we crossed since the last change while travelling to $v_{j}$. If we did not cross any zone, then $c\left(e_{i}\right)=c_{2}-c_{1}$. It equals $c\left(e_{i}\right)=c_{3}-c_{2}$ if we crossed a single zone, and if we crossed two or more zones it equals $c\left(e_{i}\right)=0$.

The determination of the value of the weight $c$ is illustrated with an example of travel to the vertex $v_{6}$ by a bus of the line whose route is presented in Fig. 1 The value of the weight $c$ of arcs $e_{1}, \ldots, e_{5}$ depends on the start vertex $v_{s}$ (Table 2). Let us consider the start vertex $v_{s}=v_{1}$. The vertices $v_{1}$ and $v_{2}$ are located in the same zone, therefore $c\left(e_{1}\right)=c_{1}$. We do not cross a zone while travelling from $v_{2}$ to $v_{3}$. Therefore it does not increase the cost of travel and the weight $c$ of arc $e_{2}$ equals $c\left(e_{2}\right)=0$. The vertices $v_{1}, v_{2}$ and $v_{3}$ are located in the same zone; and therefore the cost of travel from $v_{1}$ to $v_{2}$ equals the cost of travel from $v_{1}$ to $v_{3}$ and is equal to $c_{1}$. We crossed two zones while travelling from $v_{1}$ to $v_{5}$; therefore the next crossed zone while travelling from $v_{5}$ to $v_{6}$ does not increase the cost of travel and $c\left(e_{5}\right)=0$. Table 2 shows that the weight $c\left(e_{i}\right)$ is not constant. For example, the weight $c\left(e_{5}\right)$ takes 3 different values.

The bus route from the start stop represented by the start vertex $v_{s}$ to the final stop represented by the final

[^0]Table 2. Values of the weight $c$ of $\operatorname{arcs} e_{1}, \ldots, e_{5}$ depending on the start vertex $v_{s}$ for travel to the final vertex $v_{6}$ by a bus of the line whose route is presented in Fig. 1

| $v_{s}$ | $c\left(e_{1}\right)$ | $c\left(e_{2}\right)$ | $c\left(e_{3}\right)$ | $c\left(e_{4}\right)$ | $c\left(e_{5}\right)$ |
| :---: | :---: | :---: | :---: | :---: | :---: |
| $v_{1}$ | $c_{1}$ | 0 | $c_{2}-c_{1}$ | $c_{3}-c_{2}$ | 0 |
| $v_{2}$ | - | $c_{1}$ | $c_{2}-c_{1}$ | $c_{3}-c_{2}$ | 0 |
| $v_{3}$ | - | - | $c_{2}$ | $c_{3}-c_{2}$ | 0 |
| $v_{4}$ | - | - | - | $c_{2}$ | $c_{3}-c_{2}$ |
| $v_{5}$ | - | - | - | - | $c_{2}$ |

vertex $v_{e}$ is given by the path

$$
\begin{equation*}
p_{v_{s}, v_{e}}=\left\langle v_{0}=v_{s}, e_{1}, \ldots, v_{k-1}, e_{k}, v_{k}=v_{e}\right\rangle \tag{8}
\end{equation*}
$$

from $v_{s}$ to $v_{e}$ in the multigraph $G$ representing the bus network. For each vertex $v_{i}(i=0, \ldots, k-1)$ belonging to the path $p_{v_{s}, v_{e}}$, the time of departure $T_{i}$ is stored. Thus a path expression with reference to the multigraph $G$ representing the bus network determines the bus route in the network.

Definition 1. A partial solution is called the path $p_{v_{s}, v_{i}}$ in the multigraph $G$ from the start vertex $v_{s}$ to any vertex $v_{i}$, where $v_{i} \neq v_{s}$ and $v_{i} \neq v_{e}$. The path $p_{v_{s}, v_{e}}$ from the start vertex $v_{s}$ to the final vertex $v_{e}$ is called the final solution.

Definition 2. A path $p_{v_{i}, v_{j}}$ containing a subsequence of vertices and arcs from $v_{i}$ to $v_{j}$ belonging to $p_{v_{s}, v_{e}}$ is called a subpath of $p_{v_{s}, v_{e}}$ and it is denoted as follows:

$$
p_{v_{i}, v_{j}}=\operatorname{sub}_{p_{v_{s}, v_{e}}}\left(v_{i}, v_{j}\right)
$$

Definition 3. Assume that paths $p_{v_{s}, v_{i}}$ and $p_{v_{i}, v_{e}}$ containing a sequence of vertices and arcs described by

$$
\begin{aligned}
& p_{v_{s}, v_{i}}=\left\langle v_{0}^{\prime}, e_{1}^{\prime}, \ldots, e_{j}^{\prime}, v_{i}\right\rangle \\
& p_{v_{i}, v_{e}}=\left\langle v_{0}^{\prime \prime}, e_{1}^{\prime \prime}, \ldots, e_{k}^{\prime \prime}, v_{e}\right\rangle
\end{aligned}
$$

are given, where $v_{0}^{\prime}=v_{s}$ and $v_{0}^{\prime \prime}=v_{i}$. The start vertex of $p_{v_{i}, v_{e}}$ and the final vertex of $p_{v_{s}, v_{i}}$ are identical, and it follows that it is possible to obtain the path $p_{v_{s}, v_{e}}$ as a concatenation of $p_{v_{s}, v_{i}}$ and $p_{v_{i}, v_{e}}$ :

$$
\begin{equation*}
p_{v_{s}, v_{e}}=p_{v_{s}, v_{i}} \oplus p_{v_{i}, v_{e}} . \tag{9}
\end{equation*}
$$

The path $p_{v_{s}, v_{e}}$ consists of a sequence of vertices and arcs belonging to the paths $p_{v_{s}, v_{i}}$ and $p_{v_{i}, v_{e}}$ :

$$
p_{v_{s}, v_{e}}=\left\langle v_{s}, e_{1}^{\prime}, \ldots, e_{j}^{\prime}, v_{i}, e_{1}^{\prime \prime}, \ldots, v_{k-1}^{\prime \prime}, e_{k}^{\prime \prime}, v_{e}\right\rangle
$$

The length of the path equals the number of arcs belonging to the path. The path (8) has two weights $T\left(p_{v_{s}, v_{e}}\right)$ and $C\left(p_{v_{s}, v_{e}}\right)$ that represent the time and the cost of travel from $v_{s}$ to $v_{e}$. These weights are equal to the sum of the corresponding weights of the arcs belonging to
$p_{v_{s}, v_{e}}$, i.e.,

$$
\begin{align*}
T\left(p_{v_{s}, v_{e}}\right) & =\sum_{i=1}^{k} t\left(e_{i}\right)  \tag{10}\\
C\left(p_{v_{s}, v_{e}}\right) & =\sum_{i=1}^{k} c\left(e_{i}\right) \tag{11}
\end{align*}
$$

Additionally, the number of crossed zones in $p_{v_{s}, v_{e}}$ is denoted by $Z\left(p_{v_{s}, v_{e}}\right)$.

The time of travel of the path $p_{v_{s}, v_{e}}$ (9) obtained as a concatenation of $p_{v_{s}, v_{i}}$ and $p_{v_{i}, v_{e}}$ equals the sum of the times of travel:

$$
T\left(p_{v_{s}, v_{e}}\right)=T\left(p_{v_{s}, v_{i}}\right)+T\left(p_{v_{i}, v_{e}}\right)
$$

where the time of starting travel at $v_{i}$ equals $T_{i}=T_{s}+$ $T\left(p_{v_{s}, v_{i}}\right)$.

The cost of travel $C\left(p_{v_{s}, v_{e}}\right)$ of the path $p_{v_{s}, v_{e}}$ (9) depends on a possible change at the vertex $v_{i}$. If the travel through $v_{i}$ is done with a change (Fig. 2 a) ) then it is necessary to validate a new ticket and the cost of travel $C\left(p_{v_{s}, v_{e}}\right)$ equals the sum of the costs of travel:

$$
C\left(p_{v_{s}, v_{e}}\right)=C\left(p_{v_{s}, v_{i}}\right)+C\left(p_{v_{i}, v_{e}}\right) .
$$

Otherwise (Fig. 2(b)) it is not necessary to validate a new ticket, the sum of costs of travel should be decreased by $\Delta c$ and equals

$$
C\left(p_{v_{s}, v_{e}}\right)=C\left(p_{v_{s}, v_{i}}\right)+C\left(p_{v_{i}, v_{e}}\right)-\Delta c .
$$



Fig. 2. Paths from the start vertex $v_{s}$ to the final vertex $v_{e}$ : with a change at the vertex $v_{i}$ (a), without a change at the vertex $v_{i}(\mathrm{~b})$. The label on each arc represents the line number of the bus which runs between bus stops represented by the vertices connected by the arc.

The value of decreasing the cost $\Delta c$ depends on the vertex $v_{i}$ and the number of crossed zone borders $Z_{a}$ since the last change at $v_{a}$ while travelling to $v_{i}$ in $p_{v_{s}, v_{i}}$ and the

Table 3. Values of decreasing the cost $\Delta c$ depending on the number of crossed zone borders $Z_{a}$ and $Z_{b}$ being a result of travel through the vertex $v_{i}$ without a change.

|  | $Z_{b}=0$ | $Z_{b}=1$ | $Z_{b} \geq 2$ |
| :---: | :---: | :---: | :---: |
| $Z_{a}=0$ | $c_{1}$ | $c_{1}$ | $c_{1}$ |
| $Z_{a}=1$ | $c_{1}$ | $2 \times c_{2}-c_{3}$ | $c_{2}$ |
| $Z_{a} \geq 2$ | $c_{1}$ | $c_{2}$ | $c_{3}$ |

number of crossed zone borders $Z_{b}$ to the next change at $v_{b}$ while travelling from $v_{i}$ in $p_{v_{i}, v_{e}}$ (Fig. (2)a)). It equals

$$
\begin{aligned}
\Delta c= & C\left(\operatorname{sub}_{p_{v_{s}, v_{e}}}\left(v_{a}, v_{i}\right)\right)+C\left(\operatorname{sub}_{p_{v_{s}, v_{e}}}\left(v_{i}, v_{b}\right)\right) \\
& -C\left(\operatorname{sub}_{p_{v_{s}, v_{e}}}\left(v_{a}, v_{b}\right)\right)
\end{aligned}
$$

where $C\left(\operatorname{sub}_{p_{v_{s}, v_{e}}}\left(v_{a}, v_{i}\right)\right)$ and $C\left(\operatorname{sub}_{p_{v_{s}, v_{e}}}\left(v_{i}, v_{b}\right)\right)$ are respectively the cost of travel from $v_{a}$ to $v_{i}$ and the cost of travel from $v_{i}$ to $v_{b}$, and $C\left(\operatorname{sub}_{p_{v_{s}, v_{e}}}\left(v_{a}, v_{b}\right)\right)$ equals the cost of travel from $v_{a}$ to $v_{b}$. The values of $\Delta c$ are shown in Table 3

The objective of the BBR problem is to find, in the multigraph $G$ representing the bus network, the path $p_{v_{s}, v_{e}}$ minimizing (10) and (11) simultaneously.

The BBR problem is an example of a multiple-criteria optimization (MO) problem, where $k(k>1)$ minimized or maximized criteria $f_{i}$ $(i=1, \ldots, k)$ are given. In most cases, there does not exist a single solution for which all the critera take optimum values, because in order to improve the value of any of the functions we need to degrade those of other functions. Therefore the solution of the MO is a set of solutions called the set of non-dominated (Pareto optimal) solutions (Ehrgott, 2000; Pareto, 1896).

Definition 4. Assume that there are $k(k>1)$ minimized criteria $f_{i}(i=1, \ldots, k)$ and two solutions $A$ and $B$. The solution $A$ is said to dominate the solution $B$, which is denoted as $A \succ B$, if the following conditions are satisfied:

$$
\begin{aligned}
& \forall i \in\{1, \ldots, k\}: f_{i}(A) \leq f_{i}(B) \\
& \exists j \in\{1, \ldots, k\}: f_{j}(A)<f_{j}(B)
\end{aligned}
$$

Solving the BBR problem consists in solving the bicriterion shortest path (BSP) problem between $v_{s}$ and $v_{e}$ vertices in a multigraph with variable weights. The solution of the BBR problem consists of a set of paths in the multigraph $G$, representing the bus network, forming the set of non-dominated solutions. The weights defined by (10) and (11) are the criteria to be minimized.

The set of non-dominated solutions can contain many paths with the same values of the weights (10) and (11) (Widuch, 2012). According to Definition 4 these paths are non-dominated solutions.

Let the paths $p_{v_{s}, v_{e}}^{\prime}$ and $p_{v_{s}, v_{e}}^{\prime \prime}$ belong to the set of
non-dominated solutions and

$$
\begin{align*}
& T\left(p_{v_{s}, v_{e}}^{\prime}\right)=T\left(p_{v_{s}, v_{e}}^{\prime \prime}\right)  \tag{12}\\
& C\left(p_{v_{s}, v_{e}}^{\prime}\right)=C\left(p_{v_{s}, v_{e}}^{\prime \prime}\right) \tag{13}
\end{align*}
$$

One of the following properties is satisfied:

1. The paths $p_{v_{s}, v_{e}}^{\prime}$ and $p_{v_{s}, v_{e}}^{\prime \prime}$ differ from each other in vertices or arcs belonging to these paths.
2. The paths $p_{v_{s}, v_{e}}^{\prime}$ and $p_{v_{s}, v_{e}}^{\prime \prime}$ are identical, i.e., they contain the same sequence of vertices and arcs, and differ from each other in the times of departure from all vertices belonging to these paths.

A necessary condition for a non-dominated solution with the second property is determined by Lemma 1 .

Lemma 1. (Widuch, 2013) Consider paths $p_{v_{s}, v_{e}}^{\prime}$ and $p_{v_{s}, v_{e}}^{\prime \prime}$ which consist of the same sequence of vertices and arcs but differ from each other in the times of departure from vertices belonging to these paths. Let (12) and (13) be satisfied. Then both the paths belong to the set of nondominated solutions if we change at least once in these paths.

The multigraph $G$ representing the bus network contains many paths with the same sequence of vertices and arcs that differ from each other in times of departure from vertices. The property of a path belonging to the set of non-dominated solutions is described by Lemma 2

Lemma 2. Let $P_{T}$ be the set of all paths from $v_{s}$ to $v_{e}$ containing the same sequence of vertices and arcs that differ from each other in the times of departure from vertices. For the path $p_{v_{s}, v_{e}} \in P_{T}$, let $\Delta t$ equal the total time of waiting for changes in $p_{v_{s}, v_{e}}$. If $p_{v_{s}, v_{e}}$ belongs to the set of non-dominated solutions, then

$$
\begin{equation*}
\forall p_{v_{s}, v_{e}}^{\prime} \in P_{T}: \Delta t \leq \Delta t^{\prime} \tag{14}
\end{equation*}
$$

where $\Delta t^{\prime}$ equals the total time of waiting for changes in $p_{v_{s}, v_{e}}^{\prime}$.

Proof. The paths belonging to the set $P_{T}$ contain the same sequence of vertices and arcs, and it follows that

$$
\begin{equation*}
\forall p_{v_{s}, v_{e}}^{\prime}: C\left(p_{v_{s}, v_{e}}\right)=C\left(p_{v_{s}, v_{e}}^{\prime}\right) \tag{15}
\end{equation*}
$$

The time of travel $T\left(p_{v_{s}, v_{e}}\right)$ equals the sum of the travel times between vertices and the total time of waiting for changes. The total travel times between vertices in these paths are identical, and the time of travel $T\left(p_{v_{s}, v_{e}}\right)$ depends on the total time of waiting for changes. If $p_{v_{s}, v_{e}}$ is a non-dominated solution, then it follows that (14) and

$$
\forall p_{v_{s}, v_{e}}^{\prime} \in P_{T}: T\left(p_{v_{s}, v_{e}}\right) \leq T\left(p_{v_{s}, v_{e}}^{\prime}\right)
$$

are satisfied, otherwise it is a dominated solution.
(a)


Fig. 3. Part of paths from the start vertex $v_{s}$ to the final vertex $v_{e}$ : a non-loopless path (a), a loopless path (b).

The set of non-dominated solutions can contain paths which are not loopless (Widuch, 2012). This property is satisfied only in a multigraph with variable weights of arcs, like the multigraph $G$ representing the bus network. It has been shown that the set of non-dominated solutions contains only loopless paths if weights of arcs are non-negative and constant, and at least one is positive (Henig, 1985; Tung and Chew, 1988; 1992).

It should be pointed out that for each non-loopless path $p_{v_{s}, v_{e}}$ there exists a loopless path $p_{v_{s}, v_{e}}^{\prime}$ which contains the same sequence of vertices and arcs like $p_{v_{s}, v_{e}}$ but is devoid of the cycle. The properties of the non-loopless path $p_{v_{s}, v_{e}}$ belonging to the set of non-dominated solutions are defined by Theorem 1

Theorem 1. Consider a non-loopless path $p_{v_{s}, v_{e}}$ containing the cycle $p_{v_{c}, v_{c}}$ (Fig. 3a)) and a loopless path $p_{v_{s}, v_{e}}^{\prime}$ which contains the same sequence of vertices and arcs like $p_{v_{s}, v_{e}}$ but is devoid of the cycle (Fig. З3b)). The path $p_{v_{s}, v_{e}}$ belongs to the set of non-dominated solutions if the following conditions are satisfied:

1. The cycle $p_{v_{c}, v_{c}}$ contains only a single change.
2. At the vertex $v_{c}$ no change is made.
3. The sum of the time of travel through the cycle $p_{v_{c}, v_{c}}$ and the total time of waiting for changes in vertices of the non-loopless path $p_{v_{s}, v_{e}}$ equals the total time of waiting for changes in vertices of the loopless path $p_{v_{s}, v_{e}}^{\prime}$.

Proof. The weights $c$ and $t$ of arcs do not take negative values, and for that reason $C\left(p_{v_{s}, v_{e}}^{\prime}\right) \leq C\left(p_{v_{s}, v_{e}}\right)$ and $T\left(p_{v_{s}, v_{e}}^{\prime}\right) \leq T\left(p_{v_{s}, v_{e}}\right)$. The loopless path $p_{v_{s}, v_{e}}^{\prime}$ does not dominate $p_{v_{s}, v_{e}}$ when the following conditions are satisfied:

$$
\begin{align*}
& C\left(p_{v_{s}, v_{e}}\right)=C\left(p_{v_{s}, v_{e}}^{\prime}\right)  \tag{16}\\
& T\left(p_{v_{s}, v_{e}}^{\prime}\right)=T\left(p_{v_{s}, v_{e}}^{\prime}\right) \tag{17}
\end{align*}
$$

Thus, it is necessary to prove the fulfillment of (16) and (17) and define conditions that guarantee this.

Consider two vertices of changes $v_{a}$ and $v_{b}$, where $v_{a}$ is the vertex of the last change while travelling to $v_{c}$ and $v_{b}$ is the vertex of the next change while travelling from $v_{c}$ (Fig. 3). For the paths $p_{v_{s}, v_{e}}$ and $p_{v_{s}, v_{e}}^{\prime}$ the conditions

$$
\begin{aligned}
\operatorname{sub}_{p_{v_{s}, v_{e}}}\left(v_{s}, v_{a}\right) & =\operatorname{sub}_{p_{v_{s}, v_{e}}^{\prime}}\left(v_{s}, v_{a}\right) \\
\operatorname{sub}_{p_{v_{s}, v_{e}}}\left(v_{b}, v_{e}\right) & =\operatorname{sub}_{p_{v_{s}, v_{e}}^{\prime}}\left(v_{b}, v_{e}\right)
\end{aligned}
$$

are satisfied, and it follows that the costs of travel from $v_{s}$ to $v_{a}$ and from $v_{b}$ to $v_{e}$ are equal in both paths, i.e,

$$
\begin{aligned}
C\left(\operatorname{sub}_{p_{v_{s}, v_{e}}}\left(v_{s}, v_{a}\right)\right) & =C\left(\operatorname{sub}_{p_{v_{s}, v_{e}}^{\prime}}\left(v_{s}, v_{a}\right)\right) \\
C\left(\operatorname{sub}_{p_{v_{s}, v_{e}}}\left(v_{b}, v_{e}\right)\right) & =C\left(\operatorname{sub}_{p_{v_{s}, v_{e}}^{\prime}}\left(v_{b}, v_{e}\right)\right)
\end{aligned}
$$

For this reason the cost of travel $C\left(p_{v_{s}, v_{e}}\right)$ of the non-loopless path $p_{v_{s}, v_{e}}$ depends on the cost of travel from $v_{a}$ to $v_{b}$, and (16) holds if

$$
\begin{equation*}
C\left(\operatorname{sub}_{p_{v_{s}, v_{e}}}\left(v_{a}, v_{b}\right)\right)=C\left(\operatorname{sub}_{p_{v_{s}, v_{e}}^{\prime}}\left(v_{a}, v_{b}\right)\right) \tag{18}
\end{equation*}
$$

The subpath $\operatorname{sub}_{p_{v_{s}, v_{e}}^{\prime}}\left(v_{a}, v_{b}\right)$ of the loopless path contains a single change at the vertex $v_{c}$ (Fig. 3 b)). The condition (18) is satisfied if the subpath $\operatorname{sub}_{p_{v_{s}, v_{e}}}\left(v_{a}, v_{b}\right)$ containing the cycle $p_{v_{c}, v_{c}}$ also contains a single change at $v_{d}$ and we do not change at $v_{c}$ (Fig. 3] (a)). With each next change it is necessary to validate a new ticket. This increases the cost of travel and $C\left(p_{v_{s}, v_{e}}^{\prime}\right)<C\left(p_{v_{s}, v_{e}}\right)$, and the non-loopless path $p_{v_{s}, v_{e}}$ is dominated. This proves Properties 1 and 2 in the Theorem. In the next part of the proof, it is necessary to define the conditions of the fulfillment of (18).

The condition (18) holds if the costs of travel from $v_{a}$ to the vertices $v_{d}$ and $v_{c}$, where we change, and the costs of travel from these vertices to the vertex $v_{b}$ are equal, i.e.,

$$
\begin{align*}
C\left(\operatorname{sub}_{p_{v_{s}, v_{e}}}\left(v_{a}, v_{d}\right)\right) & =C\left(\operatorname{sub}_{p_{v_{s}, v_{e}}^{\prime}}\left(v_{a}, v_{c}\right)\right),  \tag{19}\\
C\left(\operatorname{sub}_{p_{v_{s}, v_{e}}}\left(v_{d}, v_{b}\right)\right) & =C\left(\operatorname{sub}_{p_{v_{s}, v_{e}}^{\prime}}\left(v_{c}, v_{b}\right)\right) . \tag{20}
\end{align*}
$$

The fulfillment of (19) depends on $Z\left(\operatorname{sub}_{p_{v_{s}, v_{e}}}\left(v_{a}, v_{c}\right)\right)$ and $Z\left(\operatorname{sub}_{p_{v_{s}, v_{e}}}\left(v_{c}, v_{d}\right)\right)$, where $Z\left(\operatorname{sub}_{p_{v_{s}, v_{e}}}\left(v_{a}, v_{c}\right)\right)$ equals the number of crossed zones in the subpath from $v_{a}$ to $v_{c}$ and $Z\left(\operatorname{sub}_{p_{v_{s}, v_{e}}}\left(v_{c}, v_{d}\right)\right)$ equals the number of crossed zones in the subpath from $v_{c}$ to $v_{d}$. If $Z\left(\operatorname{sub}_{p_{v_{s}, v_{e}}}\left(v_{a}, v_{c}\right)\right) \geq 2$, then the next crossed zone does not increase the cost of travel. Thus, (19) is satisfied because the cost of travel $C\left(\operatorname{sub}_{p_{v_{s}, v_{e}}}\left(v_{a}, v_{d}\right)\right)$ does not depend on $Z\left(\operatorname{sub}_{p_{v_{s}, v_{e}}}\left(v_{c}, v_{d}\right)\right)$. Otherwise, i.e., $Z\left(\operatorname{sub}_{p_{v_{s}, v_{e}}}\left(v_{a}, v_{c}\right)\right)<2$, the next crossed zone increase the cost of travel and (19) is satisfied if $Z\left(\operatorname{sub}_{p_{v_{s}, v_{e}}}\left(v_{c}, v_{d}\right)\right)=0$.

Table 4. Timetable of the path $p_{1,7}$ from $v_{s}=1$ to $v_{e}=7$ containing a cycle.

| Vertex/ <br> zone | Arrival <br> time | Departure <br> time | Bus <br> line | Cost of <br> travel |
| :---: | :---: | :---: | :---: | :---: |
| $1 / 1$ | $12: 00$ | $12: 05$ | 1 | 0.0 |
| $2 / 1$ | $12: 08$ | $12: 08$ | 1 | 2.0 |
| $3 / 1$ | $12: 11$ | $12: 11$ | 1 | 2.0 |
| $4 / 1$ | $12: 15$ | $12: 25$ | 2 | 2.0 |
| $5 / 1$ | $12: 28$ | $12: 28$ | 2 | 4.0 |
| $2 / 1$ | $12: 32$ | $12: 32$ | 2 | 4.0 |
| $6 / 1$ | $12: 35$ | $12: 45$ | 3 | 4.0 |
| $7 / 1$ | $12: 50$ |  | 3 | 6.0 |

The condition (20) is satisfied in similar cases. If $Z\left(\operatorname{sub}_{p_{v_{s}, v_{e}}}\left(v_{c}, v_{b}\right)\right) \geq 2$ then $Z\left(\operatorname{sub}_{p_{v_{s}, v_{e}}}\left(v_{d}, v_{c}\right)\right)$ does not influence the cost of travel and (20) is satisfied. Otherwise, we have $Z\left(\operatorname{sub}_{p_{v_{s}, v_{e}}}\left(v_{c}, v_{b}\right)\right)<2$, the condition (20) is satisfied if $Z\left(\operatorname{sub}_{p_{v_{s}, v_{e}}}\left(v_{c}, v_{d}\right)\right)=0$.

In the second part of the proof, (17) will be demonstrated. The condition

$$
\operatorname{sub}_{p_{v_{s}, v_{e}}}\left(v_{s}, v_{c}\right)=\operatorname{sub}_{p_{v_{s}, v_{e}}^{\prime}}\left(v_{s}, v_{c}\right)
$$

is satisfied. Then it follows that

$$
T\left(\operatorname{sub}_{p_{v_{s}, v_{e}}}\left(v_{s}, v_{c}\right)\right)=T\left(\operatorname{sub}_{p_{v_{s}, v_{e}}^{\prime}}\left(v_{s}, v_{c}\right)\right)
$$

is satisfied, too, and the total times of waiting for changes in vertices of the subpaths $\operatorname{sub}_{p_{v_{s}, v_{e}}}\left(v_{s}, v_{c-1}\right)$ and $\operatorname{sub}_{p_{v_{s}, v_{e}}^{\prime}}\left(v_{s}, v_{c-1}\right)$ are equal, where $v_{c-1}$ is the vertex which precedes $v_{c}$ in the paths. If the time of travel through the cycle $p_{v_{c}, v_{c}}$ equals the time of waiting for a change at $v_{c}$ in $p_{v_{s}, v_{e}}^{\prime}$, then the times of departure from $v_{c}$ in $p_{v_{s}, v_{e}}$ and $p_{v_{s}, v_{e}}^{\prime}$ are identical. It follows that

$$
T\left(\operatorname{sub}_{p_{v_{s}, v_{e}}}\left(v_{c}, v_{e}\right)\right)=T\left(\operatorname{sub}_{p_{v_{s}, v_{e}}^{\prime}}\left(v_{c}, v_{e}\right)\right)
$$

occurs and 17) is satisfied, and this proves Property 3 defined in the theorem.

When the time of travel through the cycle $p_{v_{c}, v_{c}}$ is longer than the time of waiting for a change at $v_{c}$ in $p_{v_{s}, v_{e}}^{\prime}$, then $(17)$ is satisfied if

$$
T\left(p_{v_{c}, v_{c}}\right)+\Delta t_{v_{c}, v_{e}}=\Delta t_{v_{c}, v_{e}}^{\prime}
$$

where $\Delta t_{v_{c}, v_{e}}$ and $\Delta t_{v_{c}, v_{e}}^{\prime}$ equal the total time of waiting for changes in the subpaths $\operatorname{sub}_{p_{v_{s}, v_{e}}}\left(v_{c}, v_{e}\right)$ and $\operatorname{sub}_{p_{v_{s}, v_{e}}^{\prime}}\left(v_{c}, v_{e}\right)$, respectively. In this case, Property 3 defined in the Theorem is satisfied, too.

We illustrate Theorem 1 with an example of determining paths from $v_{s}=1$ to $v_{e}=7$. The path $p_{1,7}$ containing the cycle $2 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 2$ is presented in Table 4 Table 5 shows the path $p_{1,7}^{\prime}$, which has the

[^1]Table 5. Timetable of the path $p_{1,7}^{\prime}$ from $v_{s}=1$ to $v_{e}=7$ without a cycle.

| Vertex/ <br> zone | Arrival <br> time | Departure <br> time | Bus <br> line | Cost of <br> travel |
| :---: | :---: | :---: | :---: | :---: |
| $1 / 1$ | $12: 00$ | $12: 05$ | 1 | 0.0 |
| $2 / 1$ | $12: 08$ | $12: 15$ | 2 | 2.0 |
| $6 / 1$ | $12: 18$ | $12: 45$ | 3 | 4.0 |
| $7 / 1$ | $12: 50$ |  | 3 | 6.0 |

same sequence of vertices and arcs as $p_{1,7}$ but is devoid a cycle. All vertices belonging to the cycle are located in the same zone, and thus we do not cross a zone border while running through the cycle and the travel through the cycle does not increase the cost of travel. Therefore the costs of travel of the paths $p_{1,7}$ and $p_{1,7}^{\prime}$ are equal and their value is 6.0 units. In the path $p_{1,7}^{\prime}$ we change at the vertices 2 and 6 , and the times of waiting for a change are equal to 7 and 27 minutes, respectively. The time of departure from the vertex 2 towards the vertex 6 in the path $p_{1,7}$ is later than in the path $p_{1,7}^{\prime}$; thus the time of waiting for change at the vertex 6 in the path $p_{1,7}$ is shorter and it equals 10 minutes. The time of making the cycle in the path $p_{1,7}$ equals 24 minutes and is longer than the time of waiting for change at vertex 2 in the path $p_{1,7}^{\prime}$. The sum of the time of making the cycle and the time of waiting for change at the vertex 6 in the path $p_{1,7}$ is 34 minutes. It equals the sum of times of waiting for a change at the vertices 2 and 6 in the path $p_{1,7}^{\prime}$. Therefore the times of travel of the paths $p_{1,7}$ and $p_{1,7}^{\prime}$ are equal and their value is 50 minutes. The path $p_{1,7}$ satisfies the conditions defined by Theorem 1 . The cycle contains only a single change at the vertex 4 , and the vertex 2 is passed without a change.

Let us consider the paths $p_{v_{s}, v_{e}}^{t}$ and $p_{v_{s}, v_{e}}^{c}$ with the minimal time and the minimal cost of travel from $v_{s}$ to $v_{e}$, respectively, and $c_{\max }=C\left(p_{v_{s}, v_{e}}^{t}\right)$ and $t_{\text {max }}=T\left(p_{v_{s}, v_{e}}^{c}\right)$. The values $t_{\text {max }}$ and $c_{\text {max }}$ determine the maximal time and the maximal cost of travel the path belonging to the set of non-dominated solutions. According to Definition 44 for the path $p_{v_{s}, v_{e}}$, if $C\left(p_{v_{s}, v_{e}}\right)>c_{\text {max }}$ is satisfied, then $p_{v_{s}, v_{e}}^{t} \succ p_{v_{s}, v_{e}}$. Similarly, $p_{v_{s}, v_{e}}^{c} \succ p_{v_{s}, v_{e}}$ if $T\left(p_{v_{s}, v_{e}}\right)>t_{\max }$ occurs. The value of $t_{\max }$ makes it possible to determine the latest time of arrival $T_{\max }^{e}$ to the final vertex $v_{e}$ in the path being a non-dominated solution, i.e.,

$$
\begin{equation*}
T_{\max }^{e}=T_{s}+t_{\max } \tag{21}
\end{equation*}
$$

### 2.4. Influence of dominated partial solutions on non-

 dominated final solutions. The partial solution $p_{v_{s}, v_{i}}$ can be extended to the final solution $p_{v_{s}, v_{e}}$ by determining the path from $v_{i}$ to $v_{e}$. There are several problems connected with this operation. Many partial solutions are determined for the given vertex $v_{i}$ during the process of finding the solutions and these partial solutions can becompared to each other according to the time and the cost of travel. If the partial solution is dominated by another partial solution, it is necessary to decide whether it should be stored and analysed or if it may be omitted. In consequence, it is important to know whether it is possible to extend a dominated partial solution and obtain from it a non-dominated final solution. An answer to this question contains conditions required to obtain a non-dominated final solution from a dominated partial solution which are presented in this subsection. They take into account, inter alia, on whether the vertex $v_{i}$ is passed with a change or without it. Therefore all possible cases are analysed. Next, it is necessary to define the conditions under which a dominated partial solution may be omitted because it is not possible to obtain a non-dominated final solution from it. The estimation is done based on the partial solutions already computed for the vertex $v_{i}$. This subsection resolves all of the mentioned problems and it contains all listed conditions.

Assume that there are two final solutions $p_{v_{s}, v_{e}}$ and $p_{v_{s}, v_{e}}^{\prime}$ obtained from the partial solutions $p_{v_{s}, v_{i}}$ and $p_{v_{s}, v_{i}}^{\prime}$, where $p_{v_{s}, v_{i}} \succ p_{v_{s}, v_{i}}^{\prime}$. If the weights of arcs are constant then it shown that the monotonicity assumption holds, i.e., a final solution $p_{v_{s}, v_{e}}^{\prime}$ obtained from a dominated partial solution $p_{v_{s}, v_{i}}^{\prime}$ is a dominated solution ( $p_{v_{s}, v_{e}} \succ$ $p_{v_{s}, v_{e}}^{\prime}$ ) and $p_{v_{s}, v_{e}}$ belongs to the set of non-dominated solutions solely if, for each $v_{i}$ belonging to $p_{v_{s}, v_{e}}$, the subpath $\operatorname{sub}_{p_{v_{s}, v_{e}}}\left(v_{s}, v_{i}\right)$ is a non-dominated solution, too (Azevedo and Martins, 1991; Carraway et al., 1990; Martins et al., 1999; Mote et al., 1991).

Lemma 3. Consider a weighed multigraph $G$, where the weights take non-negative and variable values, and two partial solutions $p_{v_{s}, v_{i}}$ and $p_{v_{s}, v_{i}}^{\prime}$, where $p_{v_{s}, v_{i}} \succ p_{v_{s}, v_{i}}^{\prime}$. Then the monotonicity assumption does not hold, and it is possible to obtain a non-dominated final solution $p_{v_{s}, v_{e}}^{\prime}$ from a dominated partial solution $p_{v_{s}, v_{i}}^{\prime}$.

Proof. According to Definition4 if $p_{v_{s}, v_{i}} \succ p_{v_{s}, v_{i}}^{\prime}$, then (22) or (23) holds:

$$
\begin{align*}
& T\left(p_{v_{s}, v_{i}}\right)<T\left(p_{v_{s}, v_{i}}^{\prime}\right) \wedge C\left(p_{v_{s}, v_{i}}\right) \leq C\left(p_{v_{s}, v_{i}}^{\prime}\right),  \tag{22}\\
& T\left(p_{v_{s}, v_{i}}\right) \leq T\left(p_{v_{s}, v_{i}}^{\prime}\right) \wedge C\left(p_{v_{s}, v_{i}}\right)<C\left(p_{v_{s}, v_{i}}^{\prime}\right) . \tag{23}
\end{align*}
$$

Let $\delta t$ and $\delta c$ be respectively the differences between the times and the costs of travel of the partial solutions $p_{v_{s}, v_{i}}^{\prime}$ and $p_{v_{s}, v_{i}}$ :

$$
\begin{aligned}
& \delta t=T\left(p_{v_{s}, v_{i}}^{\prime}\right)-T\left(p_{v_{s}, v_{i}}\right), \\
& \delta c=C\left(p_{v_{s}, v_{i}}^{\prime}\right)-C\left(p_{v_{s}, v_{i}}\right) .
\end{aligned}
$$

In order to prove the theorem, it is necessary to consider all possible cases of obtaining the final solution on the basis of a partial one. The time and the cost of travel of the final solutions $p_{v_{s}, v_{e}}^{\prime}$ and $p_{v_{s}, v_{e}}$ obtained from $p_{v_{s}, v_{i}}^{\prime}$
and $p_{v_{s}, v_{i}}$ will be analysed. Let us denote by $p_{v_{s}, v_{e}}^{\prime}$ and $p_{v_{s}, v_{e}}$ the concatenation of paths:

$$
\begin{align*}
& p_{v_{s}, v_{e}}=p_{v_{s}, v_{i}} \oplus p_{v_{i}, v_{e}}  \tag{24}\\
& p_{v_{s}, v_{e}}^{\prime}=p_{v_{s}, v_{i}}^{\prime} \oplus p_{v_{i}, v_{e}}^{\prime} \tag{25}
\end{align*}
$$

From the conditions (22) or (23) it follows that $\delta t \geq$ 0 and $\delta c \geq 0$. The relationship between the time of travel is defined by one of the following conditions:
$\mathrm{T} 1: T\left(p_{v_{i}, v_{e}}\right)=T\left(p_{v_{i}, v_{e}}^{\prime}\right)+\delta t \Rightarrow T\left(p_{v_{s}, v_{e}}\right)=T\left(p_{v_{s}, v_{e}}^{\prime}\right)$,
T2: $T\left(p_{v_{i}, v_{e}}\right)>T\left(p_{v_{i}, v_{e}}^{\prime}\right)+\delta t \Rightarrow T\left(p_{v_{s}, v_{e}}\right)>T\left(p_{v_{s}, v_{e}}^{\prime}\right)$,
T3: $T\left(p_{v_{i}, v_{e}}\right)<T\left(p_{v_{i}, v_{e}}^{\prime}\right)+\delta t \Rightarrow T\left(p_{v_{s}, v_{e}}\right)<T\left(p_{v_{s}, v_{e}}^{\prime}\right)$. If $\delta t=0$, then the relationship between the time of travel depends only on the time of travel from $v_{i}$ to $v_{e}$. A similar relationship occurs between the cost of travel, i.e.,

$$
\left.\begin{array}{rl}
\mathrm{C} 1: & C\left(p_{v_{i}, v_{e}}\right)-\Delta c+\delta c=C\left(p_{v_{i}, v_{e}}^{\prime}\right)-\Delta c^{\prime} \Rightarrow \\
& C\left(p_{v_{s}, v_{e}}\right)
\end{array}\right)=C\left(p_{v_{s}, v_{e}}^{\prime}\right) . ~\left(\begin{array}{rl}
\mathrm{C} 2: & C\left(p_{v_{i}, v_{e}}\right)-\Delta c+\delta c>C\left(p_{v_{i}, v_{e}}^{\prime}\right)-\Delta c^{\prime} \Rightarrow \\
& C\left(p_{v_{s}, v_{e}}\right)>C\left(p_{v_{s}, v_{e}}^{\prime}\right), \\
\mathrm{C} 3: & C\left(p_{v_{i}, v_{e}}\right)-\Delta c+\delta c<C\left(p_{v_{i}, v_{e}}^{\prime}\right)-\Delta c^{\prime} \Rightarrow \\
& C\left(p_{v_{s}, v_{e}}\right)<C\left(p_{v_{s}, v_{e}}^{\prime}\right),
\end{array}\right.
$$

where $\Delta c$ and $\Delta c^{\prime}$ are equal to the decrease in the cost of travel of the paths (24) and (25) when the vertex $v_{i}$ is passed without a change. According to Definition 4 the following conditions are satisfied:

$$
\begin{array}{r}
\delta c=0 \quad \Rightarrow \quad \delta t>0 \\
\delta t=0 \quad \Rightarrow \quad \delta c>0
\end{array}
$$

On the basis of T1-T3 and $\mathrm{C} 1-\mathrm{C} 3$ it is possible to determine nine relationships between the final solutions $p_{v_{s}, v_{e}}$ and $p_{v_{s}, v_{e}}^{\prime}$, denoted by R1-R9:
$\mathrm{R} 1: \mathrm{C} 1 \wedge \mathrm{~T} 1 \Rightarrow p_{v_{s}, v_{e}}$ and $p_{v_{s}, v_{e}}^{\prime}$ are non-dominated,
R2: $\mathrm{C} 1 \wedge \mathrm{~T} 2 \Rightarrow p_{v_{s}, v_{e}}^{\prime} \succ p_{v_{s}, v_{e}}$,
R3: $\mathrm{C} 1 \wedge \mathrm{~T} 3 \Rightarrow p_{v_{s}, v_{e}} \succ p_{v_{s}, v_{e}}^{\prime}$,
$\mathrm{R} 4: \mathrm{C} 2 \wedge \mathrm{~T} 1 \Rightarrow p_{v_{s}, v_{e}}^{\prime} \succ p_{v_{s}, v_{e}}$,
R5: $\mathrm{C} 2 \wedge \mathrm{~T} 2 \Rightarrow p_{v_{s}, v_{e}}^{\prime} \succ p_{v_{s}, v_{e}}$,
R6: $\mathrm{C} 2 \wedge \mathrm{~T} 3 \Rightarrow p_{v_{s}, v_{e}}$ and $p_{v_{s}, v_{e}}^{\prime}$ are non-dominated,
$\mathrm{R} 7: \mathrm{C} 3 \wedge \mathrm{~T} 1 \Rightarrow p_{v_{s}, v_{e}} \succ p_{v_{s}, v_{e}}^{\prime}$,
R8: $\mathrm{C} 3 \wedge \mathrm{~T} 2 \Rightarrow p_{v_{s}, v_{e}}$ and $p_{v_{s}, v_{e}}^{\prime}$ are non-dominated,
$\mathrm{R} 9: \mathrm{C} 3 \wedge \mathrm{~T} 3 \Rightarrow p_{v_{s}, v_{e}} \succ p_{v_{s}, v_{e}}^{\prime}$.
A possible change at the vertex $v_{i}$ (Fig.(2) determines whether a given relationship of R1-R9 is actually satisfied or just only theoretically and never occurs. For that reason the following cases are considered:

1. passing $v_{i}$ with a change in $p_{v_{s}, v_{e}}$ and $p_{v_{s}, v_{e}}^{\prime}$,
2. passing $v_{i}$ with a change only in $p_{v_{s}, v_{e}}^{\prime}$,
3. passing $v_{i}$ with a change only in $p_{v_{s}, v_{e}}$,
4. passing $v_{i}$ without a change in $p_{v_{s}, v_{e}}$ and $p_{v_{s}, v_{e}}^{\prime}$.

In the first case, travel with a change at $v_{i}$ in $p_{v_{s}, v_{e}}$ and $p_{v_{s}, v_{e}}^{\prime}$ is considered. It follows that $\Delta c=\Delta c^{\prime}=0$, and the time and the cost of travel of $p_{v_{s}, v_{e}}$ and $p_{v_{s}, v_{e}}^{\prime}$ only depend on $p_{v_{i}, v_{e}}$ and $p_{v_{i}, v_{e}}^{\prime}$, which are subpaths of $p_{v_{s}, v_{e}}$ and $p_{v_{s}, v_{e}}^{\prime}$, respectively:

$$
\begin{aligned}
p_{v_{i}, v_{e}} & =\operatorname{sub}_{p_{v_{s}, v_{e}}}\left(v_{i}, v_{e}\right) \\
p_{v_{i}, v_{e}}^{\prime} & =\operatorname{sub}_{p_{v_{s}, v_{e}}}\left(v_{i}, v_{e}\right)
\end{aligned}
$$

If $p_{v_{i}, v_{e}}$ and $p_{v_{i}, v_{e}}^{\prime}$ are identical, then

$$
\begin{aligned}
& T\left(p_{v_{i}, v_{e}}\right)=T\left(p_{v_{i}, v_{e}}^{\prime}\right) \\
& C\left(p_{v_{i}, v_{e}}^{\prime}\right)=C\left(p_{v_{i}, v_{e}}^{\prime}\right)
\end{aligned}
$$

and according to (22) and (23), the following conditions are satisfied:

$$
\begin{align*}
& T\left(p_{v_{s}, v_{e}}\right) \leq T\left(p_{v_{s}, v_{e}}^{\prime}\right)  \tag{26}\\
& C\left(p_{v_{s}, v_{e}}\right) \leq C\left(p_{v_{s}, v_{e}}^{\prime}\right) \tag{27}
\end{align*}
$$

It follows that $p_{v_{s}, v_{i}}^{\prime} \succ p_{v_{s}, v_{i}}$ never occurs and the relationships R2, R4 and R5 are not satisfied.

It should be pointed out that it is not possible to obtain $p_{v_{s}, v_{e}}^{\prime}$ for which (26) or (27) is not satisfied. This would mean that $p_{v_{i}, v_{e}}$ and $p_{v_{i}, v_{e}}^{\prime}$ are different and one of the following conditions is satisfied:

$$
\begin{aligned}
& T\left(p_{v_{i}, v_{e}}\right)>T\left(p_{v_{i}, v_{e}}^{\prime}\right), \\
& C\left(p_{v_{i}, v_{e}}\right)>C\left(p_{v_{i}, v_{e}}^{\prime}\right)
\end{aligned}
$$

In this case, $p_{v_{s}, v_{e}}$ would be created as a concatenation of paths: $p_{v_{s}, v_{e}}=p_{v_{s}, v_{i}} \oplus p_{v_{i}, v_{e}}^{\prime}$, then the subpaths $\operatorname{sub}_{p_{v_{s}, v_{e}}}\left(v_{i}, v_{e}\right)$ and $\operatorname{sub}_{p_{p_{s}, v_{e}}^{\prime}}\left(v_{i}, v_{e}\right)$ are identical and the conditions (26) and 27) occur.

In the second case, a change at $v_{i}$ in $p_{v_{s}, v_{e}}^{\prime}$ is but in $p_{v_{s}, v_{e}}$ it is passed without a change. It follows that $\Delta c^{\prime}=0$ and $\Delta c \neq 0$. If the subpaths $\operatorname{sub}_{p_{v_{s}, v_{e}}}\left(v_{i}, v_{e}\right)$ and $\operatorname{sub}_{p_{v_{s}, v_{e}}^{\prime}}\left(v_{i}, v_{e}\right)$ are identical, then

$$
\begin{aligned}
& T\left(p_{v_{s}, v_{e}}\right) \leq T\left(p_{v_{s}, v_{e}}^{\prime}\right) \\
& C\left(p_{v_{s}, v_{e}}\right)<C\left(p_{v_{s}, v_{e}}^{\prime}\right)
\end{aligned}
$$

Therefore, $p_{v_{s}, v_{i}} \succ p_{v_{s}, v_{i}}^{\prime}$ and only the relationship R7 or R9 can be satisfied. When the subpaths $\operatorname{sub}_{p_{v_{s}, v_{e}}}\left(v_{i}, v_{e}\right)$ and $\operatorname{sub}_{p_{v_{s}, v_{e}}^{\prime}}\left(v_{i}, v_{e}\right)$ are different, then it is not possible to determine the relationship between the time and the cost of travel of the paths $p_{v_{s}, v_{e}}$ and $p_{v_{s}, v_{e}}^{\prime}$, thus any of the relationships R1-R9 can be fulfilled.

Table 6. Possibility of fulfillment of the relationships R1-R9 between the final solutions $p_{v_{s}, v_{e}}$ and $p_{v_{s}, v_{e}}^{\prime}$ passing the vertex $v_{i}$ with or without a change and the subpaths $\operatorname{sub}_{p_{v_{s}, v_{e}}}\left(v_{i}, v_{e}\right)$ and $\operatorname{sub}_{p_{v_{s}, v e}^{\prime}}\left(v_{i}, v_{e}\right)$ identical or different.


In the next case, $v_{i}$ is passed without a change in $p_{v_{s}, v_{e}}^{\prime}$ obtained from the dominated partial solution $p_{v_{s}, v_{i}}^{\prime}$ and a change is performed in $p_{v_{s}, v_{e}}$, thus $\Delta c^{\prime} \neq 0$ and $\Delta c=0$. When the subpaths $\operatorname{sub}_{p_{v_{s}, v_{e}}}\left(v_{i}, v_{e}\right)$ and $\operatorname{sub}_{p_{v_{s}, v_{e}}^{\prime}}\left(v_{i}, v_{e}\right)$ are identical, then

$$
\begin{equation*}
T\left(p_{v_{s}, v_{e}}\right) \leq T\left(p_{v_{s}, v_{e}}^{\prime}\right) \tag{28}
\end{equation*}
$$

but the cost of travel depends on $\Delta c^{\prime}$ and $\delta c$, and one of the following conditions is satisfied:

$$
\begin{align*}
& \Delta c^{\prime}=\delta c \Rightarrow C\left(p_{v_{s}, v_{e}}\right)=C\left(p_{v_{s}, v_{e}}^{\prime}\right),  \tag{29}\\
& \Delta c^{\prime}>\delta c \Rightarrow C\left(p_{v_{s}, v_{e}}\right)>C\left(p_{v_{s}, v_{e}}^{\prime}\right),  \tag{30}\\
& \Delta c^{\prime}<\delta c \Rightarrow C\left(p_{v_{s}, v_{e}}\right)<C\left(p_{v_{s}, v_{e}}^{\prime}\right) . \tag{31}
\end{align*}
$$

According to (28)-(31), the relationships R2, R5 and R8 never occur. If the subpaths $\operatorname{sub}_{p_{v_{s}, v_{e}}}\left(v_{i}, v_{e}\right)$ and $\operatorname{sub}_{p_{v_{s}, v_{e}}}\left(v_{i}, v_{e}\right)$ are different, then any of the relationships $\mathrm{T} 1-\mathrm{T} 3$ and $\mathrm{C} 1-\mathrm{C} 3$ can occur, thus any of the relationships R1-R9 can be fulfilled.

In the last case, travel without a change at $v_{i}$ in $p_{v_{s}, v_{e}}$ and $p_{v_{s}, v_{e}}^{\prime}$ is considered, thus $\Delta c \neq 0$ and $\Delta c^{\prime} \neq 0$. First, the case when the subpaths $\operatorname{sub}_{p_{v_{s}, v_{e}}}\left(v_{i}, v_{e}\right)$ and $\operatorname{sub}_{p_{v_{s}, v_{e}}^{\prime}}\left(v_{i}, v_{e}\right)$ are identical is considered. It should be pointed out that $v_{i}$ in the partial solutions $p_{v_{s}, v_{i}}$ and $p_{v_{s}, v_{i}}^{\prime}$ is reached by a bus of the same line. The conditions (22) or (23) is satisfied and it follows that

$$
T\left(p_{v_{s}, v_{e}}\right) \leq T\left(p_{v_{s}, v_{e}}^{\prime}\right)
$$

The cost of travel depends on $\Delta c, \Delta c^{\prime}$ and $\delta c$, therefore one of the following conditions can be fulfilled:

$$
\begin{aligned}
& \Delta c^{\prime}=\Delta c+\delta c \Rightarrow C\left(p_{v_{s}, v_{e}}\right)=C\left(p_{v_{s}, v_{e}}^{\prime}\right), \\
& \Delta c^{\prime}>\Delta c+\delta c \Rightarrow C\left(p_{v_{s}, v_{e}}\right)>C\left(p_{v_{s}, v_{e}}^{\prime}\right), \\
& \Delta c^{\prime}<\Delta c+\delta c \Rightarrow C\left(p_{v_{s}, v_{e}}\right)<C\left(p_{v_{s}, v_{e}}^{\prime}\right) .
\end{aligned}
$$

For that reason the relationships R2, R5 and R8 never occur.

If the subpaths $\operatorname{sub}_{p_{v_{s}, v_{e}}}\left(v_{i}, v_{e}\right)$ and $\operatorname{sub}_{p_{v_{s}, v_{e}}^{\prime}}\left(v_{i}, v_{e}\right)$ are different then any of the relationships $\mathrm{T} 1-\mathrm{T} 3$ and $\mathrm{C} 1-\mathrm{C} 3$ can be satisfied and any of the relationships R1-R9 can be fulfilled.

To complete the proof, the possibility of the fulfillment of the relationships R1-R9 between the final solutions $p_{v_{s}, v_{e}}$ and $p_{v_{s}, v_{e}}^{\prime}$ in each case is presented in Table 6 If the given relationship never occurs, then it is denoted by the symbol "-", otherwise, i.e., it can be satisfied, it is denoted by the symbol " + ".

According to Lemma 3, a final solution obtained from a dominated partial solution can be a non-dominated one. Thus, during the process of finding the solutions, it is necessary to analyse dominated partial solutions and they cannot be omitted. For a given partial solution $p_{v_{s}, v_{i}}$ one can estimate whether the final solution obtained from it will be a dominated one. This can be estimated on the basis of the values $t_{\text {max }}, c_{\text {max }}, t_{\text {min }}^{i}, c_{\text {min }}^{i}$ and $\Delta c$, where $t_{\text {min }}^{i}$ and $c_{\text {min }}^{i}$ are equal respectively the minimal time and the minimal cost of travel from $v_{i}$ to the final vertex $v_{e}$. The time of travel from $v_{i}$ to $v_{e}$, depends on the time of arrival to $v_{i}$, therefore the value $t_{\text {min }}^{i}$ equals the minimal time of travel from $v_{i}$ to $v_{e}$ which guarantees arrival to $v_{e}$ no later than the time $T_{\max }^{e}$ (21). As mentioned in Section 2.3, if the time of arrival to $v_{e}$ is later than $T_{\max }^{e}$, then the solution is dominated.

Theorem 2. (Widuch, 2012) Let $p_{v_{s}, v_{i}}$ be a partial solution representing a path from the start vertex $v_{s}$ to the vertex $v_{i}$. It is possible to obtain a non-dominated final solution from $p_{v_{s}, v_{i}}$ even when the following condition is satisfied:

$$
C\left(p_{v_{s}, v_{i}}\right)+c_{\min }^{i}-\Delta c>c_{\max } .
$$

It can be estimated that the final solution obtained from the partial solution $p_{v_{s}, v_{i}}$ will be a dominated one. The partial solution $p_{v_{s}, v_{i}}$ can be omitted if it is not possible to obtain a non-dominated final solution from it. When it is estimated on the basis of the cost of travel, then the possibility of travel through the vertex $v_{i}$ without a change and travel to any vertex which is a successor of $v_{i}$ in the path of this line should be taken into account. Thus we must assume the value $\Delta c_{\max }$ being a maximal value of the decreasing of the cost of travel which takes into account the number of zones $Z_{a}$ that we crossed since the last change while travelling to $v_{i}$ (Table 3).

The partial solution $p_{v_{s}, v_{i}}$ can be omitted when

$$
\begin{equation*}
C\left(p_{v_{s}, v_{i}}\right)+c_{\min }^{i}-\Delta c_{\max }>c_{\max } \tag{32}
\end{equation*}
$$

occurs. If the vertex $v_{i}$ is the final vertex of the line by whose bus we arrive at $v_{i}$ in $p_{v_{s}, v_{i}}$, then we must change at $v_{i}$ and in this case $\Delta c_{\max }=0.0$ is assumed. The partial solution can be estimated on the basis of the time of travel and can be omitted if condition

$$
\begin{equation*}
T\left(p_{v_{s}, v_{i}}\right)+t_{\min }^{i}>t_{\max } \tag{33}
\end{equation*}
$$

is satisfied.
The partial solutions already computed can be used to estimate, given the partial solution $p_{v_{s}, v_{i}}$, whether the final solution obtained from it will be a dominated one. We can stop analysing a dominated partial solution and it can be omitted when the conditions determined by Theorem 3 are satisfied. Otherwise we have to continue analysing it.

Theorem 3. Consider partial solutions $p_{v_{s}, v_{i}}$ and $p_{v_{s}, v_{i}}^{\prime}$. The final solution obtained from $p_{v_{s}, v_{i}}$ will be a dominated one if

$$
\begin{align*}
T\left(p_{v_{s}, v_{i}}\right) & \geq T\left(p_{v_{s}, v_{i}}^{\prime}\right)  \tag{34}\\
C\left(p_{v_{s}, v_{i}}\right)-\Delta c_{\max } & >C\left(p_{v_{s}, v_{i}}^{\prime}\right) \tag{35}
\end{align*}
$$

Therefore $p_{v_{s}, v_{i}}$ can be omitted.
Proof. Let us consider the final solutions $p_{v_{s}, v_{e}}$ and $p_{v_{s}, v_{e}}^{\prime}$ respectively obtained from $p_{v_{s}, v_{i}}$ and $p_{v_{s}, v_{i}}^{\prime}$,

$$
\begin{aligned}
& p_{v_{s}, v_{e}}=p_{v_{s}, v_{i}} \oplus p_{v_{i}, v_{e}}, \\
& p_{v_{s}, v_{e}}^{\prime}=p_{v_{s}, v_{i}}^{\prime} \oplus p_{v_{i}, v_{e}}^{\prime} .
\end{aligned}
$$

From (34) it follows that the time of arrival to $v_{i}$ in $p_{v_{s}, v_{i}}$ is not earlier than the time of arrival to $v_{i}$ in $p_{v_{s}, v_{i}}^{\prime}$, and the time of departure from $v_{i}$ can be earlier in $p_{v_{s}, v_{i}}^{\prime}$. Therefore the subpaths $p_{v_{i}, v_{e}}$ and $p_{v_{i}, v_{e}}^{\prime}$ from $v_{i}$ to the final vertex $v_{e}$ in the paths $p_{v_{s}, v_{e}}$ and $p_{v_{s}, v_{e}}^{\prime}$ can be different. This case is not considered for the following reason. The subpaths $p_{v_{i}, v_{e}}$ and $p_{v_{i}, v_{e}}^{\prime}$ can be identical and this case decides on negative estimation of the partial solution $p_{v_{s}, v_{i}}$.

If the subpaths $p_{v_{i}, v_{e}}$ and $p_{v_{i}, v_{e}}^{\prime}$ are identical, then from (34) it follows that

$$
\begin{equation*}
T\left(p_{v_{s}, v_{e}}\right) \geq T\left(p_{v_{s}, v_{e}}^{\prime}\right) \tag{36}
\end{equation*}
$$

If (35) holds, then

$$
\begin{equation*}
C\left(p_{v_{s}, v_{e}}\right)>C\left(p_{v_{s}, v_{e}}^{\prime}\right) \tag{37}
\end{equation*}
$$

regardless of whether a change at $v_{i}$ is performed it is passed without a change.

From (36) and 37) it follows that $p_{v_{s}, v_{e}}$ is a dominated solution and it is not possible to obtain a non-dominated partial solution from $p_{v_{s}, v_{i}}$ when (34) and (35) are satisfied.

Consider the vertex $v_{i}$ and the list $L P S\left[v_{i}\right]$ containing $k$ partial solutions $p_{v_{s}, v_{i}}^{1}, \ldots, p_{v_{s}, v_{i}}^{k}$ determined for the vertex $v_{i}$. Let $t_{\text {min }}^{p i}, t_{\text {max }}^{p i}$ and $c_{\text {max }}^{p i}$ be respectively the minimal and the maximal time of travel and the maximal cost of travel from among all partial solutions in the list $L P S\left[v_{i}\right]$ :

$$
\begin{aligned}
t_{\min }^{p i} & =\min \left\{T\left(p_{v_{s}, v_{i}}^{1}\right), \ldots, T\left(p_{v_{s}, v_{i}}^{k}\right)\right\}, \\
t_{\max }^{p i} & =\max \left\{T\left(p_{v_{s}, v_{i}}^{1}\right), \ldots, T\left(p_{v_{s}, v_{i}}^{k}\right)\right\}, \\
c_{\max }^{p i} & =\max \left\{C\left(p_{v_{s}, v_{i}}^{1}\right), \ldots, C\left(p_{v_{s}, v_{i}}^{k}\right)\right\} .
\end{aligned}
$$

Let $c_{\text {min }}^{p i}$ signify the minimal cost of travel decreased by the value of $\Delta c_{\text {max }}^{i}(i=1, \ldots, k)$ from among all partial solutions, i.e.,

$$
\begin{gathered}
c_{\min }^{p i}=\max \left\{C\left(p_{v_{s}, v_{i}}^{1}\right)-\Delta c_{\max }^{1}, \ldots,\right. \\
\left.C\left(p_{v_{s}, v_{i}}^{k}\right)-\Delta c_{\max }^{k}\right\}
\end{gathered}
$$

The estimation of the partial solution $p_{v_{s}, v_{i}}$ can be done on the basis of the values of $t_{\max }^{p i}$ and $c_{\max }^{p i}$. According to Theorem [3] the partial solution $p_{v_{s}, v_{i}}$ can be omitted when

$$
\begin{equation*}
T\left(p_{v_{s}, v_{i}}\right) \geq t_{\max }^{p i} \wedge C\left(p_{v_{s}, v_{i}}\right)-\Delta c_{\max }>c_{\max }^{p i} \tag{38}
\end{equation*}
$$

is satisfied. The final solution obtained from it will be a dominated one. Additionally, the partial solutions stored in the list $\operatorname{LPS}\left[v_{i}\right]$ can be estimated on the basis of $p_{v_{s}, v_{i}}$ and the values of $c_{\text {min }}^{p i}$ and $t_{\text {min }}^{p i}$. If

$$
\begin{equation*}
T\left(p_{v_{s}, v_{i}}\right) \leq t_{\min }^{p i} \wedge C\left(p_{v_{s}, v_{i}}\right)<c_{\min }^{p i}, \tag{39}
\end{equation*}
$$

then all partial solutions can be removed from the list $L P S\left[v_{i}\right]$. These partial solutions can be omitted because it is not possible to obtain a non-dominated final solution from any of them. The estimation is done only by checking the (38) and (39) conditions without comparing $p_{v_{s}, v_{i}}$ with solutions stored in the list $L P S\left[v_{i}\right]$. Thus, it is done in $O(1)$ time and does not depend on the number of partial solutions in the list $L P S\left[v_{i}\right]$.
2.5. Algorithm for solving the BBR problem. The algorithm for finding all non-dominated paths from the start vertex $v_{s}$ to the final vertex $v_{e}$ and for the time of starting travel $T_{s}$ at $v_{s}$ is presented as Algorithm 1 It belongs to the group of label correcting algorithms with storing partial solutions and implements the methods of estimation of partial solutions presented in Section 2.4. During the process of finding the solutions, the following data structures are used:

- $\operatorname{LPS}\left[v_{i}\right]$ : the list of computed partial solutions for the vertex $v_{i} \in V$; for the final vertex $v_{e}$ it contains the final solutions formed as the set of non-dominated solutions;
- LFS: the list of final solutions constitute the set of non-dominated solutions, where each solution represents the path from $v_{s}$ to $v_{e}$ for the time of starting travel equal to $T_{s}$;
- $Q$ : the queue containing vertices for which the partial solutions have been computed.
The solutions stored in the list $L P S\left[v_{i}\right]$ are represented by a record $\left(t_{s i}, c_{s i}, v_{k}, l_{i}, T_{k}, L P P_{k}\right)$ containing the following data:
- $t_{s i}, c_{s i}$ : the time and the cost of travel from $v_{s}$ to $v_{i}$,
- $v_{k}$ : the vertex which precedes $v_{i}$ in the path,
- $l_{i}$ : the bus line by which we arrive from $v_{k}$ to $v_{i}$,
- $T_{k}$ : the time of departure from $v_{k}$ towards $v_{i}$,
- $L P P_{k}$ : the list of pointers to the partial solutions for the vertex $v_{k}$.

In the initial part of the algorithm (lines 1.7) the values $t_{\text {max }}, c_{\text {max }}, t_{\text {min }}^{i}$ and $c_{\text {min }}^{i}$ are computed by the procedure FInDTC. The values $t_{\text {max }}$ and $c_{\text {max }}$ are computed by determining adequately the path of the minimal cost and that of the minimal time of travel from $v_{s}$ to $v_{e}$. The path of the minimal time of travel is determined in the multigraph $G=(V, E)$ using the Dijkstra algorithm. It is not possible to find the path of the minimal cost of travel in the multigraph $G$ using this algorithm or the other shortest path one (Widuch, 2012). Thus it is necessary to create the multigraph $G^{\prime}=\left(V, E^{\prime}\right)$, $E \subset E^{\prime}$, obtained from $G$, by adding additional arcs for each bus line. Let the path of the $i$-th bus line contains following sequence of vertices:

$$
\left\langle v_{0}^{i}, v_{1}^{i}, \ldots, v_{k-1}^{i}, v_{k}^{i}\right\rangle
$$

For each pair of vertices $v_{a}^{i}$ and $v_{b}^{i}(a=0, \ldots, k-2$, $b=a+2, \ldots, k)$ an $\operatorname{arc}\left(v_{a}^{i}, v_{b}^{i}\right)$ is added to the multigraph $G^{\prime}$. The path of the minimal cost of travel is determined in the multigraph $G^{\prime}=\left(V, E^{\prime}\right)$ using the Dijkstra algorithm.

```
Algorithm 1. Algorithm SolveBBR for finding all
non-dominated paths from the start vertex \(v_{s}\) to the final
vertex \(v_{e}\) and for the time of starting travel \(T_{s}\).
Input: \(v_{s}, v_{e}, T_{s}, G=(V, E)\)
Output: LFS
    \(t_{\text {max }}, c_{\text {max }}, t_{\text {min }}^{i}, c_{\text {min }}^{i} \leftarrow \operatorname{FinDTC}\left(v_{s}, v_{e}, T_{s}, G\right)\);
    \(G \leftarrow \operatorname{MODIFYG}\left(G, t_{\text {min }}^{i}, t_{\text {max }}, c_{\text {min }}^{i}, c_{\text {max }}\right)\);
    for all \(v_{i} \in V\) do
        \(L P S\left[v_{i}\right] \leftarrow \emptyset ;\)
    end for
    \(S_{s} \leftarrow\left(0,0,0,0, T_{s}, \emptyset\right) ; \operatorname{LPS}\left[v_{s}\right] . \operatorname{ADDToList}\left(S_{s}\right) ;\)
    \(Q \leftarrow \emptyset ; Q . \operatorname{PuSh}\left(v_{s}\right) ;\)
    while not \(Q\).Empty() do
        \(v_{k} \leftarrow Q . \operatorname{POP}() ;\)
        for all \(S_{k} \in L P S\left[v_{k}\right]\) has not yet been analysed do
            for all \(\left(v_{k}, v_{i}, l_{i}\right) \in \operatorname{out}\left(v_{k}\right) ; v_{i} \neq v_{s}\) do
                if not \(\operatorname{InPath}\left(S_{k}, v_{i}\right)\) then
                    \(t_{s i}, c_{s i} \leftarrow\) the time and the cost of travel
                    from \(v_{s}\) to \(v_{i}\);
                    \(T_{k} \leftarrow\) the time of departure from \(v_{k}\);
                    \(\Delta c_{\max } \leftarrow\) the maximal decreasing of the
                    cost of travel from \(v_{s}\) to \(v_{i}\);
                    \(S_{i} \leftarrow\left(t_{s i}, c_{s i}, v_{k}, l_{i}, T_{k}, \& S_{k}\right)\);
                    if estimation of \(S_{i}\) is positive then
                    if \(v_{i}=v_{e}\) then
                        \(L P S\left[v_{i}\right] \Leftarrow \operatorname{ADDFSoL}\left(L P S\left[v_{i}\right], S_{i}\right) ;\)
                    else
                        \(L P S\left[v_{i}\right] \Leftarrow \operatorname{ADDPSol}\left(L P S\left[v_{i}\right], S_{i}\right.\),
                        \(\left.\Delta c_{\text {max }}, t_{\text {max }}, c_{\text {max }}, t_{\text {min }}^{i}, c_{\text {min }}^{i}\right)\);
                        \(Q . \operatorname{PUSH}\left(v_{i}\right)\);
                    end if
                end if
                end if
        end for
        end for
    end while
    \(L F S \Leftarrow \operatorname{CrEatEFULLPathS}\left(L P S\left[v_{e}\right], v_{e}\right)\);
    \(L F S \Leftarrow\) PathsDifferTimes \((L F S, G)\);
    \(L F S \Leftarrow\) PathsContainingCycles \((L F S, G)\);
    return \(L F S\);
```

The Dijkstra algorithm and the multigraphs $G$ and $G^{\prime}$ are also used in the procedure FIndTC for determining the minimal times and the minimal costs of travel from each vertex $v_{i} \in V$ to $v_{e}$, and these values are stored in $t_{\text {min }}^{i}$ and $c_{\text {min }}^{i}$. The value $t_{\text {min }}^{i}$ guarantees arrival to $v_{e}$ no later than the time $T_{\text {max }}^{e}$.

The values of $t_{\text {max }}, c_{\text {max }}, t_{\text {min }}^{i}$ and $c_{\text {min }}^{i}$ make it possible to determine a set of vertices that do not belong to a path being a non-dominated solution. If one of the following conditions is satisfied:

$$
\begin{align*}
t_{\min }^{i}\left[v_{i}\right] & \geq t_{\max },  \tag{40}\\
c_{\min }^{i}\left[v_{i}\right]-\Delta c_{\max }^{i} & >c_{\max }, \tag{41}
\end{align*}
$$

then a path containing the vertex $v_{i}$ does not belong to the set of non-dominated solutions. Thus $v_{i}$ can be removed from the multigraph $G$. Therefore $G$ is modified (line 2 of Algorithm (1) by the procedure ModifyG, which removes from $G$ all vertices $v_{i}$ for which (40) or (41) is satisfied. Additionally, all arcs incoming into $v_{i}$ and all arcs outgoing from $v_{i}$ are removed from $G$.

In the next steps of the initial part of the algorithm, for each vertex $v_{i} \in V\left(v_{i} \neq v_{s}\right)$ the list of partial solutions $L P S\left[v_{i}\right]$ is initialised as empty (lines 3-5). The list $L P S\left[v_{s}\right]$ of the start vertex $v_{s}$ is initialised by an initial solution (line 6) from which the algorithm starts computation and $v_{s}$ is inserted into the queue $Q$ (line7).

The paths from $v_{s}$ to $v_{e}$ belonging to the set of non-dominated solutions are computed in the while-loop (lines 8-28) by visiting vertices of the multigraph $G$ using a modified breadth first search method (Jungnickel, 1999). There are computed all paths belonging to the set of non-dominated solutions except for those containing cycles, and paths differ from each other only in the times of departure from vertices. These paths are computed by the procedures PathsContainingCycles and PathsDIFFERTIMES (lines 30-31) based on the paths computed in the while-loop.

In a single iteration of the while-loop the following operations are executed. The first vertex $v_{k}$ from the queue $Q$ is taken (line 9). In the for-loop (lines 10-27) we try to extend each unanalyzed partial solution $S_{k}$ in the list $L P S\left[v_{k}\right]$ (each partial solution is analysed only once) and obtain new solutions for all adjacent vertices of the vertex $v_{k}$. For this purpose, in the for-loop all outgoing $\operatorname{arcs}\left(v_{k}, v_{i}, l_{i}\right)$ of the vertex $v_{k}$ are analysed (lines 11-26). The arc $\left(v_{k}, v_{i}, l_{i}\right)$ corresponds to the $l_{i}$ bus line whose buses run directly from $v_{k}$ to $v_{i}$. If $v_{i}$ does not belong to the path represented by the partial solution $S_{k}$ (it is checked by the procedure InPath in line 12), then the time $t_{s i}$ and the cost $c_{s i}$ of travel from $v_{s}$ to $v_{i}$ as well as the time of departure $T_{k}$ from $v_{k}$ are computed (lines 13. 14). Additionally, a maximal decrease in the cost of travel $\Delta c_{\text {max }}$ is determined (line 15) and a new solution $S_{i}$ is created (line16). The solution $S_{i}$ represents a path from $v_{s}$ to $v_{i}$ and the vertex $v_{k}$ precedes $v_{i}$ in the path.

The solution $S_{i}$ is estimated if it is possible to obtain a non-dominated final solution from it. It is omitted if (32) or (33) is satisfied, which is checked in line 17 Otherwise, $S_{i}$ is inserted into the list $L P S\left[v_{i}\right]$. If $v_{i}=v_{e}$, then $S_{i}$ is inserted into $L P S\left[v_{i}\right]$ by the procedure ADDFSOL (line 191, otherwise the procedure ADDPSOL is used and $v_{i}$ is inserted into the queue $Q$ (lines 21, 22).

In the last part of the algorithm, for all computed solutions in the list $L P S\left[v_{e}\right]$ full paths are determined by the procedure CreateFullPaths (line 29), and these paths are stored in the list $L F S$. The full path is determined based on values stored in the record representing a solution in the list $\operatorname{LPS}\left[v_{e}\right]$. The record contains the vertex
$v_{k}$, which precedes the vertex $v_{e}$ in the path, and contains the list $L P P_{k}$, which stores pointers to the partial solutions for the vertex $v_{k}$. Thus, the path is created from $v_{e}$ by visiting the vertices which precede the current vertex until the start vertex $v_{s}$ is reached. The full path is determined on the basis of the vertex $v_{k}$ and the list of pointers $L P P_{k}$ stored in the record representing the solution.

Next, paths that differ from each other only in the times of departure from vertices belonging to the path are determined by the procedure PathsDifferTimes (line 30). According to Lemma 1 we change at least once in these paths. Thus in the procedure each path $p_{v_{s}, v_{e}}$ stored in the list LFS is analysed but only the paths containing at least single vertex of change are taken into consideration. Let $p_{v_{s}, v_{e}}$ contain $k(k>0)$ vertices of changes: $v_{0}, \ldots, v_{k}$, where $v_{0}=v_{s}$. Let us assume the times of departure from $v_{0}, \ldots, v_{k}$ are equal respectively to $T_{0}, \ldots, T_{k}$. If one can leave $v_{j}(j=0, \ldots, k-1)$ at any time later than the time $T_{j}$ and this does not increase the time of travel $T\left(p_{v_{s}, v_{e}}\right)$, then a new path $p_{v_{s}, v_{e}}^{\prime}$ is created and added to the list $L F S$. The path $p_{v_{s}, v_{e}}^{\prime}$ differs from the path $p_{v_{s}, v_{e}}$ only in times of departure from vertices belonging to these paths, and hence these paths differ in times of waiting for changes on vertices $v_{0}, \ldots, v_{k}$. The times of travel in both paths are equal, i.e., $T\left(p_{v_{s}, v_{e}}\right)=$ $T\left(p_{v_{s}, v_{e}}^{\prime}\right)$, and it follows that the total times of waiting for changes in these paths are also equal.

In the last step, paths containing cycles are determined by the procedure PathsContainingCyCLES (line 31). A necessary condition for a non-dominated solution containing cycles is determined by Theorem 1 and therefore the paths containing at least a single vertex of change are analysed. Let us consider the path $p_{v_{s}, v_{e}}$ (Fig. 3(b)), where the change is made at $v_{c}$ and we arrive at it on a bus of the line $l_{a}$ and leave it on a bus of the line $l_{b}$. The paths $p_{v_{0}^{a}, v_{k}^{a}}^{a}$ and $p_{v_{0}^{b}, v_{k}^{b}}^{a}$ of the buses of $l_{a}$ and $l_{b}$ contain the following sequence of vertices:

$$
\begin{aligned}
p_{v_{0}^{a}, v_{j}^{a}}^{a} & =\left\langle v_{0}^{a}, \ldots, v_{p}^{a}=v_{c}, \ldots, v_{j}^{a}\right\rangle, \\
p_{v_{0}^{b}, v_{k}^{b}}^{a} & =\left\langle v_{0}^{b}, \ldots, v_{q}^{b}=v_{c}, \ldots, v_{k}^{b}\right\rangle,
\end{aligned}
$$

and these paths have a common vertex $v_{c}$. If

$$
\exists x \in\{p+1, \ldots, j\} \wedge \exists y \in\{0, \ldots, q-1\}: v_{x}^{a}=v_{y}^{b}
$$

then the paths $p_{v_{0}^{a}, v_{k}^{a}}^{a}$ and $p_{v_{0}^{b}, v_{k}^{b}}^{a}$ have also a common vertex $v_{d}$, where $v_{d}=v_{x}^{a}=v_{y}^{b}$, and there exists the cycle $p_{v_{c}, v_{c}}$ which starts and ends at $v_{c}$ and contains $v_{d}$. If passing a cycle does not increase the time of travel and the conditions defined by Theorem 1 are satisfied, then a new path $p_{v_{s}, v_{e}}^{\prime}$ (Fig. 3(a)) containing the cycle $p_{v_{c}, v_{c}}$ is created and inserted into the list $L F S$.

The procedure AdDPSol is described by Algorithm 2 It tries to insert $S_{i}$ into the list of partial solutions $L P S$. In the first part of the procedure


Fig. 4. Ways of arriving to the vertex $v_{k}$.
(lines 15) the solution $S_{i}$ and solutions in the list $L P S$ are estimated on the basis of values $t_{\text {max }}^{p i}, c_{\text {max }}^{p i}, t_{\text {min }}^{p i}$ and $c_{\min }^{p i}$ without comparing $S_{i}$ with solutions stored in LPS. First, in line 1 the condition (38) is checked, and if it is satisfied then the final solution obtained from $S_{i}$ will be a dominated solution. Therefore $S_{i}$ is omitted and the function returns an unmodified list $L P S$ (line 2). The next, according to (39) condition it is estimated (line 3) if the final solutions obtained from partial solutions stored in $L P S$ will be dominated. If the estimation is positive then all solutions are removed from $L P S$ and $S_{i}$ is inserted into $L P S$ (line 4). In this case, the procedure returns the list $L P S$ containing only single solution $S_{i}$ (line5).

In the second part of the procedure, in the for-loop the partial solution $S_{i}$ is compared with each solution $S$ stored in LPS (lines 744). According to Theorem 3 there are checked the conditions (34) and (35) (lines 9 and 11). If the condition in line 9 is satisfied, then it is not possible to obtain non-dominated final solutions from $S_{i}$; therefore $S_{i}$ is omitted, the procedure ends its execution and returns $L P S$ (line 10). If the condition in line 11 is met, then a final solution obtained from $S$ stored in $L P S$ will be a dominated solution, and therefore it is removed from LPS (line 12).

If the estimation of the partial solution $S_{i}$ is positive, i.e., it is possible to obtain from it a non-dominated final solution, then it is inserted into the list $L P S$ (lines $15-19$. It should be pointed out that the vertex $v_{k}$, which precedes the vertex $v_{i}$ in the path, can be reached from many vertices by buses of different lines. In Fig. 4 it is reached from $v_{g}, v_{h}, v_{j}$, and each case represents a different path from $v_{s}$ to $v_{k}$. These paths are represented by solutions $S_{g}, S_{h}, S_{j}$ stored in the list $L P S\left[v_{k}\right]$. The solutions $S_{g}$, $S_{h}, S_{j}$ are extended by adding the vertex $v_{i}$ to obtain new solutions $S_{g}^{\prime}, S_{h}^{\prime}, S_{j}^{\prime}$ for $v_{i}$. The new solutions represent different paths from $v_{s}$ to $v_{i}$, where $v_{i}$ is preceded by $v_{k}$ in these paths and $v_{i}$ is reached by a bus of the line $l_{i}$. If the times and the costs of travel are identical, i.e., $S_{g}^{\prime} \cdot t_{s i}=S_{h}^{\prime} \cdot t_{s i}=S_{j}^{\prime} \cdot t_{s i}$ and $S_{g}^{\prime} \cdot c_{s i}=S_{h}^{\prime} \cdot c_{s i}=S_{j}^{\prime} \cdot c_{s i}$, then only a single solution $S^{\prime}$ is created and stored in the

```
Algorithm 2. Procedure ADDPSOL of adding the partial
solution \(S_{i}\) to the list \(L P S\).
Input: \(L P S, S_{i}, \Delta c_{\text {max }}, t_{\text {max }}, c_{\text {max }}, t_{\text {min }}^{i}, c_{\text {min }}^{i}\)
Output: LPS
    if \(S_{i} \cdot t_{s i} \geq L P S . t_{\max }^{p i}\) and \(S_{i} \cdot c_{s i}-\Delta c_{\max }>\)
    \(L P S . c_{\text {max }}^{p i}\) then
        return \(L P S\);
    else if \(S_{i} \cdot t_{s i} \leq L P S . t_{\min }^{p i}\) and \(S_{i} \cdot c_{s i}<L P S . c_{\min }^{p i}\)
    then
        LPS.ClEAR(); LPS.AddToList \(\left(S_{i}\right)\);
        return \(L P S\);
    else
        for all \(S \in L P S\) do
            \(\Delta c_{\text {max }}^{\prime} \leftarrow\) the maximal decrease in the cost of
            travel in the solution \(S\);
            if \(S_{i} . t_{s i} \geq S . t_{s i}\) and \(S_{i} . c_{s i}-\Delta c_{\max }>S . c_{s i}\)
            then
                    return \(L P S\);
            else if \(S . t_{s i} \geq S_{i} \cdot t_{s i}\) and \(S . c_{s i}-\Delta c_{\max }^{\prime}>\)
            \(S_{i} . c_{s i}\) then
                LPS.REmove \((S)\);
            end if
        end for
        if \(\exists S \in L P S: S . t_{s i}=S_{i} \cdot t_{s i}\) and \(S . c_{s i}=S_{i} \cdot c_{s i}\)
        and \(S . l_{s i}=S_{i} . l_{s i}\) then
            \(S . L P P_{k} \cdot \operatorname{ADDToLIST}\left(S_{i} \cdot L P P_{k}\right)\);
        else
            LPS.ADDToList \(\left(S_{i}\right)\);
        end if
        return \(L P S\);
    end if
```

```
Algorithm 3. Procedure ADDFSol of adding the final
solution \(S_{i}\) to the list \(L P S\).
Input: \(L P S, S_{i}\)
Output: \(L P S\)
    if \(S_{i} \cdot t_{s i}>L P S . t_{\max }^{p i}\) and \(S_{i} \cdot c_{s i}>L P S . c_{\max }^{p i}\) then
        return \(L P S\);
    else if \(S_{i} \cdot t_{s i}<L P S . t_{\min }^{p i}\) and \(S_{i} . c_{s i}<L P S . c_{\text {min }}^{p i}\)
    then
        LPS.CLEAR (); LPS.AddToList \(\left(S_{i}\right)\);
        return \(L P S\);
    else
        for all \(S \in L P S\) do
            if \(S \succ S_{i}\) then
                return \(L P S\);
            else if \(S_{i} \succ S\) then
                    LPS.REMOVE \((S)\);
            end if
        end for
        LPS.AddToList \(\left(S_{i}\right)\);
        return \(L P S\);
    end if
```

list $L P S\left[v_{i}\right]$. In the solution $S^{\prime}$ the list $L P P_{k}$ stores pointers to the solutions $S_{g}, S_{h}, S_{j}$, which decreases the number of stored and analysed solutions as well as the time of computation. But in the procedure CreateFullpaths (line 29 of Algorithm 11), from a single solution $S^{\prime}$ three full paths are determined. In these paths the vertex $v_{i}$ is preceded by the vertices $v_{g}, v_{h}, v_{j}$, respectively.

In line 15 of the procedure ADDPSOL it is checked if the list $L P S$ contains a solution $S$ in which the vertex $v_{i}$ is reached by a bus of the same line as in the partial solution $S_{i}$ and the time and the cost of travel of $S$ and $S_{i}$ are equal. If $S$ exists, then the pointer to the partial solution for the vertex $v_{k}$ is added to the list $L P P_{k}$ stored in $S$ (line 16) and the new partial solution $S_{i}$ is not inserted into the list $L P S$. Otherwise $S_{i}$ is inserted into $L P S$ (line 18).

The procedure ADDFSOL is described by Algorithm 3. It tries to insert the new solution $S_{i}$ to the list $L P S$ storing solutions for the final vertex $v_{e}$. The first part of the procedure (lines 145 ) is similar to lines $1 / 5$ of the procedure ADDPSoL. The solution $S_{i}$ and solutions in the list LPS are estimated on the basis of the values $t_{\text {max }}^{p i}, c_{\text {max }}^{p i}, t_{\min }^{p i}$ and $c_{\text {min }}^{p i}$ without comparing $S_{i}$ with solutions stored in $L P S$. During the estimation the solution $S_{i}$ or all solutions in the list $L P S$ are omitted as dominated ones (lines 2and (4).

In the second part of the procedure (lines $7-15$ ) the solution $S_{i}$ is compared with each solution $S$ stored in the list $L P S$. If $S_{i}$ dominates $S$ (line 10), then $S$ is removed from the list $L P S$ (line 11), or if $S$ dominates $S_{i}$ (line 8), then $S_{i}$ is omitted and the procedure ends its execution (line 9). If $S_{i}$ is a non-dominated solution, then it is inserted into the list LPS (line 14).

The time and memory complexities of the algorithm SOLVEBBR depend on the number of paths belonging to the set of non-dominated solutions. The latter is a variant of the BSP problem. The BSP problem is known to be NP-complete and, as shown by Serafini (1987) or Skriver and Andersen (2000b), in the worst case, that of solutions of the problem grows exponentially with the number of vertices representing the bus stops. Therefore any algorithm that attempts to solve it is also exponential in terms of worst-case time and memory complexities. The worst case occurs when all possible paths between a given pair of vertices belong to the set of non-dominated solutions and the number of paths equals $\prod_{i=1}^{n} x_{i} \cdot d_{i}$, where $d_{i}$ is equal to the outdegree of the vertex $v_{i}$; $x_{i}=1$ if $v_{i} \neq v_{e}$ and $x_{i}=0$ otherwise (we do not examine arcs outgoing from the final vertex $v_{e}$ and the path does not contain these arcs). Therefore the pessimistic memory complexity equals $O\left(d^{n}\right)$, where $d=$ $\max _{i=1, \ldots, n}\left\{x_{i} \cdot d_{i}\right\}$.

Time complexity is clearly dominated by execution of the while-loop (lines $8-28$ of the algorithm SolveBBR). Each path determined in the while-loop contains at most $n$ vertices. For each of $q_{k}$ solutions

Table 7. Parameters of the multigraphs $G$ and $G^{\prime}$ representing the bus network.

|  | multigraph $G$ | multigraph $G^{\prime}$ |
| :---: | :---: | :---: |
| number of vertices | 1211 | 1211 |
| number of arcs | 5549 | 35610 |
| minimum indegree of vertex | 1 | 6 |
| minimum outdegree of vertex | 1 | 6 |
| minimum degree of vertex | 2 | 13 |
| maximum indegree of vertex | 13 | 122 |
| maximum outdegree of vertex | 13 | 122 |
| maximum degree of vertex | 26 | 244 |

determined for the vertex $v_{k}$ and stored in the list $L P S\left[v_{k}\right]$ (line 10), all of $d_{k}$ outgoing arcs from $v_{k}$ are examined (line 11). Thus, in the worst case it is necessary to examine $\prod_{i=1}^{n} x_{i} \cdot q_{i} \cdot d_{i}$ arcs, and pessimistic time complexity equals $O\left(r^{n}\right)$, where $r=\max _{i=1, \ldots, n}\left\{x_{i} \cdot q_{i} \cdot d_{i}\right\}$.

## 3. Experimental test results

The SolveBBR algorithm was implemented in C++ and the test experiments were carried out on a 2.9 GHz Intel(R) Core(TM) i5-4570S CPU computer with 4 GB of RAM, running Windows 7 Enterprise x64. The results of the tests are compared with those obtained by the algorithm denoted by WID12 and presented in (Widuch, 2012). In the tests, a random generated bus network consisting of 1211 stops divided into 26 zones was used. In the network, buses of 500 bus lines are run: the shortest length of the route of a bus line equals 6 and the longest length of the route equals 29 . The parameters of the multigraphs $G$ and $G^{\prime}$ representing the bus network are presented in Table $73^{3}$ The performed tests had the following goals:

- investigate the properties of computed non-dominated final solutions, i.e., the number of computed final solutions containing cycles, the number of computed final solutions differing from each other only in the times of departure from vertices belonging to the paths, and the number of computed final solutions obtained from dominated partial solutions;
- compare the computation time and the number of computed partial solutions using the tested algorithms; all computed partial solutions inserted into the list $\operatorname{LPS}\left[v_{i}\right]$ (line [21 of the procedure SolveBBR) during a single experiment were counted; additionally, all analysed and omitted partial solutions for which the estimation is negative

[^2](this means that the final solutions obtained from them are dominated) were counted;

- count the number of computed non-dominated final solutions;
- count the number of computed final solutions satisfying the relationships R1-R9 defined in Section 2.4

We carried out 7,326,550 test experiments using the SolveBBR and Wid12 algorithms. The aim of a single test was to find all paths belonging to the set of non-dominated solutions for the given pair of the start $v_{s}$ and the final $v_{e}$ vertices and the time of staring travel $T_{s}$ at $v_{s}$. The tests were carried out for all pairs of vertices and for the times of starting travel equal to $T_{s}=0: 00,8: 00$, 12:00, 16:00 and 20:00, respectively. The results of the tests are presented depending on the number of changes in the path and the length of the path.

The results of test experiments using the algorithms are presented in Tables 8 and 9 . Here we show

- the number of all computed non-dominated final solutions during all experiments and the maximal number of computed non-dominated final solutions in a single experiment,
- the maximal number of computed and omitted partial solutions in a single experiment,
- maximal computation time in milliseconds.


## Additionally, in charts (Fig. 5) we show

- the percentage number of non-dominated solutions containing cycles,
- the percentage number of solutions with the same time and cost of travel and the same path, differing from each other only in the times of departure from the vertices,
- the percentage number of solutions obtained from dominated partial solutions.

The number of solutions differing from each other only in the times of departure and the of solutions obtained from dominated partial solutions were computed by mutual comparison of solutions obtained in a single experiment. According to Theorem 1 and Lemma 1 , the paths containing cycles and those differing from each other only in the times of departure belong to the set of non-dominated solutions if we change at least once in these paths. Therefore, there was no such path without a change. The percentage number of these determined solutions grows with the length of the path. The maximum length of a determined path equals 95 , and it should be pointed out that all paths of this length contain a cycle.

About $80 \%$ of the determined paths of the maximum length differ from each other only in the times of departure from vertices.

The computation time depends on the number of computed partial and final non-dominated solutions. With an increase in the length of the path and the number of changes, the searched space of solutions grows and so does the number of computed partial solutions. The maximal computation time using the SolveBBR algorithm equals 343 milliseconds and was a result of determining the paths in which the bus change 6 and 7 times was made while the length of the path equals 41-70 vertices. In these cases the number of computed partial solutions is maximal and equal to 10,121 .

The SolveBBR algorithm applies conditions for estimation of partial solutions described in Sections 2.3 and 2.4 but the WID12 algorithm does not apply them. For that reason, a larger space of solutions by the WID12 algorithm than by the SolveBBR algorithm is searched, the WID12 algorithm computes a larger number of partial solutions than the SOLVEBBR algorithm and a larger number of solutions are analysed by the WID12 algorithm. Therefore, the computation time is larger for the Wid12 algorithm than the for SolveBBR algorithm.

The number of computed partial solutions also depends on their representation. The partial solutions stored in the list $L P S\left[v_{i}\right]$ of the vertex $v_{i}$ contain the list of pointers $L P P_{k}$ to the partial solutions for the vertex $v_{k}$, which precedes $v_{i}$ in the path from the start vertex $v_{s}$ to $v_{i}$ (see Section 2.5). Thus the partial solution represents many paths from $v_{s}$ to $v_{i}$. It is an important difference to the WID12 algorithm, wherein the partial solution represents a single path from $v_{s}$ to $v_{i}$. Therefore, the number of computed partial solutions by the SolveBBR algorithm is smaller than the number of partial solutions computed and analysed by the WID12 algorithm. In the best case, it computes 66 times less partial solutions. For that reason, the computation time is shorter for the SolveBBR algorithm than for the WID12 algorithm, and in the best case it is 47 times less. It depends on checking by the procedure InPath if the vertex $v_{i}$ belongs to the path represented by the partial solution $S_{k}$ (line 12] of the SolveBBR algorithm). It is executed in $O(k)$ time, where $k$ equals the length of the path, but in the Wid 12 algorithm it is checked in $O(1)$ time.

The representation of a partial solution and conditions used for estimation of partial solutions have also influence the number of omitted partial solutions. It should be pointed out that each omitted partial solution was analysed, but due to negative estimation it was not inserted into the list $L P S\left[v_{i}\right]$. Therefore the conditions for estimation of partial solutions applied by the SolveBBR algorithm and the proposed representation of the partial solution decrease the number of analysed and omitted partial solutions. In the best case it is 214 times less.

Table 8. Number of all computed non-dominated final solutions during all experiments, the maximal number of computed nondominated final solutions in a single experiment, the maximal number of computed and omitted partial solutions in a single experiment and maximal computation time for the SolveBBR and Wid12 algorithms. They depend on the number of changes.

| Number of changes | Maximal number of non-dominated final solutions computed in single experiment | Numberof non-dominatedfinal solutionscomputed duringall experiments | Maximal number of partial solutions |  |  |  | $\begin{gathered} \text { Maximal computation } \\ \text { time }(\text { in } \mathrm{ms} \text { ) for algorithm } \\ \hline \end{gathered}$ |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  |  |  | computed by algorithm |  | omitted by algorithm |  |  |  |
|  |  |  | Solvebir | WID12 | Solvebir | WID 12 | Solvebir | WID12 |
| 0 | 3 | 161251 | 962 | 9789 | 18249 | 356123 | 16 | 16 |
| 1 | 52 | 2018583 | 3375 | 70275 | 46119 | 2224413 | 16 | 16 |
| 2 | 1966 | 10022638 | 5435 | 165285 | 75672 | 4299398 | 16 | 94 |
| 3 | 4772 | 28648372 | 7596 | 416546 | 80712 | 12184704 | 32 | 140 |
| 4 | 23917 | 57132301 | 9171 | 576074 | 108322 | 16522381 | 62 | 499 |
| 5 | 127259 | 96012598 | 10121 | 622354 | 118343 | 17852038 | 140 | 733 |
| 6 | 529039 | 58530211 | 10121 | 622354 | 120729 | 17852038 | 343 | 749 |
| 7 | 228614 | 7889988 | 10121 | 622354 | 120729 | 17852038 | 343 | 780 |
| 8 | 4149 | 1516711 | 10121 | 564170 | 120729 | 16549834 | 312 | 780 |
| 9 | 1410 | 375808 | 10121 | 556555 | 119128 | 16549834 | 312 | 780 |
| 10 | 300 | 113357 | 10121 | 525733 | 113660 | 14866546 | 172 | 780 |
| 11 | 270 | 28950 | 10121 | 568959 | 87476 | 16396951 | 172 | 670 |
| 12 | 180 | 5587 | 9421 | 622354 | 83366 | 17852038 | 31 | 702 |
| 13 | 72 | 860 | 6281 | 208329 | 76607 | 6343552 | 16 | 749 |
| 14 | 32 | 32 | 5557 | 208329 | 52898 | 4655803 | 16 | 276 |

Table 9. Number of all computed non-dominated final solutions during all experiments, the maximal number of computed nondominated final solutions in a single experiment, the maximal number of computed and omitted partial solutions in a single experiment and maximal computation time for the SolveBBR and Wid12 algorithms. They depend on the length of the path.

| Length of path | Maximal number of non-dominated final solutions computed in single experiment | Number of non-dominated final solutions computed during all experiments | Maximal number of partial solutions |  |  |  | Maximal computation time (in ms) for algorithm |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  |  |  | computed by algorithm |  | omitted by algorithm |  |  |  |
|  |  |  | Solvebir | WID12 | Solvebir | WID12 | Solvebir | WID12 |
| 1-5 | 106 | 295599 | 946 | 6601 | 17091 | 214944 | 16 | 16 |
| 6-10 | 545 | 1776642 | 1828 | 28007 | 28896 | 860096 | 16 | 16 |
| 11-15 | 2822 | 4792685 | 3344 | 67973 | 46990 | 2061928 | 16 | 32 |
| 16-20 | 6648 | 9626750 | 4323 | 202677 | 79282 | 5748026 | 16 | 78 |
| 21-25 | 8447 | 16043672 | 6012 | 202677 | 85086 | 5748026 | 32 | 219 |
| 26-30 | 20207 | 22892227 | 7644 | 337508 | 99581 | 9955280 | 62 | 249 |
| 31-35 | 36037 | 27976184 | 9114 | 496767 | 108817 | 14218868 | 63 | 374 |
| 36-40 | 90347 | 32729054 | 9171 | 622354 | 118343 | 17852038 | 296 | 608 |
| 41-45 | 106932 | 37001418 | 10121 | 622354 | 118343 | 17852038 | 343 | 749 |
| 46-50 | 290418 | 38250850 | 10121 | 622354 | 118343 | 17852038 | 343 | 780 |
| 51-55 | 288822 | 31927564 | 10121 | 622354 | 120729 | 17852038 | 343 | 780 |
| 56-60 | 228044 | 20002773 | 10121 | 622354 | 120729 | 17852038 | 343 | 780 |
| 61-65 | 49336 | 11397158 | 9905 | 622354 | 120729 | 17852038 | 343 | 780 |
| 66-70 | 117832 | 5403238 | 9905 | 622354 | 120729 | 17852038 | 343 | 749 |
| 71-75 | 12346 | 1780694 | 9905 | 568959 | 120729 | 16396951 | 219 | 749 |
| 76-80 | 7166 | 511479 | 9905 | 444956 | 119128 | 13732192 | 78 | 733 |
| 81-85 | 2490 | 45222 | 9905 | 438201 | 117127 | 13732192 | 47 | 530 |
| 86-90 | 240 | 3825 | 7375 | 265206 | 107377 | 6889943 | 31 | 436 |
| 91-95 | 60 | 213 | 6063 | 265206 | 80152 | 1082443 | 16 | 109 |

(a)

(b)


Fig. 5. Properties of all computed non-dominated solutions by the SOLVEBBR algorithm depending on the number of changes (a) and the length of the path (b).

In the single experiment it was not determined more than 3 non-dominated final solutions where we travel without a bus change (Table 8). The maximal number of bus changes equals 14 , and there were determined 32 paths with this property, all in a single experiment. During all experiments most paths were determined with lengths from the range $46-50$ vertices, and $38,250,850$ paths were determined (Table 9). The average number of computed non-dominated final solutions during a single experiment equals 36, where 28 solutions have the same paths and differ from each other only in the times of departure from all vertices belonging to these paths while 6 solutions contain a cycle. The average numbers of partial solutions computed and omitted by the SolveBBR algorithm are equal to 305 and 6,918 , respectively. This is about 7 and 11 times less than for the WID12 algorithm. This implies that the SolveBBR algorithm searches a smaller space of solutions than the WID12 algorithm.

The next goal of the tests was to count the number of determined final solutions satisfying the relationships R1-R9 defined in Section 2.4. It was counted by mutual comparison all pairs of paths belonging to a determined set of non-dominated solutions. The paths that differ from each other only in the times of departure from vertices belonging to these paths are not counted because each path contains at least one vertex $v_{i}$ for which one of the relationships R1-R9 is satisfied due to the time of travel from $v_{s}$ to $v_{i}$. Consider a pair of paths

$$
\begin{aligned}
& p_{v_{s}, v_{e}}=\left\langle v_{0}=v_{s}, \ldots, v_{j}, \ldots, v_{x}=v_{e}\right\rangle \\
& p_{v_{s}, v_{e}}^{\prime}=\left\langle v_{0}^{\prime}=v_{s}, \ldots, v_{k}^{\prime}, \ldots, v_{y}^{\prime}=v_{e}\right\rangle
\end{aligned}
$$

belonging to the set of non-dominated solutions. The paths are compared if the following conditions are satisfied:

1. These paths have a common vertex $v_{i}$, i.e.,

$$
\begin{aligned}
& \exists j \in\{1, \ldots, x-1\} \wedge \\
& \exists k \in\{1, \ldots, y-1\}: v_{j}=v_{k}^{\prime}=v_{i}
\end{aligned}
$$

If there exist many $j$ and $k$, then the minimal values from among $j$ and $k$ are taken into account.
2. The partial solution $p_{v_{s}, v_{i}}$ dominates the partial solution $p_{v_{s}, v_{i}}^{\prime}$, i.e., $p_{v_{s}, v_{i}} \succ p_{v_{s}, v_{i}}^{\prime}$, where

$$
\begin{aligned}
p_{v_{s}, v_{i}} & =\operatorname{sub}_{p_{v_{s}, v_{e}}}\left(v_{s}, v_{i}\right) \\
p_{v_{s}, v_{i}}^{\prime} & =\operatorname{sub}_{p_{v_{s}, v_{e}}^{\prime}}\left(v_{s}, v_{i}\right)
\end{aligned}
$$

If the solution satisfies several relationships, then all these cases are counted, and if satisfies the relationship many times, then it is counted only once. The results of the comparison of the solutions are presented in plots (Fig. 6). The determined non-dominated final solutions are compared, therefore only relationships R1, R6 and R8 are satisfied. To check satisfaction of other relationships, it is necessary to determine all possible paths from the start vertex $v_{s}$ to the final vertex $v_{e}$. Thus the dominated solutions cannot be omitted during computation in the SolveBBR algorithm. This kind of test is not possible to be carried out within reasonable time due to memory complexity. The plots contain results for all cases considered in Section 2.4 i.e.,

- when $v_{i}$ is passed without a change in both the compared paths $p_{v_{s}, v_{e}}$ and $p_{v_{s}, v_{e}}^{\prime}$ (Figs. 6(a) and (b)),
- when $v_{i}$ is passed without a change in $p_{v_{s}, v_{e}}$ and we change at $v_{i}$ in $p_{v_{s}, v_{e}}^{\prime}$ (Figs. G(c) and (d)),
- when we change at $v_{i}$ in $p_{v_{s}, v_{e}}$ and it is passed without a change in $p_{v_{s}, v_{e}}^{\prime}$ (Figs. Ge) and (f)),


Fig. 6. Percentage number of all computed non-dominated solutions by the SoLVEBBR algorithm satisfying R1, R6 and R8 in relation to the number of changes and the path length.

- when we change at $v_{i}$ in both the compared paths $p_{v_{s}, v_{e}}$ and $p_{v_{s}, v_{e}}^{\prime}$ (Figs. 6 (g) and (h)).

The number of solutions satisfying the relationships considered decreases with the length of the path, except for the case when the vertex $v_{i}$ is passed without a change in both paths $p_{v_{s}, v_{e}}$ and $p_{v_{s}, v_{e}}^{\prime}$ (Fig. 6b)). In this case it grows from the length of 76 .

In most cases the relationship R1 is satisfied by solutions if the time and the cost of travel fulfill the conditions

$$
\begin{equation*}
T\left(p_{v_{s}, v_{e}}\right)=T\left(p_{v_{s}, v_{e}}^{\prime}\right) \wedge C\left(p_{v_{s}, v_{e}}\right)=C\left(p_{v_{s}, v_{e}}^{\prime}\right) \tag{42}
\end{equation*}
$$

The largest difference between the number of solutions satisfying the relationship R1 and the number of solutions satisfying the relationships R6 and R8 occurs when we change at $v_{i}$ in $p_{v_{s}, v_{e}}$. This is due to the fact that in most cases the subpath $\operatorname{sub}_{p_{v_{s}, v_{e}}}\left(v_{i}, v_{e}\right)$ is the same as the subpath $\operatorname{sub}_{p_{v_{s}, v_{e}}^{\prime}}\left(v_{i}, v_{e}\right)$ and the time of travel $T\left(p_{v_{s}, v_{e}}\right)=T\left(p_{v_{s}, v_{e}}^{\prime}\right)$. Both paths belong to the set of non-dominated solutions, therefore the cost of travel $C\left(p_{v_{s}, v_{e}}\right)=C\left(p_{v_{s}, v_{e}}^{\prime}\right)$ and (42) occurs.

Most solutions satisfy the relationships in the case when $v_{i}$ is passed without a change in $p_{v_{s}, v_{e}}^{\prime}$ obtained from the dominated partial solution $p_{v_{s}, v_{i}}^{\prime}$ (Figs. 6(a), (b), (e) and (f)). When $v_{i}$ is passed without a change, the cost of travel $C\left(p_{v_{s}, v_{e}}^{\prime}\right)$ equals the sum of costs of travel $C\left(p_{v_{s}, v_{i}}^{\prime}\right)$ and $C\left(\operatorname{sub}_{p_{v_{s}, v_{e}}^{\prime}}\left(v_{i}, v_{e}\right)\right)$ decreased by $\Delta c^{\prime}$. This increases the possibility of obtaining a non-dominated solution from a dominated partial one. Otherwise, when we change at $v_{i}$ in $p_{v_{s}, v_{i}}^{\prime}$, we do not decrease the cost of travel $C\left(p_{v_{s}, v_{i}}^{\prime}\right)$ by $\Delta c^{\prime}$. Therefore in many cases a final solution obtained from a dominated partial one was a dominated solution. Thus, at most 20\% of non-dominated final solutions were obtained from a dominated partial solutions (Figs. 6(c), (d), (g) and (h)).

For each number of changes and each range of the path length a solution obtained from a dominated partial solution exists. Thus, these solutions would not have been determined as a result of omitting dominated partial solutions as in the case of the multigraph with constant weights (Skriver and Andersen, 2000b). It can be seen that the number of solutions obtained from a dominated partial solution decreases with the path length. It should be noted that about $70 \%$ solutions where in the path a change is performed 11 times and about $5 \%$ solutions where the journey is performed without a change were obtained from dominated partial solutions. Additionally, in each case there exists pair of solutions $p_{v_{s}, v_{e}}$ and $p_{v_{s}, v_{e}}^{\prime}$ satisfying any relationship,

$$
\begin{aligned}
& T\left(p_{v_{s}, v_{e}}\right)<T\left(p_{v_{s}, v_{e}}^{\prime}\right) \wedge C\left(p_{v_{s}, v_{e}}\right)>C\left(p_{v_{s}, v_{e}}^{\prime}\right), \\
& T\left(p_{v_{s}, v_{e}}\right)=T\left(p_{v_{s}, v_{e}}^{\prime}\right) \wedge C\left(p_{v_{s}, v_{e}}\right)=C\left(p_{v_{s}, v_{e}}^{\prime}\right), \\
& T\left(p_{v_{s}, v_{e}}\right)>T\left(p_{v_{s}, v_{e}}^{\prime}\right) \wedge C\left(p_{v_{s}, v_{e}}^{\prime}\right)<C\left(p_{v_{s}, v_{e}}^{\prime}\right),
\end{aligned}
$$

that takes place between the time and the cost of travel, where $p_{v_{s}, v_{e}}$ is obtained from the partial solution $p_{v_{s}, v_{i}}$ and $p_{v_{s}, v_{e}}^{\prime}$ is obtained from the dominated partial solution $p_{v_{s}, v_{i}}^{\prime}$, i.e., $p_{v_{s}, v_{i}} \succ p_{v_{s}, v_{i}}^{\prime}$.

For $1,838,360$ tests all non-dominated solutions were obtained from non-dominated partial solutions, and this is about $25 \%$ of tests carried out. Thus about $75 \%$ of conducted tests contain at least one solution obtained from a dominated partial solution, and for 311,309 tests (which is about $4 \%$ of all executed tests) all non-dominated solutions were obtained from dominated partial solutions. The results demonstrate that the monotonicity assumption does not hold in the BBR problem and the proposed estimation of partial solutions applied by the SolveBBR algorithm makes it possible to determine these solutions. Otherwise, i.e., when dominated partial solutions are omitted, it would not determine any of these solutions.

The maximal number of computed non-dominated final solutions in the single test equals 529,183 . There are only 145 solutions differing from each other in vertices or arcs belonging to these paths, but 143 solutions satisfying the relationships R1 and R6 from among them. Thus, 143 were obtained from dominated partial solutions. The maximal number of non-dominated solutions obtained from dominated partial ones in a single test equals 2,627 and these solutions satisfying the relationships R1, R6 and R8. The number of all non-dominated solutions determined in this test equals 4,962, and all solutions obtained from dominated partial ones differ from each other in vertices or arcs belonging to these paths. In these two tests the omission of dominated partial solutions would cause a failure in finding valid non-dominated final solutions.

## 4. Conclusions

In this paper the bicriterion bus routing (BBR) problem was considered and its theoretical analysis was presented. It was shown that a non-dominated final solution can be obtained from a dominated partial one. This is an important property used during the process of determining solutions. We presented a detailed analysis of the conditions that must be fulfilled in order to obtain a non-dominated solution from a dominated partial one.

We proposed a new label correcting algorithm for solving the BBR problem. The proposed representation of a partial solution and methods of estimating partial solutions decrease the number of computed and analysed partial solutions in comparison with the algorithm presented by Widuch (2012). The methods of estimating partial solutions make it possible to determine all non-dominated solutions, which was confirmed by experimental tests. The tests were carried out for all pairs of vertices in a multigraph representing the bus network and for 5 different times of starting travel.

The experimental tests showed that about $75 \%$ of the conducted tests contain a non-dominated final solution obtained from a dominated partial one and about $4 \%$ of conducted tests contain all non-dominated final solutions obtained from a dominated partial solution.

The problem is known to be NP-complete, and in the worst case the number of solutions grows exponentially with the number of vertices representing the bus stops. On the basis of the test results we found that our procedure exhibits a reasonable execution time for a bus network containing about 1200 stops. Additionally, a test results demonstrate that the number of non-dominated solutions is not exponential in reality. The set of non-dominated solutions determined by the algorithm may be a basis for choosing by a passenger or an application a single bus route. It is chosen on the basis of additional criteria, for example, the length of the route, the number of changes, the total waiting time at bus stops, etc.

## References

Addor, J.A., Amponsah, S.K., Annan, J. and Sebil, C. (2013). School bus routing: A case study of wood bridge school complex, Sekondi-Takoradi, Ghana, International Journal of Business and Social Research 3(12): 26-36.
Arias-Rojas, J.S., Jiménez, J.F. and Montoya-Torres, J.R. (2012). Solving of school bus routing problem by ant colony optimization, Revista EIA 9(17): 193-208.

Azevedo, J.A. and Martins, E.Q.V. (1991). An algorithm for the multiobjective shortest path problem on acyclic networks, Investigação Operacional 11(1): 52-69.
Bronshtein, E.M. and Vagapova, D.M. (2015). Comparative analysis of application of heuristic and metaheuristic algorithms to the school bus routing problem, Informatics and Its Applications 9(2): 56-62.
Brumbaugh-Smith, J. and Shier, D. (1989). An empirical investigation of some bicriterion shortest path algorithms, European Journal of Operational Research 43(2): 216-224.
Caceres, H., Batta, R. and He, Q. (2014). School bus routing with stochastic demand and duration constraints, Transportation Research Board 93rd Annual Meeting, Washington, DC, USA, pp. 1-23.
Carraway, R.L., Morin, T.L. and Moskowitz, H. (1990). Generalized dynamic programming for multicriteria optimization, European Journal of Operational Research 44(1): 95-104.
Chalkia, E., Grau, J.M.S., Bekiaris, E., Ayfandopoulou, G., Ferarini, C. and Mitsakis, E. (2014). Routing algorithms for the safe transportation of pupils to school using school buses, Transport Research Arena (TRA) 5th Conference: Transport Solutions from Research to Deployment, Paris, France, pp. 1-10.
Chen, P. and Nie, Y.M. (2013). Bicriterion shortest path problem with a general nonadditive cost, Transportation Research B: Methodological 57: 419-435.

Chen, X., Kong, Y., Dang, L., Hou, Y. and Ye, X. (2015). Exact and metaheuristic approaches for a bi-objective school bus scheduling problem, PLoS ONE 10(7): 1-20. DOI:10.1371/journal.pone. 0132600 .
Climaco, J.C. and Martins, E.Q.V. (1982). A bicriterion shortest path algorithm, European Journal of Operational Research 11(4): 399-404.
Corley, H.W. and Moon, I.D. (1985). Shortest paths in networks with vector weights, Journal of Optimization Theory and Application 46(1): 79-86.
Daellenbach, H.G. and De Kluyver, C.A. (1980). Note on multiple objective dynamic programming, Journal of the Operational Research Society 31(7): 591-594.
Dell'Olmo, P., Gentili, M. and Scozzari, A. (2005). On finding dissimilar Pareto-optimal paths, European Journal of Operational Research 162(1): 70-82.
Díaz-Parra, O., Ruiz-Vanoye, J.A., Buenabad-Arias, A. and Cocón, F. (2012). A vertical transfer algorithm for the school bus routing problem, 4th World Congress on Nature and Biologically Inspired Computing (NaBIC), Mexico City, Mexico, pp. 66-71.
Ehrgott, M. (2000). Multicriteria Optimization, Springer-Verlag, Berlin.
Ellegood, W. A., Campbell, J. F. and North, J. (2015). Continuous approximation models for mixed load school bus routing, Transportation Research B 77: 182-198.
Euchi, J. and Mraihi, R. (2012). The urban bus routing problem in the Tunisian case by the hybrid artificial ant colony algorithm, Swarm and Evolutionary Computation 2: 15-24.
Garey, M. and Johnson, D. (1990). Computers and Intractibility: A Guide to the Theory of NP-Completeness, W.H. Freeman \& Co., New York, NY.
Hansen, P. (1980). Bicriterion path problems, in G. Fandel and T. Gal (Eds.), Multiple Criteria Decision Making: Theory and Application, Springer-Verlag, Berlin, pp. 109-127.
Henig, M.I. (1985). The shortest path problem with two objective functions, European Journal of Operational Research 25(2): 281-291.
Huang, L.C., Guan, W. and Xiong, J. (2014). Routing design optimization of bus joint for passenger transfer centers, in M. Sun and Y. Zhang (Eds.), Renewable Energy and Environmental Technology, Applied Mechanics and Materials, Vol. 448, Trans Tech Publications, Zurich, pp. 4140-4149.
Jungnickel, D. (1999). Graphs, Networks and Algorithms, 2nd Edition, Springer-Verlag, Berlin.
Kang, M., Kim, S.K., Felan, J.T., Choi, H.R. and Cho, M. (2015). Development of a genetic algorithm for the school bus routing problem, International Journal of Software Engineering and Its Applications 9(5): 107-126.
Kim, B.I., Kim, S. and Park, J. (2012). A school bus scheduling problem, European Journal of Operational Research 218(2): 577-585.
Kim, T. and Park, B.J. (2013). Model and algorithm for solving school bus problem, Journal of Emerging Trends in Computing and Information Sciences 4(8): 596-600.

Kinable, J., Spieksma, F.C.R. and Berghe, G.V. (2014). School bus routing-a column generation approach, International Transactions in Operational Research 21(3): 453-478.
López, E.R. and Romero, J. (2015). A hybrid column generation and clustering approach to the school bus routing problem with time windows, Ingeniería 20(1): 111-127.
Machuca, E., Mandow, L. and Pérez de la Cruz, J.L. (2009). An evaluation of heuristic functions for bicriterion shortest path problems, Proceedings of the 14 Portuguese Conference on Artificial Inteligence (EPIA 2009), Aveiro, Portugal, pp. 205-216.
Mandow, L. and Pérez de la Cruz, J.L. (2008). Path recovery in frontier search for multiobjective shortest path problems, Journal of Intelligent Manufacturing 21(1): 89-99.

Manumbu, D.M., Mujuni, E. and Kuznetsov, D. (2014). A simulated annealing algorithm for solving the school bus routing problem: A case study of Dar es Salaam, Computer Engineering and Intelligent Systems 5(8): 44-58.

Martí, R., González Velarde, J.L. and Duarte, A. (2009). Heuristics for the bi-objective path dissimilarity problem, Computers \& Operations Research 36(11): 2905-2912.
Martins, E.Q.V. (1984). On a multicriteria shortest path problem, European Journal of Operational Research 16(2): 236-245.

Martins, E.Q.V., Pascoal, M.M.B., Rasteiro, D.M.L.D. and Santos, J.L.E. (1999). The optimal path problem, Investigação Operacional 19(1): 43-60.
Mote, J., Murthy, I. and Olson, D.L. (1991). A parametric approach to solving bicriterion shortest path problems, European Journal of Operational Research 53(1): 81-82.
Newton, R.M. and Thomas, W.H. (1969). Design of school bus routes by computer, Socio-Economic Planning Sciences 3(1): 75-85.

Pacheco, J., Caballero, R., Laguna, M. and Molina, J. (2013). Bi-objective bus routing: An application to school buses in rural areas, Transportation Science 47(3): 397-411.
Pareto, V. (1896). Course d'Economie Politique, F. Rouge, Lausanne.
Park, J. and Kim, B.-I. (2010). The school bus routing problem: A review, European Journal of Operational Research 202(2): 311-319.
Raith, A. and Ehrgott, M. (2009). A comparison of solution strategies for biobjective shortest path problems, Journal Computers and Operations Research 36(4): 1299-1331.

Riera-Ledesma, J. and Salazar-González, J.J. (2012). Solving school bus routing using the multiple vehicle traveling purchaser problem: A branch-and-cut approach, Computers \& Operations Research 39(2): 391-404.
Riera-Ledesma, J. and Salazar-González, J.J. (2013). A column generation approach for a school bus routing problem with resource constraints, Computers \& Operations Research 40(2): 566-583.
Schittekat, P., Kinable, J., Sörensen, K., Sevaux, M. and Spieksma, F. (2013). A metaheuristic for the school bus routing problem with bus stop selection, European Journal of Operational Research 229(2): 518-528.

Serafini, P. (1987). Some considerations about computational complexity for multi objective combinatorial problems, in J. Jahn and W. Krabs (Eds.), Recent Advances and Historical Development of Vector Optimization, Lecture Notes in Economics and Mathematical Systems, Vol. 294, Springer, Berlin/Heidelberg, pp. 222-232.
Sghaier, S.B., Guedria, N.B. and Mraihi, R. (2013). Solving school bus routing problem with genetic algorithm, International Conference on Advanced Logistics and Transport (ICALT'2013), Sousse, Tunisia, pp. 7-12.
Siqueira, V.S., Silva, F.J.E.L., Silva, E.N., Silva, R.V.S. and Rocha, M.L. (2016). Implementation of the metaheuristic grasp applied to the school bus routing problem, International Journal of e-Education, e-Business, e-Management and e-Learning 6(2): 137-145.
Skriver, A.J.V. and Andersen, K.A. (2000a). A classification of bicriteria shortest path (BSP) algorithms, Asia-Pacific Journal of Operational Research 17(2): 199-212.

Skriver, A.J.V. and Andersen, K.A. (2000b). A label correcting approach for solving bicriterion shortest-path problems, Computers \& Operations Research 27(6): 507-524.

Tung, C.T. and Chew, K.L. (1988). A bicriterion Pareto-optimal path algorithm, Asia-Pacific Journal of Operational Research 5(2): 166-172.

Tung, C.T. and Chew, K.L. (1992). A bicriterion Pareto-optimal path algorithm, European Journal of Operational Research 62(2): 203-209.

Widuch, J. (2012). A label correcting algorithm for the bus routing problem, Fundamenta Informaticae 118(3): 305-326.

Widuch, J. (2013). A label correcting algorithm with storing partial solutions to solving the bus routing problem, Informatica 24(3): 461-484.

Worwa, K. (2014). A case study in school transportation logistics, Research in Logistics \& Production 4(1): 45-54.

Yigit, T. and Unsal, O. (2016). Using the ant colony algorithm for real-time automatic route of school buses, International Arab Journal of Information Technology 13(5): 559-565.


Jacek Widuch was awarded the PhD degree at the Silesian University of Technology in 2008. At present he is an assistant professor in the Institute Informatics there. His main research interests are connected with multicriteria optimization algorithms in transportation problems and parallel computing. He is a member of the Upper Silesian Regional Committee of the Polish School Contest in Informatics.

Received: 25 January 2016
Revised: 13 July 2016
Accepted: 10 November 2016


[^0]:    ${ }^{1}$ Travel by a bus of a regular line is assumed, otherwise the value of $c\left(e_{i}\right)$ must be multiplied by 2 .

[^1]:    ${ }^{2}$ The column "Cost of travel" contains the cost of travel from the start vertex $v_{s}=1$ to the given vertex, and the column "Bus line" contains the bus line by which we leave the given vertex.

[^2]:    ${ }^{3}$ An indegree, an outdegree and a degree of a vertex are defined by Jungnickel (1999).

