ISSN (e): 2250 – 3005 || Volume, 14 || Issue, 1|| Jan. - Feb. – 2024 ||
International Journal of Computational Engineering Research (IJCER)

An Adaptive Large Neighborhood Search Metaheuristic for the School Bus Routing Problem with Mixed-Load Consideration

Mohammad Saied FallahNiasar
University of Antwerp Operations Research Group,Faculty of Applied Economic

Abstract
This paper introduces a new variant of the School Bus Routing Problem (SBRP) in which the following are considered: determining the set of possible stops to be visited, allocation of students to possible stops, and making each route while riding students of different schools in a bus is possible (mixed load effect). A new mathematical formulation is proposed to solve the problem optimally. In order to evaluate the developed mathematical model, random instances in small, medium and large instances are generated. To solve instances in medium and large sizes, an Adaptive Large Neighborhood Search (ALNS) and different configurations of large neighborhood search heuristic are proposed. The attained results reveal that among the proposed heuristics, ALNS is a reliable metaheuristic and provides solutions with lower cost. We compare the main characteristics (total travel time and total number of buses) when using a single and a mixed load strategy. The results demonstrate overall reduction in number of routes (up to 11.36%) and total travel time and (up to 14.56%) when utilizing mixed load effect. Additional analysis is performed in order to analyze the performance of different configurations of large neighborhood search heuristics.

Key words: *School bus routing problem, adaptive large neighborhood metaheuristic, mixed-load problem*

Date of Submission: 08-01-2024 Date of acceptance: 20-01-2024

I. Introduction

A safe, reliable and available supply of fair transport for students has consistently appeared as a top concern for any community. Lack of bus network planning leads to air pollution, noise, accident, user's dissatisfaction, and consequently, extra transportation costs. Indeed, the rise of fuel prices and the increased time (due to congestions) required to take students to and from schools have forced families to use public bus transportation system. These assumptions have led communities to concentrate on providing an efficient school bus service.

The issue is addressed in the literature as the**School Bus Routing Problem** (SBRP). Generally, SBRP addresses the problem of how to transport students to and from schools in the safest, the most economical and convenient approach (Corberán et al. (2002)). Each bus has to transfer students to and from schools while satisfying predefined constraints such as bus capacity, student's maximum allowable riding time in a bus, and time windows of the school.

Tehran, the capital of Iran, is one of the40 largest cities in the world with a population of about 9.4 million[1] in 2023. It is estimated that more than 1 million students in Tehran use public transport on a daily basis (data is compiled by author).Additionally, it is estimated that buses used only in Tehran to transport students to schools travel about 412 million km per year with a total transport cost of $170 million. It should be mentioned that transporting students to school by bus is definitely much more efficient and environmentally friendly than using private vehicles. Due to the high population of students in Tehran, handling school transportation service has been an important challenge for educational authorities. However, certain features of student transportation in Tehran (e.g., safety and existing security restrictions, multiple routes per bus, inconsistency in loading and

[1]https://worldpopulationreview.com/world-cities/tehran-population

unloading times of student) result in tremendously challenging schoolbusrouting problems that require additional efforts to plan and organize cost-effective transport services.

To support these issues, on the one hand, authorities have always attempted to provide an efficient transport system by considering the limited resources so that a large amount of money can be saved. From user's perspective, however, student convenience needs to be guaranteed as well while designing a transportation plan. In former cases, due to the lack of available resources, it was not possible to dedicate one bus to a single school (see e.g. Park and Kim, (2010)). In those cases, it would be beneficial to pick up different students of different schools using the same bus (This is called "mixed load approach",differing from "single load approach" which does not allow for transporting students of different schools in the same bus). The sharing of resources between schools helps boost the efficiency of school-bus system. Nevertheless, it increases the complexity of the problem, resulting in overcrowded buses and long routes. For this reason, it seems that designing bus routes based on a mixed load plan can be an effective solution approach, in which appropriate objective,assumptions, and constraints should be considered.

In order to address these issues, a number of desirable criteria, including objectives and assumptions, needs to be considered. Undoubtedly, these criteria can significantly affect the performance of public transportation.

As Savas (1978) has proposed, three criteria for assessing the performance of public facilities should be considered: **efficiency,effectiveness, and equity**. Each criterion incorporates its own constraints and objectives to be fulfilled.

Efficiency criterion is expressed as the ratio between service level and the cost of resources needed to provide the service in question. According to Bowerman et al. (1995), there exist two kinds of perspectives on SBRP; one is the cost per distance travelled and the other is the capital cost required to handle school bus. Thus, an efficient service level follows the solution with a fewer number of buses and a shorter travel distance. The **effectiveness** indicator measures how well the demand is satisfied. More precisely, effectiveness necessitates that SBRP system must be available for eligible students and provide them with adequate service. Eventually, the **equity** criterion calculates the equality or fairness of serviceprovision. First, on first off approach to board students and load balancing the routes can be cited as good ways of improving this criterion. The objectives and assumptions to be adopted in this study must ensure that limited resources are being employed in the best way to pick up and deliver students, thereby reducing costs and increasing users' convenience. To this aim, in term of **efficiency**, we strive to minimize the bus's travel distance. The e**ffectiveness** indicator considers the maximum allowable travel distance for each bus and bus capacity constraints.

Two representative examples of equity criterion are (1) making a balance between routes to avoid too large variations in route loading (i.e. making a desirable distribution of students between routes) and (2) assigning a reasonable number of students to each stop.

The first research in which the phrase "school bus routing problem" appeared is attributed to Newton and Thomas (1969). Detailed explanations on SBRP and a survey of the relevant literature can be found in Park and Kim (2010). They classified SBRP into five categories: data preparation, bus stop selection, bus route generation, and school bell time. **Data preparation** deals with preparing necessary routing data, including student residence locations, schools' geographical locations, and types of fleet used. **Bus stop selection** refers to defining reachable bus stops for students (pick-up or drop-off locations for students) such that these stops will be visited by buses. In rural cases, thebus stop is located at the student residence, whereas in urban areas, students should walk to the bus stop. It is worth mentioning that, for some cases,bus accessibility is denied due to road conditions. Thus, the boarding point is fixed on picking up or dropping off students. **Bus routing generation** constructs the order based on which selected stops are to be visited. There are two types of heuristics employed to generate routes: route-first cluster-second and cluster-first route-second. **Bus scheduling problem** attempts to assign a chain of trips to the same bus while considering the time window of the school. Finally, **school bell time** introduces different starting and ending time constraints for schools. In some cases, different school time windows are considered for a school. This leads the bus schedule to take up more than one trips, resulting in a decrease in travel distance.

The SBRP presented in this paper consists of two interrelated sub-problems including **bus stop selection** and **route generation**,**which are two of the five sub-problems** mentioned in Park and Kim (2010). Each student has to walk to the bus stop from his/her home by considering themaximum walking distance of students from the possible bus stops. Afterward, the bus starts from the garage and picks up students from the visited bus stops while satisfying assumptions such as bus capacity and maximum route length.To maintain convenience for students, student's maximum walking distance to bus stop and maximum allowable number of students in each bus stop are also considered. To the best of the authors' knowledge, the current study is the first to jointly address **bus stop selection** and **routing generation** while **considering both mixed loading and load balancing effects**. More precisely, this paper is constructed based the original paper addressed by

Schittekat(2013) and then it is tried to develop this paper corresponding real case sitation(mixed load effect, school time window, etc.)

Not surprisingly, adaptive large neighborhood search(ALNS) has been successfully used in a variety of vehicle routing problems (VRPs) (Goeke and Schneider (2015), Hiermann et al. (2016)), but it still new for SBRP presented in this paper

The main contributions of our study are as follows:

- Considering a new set of benchmark instances generated randomly, but the problem characteristic is made close to the real situation in Tehran
- Proposing a new mathematical formulation of SBRP, which considers thedefined objectiveand assumptions.
- Analyzing the performance of each insertion and removal operator separately.
- Suggesting an ALNS metaheuristic for solving medium and large instances of SBRP, tuned using a statistical experiment and then compared with existing benchmark.
- Comparing the influence of mixed load and single load approaches on reducing the number of buses, total traveled distance, average weighted riding distance of students, and bus occupation.

II. Literature Review

Over the last decades, numerous researchers have provided significant contributions tothe field of school bus routing problem. Only the most interesting works in literature, although not exhaustively, will be reviewed in this paper. Regarding problem description, bus stop selection, bus routing, and bus scheduling studies will be appraised. In addition, concerning problem characteristics, the authors will survey features such as single load and mixed load contents and rural and urban cases. Comprehensive explanations of the general SBRP and a review of associated literature can be found in Park and Kim (2010).

The majority of SBRP studies have concentrated on one or two objectives: minimizing bus route travelled and minimizing the number of required buses (Li and Fu, 2002) and Pacheco and Martí (2006)). However, a few studies have addressed other objectives that describe maximum route length (Park and Kim (2010)), student riding times (Bennet et al. (1972), Thangiah et al. (1992), Li and Fu (2002)), and student walking distance to a bus stop (Bowerman et al. (1995)).

SBRP can be employed for one or multiple schools at the same time. It is worth mentioning that real-life problems usually deal with several schools, though for simplicity's sake some authors consider only one school. Schittekat et al. (2006), Pacheco and Martí (2006), Martinez and Viegas (2011), Ledesma and Gonzalez (2012), and Euchi and Mraihi (2012)) all have developed single school models.

A large and growing body of literature has investigatedbus stop selection approach. In fact, bus stop selection attempts to select a set of bus stops and then allocate students to these defined stops. In the case of rural areas, a student is picked up from his/her home. Conversely, in urban regions, students are picked up at certain bus stops. The heuristic solution methods for bus stop selection are categorized into **location-allocation-routing** (LAR) strategy and **allocation-routing-location** (ARL) strategy. In the first case (LAR), a set of bus stops are defined initially, and then students are allocated to these stops. Afterward, a number of routes connecting a selection of bus stops are created. In practice, as bus stop selection and student allocation occur before route generation, this approach leads to the production of an excessive number of routes. In the second approach (ARL), students are first allocated to clusters while satisfying capacity constraint. After that, bus stops are selected for each cluster and a route is constructed. As a final step for each existing cluster, students are allocated to the bus stops in which predefined assumptions (including maximum walking distance of students from their residence, and maximum number of students that can be assigned to a bus stop) are considered. Applying this method creates several advantages as follows: firstly, effective load balancing can be carried out during the allocation phase; secondly, it allows the possibility of keeping the number of buses at minimum levels. Chapleau et al. (1985) and Bowerman et al. (1995) have studied ALR approach.

Schittekat et al. (2013) classified SBRP in three sub-problems: finding a set of bus stops to be visited, determining the bus stop to which each student should walk, and determining routes to visit the defined bus stops while minimizing the total distance traveled by each bus.

A number of researchers haveexplored bus routing generation mechanism. According to Park et al. (2012), the algorithm for bus route generation is classified into ''route-first, cluster-second" approach and the ''cluster-first, route-second" approach. In the former, a long route is first created and then divided into smaller ones considering predefined constraints (Newton and Thomas (1969) and Bodin and Berman (1979 b)). In the latter, students are grouped into clusters in such a way that each cluster covers a route respecting some pre-defined constraints (Dulac et al. (1980), Chapleau et al. (1985) and Bowerman et al. (1995)).

In both cluster-first and route-first approaches, after constructing an initial solution, an improvement phase is applied in order to enhance the solution. Newton and Thomas (1969), Dulac et al. (1980), Chapleau et al. (1985) and Desrosiers et al. (1986) have investigated a 2-opt method, and Bennett and Gazis (1972) and Bodin and Berman (1979b) have employed a 3-opt approach.

An interesting field of application which has not received much attention so far concerns transporting students using a mixed load framework, called mixed loading plan. It means moving students from different schools by the same bus simultaneously.

A considerable amount of literature has been published on various kinds of mixed loading school bus routing problem (Bodin and Berman, (1979), Chen et al. (1988), Hargroves et al. (1981), Braca et al. (1997), Spada et al.(2005), Simchi et al. (2005), Hernan Caceres et al. (2015), Bogl et al. (2015), Campbell et al. (2015), Ellegood et al. (2015), Kang et al. (2015), Chen et al. (2015), Maciel Silva et al. (2015), Fátima Machado et al. (2016), and Yao et al. (2016)). To better understand mixed load effect in the current study, Table 1 summarizes the main features considered in rural and urban school bus routing.

The problem of mixed load was first introduced by Bodin and Berman (1979) who highlighted that this method is commonly encountered in rural areas in order to enhance the flexibility of school bus service and to decrease operation costs. Chen et al. (1988) claimed that considering a single load assumption results in an excessive number of buses required to transport students, especially when low density areas are under consideration.

The first computation algorithm for mixed load problem was proposed by Braca et al. (1997). In this paper, an insertion heuristic was developed in which each bus stop and its respective school are inserted in the cheapest position while satisfying time window and bus capacity constraints. The authors also claimed that mixed load problem leads to improving flexibility and producing a significant cost saving. Braca et al. (1997) reported that mixed load has been permitted for most parts of New York City.

In a similar study, Spada et al. (2005) considered multiple schools and proposed a heuristic procedure to solve the problem. The recommended structure improves the service level provided by the bus operator,while allowing mixed load case. The schools are sorted according to their starting time, and correspondingly the routes are created using a greedy method. Subsequently, local search frameworks (simulated annealing and Tabu search) are used to improve the initial solution.

Park and Kim (2010) improved the model proposed by Bracaet al. (1997) by implementing post-improvement procedures. They also performed a quantitative study to measure the special effects of using mixed loading method. The problem incorporates several features: the homogeneous fleet, different starting times, time window, and capacity constraints.

Bogl et al. (2015) studied bus stop selection, pupil assignment, bus routing, and bus scheduling. They compared the results using two different modeling approaches, namely DARP (Dial-A-Ride Problem) and OVRP (Open Vehicle Routing Problem). Campbel et al. (2015) made use ofa strategic continuous approximation to investigate the value of mixed loading for school bus routing problems and also developed three-phase heuristics to evaluate mixed bus trips. The results revealed that mixed trips are more favorable when (1) students are sparsely distributed, (2) there exist many bus stops, and (3) there is the possibility of sharing large percentage of stops. The results also emphasized that mixed routing is more beneficial when there is appropriatestudent distribution between schools and large percentage sharing of stop between schools. Kang et al. (2015) examined the assumptions of mixed loading, homogeneous vehicles, and schools with different starting times. The process was constructed as follows: students are clustered by using a covering approach aiming at minimizing the number of stops; afterward, the genetic algorithm will construct a route with the objective of minimizing total travelled distance. In case an infeasible solution appears after either mutation or crossover operator, time-consuming repair operators come into play in order to return the solution to a feasible one. In the same context, Chen and Kong et al. (2015) solved a bi-objective (fleet's fixed cost and the routing cost) school bus routing problem that takes account of different school starting times and heterogeneous fleet assumptions. Another similar study is that of Maciel Silva et al. (2015). They solved the problem considering assumptions such as mixed load, heterogeneous fleet, and schools starting at the same time. The authors did not consider different fixed costs for different bus types. They proposed the GRASP heuristic to solve the problem of real instance. Present study offers fewer fleet size (up to 37%) and lower traveled distance (up to 20%) while considering mixed load effect. Chen et.al (2016) introduced the problemcharacteristic with considering split demand for each stop. That means there is possibility of visiting each stop by several buses. Lima et al. (2016) developed five metaheuristic-based algorithms while considering mixed load and heterogeneous fleet. They also performed a comparison between the results of the proposed algorithms that have been considered in their paper. The solutions demonstrated that the mixed load approach entails a greater cost saving and a lower fleet size compared to the single load approach.

Additionally, Rodríguez-Parra et.al (2017) studied the school bus routing and scheduling in version of mixed load and single load effect. Riding special education student in mixed load case considered by Caceresa et .al (2015). This problem has significant different from regular SBRP and needs to consider careful attention.

In practice, considering bus with different seat configuration (the bus needs to be equipped with wheelchairs), picking up of student occur in their homes, etc. More over based on problem characteristics, they involve different bell time and more disperse location for considered school. They reveal that benefit of combining mixed load and different starting time can result in less number of buses. Recently, Lima et.al (2017) addressed the multi-objective meta-heuristic algorithms for multi objective SBRP featuring mixed load and heterogeneous fleet. The proposed objectives include total traveling time of students, balance of routes between drivers, routing cost. Four-multi objective ILS metaheuristics are developed and the attained result draw better performance respect to current literature. Another interesting paper in the context of mixed load SBRP is addressed by Miranda et.al (2018). Theyintroduced the scope of research that consider both mixed load (students of different schools can be in same bus) and multi load problem (pickup and delivery of students occur simultaneously, irrespective of their shift or commuting direction). They devised four version of heuristics based on an Iterated Local Search (ILS) framework with different strategies and features. The attained results reveal that local search with small time window strategy provide better results than other versions used. Bi-objective mixed integer linear programming formulation for mixed load SBRP is proposed by Mokhtari et.al (2018).

Many studies have addressed bus-scheduling approach (Desrosiers et al. (1981, 1986a), Swersey and Ballard (1984), Graham and Nuttle (1986), Fügenschuh (2009), Kim et al. (2012)). Fügenschuh (2009) considered a school bus scheduling problem that allows the adjustment of school starting times and transshipment of students among trips. An integer programming model based on VRPTW (vehicle routing with time windows) was introduced and solved through branch-and-cut method with several pre-processing procedures and valid cuts.

Kim et al. (2012) proposed a bus scheduling problem in which trips for each school are given separately. Each trip contains a sequence of stops and a related school. The problem is formulated as a vehicle routing problem with time windows. To assign buses to predefined trips, the authors adopted two approaches, namely the exact method for small cases and the heuristic approach for large cases. The findings derived from the above literature give a fairly good idea about the various aspects of bus routing problem while considering mixed load planning.

Table 1. Features studied in the literature

Reference	Urban/ Rural	Mixed load	Fleet mix	Cost	Constraint	Area	Load balancing	Starting and ending location of the bus	Share flexible depot	Sub problems considered
Bodin and Berman (1979)	Rural	✓	HO	FC	C, MRT TW	Brentwood New York		School		BSS BRG RS
Hargroves et al. (1981)	Urban	✓	HT	FC RC	C MRT MNS	Albemarle, Virginia		School		BRG RS
Boweman et al. (1995)	Urban		HO	FC SWD RC LB	C MWT	Ontario Canada	✓	School		BSS BRG
Desrosiers et al. (1981-1986)	Both		HO	FC RC	C, MRT, MWT	Drummondville, Canada		Depot		BSS BRG RS
Chen et al. (1988)	Rural	✓	HO	FC, RC	C, MRT	Choctaw Alabama		Depot		BRG BS

Li and Fu (2002)	Urban		HT	FC, TSD RC	C	Hong Kong	✓	Depot		BRG
Braca et al. (1997)	Urban	✓	HO	FC	C, MRT, TW, EPT, MSN	Manhattan, New York		Depot		RG RS
Spada et al. (2005)	Rural	✓	HT	TL	C, TW	Switzerland		School		BRG RS
Fügenschuh et al. (2009)	Rural		HO	FC RC	TW	Germany		Depot		RS SBT
Park et al. (2012)	Rural	✓	HO	FC	MRT TW C	Artificial		Depot		BRG RS
Campbell et al. (2015)	Rural	✓	HO	RC FC	MRT C TW	Missouri USA		Depot		BRG RS
Kang et al. (2015)	Urban	✓	HT	RC FC TSD	MWT TW DDT C	USA		Depot		BSS BRG RS
Bögl et al. (2015)	Urban	✓	HO	RC	TW C MWT	-		Depot School		BRG RS
Hernan Caceres (2015)	Sub Urban	✓	HO	RC FC	TW C MRT	New York United States		Depot		BRG RS
Ellegood et al. (2015)	Semi-rural	✓	HO	RC	C TW	Missouri USA		Depot		BRG RS
Silva et al. (2015)	Rural	✓	HO	RC	C MWT MRD	Brazilian city		School		BSS BRG
Chen et al. (2015)	Urban		HO, HT	RC FC	TW C	from literature		Depot		BRG RS

					MRT					
Yao et al. (2016)		✓	HO	RC	MRD C	-		School		BRG
Lima et.al (2016)	Rural	✓	HT	FC RC	C	Minas Gerais, Brazil		Depot		BRG
Lima et.al (2017)	Rural	✓	HT	TSD LB FC	C	Artificial and from the literature (Park et. (2010)		Depot		BRG
Caceresa et .al (2015)	Urban		HO	N RC	C MRT MWT	Western New York		Depot		BRG
Rodríguez-Parra et.al (2017)	Urban	✓	HO		C	Bogota		School		BRG RS
Mokhtari et.al (2018)	Rural	✓	HT	N TSD	C MRT	-		Depot		BRG
Miranda et.al (2018)	Rural	✓	HT	FC RC	C MWT MRT	Esp´ırito Santo, Brazil		Depot		BRG BSS
Our study	Urban	✓	HO	RC	C MWT TW MNS	Tehran Iran	✓	Depot	✓	BSS BRG

- Fleet mix
. Homogeneous fleet (HO)
. Heterogeneous fleet (HT)
- Constraint
. Vehicle capacity (C)
. Maximum riding time (MRT)
. School time window (TW)
. Maximum walking time or distance to bus stop (MWT)
. Earliest pick-up time (EPT)
. Minimum student number to

- Objective
Fleet cost (FC)
Routing cost (RC)
Total student riding distance (TSD)
Student walking distance (SWD)
Load balancing (LB)
Maximum route length (MRL)
Child's time loss (TL)

- Sub-problems considered in the literature
Bus stop selection (BSS)
Bus route generation (BRG)
Route scheduling (RS)
School bell time adjustment (SBT)

create a route (MSN) . Maximum riding distance of bus (MRD) . Depot departure time (DDT) . Maximum number of students in each stop (MNS) . Maximum route length (MRL)	Number of bus (N)	

III. Problemdescription and mathematical model

In this research, multiple schools, one type of student, potential bus stops, a set of garages, and identical buses (each of them with the same capacity) are taken into consideration. This study arose from the need to develop a daily transportation plan for bringing students from their home to their school. To do so, each student is allocated to an allowable bus stop while considering a defined walking distance. After that, each bus starts from the garage (starting location), picks up students from bus stop(s), delivers them to school, and finally returns to the garage (ending location). Each student has to be delivered to his/her respective school. Since the problem selects a set of bus stops and generates routes while considering mixed load effect, it is possible to assign students from different schools to the same bus. It is not necessary that starting and ending locations of a bus be the same, therefore, it potentially avoids long trips to return to the same garage. However, in order to prevent bus crowding in the same garage, an allowable number of parking spaces for each garage will be considered in the model. In order to make sense of the real case, two time constraints are presented in our model as follows: each bus should arrive at its related school before a defined latest arrival time and each stop cannot be visited by the bus before an earliest time.

Let G, P^{+} and P^{-} respectively define starting and ending locations of a bus, potential bus stops, and potential schools. P is the union of potential schools and bus stops($P = P^{-} \cup P^{+}$), and $N$is the set of all nodes ($N = P \cup G$).Travel time from node $i$to node j is calculated by travel distance between two nodes multiplied by the speed of the bus. For the sake of simplicity, all buses move at the same speed.

The objective function is to minimize the total travel distance of all routes. In our problem, school data consists of school location and only the latest possible times of bus arrival. More specifically, this study intends to pick up primary and secondary school students in such a way that each school can have a different time window.

The most important constraints of our problem are as follows:

1- Every student should walk from his/her home to one of the possible bus stops within a maximum walking distance.

2- Each bus starts from a garage and ends in the garage which is closest to the last school it visited.

3- Maximum allowable number of students for each bus stop does not exceed the limitms.

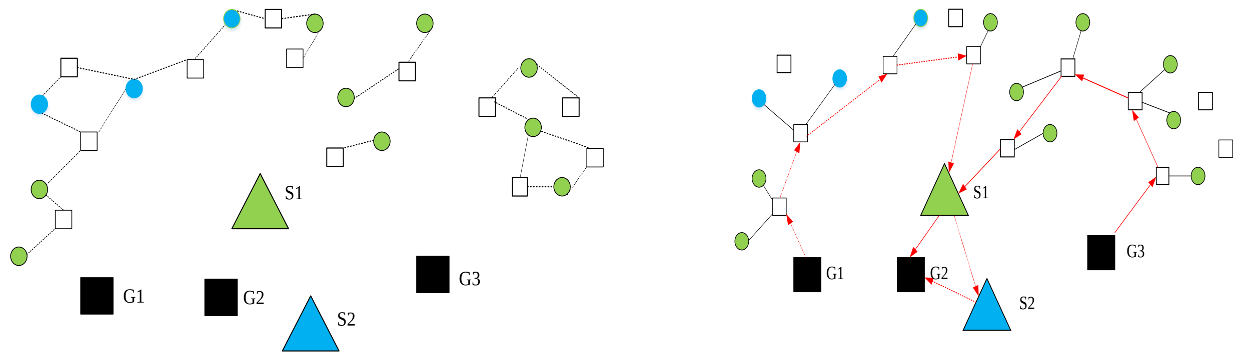
4-The number of buses coming back to a garage cannot violate the number of parking spaces at the garage P_g.

5- Each bus should arrive at its related school$i \in P^{-}$ before the latest timeb_i. We set upper bound on the time which a bus can deliver a student to his/her respective school.

6- Service time of each stop must be after earliest timea_i, and if the bus arrives at the first stop before a_i, it must wait.

7-The load of each bus along the path does not exceed its given capacity.

Among these items, constraints (1) and (3) take account of student convenience. Table 2 discusses the symbols used in the model.Figure 1a depicts an example of this problem. A student is denoted by a circle, anda potential bus stop is specified by a small square. A large black square represents a garage, while a trianglerepresents a school. Students are shown with the same color as their school. In this problem, a dotted line features the stop that a student can reach.A feasible (though not the optimal) solution is given in Figure 1b. In this figure, there are two routes denoted by red lines.Each route must start from a garage, pick up anumber of students from each stop, and transport them to the associated school prior to returning to the garage. The following assumptions have been considered in our problem:(1)each bus may carry the students of different schools at the same time;(2) each student must be picked up before being deliveredto the respective school;(3)each school can be visited by more than one bus, but each bus must visit each school once.



(a) Instance with possible allocation of students to the potential bus stops

(b) Possible feasible solution

Figure 1. Example school bus routing problem with mixed load effect

Table 2. Indices, sets, parameters, and decision variablesused in mathematical model

Indices	
k	Bus index
i, j	Node indices
l	Student index
Sets	
G	Set of starting and ending depot locations (garage locations)
K	Set of buses
s	Set of students
P^{+}	Set of potential pickup locations (bus stop locations)
P^{-}	Set of delivery locations (school locations)
$P = P^{-} \cup P^{+}$	Set of stops and schools
$N = P \cup G$	Set of nodes
Parameters	
c	Bus capacity
$big\ M$	Large constant
a_i	Earliest arrival times to stop $i \in P^{+}$
b_i	Latest arrival times to school $i \in P^{-}$
ap	Average pickup time at pickup points for each student
ad	Average delivery time at delivery points for each student

C_{ij}	Travel distance from node i to node j $(i, j \in N)$
t_{ij}	Travel time from node i to j $(i, j \in N)$
s_{il}	A parameter equal to 1 if student l can reach stop $i \in P^+$, and 0 otherwise
q_{il}	A parameter equal to 1 if student l is related to the school $i \in P^-$, and 0 otherwise
P_g	The number of parking spaces at the garage g
ms	The maximum number of allowable students for each stop
$O_i = \{S \mid s_{il} = 1\}$	The set of students that can be assigned to stop i
$W_i = \{S \mid q_{il} = 1\}$	The set of students that should be delivered to school i
Decision variables	
X_{ijk}	1 if bus k traverses the arc from node i to j $(\forall i, j \in N)$, and 0 otherwise
y_{ik}	1 if the bus k visits stop i, 0 otherwise
z_{il}^{k}	1 if student l is picked up by bus k from stop i, and 0 otherwise
T_{ik}	Arrival time of bus k to node i $(\forall i \in N)$
L_{ik}	The load of bus k after leaving node i $(\forall i \in P)$
h_{ik}	1 If bus k visits school $i \in P^-$, and 0 otherwise
D_{jl}^{k}	1 if student l is delivered by bus k to school j, and 0 otherwise

The mathematical programming formulation of the school bus routing problem is as follows:

$$Min \sum_{i \in N} \sum_{j \in N} \sum_{k \in K} t_{ij} X_{ijk} \qquad (1)$$

S.t.

$$\sum_{j \in N} X_{jik} = \sum_{j \in (P^+ \cup P^-)} X_{ijk} = y_{ik} \qquad \forall i \in P^+, k \in K \qquad (2)$$

$$\sum_{j \in (P^+ \cup P^-)} X_{jik} = \sum_{j \in N} X_{ijk} = h_{ik} \qquad \forall i \in P^-, k \in K \qquad (3)$$

$$\sum_{i \in G} \sum_{j \in P^+} X_{ijk} \leq 1 \qquad \forall k \in K \qquad (4)$$

$$\sum_{j \in P^-} \sum_{i \in G} X_{jik} \leq 1 \qquad \forall k \in K \qquad (5)$$

$$\sum_{i \in G} \sum_{j \in G} X_{ijk} = 0 \qquad \forall k \in K \qquad (6)$$

$$\sum_{k \in K} y_{ik} \leq 1 \qquad \forall i \in P^+ \qquad (7)$$

$$\sum_{k \in K} Z_{il}^k \leq s_{il} \qquad \forall l \in S, j \in P^- \qquad (8)$$

$$Z_{il}^k \leq y_{ik} \qquad \forall l \in O_i, i \in P^+, k \in K \qquad (9)$$

$$y_{ik} \leq \sum_{l \in S} Z_{il}^k \qquad \forall i \in P^+, k \in K \qquad (10)$$

$$\sum_k D_{jl}^k \leq q_{jl} \qquad \forall l \in S, j \in P^- \qquad (11)$$

$$D_{jl}^k \leq h_j^k \qquad \forall l \in W_j, j \in P^-, k \in K \qquad (12)$$

$$h_{jk} \leq \sum_{l \in S} D_{jl}^k \qquad \forall j \in P^-, k \in K \qquad (13)$$

$$\sum_{i \in P^+} Z_{il}^k = \sum_{j \in P^-} D_{jl}^k \qquad \forall l \in S, k \in K \qquad (14)$$

$$\sum_{i \in P^+} \sum_{k \in K} Z_{il}^k = 1 \qquad \forall l \in S \qquad (15)$$

$$\sum_{l \in S} Z_{il}^k \leq ms \qquad \forall i \in P^+, k \in K \qquad (16)$$

$$L_{ik} = 0 \qquad \forall i \in G, k \in k \qquad (17\text{-a})$$

$$L_{ik} + \sum_{l \in S} Z_{jl}^k \leq L_{jk} + bigM(1 - X_{ijk}) \qquad \forall i \in P, j \in P^+, k \in k \qquad (17\text{-b})$$

$$L_{ik} - \sum_{l \in S} D_{jl}^k \leq L_{jk} + bigM(1 - X_{ijk}) \qquad \forall i \in P, j \in P^-, k \in k \qquad (17\text{-c})$$

$$\sum_{l \in S} Z_{il}^k \leq L_{ik} \leq C \qquad \forall i \in P^+, k \in k \qquad (17\text{-d})$$

$$T_{ik} + ap.\sum_{l \in S} Z_{il}^k + ad.\sum_{l \in S} D_{il}^k + t_{ij} \leq T_{jk} + bigM(1 - X_{ijk}) \qquad \forall i \in P, j \in P, k \in K, i \neq j \qquad (18)$$

$$T_{ik} + t_{ij} \leq T_{jk} + bigM(1 - X_{ijk}) \qquad \forall i \in G, j \in P^+, k \in K \qquad (19)$$

$$T_{ik} \leq T_{jk} + bigM(1 - Z_{il}^k) \qquad \forall i \in P^+, j \in P^-, l \in S, k \in K \qquad (20)$$

$$T_{ik} \geq a_i - (1 - y_{ik})bigM \qquad \forall i \in P^+, k \in K \qquad (21)$$

$$T_{ik} \leq b_i + (1 - h_{ik})bigM \qquad \forall i \in P^-, k \in K \qquad (22)$$

$$\sum_{i \in P^-} \sum_{k \in K} X_{ijk} \leq Pg \qquad \forall j \in G \qquad (23)$$

$$y_{ik} \in \{0,1\} \qquad \forall i \in P^+, k \in K \qquad (24)$$

$$X_{ijk} \in \{0,1\} \qquad \forall i, j \in N, i \neq j, k \in K \qquad (25)$$

$$Z_{il}^k \in \{0,1\} \qquad \forall i \in P^+, l \in S, k \in K \qquad (26)$$

$$D_{jl}^k \in \{0,1\} \qquad \forall j \in P^-, l \in S, k \in K \qquad (27)$$

$$h_{ik} \in \{0,1\} \qquad \forall i \in P^-, k \in K \qquad (28)$$

$$r_{lk} \in \{0,1\} \qquad \forall l \in S, k \in K \qquad (29)$$

The objective function (1) minimizes the total travel time traversed by all buses. Constraints (2) ensure that a bus entering the stop node should leave it as well. The same constraints for school node are shown in Equation (3). Constraints (4) represent that a bus cannot start more than once from its home location (garage). Similarly, constraints (5) guarantee that a bus cannot arrive at its final location (garage) more than once. Infact, some buses could remain unused. Constraints (6) demand that no transferring from garage to garage is possible directly. Constraints (7) impose that each stop is visited no more than once. Constraints (8) enforce that each student is picked up from the stop to which he/she walks. Constraints (9) specify that picking up a student from a non-visited stop by bus$k$ is not possible. Constraints (10) guarantee that stops are not visited unnecessarily. Constraints (11) ensure that each student is delivered to its respective school. Constraints (12) guarantee that whenever a student is assigned to a bus, the school associated with this student is also visited by the same bus. Constraints (13) guarantee that schools are not visited unnecessarily. Constraints (14) impose that number of pickup and delivery students in each route is equal. Constraints (15) state that each student should be picked up exactly once. Constraints (16) ensure that number of allocated students to each allowable stop must not be more thanm_s. The next four sets of constraints (17-a), (17-b), (17-c) and (17-d) are load constraints. Constraints (17-b) state that when a node i is followed by a pickup node $j \in P^+$, the number of students after visiting node j is greater than or equal to the summation of the number of students after servicing node$i$ and number of picked up students in the node j. Similar to constraints (17-b), inequality (17-c) proves that when the node i is followed by the delivery node, $j \in P^-$ the number of students after visited node j is greater than or equal to the number of students after visiting node i minus the number of students delivered to the node j. In practice, constraints (17-b) and (17-c) determine load on a bus just after leaving each node on its route. Constraints (17-d) ensure the capacity of buses. Constraints (18)-(23) are time-related constraints. The arrival time of each bus to a node in p is calculated in Constraints (18). Constraints (19) are similar to constraints (18), but they are for the routes from garage to stop. Constraints (20) ensure that pick up of students by a bus is before his/her delivery. Constraints (21) and (22) indicate the time window for stops and schools, respectively. Constraints (23) restrict the number of available parking places in each garage. Finally, variables and their types are declared in (24)-(29).

IV. Solution strategy

SBRP is a generalization of vehicle routing problem (VRP), known to be an NP-hard problem. Although the exact methods suggested in the literature are capable of obtaining optimal solutions, they can be used only to solve problems with a relatively small number of stops/students. This is relatively far from real cases that involve hundreds of stops/students.Therefore, heuristic approaches are needed to cope with large instances and to achieve near-optimal solutions in a reasonable amount of time. Several variants of heuristics, based on local search contexts, have been applied to VRP. Local search operators create regular moves that slightly alter the current solution. These moves could change the requests within one or two different routes at the same time. This kind of operator can search through a large number of solutions in a short timewhilecausing a small change at each iteration. This approach also has some limitations, so that, for instance, applying a tight constraint to the problem and implementing local search operator are not sufficiently profitable (Ropke&Pisinger, 2006). Thus, in this case, moving from one promising area to another favorable point is problematic. There are alternative strategies to cope with this problem. To tackle this issue, one strategy is to use the large standard move instead of incorporating small moves. Unsurprisingly, employing this case requires an expensive computing time, compared with the small standard move strategy; nevertheless, it helps to obtain more desirable results in terms of the quality of thesolution. Therefore, instead of applying small changesto the solution, it is more effective to use large moves, resulting in more exploration in the solution space. To the best of our knowledge, the application of largeneighborhood search in an SBRP context is novel, and its methodology is known to have managed to solve a variety of VRPs.Hence, an argument can be made in favor of exploring very large spaces in the solution using large neighborhood search (LNS). LNS simply consists of a set of neighbourhoods meant to destroy and rebuild the solution. Destroy and repair operators canbe employed in different ways in an incumbent solution. The enriched version of LNS, i.e.,adaptive large neighborhoodsearch (ALNS), was introduced byRopke&Pisinger, 2006). ALNS extends LNS by considering a set of removal and insertion heuristics in the search space. In each iteration, removal and insertion operators are selected on the basis of probabilities. This probability is updated dynamically according to the performance reached by each operator in a preceding iteration. This inventive aspect of LNS, proposed in this paper,provides the possibility forexploring the whole neighborhood in the search space. Themechanism of ALNS algorithm is shown in Algorithm 1.The ALNS metaheuristic presented here is based on two stages, namely construction stage and improvement stage. In the first stage, a student allocation problem is solved for each stop. After student allocation heuristic is implemented, a variant of the nearest neighborhoodconstructive heuristic is applied to generatea feasibleinitial solution. The solution obtained through the initial solution serves as input for thesecond stage, i.e. improvement stage (Line 7 of pseudo-code). The improvement stage consists of two levels that are executed sequentially through a number of iterations. More specifically, the algorithm tries to improve the solutionby *using adaptive large neighborhood*heuristics in the primary level (Line 8 of pseudo-code). More precisely, at each iteration, a number of$q$ stops are disconnected via the removal heuristic and are placed into the stop pool, called U bank list. Then, using insertion heuristic, it inserts the stops from Ubank into the solution. Note that during removal and insertion operations, it might happen that different students from different schools are inserted in the same route. In this case,while preserving feasibility, the school associated with the student also needs to be inserted in the cheapest position of thecurrentroute.

The value of q is the key parameter that determines the scope of our solution approach. In other words, parameter q indicates neighborhood's size. If the size of q is equal with zero, no search will occur in solution space. In contrast, when the value of q is equal to the cardinality of P^+, the algorithm acts similar to a multi-start, and the problem is solved from scratch. In addition, this value can be dependent on the behavior of solution at each iteration. In order to make a balance between diversification and intensification mechanisms, the following procedure is applied for updating the value of q. Initially, the value of q is set to q_{min} (line 17 of pseudo-code) and systematically altered during the adaptive large neighborhood search algorithm. More precisely, the value of q is modified based on the solution created during the previous iteration. For instance, if for a number of iterations, an acceptable solution is obtained, the value of q should be kept in the low level,q_{min},in order to keep intensification. In contrast, for number of iterations worse solution is appeared, the value of $q$must be increased in order to investigate the solution space more efficiently. As a result, large numbers of q are removed and then re-inserted. It can help in achieving a suitable diversification strategy during the search.

Due to considering a set of removal and insertion heuristics, the choice of these heuristics are governed by a roulette wheel mechanism according to their past successfulbehavior. We also implement the *Meta-destroy* operator when after δconsecutive iterations; no improvement is made in the best solution. This operator works byimplementing two destroy operators sequentially, creating more diversification. It should be noted that this procedure is independent of performing aset of removal and insertion steps in each segment. More precisely, in each segment,the number of successive non-improving solutions is counted from the beginning of the improvement stage, and if this number is greater than $\delta$the Meta-destroyoperator isexecuted (the value of $\delta$is lower than the number of iterations in each segment).

Once the removal and insertion operators are applied, *Redistribution operator* is considered in the second level whenever a new best solution is found (line 15 of pseudo-code). The reasoning is that applying both removal and insertion heuristicshelps reconstruct a large part of the solution, such thata dispersed distribution of students between routes occurs. In order to cope with these situations, redistribution operator attempts to optimize the current load distribution. This operator tries to transfer students between routes while preserving feasibility. In practice, this operator is meant to minimize the corresponding deviation of the loading value of the routes through making a desirable distribution of students among the routes.

ALNS algorithm demonstrates a very good performance for large scale optimization problems.It has provided especially great resultsfor vehicle routing problems. Such successful applications of ALNS on VRP have inspired the present study to employ it for school bus routing problem with the mixed load plan.

Algorithm 1. AdaptiveLarge Neighborhood Search metaheuristic

Input:U: set of all potential stops, G: setof all garages,P^-:set of all schools, s: set of all students

R(set of Removal heuristics), I (set of Insertion heuristics), q (number of stops/ requests to be removed $q \in \{1,\ldots n\}$),q_{max} (maximum number of stops to be removed), P^+(Listof stops to which students are allocated), π(initial score of heuristic $(I \cup R)$),w(initial weight of removal and insertion heuristic $(I \cup R)$)

ρ(the number of iterations), η(parameter to set q_{max}**)**

1 ***// Stage 1: Construction phase***

2 P^+=all student allocated to the bus stop // Allocating using student allocation heuristic

3 x_o =Route generation (P^+, s, G, P^-) // Generating route using NNg heuristic

4 $x_{best} = x_o$

5 $f_{best} = f(x_0)$

$x_{act} = x_o$

// Stage 2: Improvement phase

/// 2.1 Set of removal and insertion heuristics in the first level

$q = q_{min}$ initialize the roulette wheel; initialize the adaptive parameters(π, w)

While *S*topping criterion ρ is not met **do**

Roulette wheel mechanism: Select **one** Removal heuristic$\boldsymbol{h_{rem}} \in \boldsymbol{R}$and **one** Insertion heuristic$\boldsymbol{h_{ins}} \in \boldsymbol{I}$or **two**Destroy operators (ifx_{best}has not been improved in last consecutiveδ iterations)

Remove $q$requestsfrom solution x_{act} using h_{rem},creating a partial solution

Insert q customers into the partial solution usingh_{ins},creating a solution x^*_{act}

If accept (x_{act}, x^*_{act}) then

/// 2.2 Redistribution operator in the second level

$x^{**}_{act} = Redistribution(x^*_{act})$ // Applying Redistribution heuristic to x^*_{act}

$x_{act} = x^{**}_{act}$

$q = q_{\min}$

Else

$$q = q + 1$$

If $q = q_{\max}$

$q = (\frac{q_{\max}}{\eta})$

End if

End if

If $f\ (x_{act}^{**}) < f_{best}$

$x_{best} = x_{act}^{**}$

End if

Update the roulette wheel(π, w)

End while

End if

4.1. Constructingan initial solution

The key concept of the construction stage is togenerate a feasible initial solution.Constructing an initial solution for the SBRP consists of three steps: assigning each student to an allowable stop, grouping allowable stops based onthe closest garage, and generating routes for allowable stops of each garage. To do so, three stages are sequentially performedas follows. Each student is first allocated to anallowable stopusing student allocation heuristic in the first step. After allocatingall existing students to the possible bus stop, the potential stops with their respective students are specified. In thesecondstep, each identified stopis assigned to the closest garage.Thus, the distribution of each stop to a given garage is obtained in advance.Then, the modified nearest neighborhood with the greedy randomized adaptive procedure (NNgr) is employed to generate a route. Greedy randomized selection reflects a balance between greediness and randomness approaches. To this aim, instead of using simple greedy nearest neighborhood heuristic, our version considers some modifications to construct routes in the following twoways.(1) For each route (i.e., bus) started from a given garage, the next node (i.e., stop) is selected randomly from the restricted candidate list(RCL) containing α first closest non-visited stops,belonging to that garage.It is worth mentioning that the non-visited stops for each given garage are those that students are assigned toit in advance. The size of RCL, i.e. the value ofα, is a parameter that controls the value of greediness and randomness.If$\alpha$is set to a small value, the construction is extremely in the greedy fashion. In contrast, ifα is large (equal to a number of non-visited stops in the solution), the construction is completely random.(2) Feasibility checkis carried outwith respect to bothallowablecapacity of thebus and school time window constraints. If afeasible solution is generated without violating the above constraints, the candidate stop will be addedtothe route. Otherwise, the generated route returns to the associated school to deliverstudentsand,finally, returns to the closest garage.Returning to the closest garage prevents long trips that may result from returning to the same garage the bus started from.If the capacity of the nearest garage is already filled, the next closest garage is selected.As the starting and ending locations of the bus are not necessarily the same, our problem is the Open Vehicle Routing Problem.

A new attempt to apply the twoways is made until all non-visited stops are considered for each garage. Since each bus might pick up students from different schools, all associated schools must be inserted at the end of the considered routeinthe cheapest way possible.More importantly, to make effective time saving,data structure is employed andupdated throughoutthe operation of NNgrheuristic,which consists of information related to load and travel time of a busriding along route k. The example in Figure 2 clarifies the operation of thedata structure. Along the operation of NNgr for each stop,the following information isupdated for each stop as follows:visited stop (S), number of studentsallocated to the stop (AS), load of bus after visiting the currentstop (LS),load of bus after visiting the next stop (LN),remaining capacity after visiting the current stop (R), remaining capacity after visiting the nextstop (RC), arrival time at the current stop(AT), andallowable remaining time to reachthe respective school(RT). This simple data structure procedure allows to efficiently check both capacity and school time window constraints prior to selectingany stop. In this example, it is assumed that thecapacity of the bus is equal to 6. The solid line shows the generated route.It is supposed that stop B is acandidate to be inserted in the route. As seen from Table3, the value of LA and LB is 5 and 7, respectively. This indicates that inserting stop B inthe route contributes to making an infeasible solution.As a result, instead of visiting stop B,the bus must come back to the respective schools.

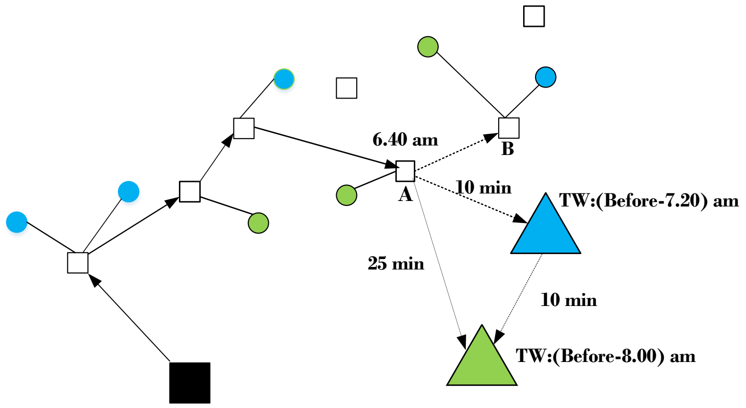


Figure 2. Example for constructing an initial solution

Table 3. Data structure

Stop(S)	AS	LS	LN	R	RC	AT	(RT)Travel time to school 1	(RT) Travel time to school 2
A	1	5	7	1	-1	6.40	10 min	25 min

4.2. Adaptive large neigbourhood search (ALNS)

In general, ALNS heuristic is aniterative process that consists of destruction and insertion operators. In practice, at every iteration,a removal heuristic is utilized to remove a number of stops (i.e. requests) from the current solution;then aninsertion heuristic is employed to insert them back to the current solution so as to construct a new solution(for more information, the reader is referred toRopke&Pisinger, 2006). Removal heuristics are described inSection 4.2.1, and insertion heuristics are discussed in Section 4.2.2.

4.2.1. Removal operators

Removal operator is the backbone of the algorithm, where at each iteration, q stops are removed and added to the list U. These operators must be selected in such a way that they efficiently explorethe whole search space or at least its interesting parts. Thus, it would make no sense to only focus on special kinds of destroyoperators. Therefore,it is needed to consider diversification and intensification operators in a structured way.Thisstudyemploysvarious kinds of removal heuristics allowing both diversification and intensification strategies. Three removal heuristics have been inspired by Ropke, and others are new and adapted according to the problem'sconsiderations (SBRP).

Shaw removal

The basic idea of Shaw removal was proposed byRopke&Pisinger (2006). This heuristic operatesbased on the similarity idea.In this approach, it seems reasonable to select stops that are somehow similar, thus making iteasier to replace them inanother place in the hope of improving the solution.If we insert the requests that are different from others, it might not obtain a gain when inserting themin thecurrentsolution, becauseit may be inserted in a bad or the original position.

The degree of similarity between stops$i$ and$j$ is calculated by relatedness measure$R(i,j)$that correspondsto their distance from each other as follows: $R(i,j)=\sqrt{(x_i-x_j)^2+(y_i-y_j)^2}$ .To do so, one request is selected randomly from the list U. If the list is empty, the following procedure is applied tothe selected candidate stop. In fact, instead of selecting one stop randomly, as presented by Shaw etal. (2006), we select the stop that produces a higher possibility ofcost saving.To do so, one route is randomly selected in advance. After that, for each stop$i$thatis included in the considered route, the value of saving is calculated as follows:

$s(i) = dist(prev(i), i) + dist(i, next(i)) - dist(next(i) - prev(i))$,where$prev(i)andnext(i)$arerespectively the predecessor and successor of anode$i$in the considered route.Node candidatesare sorted indecreasing order.The underlying idea is that the request resulting in a greater saving for the solution has more potential to be selected, and it canimprove the solution quickly if inserted in another position. Afterward, the stop with the maximum saving is selected to be transferredto list U. In the second step, the degree of similarity between the selected stop in the list U and the other $q-1$ stops that are not removed from the current solution is calculated.The process of Shaw removal ispresented in Algorithm 2.Letybe a random parameter between $[0, 1]$ and$p$define the degree of randomness to the selected request. A lower value of$p$forces the heuristic to choose more similar stops whereas a high value allows selectingless similar requests.

Algorithm 2. Shaw Removal (inspired by Ropke)

Function Shaw removal $\{x_{act} \in solution, p \in R^{+}, q \in P^{+}\}$

Request: r = selected request from x_{act} using the saving method;

Set of Request: $U = \{r\}$;

While$U \leq q$**do**

r = selected request from U;

Array: L = an array containing all request from x_{act} not in U;

Sort L such that $i \leq j \rightarrow R(r, L[i]) < R(r, L[j])$;

Choose a random number y from the interval [0,1);

$U = U \cup \{L([y^{p} |L|])\}$;

End while

Remove the requests in U from x_{act};

Shaw removal based on similarity of school of the student

This operator uses some of ideas from the Shaw removal heuristic. The only difference is that instead of using relatedness measurement $R(i,j)$between two stops based on distance, we consider the degree of similarity with respect to schools. More formally, this heuristic attempts to remove a set of stops that are similar based on the variety of schools with respect to their assigned students, as it is expected to be rationally easy to reshuffle these stops and thereby avoiding the unnecessary inclusion of a school. In practice, the degree of similarity indicates how much two stops are similar based on the schools of their students. In doing so, like Shaw removal the procedure primarily selects a random stop to remove and in the subsequent iterations it selects stops that are similar to the already removed requests based on the on the schools of their students. The degree of similarity between two stops is given by:

$$d(S_1, S_2) = \sum_{k} \left| y_{s1}^{k} - y_{s2}^{k} \right| \qquad (30)$$

Let y_{s1}^{k} determines whether stop$s1$ has any student of the school$k$. Therefore y_{s1}^{k}takes value of 1 if stop $s1$ has a student of school k, and 0 otherwise. Usingy_{s}^{k}, we can easily obtain the degree of similarity between two requests. The lower $d(S_1, S_2)$indicate that two stops ($s1$ and $s2$) are more related. This procedure proceeds until q stops are selected and transferred to the U bank.

Worst removal

This heuristic is inspiredby Ropke&Pisinger(2006). It removes q stops withhighest gains. In practice, it removesrequests that are rather expensive and thus inserts them in anotherpositionin the hopeof findinga better solution. Let the value of f_i^{+}bethe cost of thesolutionwhen therequest$i$isin the current solution, and f_i^{-}be the cost of solution without considering therequest$i$. $cost(i, x_{act}) = \Delta f_i$defines the difference between f_i^{+} and f_i^{-}. This heuristic is randomized through setting the parameter p. More precisely, a lower value of p forces the heuristic to select a requestfeatured with a high value of gain, while lower values of $p$allowschoosing requests with low values of gain.

Algorithm 3.Worst Removal (inspired by Ropke)

Function Worst Removal $\{x_{act} \in solutionS, p \in R^{+}, q \in P^{+}\}$;

while $q > 0$**do**

Array: L = All planned requests i, sorted by descending $cost(i, x_{act}) = \Delta f_i$;

Choose a random number y from the interval [0,1);

Request: $r = L([y^{p}|L|])$

remove r from solution S;

$q = q - 1$;

end while

Random removal

Random removal operator - introduced by Ropke&Pisinger(2006) is used to create appropriate diversification. The idea behind this operator is to introduce a degree of randomization in the solution space.It selects some stops randomly and inserts them into the list U. Since this operator allows one to diversifythe search, it would be very helpful to overcome local optima.

Least used bus removal (LUB)

This operator tends to remove the bus, or the route, with the smallest load.A load of thebus is the number of picked up students in the route. In fact, the route with the least occupied capacity is selected, and all stops contained in the route are removed. This removal operator aims at completely destroying the route.

Single load route removal (SLR)

This heuristic randomly selects a route that only containsstudentsbelonging toa single school(i.e. single load route)and attempts to remove the $q$stops included in the route. This heuristic pursuesa similar strategy as that adopted in the least used bus removal. For both single load route and the least used bus removal heuristics, if the number of request$i$ in the candidate route k is less than q, another route will be selected. It should be stated that this operator strives to reducesingle load routes.

4.2.2. Insertion operators

The purpose of these heuristics is to construct the partially destroyed solution through inserting requests from the list U into the existing route if possible.Two key points need to be taken into account while conducting the insertion procedure. Firstly, the request shouldbe inserted in any feasible position.Feasibility should be maintained with respect to capacity constraint and school time window during the insertion process.Secondly, while insertion happens for the stop inthe candidate route,the heuristic must check the stop to see whether the related school is already inthe new route or not. If it is not, theschool needs to be inserted in the considered route in thecheapestpossible position. More importantly, during the insertion operation when the candidate stop to be inserted has the studentsfrom other schools, the cost of insertion is the summation of cost created by inserted that stop and the respective school(s).

Althoughpursuing the above procedures is time-consuming and increases the complexity of the algorithm, it promotes diversification. To this aim, our insertion heuristic follows two strategies, namely local and global insertions. The first one means that astop of the list U is only allowed to be inserted in those routesin which their related school is located.In contrast, in the global insertion,regardless of the existence or absence of a related school, unplanned stops can be inserted at any best position of the existing route. In this study,basic greedy, regret 2 and regret 3 operators are employed based on global insertion method, while basic greedy based on the largest demandand the second-bestinsertion areexecuted usinglocal insertion.

Basic greedy heuristic

This heuristic has a greedy nature. The basic greedy heuristic looks for the cheapest insertion position for all unserved requests in the listU. Let Δf_{ik}be the value of theobjective function when therequest$i$is inserted in the route k. It should be carried outin such a way that the cost of thesolution is increased minimally.Thus, the cheapest insertion cost is formulated as $C_{(i,x)}{}^{+} = min_{k \in R}(\Delta f_{ik})$. This procedure continues until all unserved stops in the listU have been inserted.

Basic greedy based on the largest demand insertion

This heuristic only differsfrom the basic greedy in selecting a request from the U-bank. In fact, the basic greedy heuristic simply selects the first stop from the list U and inserts it in the cheapest position, whereas basic greedy based on the largest demand tries to select unplanned stops from the list Ubased on the amount of demands.As a result,the request with the largest demand in the list U is the first to be inserted.

Second-best insertion

This heuristic is slightly different from basic greedy, for it tries to insert the request in thesecond-bestposition in order tocreate diversification.

Regret-K heuristic

Dissimilar to above greedy insertion, the regret heuristic does not pursue the best insertion position policy. In fact, this heuristic is devised to improve the myopic behavior of greedy insertion heuristic. This heuristic attempts to add the stops that create maximum difference, concerning the cost of the inserting in best position tok^{th}best route position. The main concept in this method is to make a priority to insert a stop in the first stage when it will lead to more costs if inserted later. In doing so, the least insertion cost for all demands in U is calculated in advance, same step like basic greedy heuristic can be pursued. Afterward, the summation of the difference between best insertion (first element) and second best to k^{th}best route (k^{th}element) is calculated. This difference is addressed in the literature as regret value. (for more detail see Ropke&Pisinger (2006)).The regret-K heuristicadopted in this paper assumes the values of 2 and 3 for k, meaning that we calculate the difference between the second best and the best insertion positions for k equal to 2, and the summation of differences between the third bestthe second best, and the best insertion positions for k equal to 3. In the all insertion operators, if there is not any feasible route for insertion, new routes are created to visit remaining stops in the list U.

4.3. Redistribution operator

Due to employing a set of large neighborhood search heuristics in the previous stage, distribution of students between routes could lose its balance. This means that some routes contain a large number of students, and othersinclude smallernumbers. To tackle this situation, a redistribution operator helps balance the current capacity to the current solution. At the beginning, a list of routesis developed in decreasing order on the basis of the occupied capacity. After that, for the first β routes in the list,an attempt is made to move the student to another allowable stop in another route, if possible.In this study, this value is set to 25% (it is found through pilot study). The only exception is when the number of routes generated in the incumbent solution is less than 4.In this case,redistribution operator is deactivated since a lower number of routes causes redistribution operator to have acorrespondingly lower efficiency in transferring students. Transferring studentsto any new route is possible when two conditions are met:first, ensuring that an allowable stopexists for the candidate student;second, making sure that a respective school is considered too. The latter means that once there is the possibility to transfera student, a related school must also be present at the end of the route. If this is not fulfilled, an associated school needs to be inserted.

4.4. Adaptive search engine

Adaptive weight adjustment assesses the importance of each removal and insertion on the basis of its performance to produce a profitablesolution. At each iteration of ALNS heuristic, one removal and one insertion operatorneed to be selected. Choosing different removal and insertion operators at each iteration has some advantages. Firstly, it prompts to diversify the search in an efficient way. Secondly, it leads the algorithm to find better results. It happens that the combination of one insertion with one removal might perform well for some instances, and in other instances some other removal and insertion operators might behave better. This alternation between different removal and insertion heuristics yieldsan experimentally robust heuristic. Finally, it can help to make a good balance between computing time and solution quality, since implementing a number of insertion or removal operators individuallyis time-consuming. The question is how the algorithm selects removal and insertion operators. The selection of removal and insertion is governed by a roulette-wheel mechanism whereby each of the operators is assigned a weight. The probability of selecting each heuristic is dependent on how successfully it has performed in previous iterations. More precisely, each operator is assigned a score and the operator that yields a better solution has a higher probability of being selected again. It means that the operator with a poor performance still has a small chance of being chosen. In our study, selecting removal and insertion heuristics at each iteration is carried out through pairwise selection mechanism.

4.4.1. Adaptive weight adjustment

In this section, we explain the methods ofchoosing removal and insertion heuristics thatisbased on pairwise selection method. More precisely the majority of literature concentrate on independent selection of removal and insertion operators (see Ropke&Pisinger, 2006). This leads to missing the opportunity of identifying joint performance of removal and insertion on the performance of metaheuristic. Because of this, it attempts to consider the joint performance of a pair of operators and correspondingly assigns weightρ_{dr} to the operators based on their performance. At the outset of a segment, all pairs have the same weight $\rho_{dr} = 1$andall scoresare set to 0. During each segment, every time that a pair of removal and insertion is applied, its score is increased by the parameters $\sigma_1^*, \sigma_2^*, \sigma_3^*$depending on its performance. If the pair finds a solution that improvess_{best} , the score of thepair is increased by σ_1^*; in case the solution is improved, though not better than the best solution, the score of thepair is increased by σ_2^*, and finally the worse solution results in the scoreincrease byσ_3^*. After each segment is terminated, the weight is updated as follows:

$$\rho_{dr} = \gamma \frac{\Psi_{dr}}{\max(1, O^*_{ij})} + (1-\gamma)\rho_{dr} \qquad (31)$$

Like Ropke&Pisinger(2006) method, the value ofγreflects the reaction factor,O_{ij}^* specifies the number of times the pair of removal and insertion i is applied tosegment$j$, and Ψ_{dr} represents the score of each pair of removal and insertion. Better results achieved by each pair are assigned greater weights and,therefore,the pair has a higher likelihood for selection. Let n_d and n_r be the number of destroy and repair operators, respectively. At each iteration, the roulette wheel mechanism is utilized to choose one pair of removal and insertion operators with probability $\Phi_{dr} = \frac{\rho_{dr}}{\sum_{d=1}^{nd}\sum_{r=1}^{nr}\rho_{dr}}$.

4.5. Acceptance and stopping criteria

Another important component in an ALNS metaheurisitc is related to the solution acceptance rule. Once a new solution is generated through destroying and rebuilding operators, the acceptance criterion rule is used to decide whether the new solution is accepted or not. There are different types of acceptance methods. The better acceptance method accepts a solution only when it is better than the previous one. This simple acceptance rule promotes intensification; however, it has a tendency to be stuck in local optima. To this aim, not limiting algorithm by accepting only improving solutions seems to be a reasonable idea for escaping local optima. To make a balance between intensification and diversification, it seems sensible to occasionally consider the worse solution as well. In this regard, the judgment of accepting a new solution is made according to simulated annealing method (Kirkpatrick, Gelatt, and Vecchi 1983). If a new solution $f^{/}(s)$ is better than the previous one $f(s)$, search continues with asolution$f^{/}(s)$. Otherwise, the worse solution $f^{/}(s)$ is accepted with probability: $P = exp(\frac{f(s)-f(s^{'})}{T})$, where $f^{/}(s)$and $f(s)$respectively denote the objective function of the new and incumbent solutions, and $T > 0$signifies the temperature parameter. The process starts from the initial temperature T_{start} and the temperature is gradually decreased by replacing $T = T \times C$at each iteration, where $0 < c < 1$ and is used to represent a cooling factor parameter. This decline during the operation of algorithm implies that a non-improving solution is less likelyto be selected in subsequent iterations. It is important to note that the desirable value of T_{start} depends directly on the problem in question. Hence, instead of considering T_{start} a fixed parameter, we calculate its initial value using the results obtained in the initial solution x_o (this idea was suggested by Dayarian etal. (2013)). In effect, the initial temperature is set to$\frac{-w x_0}{\ln(0.5)}$. This formulation allows that w% worse solutions, here it is set to 5%, be accepted with a probability of 50%. The value of parameter$w$needs to be determined.

5. Experimental analysis

Our experimental computations consist of two parts. On primary stage, (sections 5.2 and 5.3) calibration is carried out in order to determine best parameters and also appropriate operators that significantly influence the metaheuristic's performance. For both sections 5.1 and 5.2, the testing set was made up of 10 instances (4 instances from set S, 4 instances from set M, and 2 instances from set L). Having obtained best parameter settings the analysis is performed to investigate the effect of single load and mixed load structure on the quality of the solution (minimizing total travelling time). Finally, a comparison is carried out with current literature in this area forunderstanding the main effect of mixed load strategy.

5.1. Instance generation

Since the presented problem has not been considered earlier, no test instances were available in the literature. For this purpose, new data sets were generated to conduct experiments. The data set contains 100 instances. The problem size of this data set varies according to the number of garages ranging from 2 to 5, the number of schools ranging from 1 to 7, the number of stops ranging from 5 to 50, the number of students ranging from 25 to 250, and the walking distance ranging from 5 to 25. Our data set consists of three sets, including small, medium and large instances. Small instances have 5 to 10 stops, while medium and large instances have between 15 to 30 and 35 to 55 stops, respectively. In order to avoid complexity to generate instances, the number of garage and number of students are calculated based on the number of stops.

To generate the data set, 6 parameters per instance should be defined in the primary stage: n_g(the number of garages),n_h (the number of schools), n_p (the number of stops), n_s(the number of students), and w_{max} (maximum walking distance for each student to reach a bus stop). All instances are generated and scattered in the Euclidean square between (0,0) and(x_{max} , y_{max}). In order to make dataset similar to the real world, the values of (x_{max} , y_{max})are set to (80 *80 km). Each school's coordinates are generated in the area of (20km ×60km) with respect to the center of Euclidian square. Correspondingly,the coordinates of each stop are generated in the interval of $(w, x_{max} - w), (w, y_{max} - w)$. For each generated stop, the coordinates of each student is obtained based on the angle $\alpha_j \in [0,2\pi]$and walking distance$w$ from the stop. Thus, the coordinates of each student are obtained by $x = x_s + w \cos \alpha_j$ and$y = y_s + w sin\alpha_j$. The last part in our calculation suggests the allocation of students to a school. To handle this issue, the average number of students for each school is calculated,and thenit is attempted to assign students to the closest school as long as the number of assigned students to each school reachesthe averagevalue. In this case, the students are assigned to the second closest school. This procedure continuesuntil all students are assigned properly. The garage departure time was fixed to 6:20 for all buses, and maximum arrival time to each school is randomly generated within the period (7:45 to 8:15) a.m.

5.2. Calibration of themetaheuristic parameters

The proposed metaheuristic comprises of parameters that are essential to be set and tuned. This stage consists of statistical analyses to obtain a best parameter setting. This is conducted by full factorial experimental design of parameters on a subset of instances. The parameters considered for analysis are summarized in Table 4 and include number of iterations (ρ), number of iterations without improvements (δ), minimum and maximum percentage of requests to be removed(ξ_{min} , ξ_{max}),parameter to control value of q_{max} (η),weight adjustment in roulette wheel mechanism($\sigma_1, \sigma_2, \sigma_3, \gamma$), size of restricted candidate listα, and randomness parameter in the removal procedure (p). The findings of the analysis are shown in Table 5. The output of Multi Anova highlights that number of iterations, number of iterations without improvements, the minimum and maximum number of requests to be removed, and the reaction factor for roulette wheel weight are all important factors that have an impact on both quality of solution and computing time (with P_value lower than 0.05). Among the results, it is clear that bothΦ and$\sigma_1$are the parameters that significantly affect the quality of solution. The best parameter setting for further analysis is shown in the last column of Table 4.

Table 4. Heuristic parameters

Paramete r	Description	Values	Selected value
ρ	Defines the number of iterations	300,400,500	400
δ	Define number of iterations without improvements	10,20	10
ξ_{min}	Introduces minimum percentage of request, stops, to be removed at each ALNS iteration	2%,5%,10%	5%
ξ_{max}	Introduces maximum percentage of requests, stops, to be removed at each ALNS iteration	15%,20%,25%,30%,35%,40%,45%	25%
η	Introduces the parameter to control the value of q_{max}	2,3	2
p	Is responsible for randomness in the removal process	2,4,6	4
σ_1	Is the weight adjustment of algorithm in roulette wheel mechanism	40,50,60	50
σ_2	Is the weight adjustment of algorithm in roulette wheel mechanism	20,30,40	20

σ_3	Is the weight adjustment of algorithm in roulette wheel mechanism	1,5,10	5
γ	Is the reaction factor of the weights in roulette wheel mechanism	0.25,0.5, 0.75	0.5
α	Size of the restricted candidate list	1,2,3,4	2

Table 5. Best parameter setting

Parameters	Computing time	Average solution cost
ρ	**p<0.05**	**p<0.05**
δ	**p<0.05**	**p<0.05**
ξ_{min}	**p<0.05**	**p<0.05**
ξ_{max}	**p<0.05**	**p<0.05**
η	**0.162**	0.084
p	**0.082**	0.079
σ_1	**0.093**	**p<0.05**
σ_2	**0.054**	0.809
σ_3	**0.115**	0.320
γ	**p<0.05**	**p<0.05**
α	**0.193**	**p<0.05**

5.3. Heuristic calibration

As mentioned in earlier stage, set of removal and insertion heuristics are considered for our problem. During the metaheuristic operation, a case might happen that some removal and insertion operators cannot improve the solution directly; but they provide the opportunity of escaping from local optima for other operators in next iterations, and as a result, a better quality of solutions is obtained toward the end of the search. More precisely, there may be an operator that delivers the weakest performance, but its presence triggers other operators to easily escape local optima. On the other hand, choosing a large number of removal and insertion operators demands more computing time, more time to explore in the solution space, and results in computational complexity. These evidences suggest that appropriate selection of removal and insertion operators is not straightforward and needs in-depth analysis. This helps in making a balance between computing time and quality of solution. To do so, similar to the section 5.2, full factorial experimentaldesign is conducted with levels shown in Table 6. It is worth mentioning that the other heuristic parameters are fixed at this stage and taken from section 5.2. A graphical output of results is shown in Figures 3 and 4. The analysis of variance (ANOVA) reveals that among the removal and insertion operators,shawremoval, worst removal and random removal heuristics with both basic greedy and regret k-heuristic provide a remarkable influence on the quality of solution. Moreover, the SLR and least used bus removal, basic greedy based on largest demand, and second-best insertion operators slightly improve the solution and display poorerperformance than other considered operators. As a result, the combination of both shawremoval, worst removal and random removal heuristics with both basic greedy and regret k-heuristic as insertion heuristicsis suggested for further analysis (sections 5.3 and 5.4).

Table 6.Removal and insertion heuristics setting

Heuristic	Value	No. of levels

Shaw removal (based on distance)	On –off	2
Shaw removal (based on demand)	On –off	2
Worst removal	On –off	2
Random removal	On –off	2
Least bus removal	On –off	2
SLR removal	On –off	2
Basic greedy	On –off	2
Basic greedy based largest demand	On –off	2
Second best insertion	On –off	2
Regret-k heuristic	On –off	2

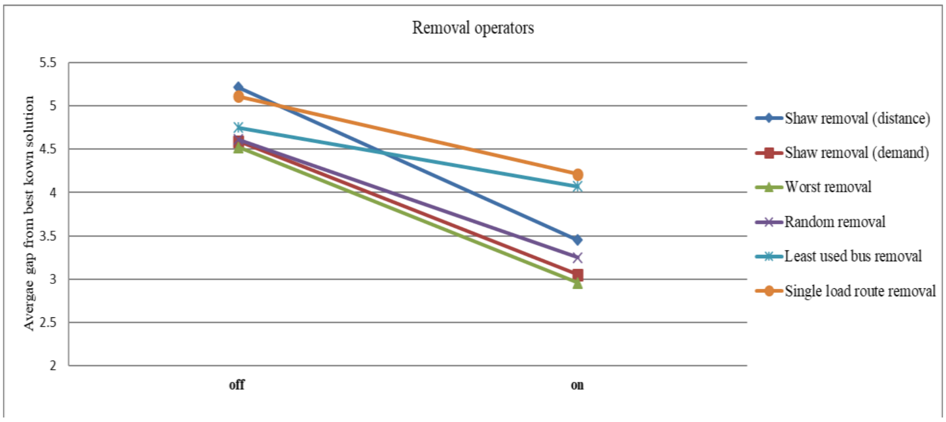


Figure 3. Removal operators

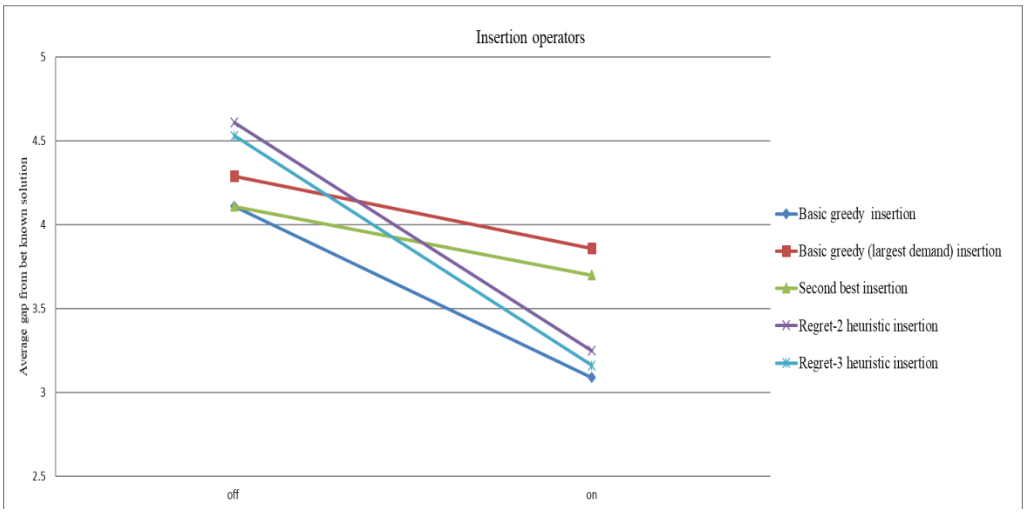


Figure 4. Insertion operators

5.4.Computational experiments

The experimentsare handled in two categories:1)analysis of metaheuristic configuration and 2) analysis of main characteristics.

5.4.1. **Experimentson the configuration of metaheuristic**

In this section experimentsare carried out in order to understand the effect of each pair of removal and insertion operators embedded in ALNS metaheuristic on the quality of solution. In doing so, at the end of each segment, the weight of each pair of removal and insertion is calculated based on its attained score. This weightis related to the quality of solution. In practice, a pair of heuristics with higher weight will be chosen with higher probability and has the ability of providingbetter solutions throughout the search.

Regarding this mechanism, the results suggest that shawremoval with any insertion heuristic producesthe highest weight.That interestingly demonstrates that what is more important is the similarity idea whether based on demand or distance. The random removal is ranked second.Therefore, it can be said that the Shaw removal heuristic orients the intensification stage and the random removal justifies the diversification. These findings further support the idea of using ALNSmetaheuristic. In practice, the ALNS enjoys a set of intensification and diversification heuristics that in case, some heuristics produce weak performance, while others can help to escape local optima properly.

Table 7. Weight values for the pairs of removal and insertion heuristics

Pair of removal and insertion heuristics	Weight	Pair of removal and insertion heuristics	Weight
Shaw removal (distance) and basic greedy	46.37	Random removal and basic greedy	42.27
Shaw removal (distance) with regret-2	53.19	Random removal and regret-2	41.10
Shaw removal (distance) with regret-3	41.18	Random removal and regret-3	36.19
Shaw removal (demand) and basic greedy	41.13	Worst removal and basic greedy	19.45
Shaw removal (demand) with regret-2	44.28	Worst removal and regret-2	23.65
Shaw removal (demand) with regret-3	37.17	Worst removal and regret-3	26.17

5.4.2. Experiments on main characteristics

In this section, we aim to compare the performance of proposed metaheuristic against the solution given by CPLEX solver. The characteristics of the test problem sizes and the results of metaheuristic and exact solutions are summarized in Appendix 1. The exact solution is reported as long as the optimal solution was found within 30 minutes. Since the combination of removal and insertion heuristic cansuggest better performance (results of Section 5.4.1), we only consider it in this section. For the proposed approach (ALNS metaheuristic), each instance is run 10 times and finally, the amount of gap is presented. Two percentage gaps are reported; Percentage gap between the average costs of the solutions calculated after 10 runs and exact solution, so called average gap and the percentage gap between the best solutions calculated after 10 runs, and exact solution, which is called best gap. As expected, while the problem size rises, approximately after instance 20, the exact method cannot find feasible solution within considered time. Thus, the exact method hasthe ability to optimally solve up to instance 20 in reasonable computing time. In comparison with the exact method, ALNS produces solutions with the average percentage gap from exact method lower than 2.5%, and finds optimal solutions in 7 instances. In Figure 5, we compare the main characteristics (total travel time and total number of buses) when using a single and a mixed load strategy. It is worth mentioning that in this paper, the defined objective function is minimizing total travel time, but one of the main effects of mixed load is to reduce the number of buses. Thus, we also consider the behavior of total number of buses in our analysis. Our experiments clearly demonstrate that mixed load effect can significantly reduce the number of buses and total travel time. In case of total travel time, this deviation is more highlighted in large instances. The reason is that as the size of problem increases, there is more tendency to use small number of routes, and as a result more savings in travel time. On the other hand, this graph indicates that small instances have the lowest deviation. The reason is that for small sizes, the number of schools is small and the performance of metaheuristic to take account of mixed load effect is not considerable. Opposite behavior is revealed for the number of routes, as it is clearly seen that for small instances, there is more deviation. Overall reduction in terms of the number of routes and total travel time is 11.36% and 14.56%, respectively, through utilizing mixed load effect. Additionally, in Figure 6 it is observed that average weighted riding time and route length are much smaller than mixed load method. On average, they were 7.8% and 8.43% smaller, respectively.

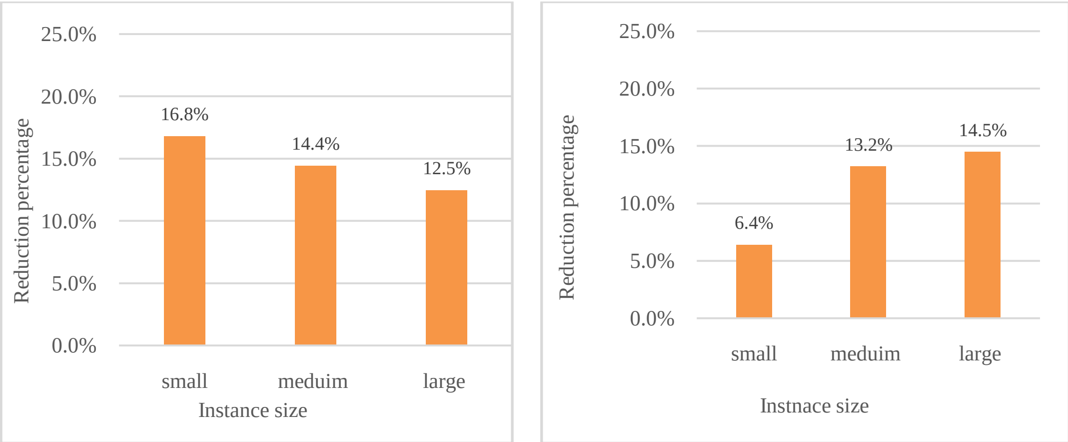


Figure 5.(a) Reduction percentage in the number of routes (left side) and (b) Reduction percentage in total travel time (right side) while considering mixed load effect

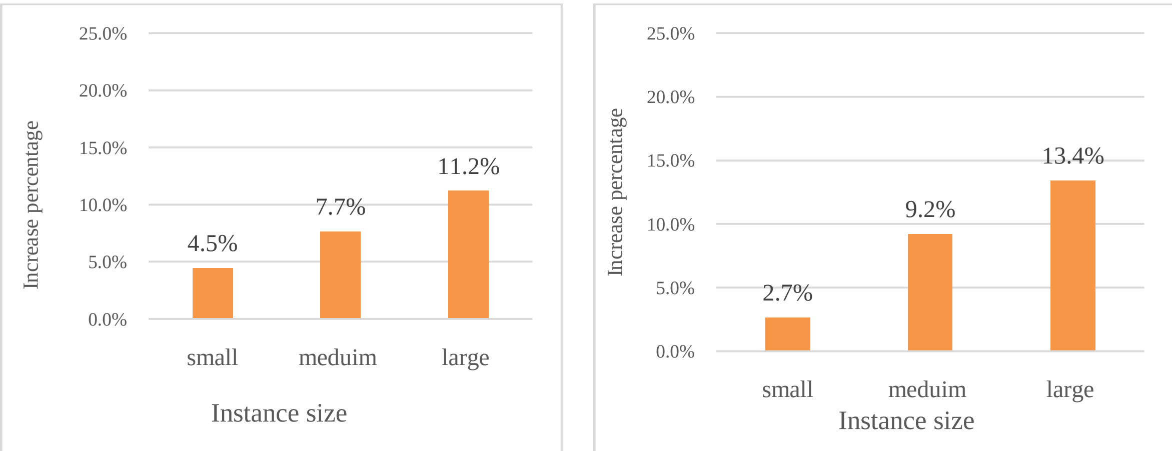


Figure 6. (a) Increase percentage in average riding time (left side) and (b) total route length (right side) while considering mixed load effect

Additionally, we analyze the rate of bus occupation (Figure 7) while considering mixed load effect. It reveals that the rate of bus occupation for small instances has the lowest deviation to the single load. In contrary, there is improvement in bus occupation while size increases.

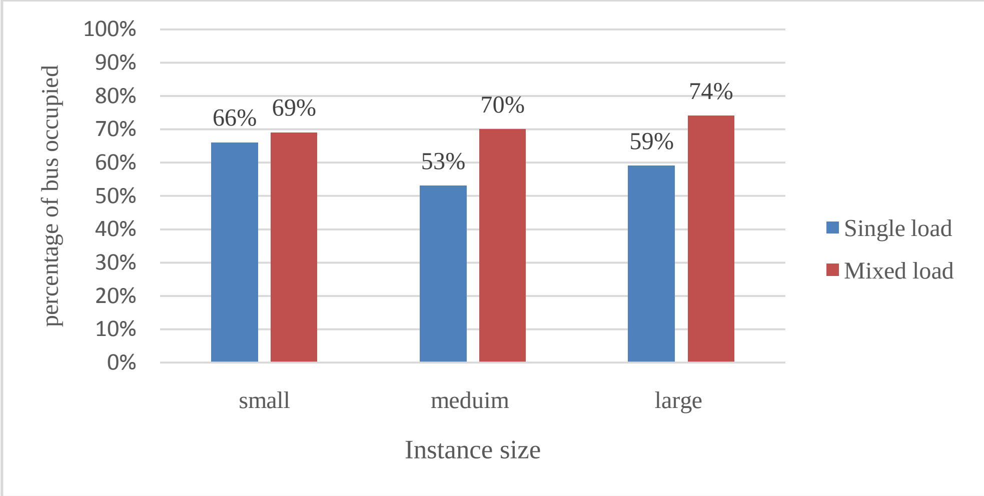


Figure 7. Bus occupation percentage while considering mixed load effect

To conclude, there are several factors affecting whether mixed load effect leads to savings in total travel time or not, and how much it benefits in terms of reduction in total time traveled compared to the case when mixed load is not considered. One determinant is the distance between schools. Specifically, when the schools are close together, there may be much savings since a bus can pick up students of multiple schools from stops and deliver them in a single route from the set of stops to the set of schools. On the other hand, nearness of stops can also affect the merit of mixed load but the distribution of students in each stop is greater importance. For instance, if stops with students of the same school are closer together, the benefit of mixed load diminishes. However, when stops are further apart and there are students from different schools in each stop, the mixed load would provide better opportunities for reduction in total travel time.

5.4 Comparison with best-known solution and previous studies

In order to evaluate the efficiency and effectiveness of the proposed metaheuristics, a comparison is carried out with a previous study in this field. In doing so, different configurations of the proposed metaheuristic (including simple LNS that contains one removal and insertion operator, and full adaptive configuration, ALNS)are compared with best-known solution in the literature. Five simple LNS heuristic configuration areShaw removal(based on demand) with basic greedy, Shaw removal(based distance) with Regret-2, random removal with Regret-2, worst removal with basic greedy and worst removal with Regret-3.For the sake of simplicity, the aforementioned LNS configurations are abbreviatedas LNS-1 to LNS-5, respectively.

Since a new version of school bus routing problem is proposed in this paper, some adaptations need to be considered in our assumptions. In practice, in order to make a fair comparison with the study ofLima et al., (2016).,the following assumptionis set according to the benchmarked study: a unitary routing cost is set to 100$. More importantly, the location of students is considered in stops, so the student's home location is set in the candidate bus stops. The value of routing cost ($)rising from different kinds of metaheuristic (LNS and ALNS) are compared with the best known solution. Table 8 reports the routing cost of the proposed metaheuristics and the algorithm proposed by Fatima’s study.

Among the proposed configurations, ALNS generally results in the best values and less deviation to the study of Lima et al., (2016), such that for four instances, better results are obtained, and for other cases, the deviation is negligible (the average deviation percentage is about 2%).

Table 8. Comparison of different kinds of metaheuristic (LNS and ALNS) with best known solution

Instance	P(student)	H(school)	ILS (best-known solution)	LNS-1	LNS-2	LNS-3	LNS-4	LNS-5	ALNS
1	250.0	6.0	7,024.4	7,687.4	7,656.6	7,895.4	7,326.5	7,649.6	7,171.9
2	250.0	12.0	10,575.0	11,697.0	11,315.2	11,732.6	10,797.0	11,283.5	10,839.3
3	500.0	12.0	19,368.2	21,640.0	20,396.5	21,705.9	19,775.0	21,498.7	19,329.5
4	500.0	25.0	27,066.3	30,520.1	29,231.6	30,612.9	29,502.3	29,889.3	26,714.4
5	1,000.0	25.0	25,622.6	29,167.5	27,491.5	29,256.2	28,774.2	28,653.7	24,725.8
6	1,000.0	50.0	65,139.5	70,350.7	66,308.2	70,564.7	72,891.1	73,021.4	68,982.8

7	2,000.0	50.0	89,398.6	106,309.5	99,232.5	106,632.9	100,394.7	104,381.9	92,536.5
8	2,000.0	100.0	105,215.4	124,501.4	117,347.3	113,632.7	117,841.3	127,415.9	108,161.5
9	250.0	6.0	7,930.6	8,872.9	8,363.0	8,899.9	8,873.6	9,429.5	8,097.2
10	250.0	12.0	12,224.6	14,318.4	13,569.3	12,591.3	13,153.6	13,202.5	12,399.4
11	500.0	12.0	17,681.6	20,646.2	19,459.8	20,709.0	18,830.9	18,406.5	18,450.7
12	500.0	25.0	23,037.7	25,802.3	24,189.6	25,880.8	23,498.5	25,721.6	23,751.9
13	1,000.0	25.0	50,627.1	52,804.0	52,804.0	52,964.7	55,335.4	56,403.6	51,690.2
14	1,000.0	50.0	66,558.9	71,218.0	71,218.0	71,434.7	68,089.8	81,135.3	66,560.4
15	2,000.0	50.0	94,661.1	118,326.4	111,700.1	118,686.3	98,447.5	108,216.5	97,747.0
16	2,000.0	100.0	88,846.8	100,183.9	96,843.0	100,488.7	90,001.8	99,339.6	92,898.2
17	250.0	6.0	10,812.2	12,308.0	11,785.2	12,345.5	11,298.7	11,989.6	11,162.5
18	250.0	12.0	14,645.7	16,858.0	15,231.5	16,909.3	14,953.3	16,205.5	15,004.5
19	500.0	12.0	21,840.4	24,898.0	23,467.3	24,973.8	22,779.5	23,996.0	22,342.7
20	500.0	25.0	24,723.6	29,231.3	26,948.7	29,320.2	26,330.6	25,799.1	24,691.4

Another interesting result worth nothing is that, on average,all considered LNS heuristicshave worse performance than the ILS algorithm in the literature.Thishighlightsthatfor this type of problem with the defined characteristics, relying only on a removal-insertion pair is not trustworthy, and a suitable combination of operators creates an effective outcome. Clearly speaking, random removal heuristic pursues a diversification strategy.Both Shaw and worst removal operators only concentrate on a small portion of solution space, as they can obtain better results in early iterations compared with other removal heuristics. However, while the solution reaches a high-quality level, the probability of being stuck in local optima rises. However, when there is no possibility of operating the above-mentioned heuristics, there is a great likelihood of being trapped in a local optimum. This demonstrates that considering only one of the above set of removal operators individually cannot reduce the risk of being trapped in local optima. More importantly, some operators can improve the solution in the preliminary stage, while others can produce a profitable solution toward the end of the procedure. Thus, clever adoption of diversification and intensification operators could be helpful in the search, where in case one operator performs poorly another one will act effectively. The ALNS takes advantage of this situation and as a result finds better results.

Conclusion

This study aimed to introduce a novel mathematical formulation and solution methodologies for the urban school bus routing problem while considering mixedload effect. The attained results confirm that the proposed framework was effective, because it brings savings in cost than the single load framework. The characteristics of the SBRP in this study includehomogeneous buses, maximum allowable students to each stop, school arrival time, and multiple garages. In primary stage, the generated instance set,was solved using the formulation by CPLEX solver in GAMS. Since the CPLEX solver has the ability of solving 20 instances in a reasonable computing time, the ALNS with a different configuration was proposed to solve all generated instances in a reasonable time.To investigate the algorithm efficiently, four lines of experiments are proposed. First,comparison of the single load and mixed load while investigating different outputs, including total traveled time and the number of buses. Second,analysis of different configurations of metaheuristic (e.g. each pair of removal and insertion operators embedded in ALNS metaheuristic on the quality of solution) is carried out.In the third line, the performance of proposed algorithm was compared with the solution given by GAMS/CPLEX.Finally, make a fair comparisonwith different study is conducted. In the earlier case, it is shown that consideringmixed load effect provides better solutions. With respect to student maximum riding time and total rough length, the results demonstrate that, on average, single load strategy can lead to producing lower values. This suggests that constraints for maximum riding time and total route length are necessary to be applied in further action. In the second analysis, among different combinations, the Shaw removal with Regret -2 gains more weight. In the third line,we compared the solution of the proposed metaheuristic against the solution of GAMS/CPLEX. The attained results show that average percentage gap from GAMS/CPLEX was lower than 2.5%,and in 7 instances optimal solution is found . Finally, in comparison with the best study, the promising result observed whileconsidering the ALNS metaheuristic. The results of other LNS configurations were not very encouraging.

Several promising directions for further researches are available from the current study. Regarding the earlier direction, researchersmay be interested to include additional constraints and features in the proposed model to make it more closer to reality. More precisely taking account of simultaneous morning and afternoon delivery and pickup of students. A second research line is using a data structure to take account of the problem characteristics to reduce the complexity of the problem solving. This data structure can keep information about the neighborhood of the current solution while preserving problem characteristics (mixed load and morning-afternoon consideration).

Reference

[1]. Bodin L. D. and Berman L. (1979). Routing and scheduling of school buses by computer. Transportation Science, 13, 113-129.

[2]. Bögl M., Doerner K.F. and Parragh S.N. (2015). The school bus routing and scheduling problem with transfers. Networks, 65(2), pp.180-203.

[3]. Bowerman R., Hall B. and Calamai P. (1995). A multi-objective optimization approach to urban school bus routing: Formulation and solution method. Transportation Research Part A: Policy and Practice, 29, 107-123.

[4]. Braca J., Bramel J., Posner B. and Simchi-levi D.(1997). A computerized approach to the New York City school bus routing problem. IIE transactions, 29, 693-702.

[5]. Bennett, B.T., Gazis, D.C., (1972). School bus routing by computer. Transportation Research 6, 317–325.

[6]. Campbell, J. F., North, J. W., &Ellegood, W. A. (2015). Modeling mixed load school bus routing. In Quantitative approaches in logistics and supply chain management (pp. 3–30). Springer

[7]. Chapleau L., Ferland J. A. and Rousseau J. M. (1985). Clustering for routing in densely populated areas. European Journal of Operational Research, 20, 48-57.

[8]. Corberán, A., Fernández, E., Laguna, M., Martí, R., (2002). Heuristic solutions to the problem of routing school buses with multiple objectives. Journal of Operational Research Society 53, 427–435.

[9]. Chen, D., Kallsn, H.A., Snider, R.C., (1988). School bus routing and scheduling: an expert system approach. Computers and Industrial Engineering 15, 179–183.

[10]. Chen, X., Kong, Y., Dang, L., Hou, Y., & Ye, X. (2015). Exact and metaheuristic approaches for a bi-objective school bus scheduling problem. PLoS One, 10(7), e0132600

[11]. Desrosiers J., Commerciales É. D. H. É. Transports, U. D. M. C. D. R. S. L. and Opérationnelle, U. D. M. D. D. I. E. D. R. (1986a). TRANSCOL: A multi-period school bus routing and scheduling system, Montréal: Université de Montréal, Centre de recherche sur les tranports.

[12]. Desrosiers J., Transports U. D. M. C. D. R. S. L. and Opérationnelle, U. D. M. D. D. I. E. D. R. (1981). An Overview of School Busing System, Montréal: Université de Montréal, Centre de recherche sur les transports.

[13]. Dulac G., Ferland J. A. and Forgues P. A.(1980). School bus routes generator in urban surroundings. Computers & Operations Research, 7, 199-213.

[14]. 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.

[15]. Fügenschuh, A. (2009). Solving a school bus scheduling problem with integer programming. European Journal of Operational Research, 193(3), 867–884.

[16]. Ellegood W. A., Campbell, J. F. and North, J. (2015). Continuous approximation models for mixed load school bus routing. Transportation Research Part B: Methodological, 77, 182-198.

[17]. Goeke, D. & Schneider, M. (2015), 'Routing a mixed fleet of electric and conventional vehicles', European Journal of Operational Research 245 (1), 81 – 99. 13, 41, 65, 66

[18]. Graham, D., Nuttle, H.L.W., (1986). A comparison of heuristics for a school bus scheduling problem. Transportation Research B 20 (2), 175–182

[19]. Hargroves, B.T., Demetsky, M.J., (1981). A computer assisted school bus routing strategy: a case study. Socio-Economic Planning Sciences 15 (6), 341–345.

[20]. Hernan Caceresa, RajanBattaa, Qing Hea, (2015),Special Need Students School Bus Routing: Consideration for Mixed Load and Heterogeneous Fleet

[21]. Hiermann, G., Puchinger, J., Ropke, S. &Hartl, R. F. (2016), 'The electric fleet size and mix vehicle routing problem with time windows and recharging stations', European Journal of Operational Research 252 (3), 995 –1018.

[22]. Iman Dayarian, Teodor Gabriel, Crainic Michel, Gendreau Walter Re. (2013).An Adaptive Large Neighborhood Search Heuristic for a Multi-Period Vehicle Routing Problem.

[23]. Kinable J., Spieksma F. and Vanden berghe G.(2014). School bus routing—a column generation approach. International Transactions in Operational Research, 21, 453-478.

[24]. Kang, M., Kim, S.-K., Felan, J. T., Choi, H. R., & 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

[25]. Lourenço H. R., Martin O. C. and Stützle T.(2010). Iterated local search: Framework and applications. Hand book of Metaheuristics. Springer.

[26]. de Souza Lima, F. M., Pereira, D. S., da Conceição, S. V., & de Camargo, R. S. (2017). A multi-objective capacitated rural school bus routing problem with heterogeneous fleet and mixed loads. 4OR, 15(4), 359-386.

[27]. Lima, F. M. S., Pereira, D. S., Conceição, S. V., & Nunes, N. T. R. (2016). A mixed load capacitated rural school bus routing problem with heterogeneous fleet: Algorithms for the Brazilian context. Expert Systems With Applications, 56, 320-334.

[28]. Lima, F. M. d. S., Pereira, D. S., da Conceio, S. V., de Camargo, R. S., (2017). A multi-objective capacitated683 rural school bus routing problem with heterogeneous fleet and mixed loads. 4or, 1–28.684

[29]. Li, L., Fu, Z., (2002). The school bus routing problem: a case study. Journal of the Operational Research Society 53, 552–558.3:75-85.

[30]. Maciel Silva, C., Sarubbi, J., Fonseca Silva, D., Porto, M., & Nunes, N. (2015). A mixed load solution for the rural school bus routing problem. In Proceedings of the IEEE 18th international conference on Intelligent transportation systems (ITSC), 2015 (pp. 1940–1945).
[31]. Martinez, L. M. and Viegas, J. M. (2011). Design and Deployment of an Innovative School Bus Service in Lisbon. Procedia Social and Behavioral Sciences, pages 120–130.
[32]. Miranda, D. M., de Camargo, R. S., Conceição, S. V., Porto, M. F., & Nunes, N. T. (2018). A multi-loading school bus routing problem. Expert Systems with Applications, 101, 228-242.
[33]. Mokhtari, N. A., &Ghezavati, V. (2018). Integration of efficient multi-objective ant-colony and a heuristic method to solve a novel multi-objective mixed load school bus routing model. Applied Soft Computing, 68, 92-109.
[34]. Newton RM, Thomas WH (1969) Design of school bus routes by computer Socio-Economic Planning Sciences
[35]. Pacheco, J., Martí, R., (2006). Tabu search for a multi-objective routing problem. Journal of the Operational Research Society 57, 29–37.
[36]. 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, 397-411.
[37]. Park J. and Kim B. I.(2010). The school bus routing problem: A review. European Journal of operational research, 202, 311-319.
[38]. Park J., Tae H. and Kim B. I. (2012). A post-improvement procedure for the mixed load school bus routing problem. European Journal of Operational Research, 217, 204-213.
[39]. Riera-ledesma J., Salazar-gonzález J. J. (2012). A column generation approach for a school bus routing problem with resource constraints. Computers & Operations Research, 40, 566-583.
[40]. Rodríguez-Parra, G. R., Guerrero, W. J., & Sarmiento-Lepesqueur, A. (2017). Cooperation strategies featuring optimization in the school transportation system in Bogota. Dyna, 84(202), 164-174.
[41]. Ropke, S., &Pisinger, D. (2006). An adaptive large neighborhood search heuristic for the pickup and delivery problem with time windows. Transportation science, 40(4), 455-472.
[42]. Silva, C. M., Sarubbi, J. F., Silva, D. F., Porto, M. F., & Nunes, N. T. (2015, September). A mixed load solution for the rural school bus routing problem. In Intelligent Transportation Systems (ITSC), 2015 IEEE 18th International Conference on (pp. 1940-1945). IEEE.
[43]. de Siqueira, V. S., da Silva, E. N., da Silva, R. V. S., & 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.
[44]. Schittekat, P., Sevaux, M., & Sorensen, K. (2006). A mathematical formulation for a school bus routing problem. In Service Systems and Service Management, 2006 International Conference on (Vol. 2, pp. 1552-1557). IEEE.
[45]. Schittekat P., Kinable J., Sörensen K., Sevaux M., Spieksma F. and Springael J.(2013). A metaheuristic for the school bus routing problem with bus stop selection. European Journal of Operational Research, 229, 518-528.
[46]. Spada M., Bierlaire M. and Liebling T.M., (2005). Decision-aiding methodology for the school bus routing and scheduling problem. Transportation Science, 39(4), pp.477-490.
[47]. Simchi-Levi, D., Chen, X., Bramel, J., (2005). A case study: school bus routing. In: Simchi-Levi, D., Chen, X., Bramel, J. (Eds.), The Logic of Logistics: Theory, Algorithms, and Applications for Logistics and Supply Chain Management. Springer, New York, pp. 319–335
[48]. Thangiah, S.R., Nygard, K.E., 1992. School bus routing using genetic algorithms. In: Proceedings of the SPIE Conference on Applications of Artificial Intelligence X: Knowledge-Based Systems, Orlando, Florida. doi:1-.1117/12.
[49]. Savas, E., 1978. On equity in providing public services. Management Science 24, 800–808
[50]. Swersey, A. J., & Ballard, W. (1984). Scheduling school buses. Management Science, 30(7), 844-853.
[51]. Yao, B., Cao, Q., Wang, Z., Hu, P., Zhang, M., Yu, B., (2016). A two-stage heuristic algorithm for the school740 bus routing problem with mixed load plan. Transportation Letters 8(4),205-219

Appendix 1: Results obtained by solving the instances contained in Sets S, M, and L

In this section, detailed results of GAMS/CPLEX solver andthe ALNS metaheuristic are shown. Each table describes the details of the problem instances and also the results of GAMS/CPLEX and metaheuristic methods

Instance #	Problem characteristics					Results			
	No. of schools	No. of stops	No. of students	Capacity	Walking distance	GAMS solution	Best sol_meta	Avg time (ms)	% Best gap
1	1	5	25	25	5	135.43	135.43	35	0.00
2	1	5	25	50	5	242.62	247.56	97	2.04
3	1	5	25	25	10	151.34	155.19	71	2.54
4	1	5	25	50	10	224.68	224.68	46	0.00
5	1	5	25	25	15	219.45	223.12	71	1.67
6	1	5	25	50	15	177.50	177.50	50	0.00
7	1	5	25	25	20	103.45	105.19	62	1.68
8	1	5	25	50	20	155.47	155.47	69	0.00
9	1	5	25	25	25	136.35	138.90	65	1.87
10	1	5	25	50	25	65.98	68.34	113	3.58
11	2	10	50	25	5	161.45	161.45	186	0.00
12	2	10	50	50	5	267.21	272.12	214	1.84
13	2	10	50	25	10	226.06	228.90	242	1.26
14	2	10	50	50	10	326.34	330.23	252	1.19
15	2	10	50	25	15	244.49	244.50	260	0.00
16	2	10	50	50	15	234.68	243.12	244	3.60
17	2	10	50	25	20	247.42	250.21	220	1.13

18	2	10	50	50	20	171.99	178.23	212	3.63
19	2	10	50	25	25	167.22	169.12	174	1.14
20	2	10	50	50	25	187.82	190.45	272	1.40
21	3	15	75	25	5		794.80	383	
22	3	15	75	50	5		1012.46	345	
23	3	15	75	25	10		958.37	390	
24	3	15	75	50	10		1306.17	442	
25	3	15	75	25	15		1243.76	347	
26	3	15	75	50	15		1105.14	440	
27	3	15	75	25	20		863.20	428	
28	3	15	75	50	20		894.34	395	
29	3	15	75	25	25		835.51	314	
30	3	15	75	50	25		321.89	464	
31	4	20	100	25	5		1760.12	891	
32	4	20	100	50	5		2872.21	852	
33	4	20	100	25	10		1919.67	967	
34	4	20	100	50	10		1807.95	987	
35	4	20	100	25	15		2456.78	659	
36	4	20	100	50	15		2035.56	869	
37	4	20	100	25	20		1169.80	1,141	
38	4	20	100	50	20		1351.13	1,082	
39	4	20	100	25	25		1431.12	659	
40	4	20	100	50	25		912.34	904	
41	5	25	125	25	5		1897.23	3,016	
42	5	25	125	50	5		2182.69	3,000	
43	5	25	125	25	10		1987.59	2,667	
44	5	25	125	50	10		2878.45	2,723	
45	5	25	125	25	15		3114.09	2,104	
46	5	25	125	50	15		2304.29	2,396	
47	5	25	125	25	20		1455.34	3,903	
48	5	25	125	50	20		1678.32	3,345	
49	5	25	125	25	25		1790.32	1,785	
50	5	25	125	50	25		1903.34	2,585	
51	6	30	150	25	5		1900.34	11,912	
52	6	30	150	50	5		2234.19	14,565	
53	6	30	150	25	10		2543.19	8,142	
54	6	30	150	50	10		2664.45	9,970	
55	6	30	150	25	15		2732.21	7,805	
56	6	30	150	50	15		2021.45	9,801	
57	6	30	150	25	20		1450.39	15,648	
58	6	30	150	50	20		2097.34	13,211	
59	6	30	150	25	25		1891.13	7,219	
60	6	30	150	50	25		2793.12	9,466	
61	7	35	175	25	5		2121.34	43,367	
62	7	35	175	50	5		3043.12	63,767	
63	7	35	175	25	10		2570.57	38,718	
64	7	35	175	50	10		4096.50	42,285	
65	7	35	175	25	15		3362.22	30,057	
66	7	35	175	50	15		3021.54	34,506	
67	7	35	175	25	20		3098.45	69,988	
68	7	35	175	50	20		2560.87	62,995	
69	7	35	175	25	25		2272.12	32,626	
70	7	35	175	50	25		2341.34	45,017	
71	8	40	200	25	5		3023.32	208,465	
72	8	40	200	50	5		3957.43	330,594	
73	8	40	200	25	10		3652.32	164,202	
74	8	40	200	50	10		4432.12	178,967	
75	8	40	200	25	15		5034.21	127,729	
76	8	40	200	50	15		4321.14	146,634	
77	8	40	200	25	20		4567.23	347,841	
78	8	40	200	50	20		3457.21	309,300	
79	8	40	200	25	25		3094.32	155,156	
80	8	40	200	50	25		3101.15	240,337	
81	9	45	225	25	5		3987.21	763,417	
82	9	45	225	50	5		4976.54	1,056,142	
83	9	45	225	25	10		4674.23	601,323	
84	9	45	225	50	10		4867.12	692,230	
85	9	45	225	25	15		4523.13	523,622	
86	9	45	225	50	15		3987.12	614,957	

87	9	45	225	25	20			4231.23	1,418,512	
88	9	45	225	50	20			4578.20	1,328,981	
89	9	45	225	25	25			4309.21	673,320	
90	9	45	225	50	25			3211.98	1,039,881	
91	10	50	250	25	5			4219.20	2,396,318	
92	10	50	250	50	5			5396.95	4,116,779	
93	10	50	250	25	10			4748.18	1,949,401	
94	10	50	250	50	10			5763.43	2,938,713	
95	10	50	250	25	15			5626.40	1,821,001	
96	10	50	250	50	15			4473.65	2,114,905	
97	10	50	250	25	20			4514.68	4,276,218	
98	10	50	250	50	20			4352.37	4,091,803	
99	10	50	250	25	25			4228.80	2,408,016	
100	10	50	250	50	25			4589.78	3,968,672	