A Local Search-based Metaheuristic Algorithm Framework for the School Bus Routing Problem

Yane Hou, Bingbing Liu, Lanxue Dang, Wenwen He and Wenbo Gu

***Abstract*—School bus service is divided into many kinds of operation modes, such as single school, single load multiple schools and mixed load multiple schools and so on. When it comes to plan bus routes satisfying various constraints and different objectives, there are many applications of school bus routing problem (SBRP). It is generally recognized that middle and large sacle SBRP applications are resolved by heuristic algorithms, because SBRP is a NP-hard problem. This paper proposes a local search-based metaheuristic algorithm framework for SBRP on the basis of analysis of SBRP problem models. The framework, which is implemented by C#, is made up of basic data structures, operation functions, neighborhood operators, initial solution construction algorithm components and heuristic strategies, etc. The neighborhood-centered metaheuristic algorithms, such as simulated annealing (SA), iterated local search (ILS), variable neighborhood search (VNS), can be designed based on the framework to solve a certain problem type of SBRP. Using this framework, we can solve three operation modes SBRP with homogeneous or heterogeneous bus fleet. Moreover, the proposed metaheuristic framework also provides the development of metaheuristic for the capacitated vehicle routing problem(CVRP) by reducing or modifying the constraints of SBRP. The experimental results of a set of instances show that the metaheuristic algorithm built based on the proposed framework, can be quickly realized and applied to different SBRP applications. Meanwhile, the designed metaheuristic framework is also very effective and extensible.**

***Index Terms*—algorithm framework, school bus routing problem, local search metaheuristic, neighborhood-centered, general-purpose solver.**

I. Introduction

WITH the development of compulsory education in China, providing school bus service for primary and secondary school students is a new requirement for local governments and schools. Thus, the Chinese government has formulated relevant laws and regulations, and carried out pilot work on school bus service in some counties and districts. Planning school bus routes is the important part of the school bus operation management and it is also a very challenging job, which needs consider many factors in the actual school bus routes planning scenarios. It is not only necessary to meet the constraints of school bus capacity, the maximum travel time, students and school time windows, but also to decrease the number of school buses and operating costs as much as possible.

Manuscript received July 7, 2021;revised November 12, 2021. This research has been supported by National Natural Science Foundation of China (Grant No.41801310).

Yane Hou is an associate professor of College of Computer and Information Engineering, Henan University, Kaifeng, Henan, China (e-mail: houyane@henu.edu.cn)

Bingbing Liu is a graduate student of College of Computer and Information Engineering, Henan University, Kaifeng, Henan, China (e-mail: liubingbing628@qq.com

Lanxue Dang is an associate professor of Henan Key Laboratory of Big Data Analysis and Processing, Henan University, Kaifeng, Henan, China (corresponding author, e-mail: danglx@foxmail.com)

Wenwen He is a graduate student of College of Computer and Information Engineering, Henan University, Kaifeng, Henan, China (e-mail: ww82323@henu.edu.cn)

Wenbo Gu is a graduate student of College of Computer and Information Engineering, Henan University, Kaifeng, Henan, China (e-mail: henuguwenbo@qq.com)

It is quit inappropriate to arrange the routes of the school bus manually, on account of the complexity of bus routing planning. School bus routing problem (SBRP) was first introduced by Newton and Thomas [1] in 1969, and the school routes were designed by the computers. Later, many scholars have continuous research on SBRP, mainly focusing on the mathematical model exploration and optimization algorithm design. SBRP is made up of five sub-problems, such as data preparation, bus stop selection, bus route generation, bus route scheduling, and school bell time adjustment [2]. Most SBRP literatures address one or more sub-problems, and generation of bus routes is still the focus of SBRP research. The recent reviews of SBRP are described in [2],[3].

As a special variant of vehicle routing problem (VRP), SBRP is also a NP-hard combination optimization problem [2],[3]. According to the number of schools served by school bus, SBRP can be divided into single-school SBRP and multi-school SBRP. The multi-school SBRP also include single load SBRP and mixed load SBRP, and they have the different order of school bus visiting stops. The single-school SBRP is usually considered as the variant of capacitated vehicle routing problem (CVRP) or vehicle routing problem with time window (VRPTW) [2],[3],[4],[5]. While for the multi-school SBRP, it could be modeled as a continuous approximation model [6], or regarded as a kind of pickup and delivery vehicle routing problem with time window (PDPTW) [7],[8]. When considering the heterogeneous bus fleets with different capacity and cost, it will produce more complex variants of SBRP.

Like VRP, the algorithms solving for SBRP also include exact algorithms, heuristic algorithms, and metaheuristic algorithms [2],[3]. The design of SBRP algorithms usually learn from the algorithms that are usually applied in VRP. Although SBRP is similar to the VRP variants, the algorithms used in classical VRP still need to be modified and extended to solve the SBRP. A lot of metaheuristics have been put forward for VRP in recent years, such as genetic algorithm (GA), iterated local Search (ILS), variable neighborhood search (VNS), tabu search(TS) and so on. They have been successfully applied to solve VRP. Therefore, it is necessary to take advantage of them to solve the SBRP.

With the continuous development of algorithm technology, some unified algorithms [9],[10],[11] and general-purpose algorithm framework [12],[13] for VRP have sprung up exuberantly. The unified algorithms are different heuristics for VRP variants, which have diverse combinations of sets of attributes [9],[10],[11]. While for the general-purpose algorithm framework, it is usually a library or a heuristic framework. Groër [14] provided a library of local search, including some constructive heuristic algorithms and seven local search operators. On the basis of the library, the algorithm for VRP with capacity constraint has been implemented, and the attribute of stops time window has been reserved to extend the solved problem type. However, it is still difficult to realize the PDPTW based on this library, which requires a lot of modifications and extensions from bottom to up. Vidal [12] addressed the multi-attribute vehicle routing problems, and then proposed a component-based heuristic framework. The framework have some problem-independent components and they are can be self-adapted according to the attributes of the problem. Vogel [13] implemented a metaheuristic framework for VRP, and gave an algorithm for solving various variants of VRP based on this framework. Unlike the library designed in [14], the implementation of this framework starts from the complex pickup and delivery vehicle routing problem, so it is easier to support other simple VRP through transformation.

These successful experience of developing a unified algorithm framework for VRPs has promoted the researchers to design the algorithm framework for SBRP. The development of SBRP algorithm framework not only can reuse existing data structures and algorithm components to reduce the difficulty of algorithm design, but also can enhance the flexibility and extensibility of algorithm design.

This paper aims to propose a local search-based metaheuristic algorithm framework for SBRP. The framework provides basic data structures, operation functions, neighborhood operators, initial solution construction algorithm components, and heuristic strategies from bottom to top. It enables the algorithm designer to construct the algorithm for SBRP in different application scenarios based on the framework. The solving problems of SBRP cover three bus service operation modes and two types of bus fleets including homogeneous and heterogeneous fleets.

The remainder of this paper contains a problem description and general methodology of SBRP in Section II. Section III describes the design of neighborhood-centered metaheuristic algorithm framework. Section IV give the process of application development based on this framework and the experiment results. Finally, the remarks of this paper are offered in Section V.

II. Problem Description and General Methodology

A. School Bus Routing Problem

SBRP tries to make an efficient schedule for school buses to pick up students from the student stops and then delivery them to their school, while meeting all kinds of constraints. The common constraints of SBRP are bus capacity, maximum ridding time of students (MRT), time window of stops or schools, and other constraints. The school buses can pass through student stops and school stops in order, and then send students to school or return from school. So, the solution of SBRP is a number of school bus routes, and each route is a sequence of stations including student bus stations and school stations.

According to the number of serviced school and their operation modes, there are three modes of school bus route are shown in Fig 1. Single-school SBRP (Fig 1 (a)) only provides school bus service for one school, there is only one school node on the route. Multi-school SBRP is classified into singe load SBRP and mixed load SBRP. For the former (Fig 1 (b)), the buses serve different schools in the order of school opening time. In mixed load SBRP (Fig 1 (c)), it allows school buses to serve multiple schools at the same time, that is to say, there may be exist students belonged to different schools staying simultaneously on the same bus. The students will be delivered to their respective schools according to the school opening time.

As shown in Fig 1, these three operation modes are not completely independent. Form the view point of multi-school SBRP, it can be simplified to single-school SBRP when there is only one school station on the route. We can consider single-school SBRP as a special instance of multi-school SBRP. In additional, mixed load SBRP can be converted to singe load SBRP, when without allowing the students from different schools staying simultaneously on the bus. Therefore, it is possible to solve the application of SBRP in other modes with the help of mixed load multiple schools mode.

The model of mixed load SBRP has been defined as m-1 PDPTW problem in our former study [8]. Because students from multiple stops may go to the same school, there are multiple m-1 point-pair relationships on a route. Adding limitation to the model without allowing mixed load is the model of single load multi-school SBRP. If there is only one school station, the model of mixed load SBRP can be simplified as single-school SBRP. When they are extended to solve the heterogeneous SBRP, the routes of SBRP should be added some fleet attributes, such as the capacity of fleet, total cost and bus fleet type. The solvers for these problems can also be transformed by modifying attributes or constraints. The transformation and relationship of SBRP discussed above are shown in Fig 2.

B. General Methodology of SBRP

As a NP-hard combination optimization problem, SBRP is difficult to be solved. Among of the SBRP solving methods, exact algorithms (eg.column generation, branch and pricing) can solve small-scale SBRP problem to obtain the optimal solution[15],[16]. However, exact algorithms are usually very time-consuming.

Heuristic algorithms can usually find approximate optimal solutions for large-scale SBRP problems [17],[18],[19],[20]. In general, the traditional heuristic algorithms consist of constructive heuristic and improved heuristic. The constructive heuristic algorithms have a simple solution construction strategy, and they are usually implemented to generate initial solutions. The savings method, insertion method, location based heuristic (LBH)[18] and random location based heuristic(RLBH)[19] all belong to the kind of constructive algorithm. Improved heuristic algorithms use inter-route or intra-route neighborhood operators to improve the solution, such as 2-opt and or-opt. The traditional heuristic algorithms are easy to be implemented, but they are also easy to trap in local optima.

To overcome the shortcoming of traditional heuristic algorithms, metaheuristic algorithms with more intelligent strate-

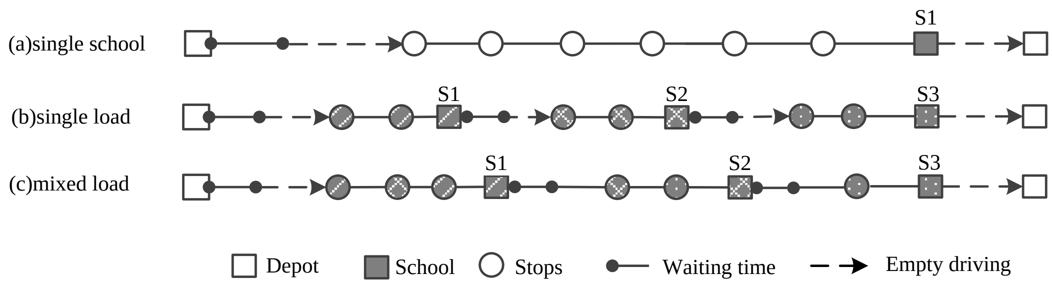

Fig. 1. illustration of three types of SBRP routes

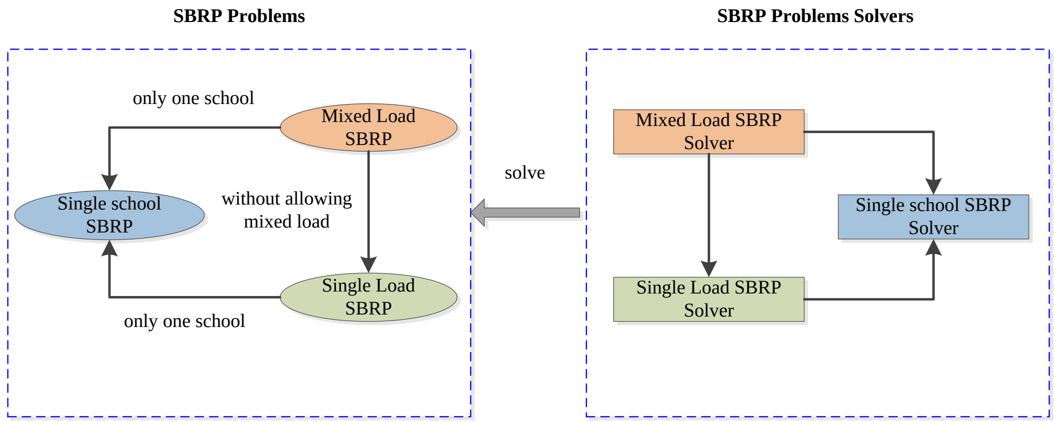

Fig. 2. Relationship between three SBRP problems and their solvers

gies are gradually developed and widely applied in solving SBRP [4],[5],[8],[21],[27]. Metaheuristic algorithm has a certain higher level algorithmic strategy that is capable of escaping from local optima. Single-solution based metaheuristic approaches improve a single solution by some local search-based neighborhood operators, such as TS, ILS, VNS and so on. The most commonly implemented of them to SBRP are TS [21],[22] and ILS [5],[23],[24]. Population-based evolutionary approaches generate a population of solutions, which are iteratively improved to obtain a high-quality solution. Genetic Algorithm is widely used for SBRP [25],[26],[27]. In additional, other population-based approaches have also been implemented to solve SBRP, such as ant colony optimization (ACO)[4],[28] and scatter search[29]. The population-based methods (eg.GA and ACO) and local search-based methods (eg. TS and VNS) are successfully mixed to solve SBRP [4].

Single-Solution metaheuristics and population-based metaheuristics are both provide a unified view of the common concepts for this kind of metaheuristics. Single-solution metaheuristics iteratively apply the procedure of generation and replacement from the current single solution. Population-based metaheuristics share the common concepts, which can be viewed as an iteratively improvement procedure in a population solutions. The common characteristics of these both kinds of metaheuristics make them possible to be as a general-purpose approaches, and there are also some successful applications in VRP [9],[10],[11],[12],[13].

From above all the analysis, we tend to develop a metaheuristic framework for SBRP. The framework is based on single-solution metaheuristics, because it is easy to be implemented and only one single solution is improved by several local search operators. Moreover, this kind of metaheuristic has fewer parameters and without requiring complex parameter tunning.

III. Metaheuristic Algorithm Framework

A. *General Description of Framework*

Framework is a kind of micro architecture, which provides incomplete template for software system in specific domain. It can be a subsystem that will be extended or (and) reused. According to the standards of software engineering, framework design should follow the principles such as availability, reusability, flexibility and extensibility [13].

In this paper, an open and extensible algorithm design framework (as shown in Fig 3) is designed for SBRP, which mainly supports the metaheuristic algorithm of trajectory based local search methods. The framework is programmed in C#, which is known as an orient-object programming language. On the basis of the basic data structure, it provides basic functions and related operations, some initial solution construction algorithm components, neighborhood operators and heuristic strategies. At the top level, some local search block components are provided with a combination of multiple neighborhood operators. Based on the framework, a neighborhood-centered metaheuristic can be designed or combined to solve a certain SBRP problem as required.

B. *Basic Data Structure*

The main data structures defined in the framework include solution, route, stops, school node and fleet type, etc. For a

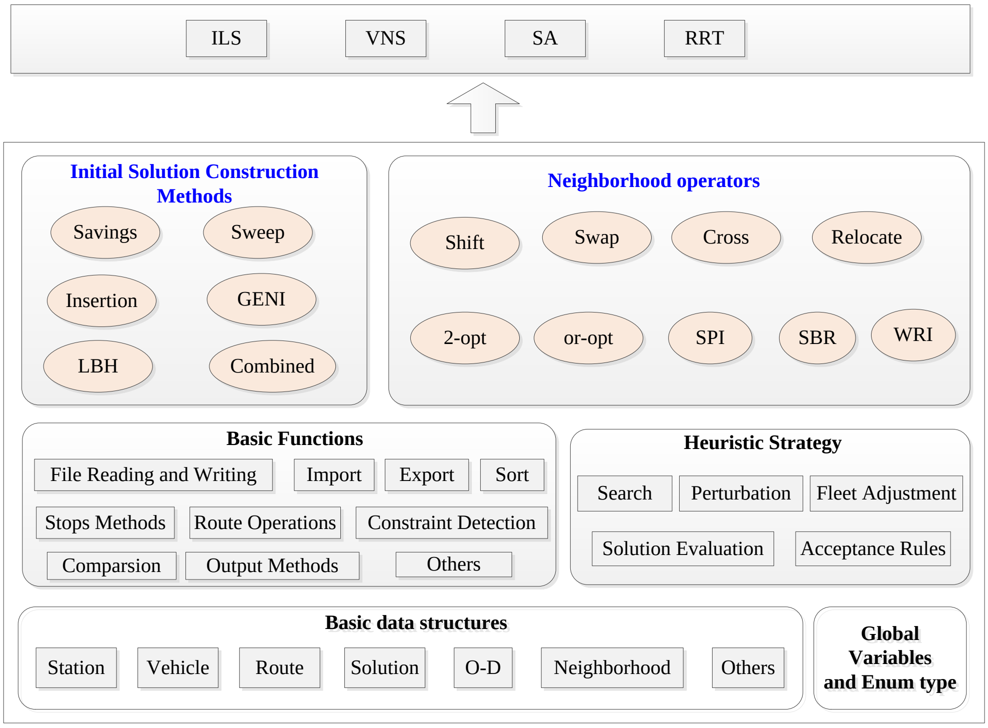

Fig. 3. Metaheuristic framework for SBRP

solution of SBRP, it includes many routes and it also has some attributes, such as total cost, total number of routes, total travel length, list of routes and other related parameters. While for each route of a solution, there are depot, student stops and school nodes on it, and the route also has a list of stops and some attributes, including total student load, the length of route, the travel time of route, and the information about fleet type. Moreover, each route has a detection list, which is made up of school stops on the route, to realize constraints detection of the route. From the point of view, we can find that the relationship between solution and route is many-to-one as well as route and stops. So, we use object-oriented technology to design several classes to represent the basic data structure and the relationship between them.

In addition to the data structures mentioned above, there are O-D matrix, time matrix, neighborhood search space and other data structures used to store data or assist neighborhood search. We also define data structures about the problem types of SBRP and VRP solved by the framework to keep its expansibility.

C. Basic Functions

In the framework, there are some basic functions directly based on the data structure, such as files reading and writing functions, import and export of solutions, stops operation, constraint detection, sorting methods, comparison methods and so on. Every basic functions are defined in the different class files in accordance to the function of them.

File reading and writing class is mainly responsible for importing the data of benchmark instances, and outputting the information about solution in the process of problem solving. The data, which needs to read before solving problem, includes road network data, stops (such as student stops, school stops and depot) data, vehicle data, and other basic data. Some functions are also provided to import the solution meeting the file format requirements. Furthermore, the output of auxiliary information in the running process, such as intermediate process, final results, and some statistical results, are also be recorded by the operating functions into the files.

The stops operation functions are mainly the creation, insertion, deletion and movement of the stops. The creation of stops is to create an instance and assign the values to its properties, including unique identification, coordinate values, stops demand, time window, designated school and other property of the station. In the process of route construction and optimization, the stops on the route will be inserted or deleted. These manipulating functions of stops must be executed in the circumstances.

Constraint detection is one of the most frequently used operations in the process of solving SBRP. Once a stop on the route has been moved, inserted, or removed, we must verify whether the route is valid and does not violate any constraints. Because the framework supports multiple problem types of SBRP, multiple constraint detection functions should be provided with different execution plans due to different constraints. The constraint detection mechanism based on route segments is designed for three operation modes of SBRP. In additional, there are also some detection functions about importing solutions.

Sorting and comparison functions classes are inherited from the standard generic classes in C#, which are applied to compare and sort the instances in a certain order. For example, when the routes need to be sorted by the number of student stops in ascending order, these functions will be called.

D. Neighborhood Operators

For single-solution metaheuristic algorithms, several neighborhood structures are used to find the better neighborhood solution in the stage of local search. These neighborhood operators are applied to explore solution space to find the local optima solution in the local search phase. Therefore, in our framework, some neighborhood operators for SBRP are redesigned and modified from the neighborhood operators that are originally designed for VRP. These neighborhood operators are divided into two classes of neighborhood structures, one is based on point or edge, the other is based on the pair of points. They are used for single-school SBPR and multi-school SBRP respectively.

The neighborhood operators based on point or edge include inter-route operators and intra-route operators, which usually consist of shift, swapping and cross. In every neighborhood structure, the moved nodes or edges do not include the depot and school node, and the operation of shift or swapping can not violate problem constraints. In the next moment, we will describe the operations of them respectively.

The inter-route neighborhood structures are described briefly as following and their operation diagrams are shown in Fig 4.

(1)Shift(1,0). A student station is moved from a route to another different route in the solution.

(2)Shift(2,0). Two adjacent student stops, which are regarded as an edge with two student stops, are moved to another route.

(3)Swap(1,1). A pair of student stops are swapped between two routes. The procedure will choose a random student stops and then try to swap it with another student node in different route.

(4)Swap(1,2). A student node is exchanged with two continuous student stops on another route.

(5)Swap(2,2). Two continuous student stops on the two different routes are exchanged to obtain two new routes.

(6)Cross. This operator deletes one edge from each of the two routes, and then adds two edges to realize the cross of route segments.

(7)Cross Exchange. The operator is the extension of Cross operator defined in above. Different from Cross operator, it deletes two non-adjacent edges from each of the two different routes and then add two edges for each route to obtain the two new routes.

The intra-route neighborhood operators move student stops or edges on the same route. It includes the several neighborhood operators common used in VRP, and the illustration operation of them is shown in Fig 5.

(1)Relocate. Relocate changes the position of one student station in the same route. Relocate operator can be considered as a special case of Shift(1,0), when the move occurs in the same route.

(2)Exchange. Two different student stops on the same route are exchanged.

(3)Three point move. It likes Swap(1,2) operator, but it occurs on the same route.

(4)2-opt. 2-opt is a simple and effective improvement procedure. 2-opt deletes two non-adjacent edges and then the student stops between these edges are all reversed.

(5)Or-opt. This move aims at shifting a sequence of consecutive student stops into another position on the same route. The number of student stops is randomly generated between two and four. After determining the length of consecutive student stops, the neighborhood operator attempts to shift these student stops to another position.

For multi-school SBRP, especially for mixed load SBRP, the fore-mentioned neighborhood operators are not used directly. Thus, three point pair neighborhood operators are provided in the framework, which are single paired insertion (SPI), swapping pairs between routes (SBR), and within route insertion (WRI), which were designed and applied successfully to solve PDPTW [7]. The basic description of them are in the following and operation illustrations are shown in Fig 6.

(1)SPI. SPI is an inter-route neighborhood operator, which moves a pair of student stops and its designated school stops to another route. When the student node is removed from original route and inserted to target route, it is necessary to determine whether delete its school node from the original route or insert its school node into the target route. It makes sure to keep the original route and target route are both valid. SPI is mainly used for reducing the number of routes.

(2)SBR. SBR is a kind of swap neighborhood operator, which exchanges two pair of students stops and its destined school nodes between two routes. Like SPI, SBR also needs to check and determine the school nodes to remove from original route or insert into the target route. SBR cannot reduce the number of routes, but it can change the permutation of nodes on the route.

(3)WRI. WRI occurs on the same route. It changes the position of student stop and its school node on the route. WRI can decrease the length of the route by moving the stops.

These neighborhood operators mentioned above can be adapted to solve different problem type of SBRP, and they are also implemented successfully in our former studies [5],[8],[30]. They can be executed sequentially or randomly, and also be organized by variable neighborhood descent (VND).

E. Initial Solution Construction Components

The initial solution construction algorithm components provide an initial solution for SBRP. They are implemented in the corresponding class files, including saving methods, sweep algorithm, LBH algorithm [18], gain tour methods, and several insertion methods. For every initial solution algorithm implemented in this framework, it also is adopted to the heterogeneous SBRP by adding the attributes about bus types to the route.

For single-school SBRP, an initial solution could be obtained by these algorithms quickly. While for multi-school SBRP, owing to its large scale, the initial solution is not easy to generate. Therefore, the two-stage initial solution generation method is designed. The first stage is to divide the multiple schools SBRP into several single school SBRP problems according to the school, and then generate an initial solution for each single school SBRP. Finally, these initial solutions are combined in the second stage.

These algorithm components are all implemented based on the basic data structure. Each algorithm component is encapsulated in an object-oriented class. There are some overloading methods to be called by different SBRP problem types.

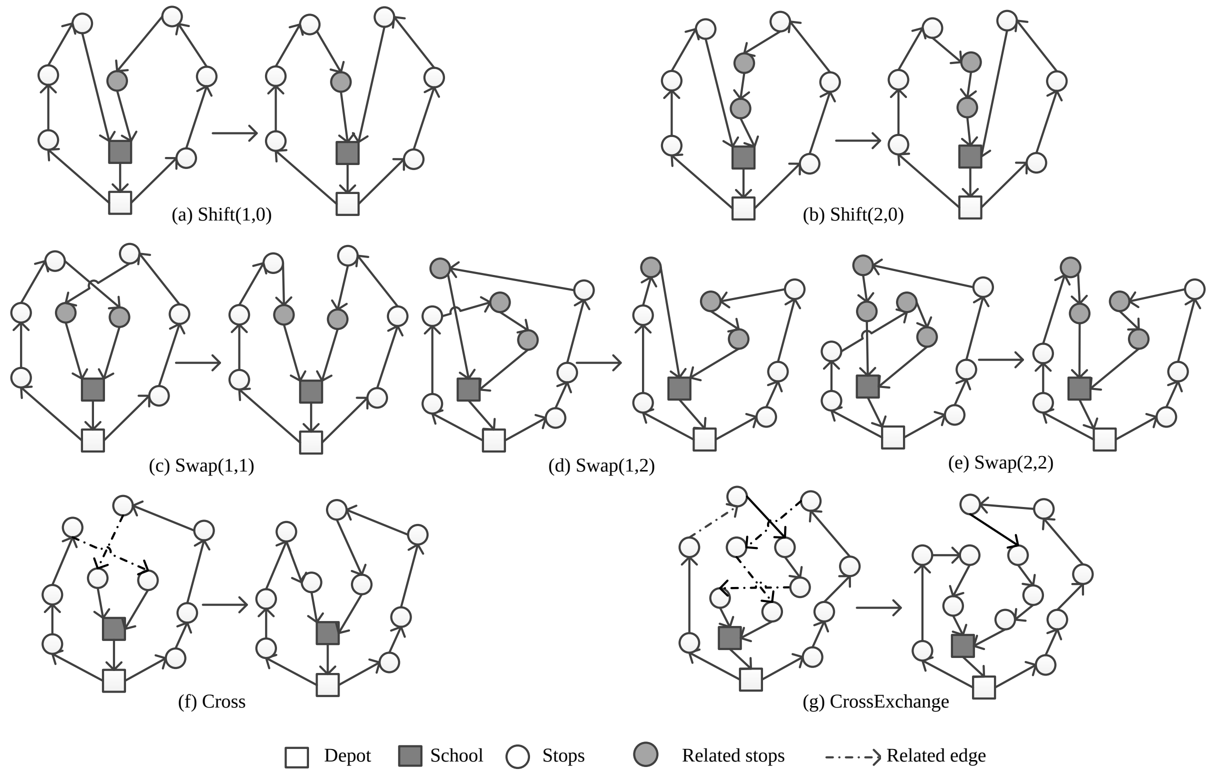

Fig. 4. inter-route neighborhood operators

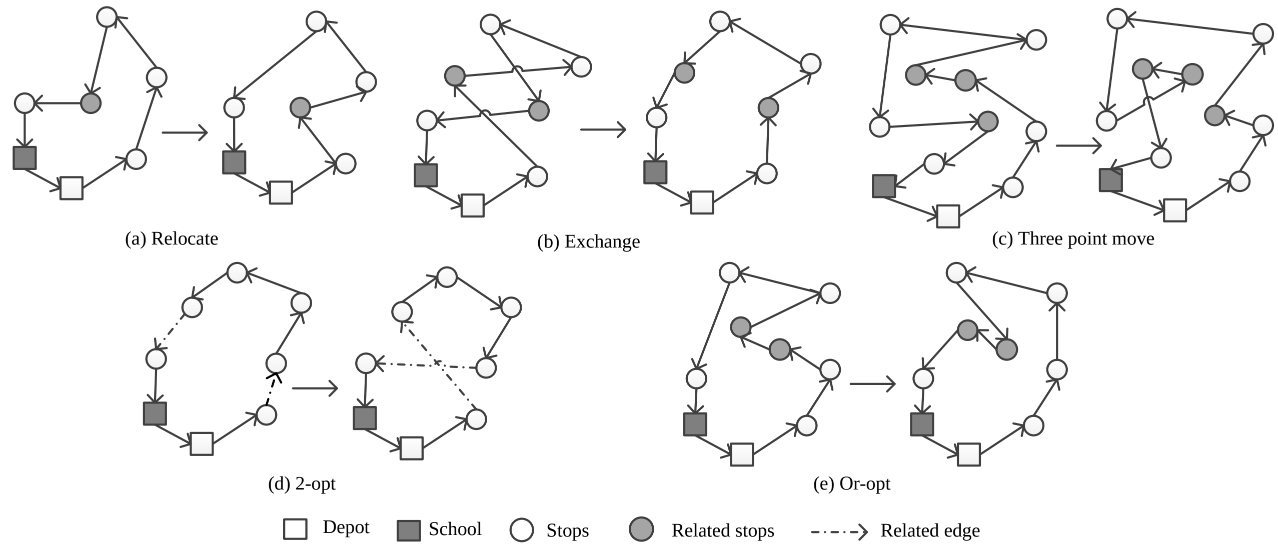

Fig. 5. intra-route neighborhood operators

F. Heuristic Strategies

To enhance the equality of algorithms, we also design several heuristic strategies, including search strategy, perturbation strategy, fleet adjustment strategy and neighborhood solution evaluation functions and acceptance rules. They are used in the local search procedure to lead the search trajectory of single-solution metaheuristics.

Search strategy is a kind of strategy that is used to specify the rule of point or edges in the local search phase. Random search strategy is to select randomly one stops in the stops list or neighborhood list. In the fixed search strategy, every student stop is selected to be executed in a fixed sequence in the stops list or neighborhood list. While for shorted route first strategy, all the routes are sorted in ascending order by the number of student stops, and the route having fewest student stops will be firstly executed in the local search.

Perturbation strategy is usually applied in the shaking procedure of VNS or perturbation procedure of ILS. The commonly used perturbation methods are shift or swapping of multiple points, and the cross of route segments. In additional, the framework also provides some large neighborhood space perturbation methods, such as ejection chain-based

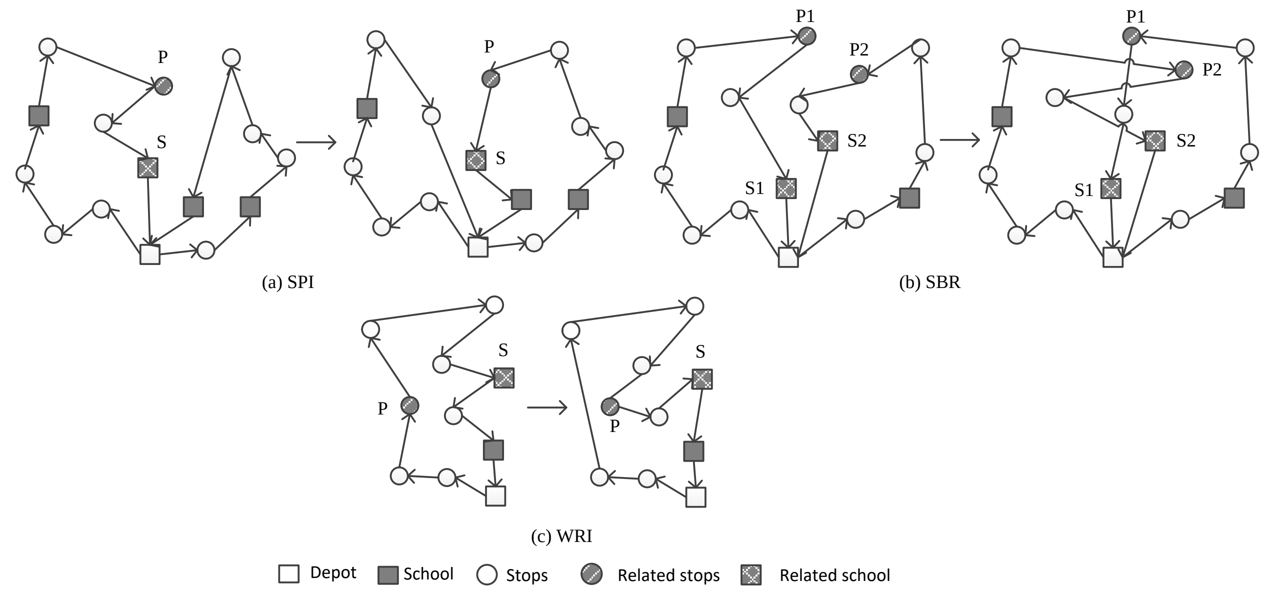

Fig. 6. neighborhood operators for multi-school SBRP

methods and ruin-and-recreated methods [5],[30]. Generally speaking, these perturbation methods have two parameters, which are the number of trials and the strength of disturbance measured by the number of points in the perturbation procedure. Once they could find a feasible solution, it will break the loop and return the solution.

For heterogeneous SBRP, reasonable fleet adjustment of the routes could have lower total cost of the solution. Fleet adjustment strategy tries to change the bus type of route to decrease the total cost of the route. The inter-route neighborhood operators are followed by the fleet adjustment strategy, only when without violating other constraints of the route except the capacity constraint. The successful of fleet adjustment depends on whether the type of bus that meets the conditions can be found. The fleet adjustment strategy is designed based on the route segment. For multi-school SBRP, the student stops and schools appear alternately on the route, so we cannot change directly the fleet type of whole route. We divide the whole route into several route segments by the school stops, and then find the reasonable fleet to meet the needs of all the route segments. For every route segment, the determination rules of the fleet type of bus can be found in [30].

Neighborhood solution evaluation functions and acceptance rules determine whether a new obtained neighborhood solution in the local search could be accepted or not, which forces the algorithm to search in the direction of decreasing objective value. Among the objectives of SBRP, cost is the main optimization objective, which is related with the number of routes, total travel distance, total cost and so on. For homogeneous SBRP, the optimization objectives of SBRP usually are the number of school buses and (or) total travel distance. While for heterogeneous SBRP, the total cost of solution is the main optimization objective. For different optimization objectives of SBRP, the neighborhood solution evaluation functions and acceptance rules are designed in our framework and described as follows.

(1) Minimizing the number of routes strategy. It is suitable for the SBRP that the number of school buses are the first optimization objective. This strategy uses a lexicographical evaluation mechanism and it is defined in Equation(1).

$$Eval(S) = \langle |S|, -\sum_{r\in S} |r|^2, \sum_{r\in S} d(r)\rangle \tag{1}$$

In the function, S is a solution, $|S|$ indicates the number of routes in the solution S, $|r|$ is the number of nodes in route r , and $d(r)$ indicates the distance of route r. The fewer routes of solution is first accepted, and then consider other conditions. The second component is to maximize $\sum_{r\in S} |r|^2$ to encourage the search operators to shift stops from shorter routes to longer routes. It will make the algorithm walk along the direction of reducing the number of routes more and more easily. Finally, the total distance of solution is evaluated, and it also can be combined with the other strategies defined below. It is very effective in minimizing the number of routes [5],[8].

(2)Increasing utilization of vehicle strategy. This strategy tends to accept the neighborhood solution whose vehicle utilization is increased. It is always combined with other neighborhood solution acceptance rules as a kind of compensation mechanism. When a new neighborhood solution is obtained by inter-route neighborhood operators, it could be regarded as an evaluation function to determine whether the solution could be accepted or not. It is defined in Equation(2).

$$f = min\{|Q_m{-}D_m|, |Q_p{-}D_p|\}{-}min\{|Q_k{-}D_k|, |Q_l{-}D_l|\} \tag{2}$$

In equation (2), we assume that the current bus types of route r and route t are m and p, and the actual load of them are D_m and D_p. The capacities of bus types of m and p are Q_m and Q_m. After the neighborhood operators executing and fleet adjustment, the new bus types of these two routes are k and l. If $f > 0$, it means that the vehicle utilization of route r or t is increased, the new solution is accepted. Otherwise, the cost saving in traveling distance will be considered.

(3) Record-to-record acceptance rule. This strategy uses the idea of record-to-record travel algorithm (RRT), which is a variant of SA. RRT uses *record* variable to represent the

fitness of current best solution, and a deviation parameter *dev* to represent the range of deviation between the current best solution and the new neighborhood solution. The definition of the rule is in Equation (3).

$$S = \left\{ \begin{array}{l} R, f(R) < record \\ R, f(R) < (1 + dev) \times record, dev \in [0, 1] \end{array} \right\} \tag{3}$$

In equation (3), R is the new neighborhood solution, S is the current best solution, *record* is the fitness value of solution, and *dev* is the relative coefficient. If the objective value of R is better than *record* or less than $(1 + dev) \times record$, R is selected as the new best solution and *record* will be updated. Otherwise R is rejected.

(4) Probability acceptance rule. It uses the probability acceptance rules of SA and its definition is in Equation (4).

$$S = \left\{ \begin{array}{l} R, f(R) < f(S) \\ R, e^{[-(f(R)-f(S))/T]} > p, p \in (0, 1) \end{array} \right\} \tag{4}$$

In Equation (4), R is the new neighborhood solution, S is the current best solution, $f(S)$ is the fitness value of solution, and T is the current temperature. If the objective value of R is better than $f(S)$, R is accepted. Otherwise, R is accepted by a certain probability.

These neighborhood solution evaluation and acceptance heuristics strategies are used as the parameters of algorithm. Like used in Vogel [13], the heuristic rules and the optimization objective of SBRP are also defined as an enumerated type. The value of each constant in this enumerated type is a number that is power of 2. Each bit of the binary represents a heuristic strategy to determine whether to enable the corresponding strategy through the *and* operation.

IV. Application Development and Experiment Results

In this section, we development the applicaitons of framework for three SBRP problems and CVRP. We first design an ILS metaheuristic algorithm on the basis of the framework for homogeneous SBRP with three operation modes, and compare it with existing algorithms. And then, we extend the ILS algorithm to CVRP and test on the benchmark instances.

A. An Iterated Local Search Metaheuristic

We design an ILS algorithm based on the proposed framework for three operation mode of homogeneous SBRP problems, including single-school SBRP, single load SBRP and mixed load SBRP. The optimization objective of three SBRP problems is to minimize the number of school buses. Because of the ILS algorithm for three SBRP problems, an adaptive selection mechanism is developed to select the initial solution construction method, neighborhood operators, perturbation and search strategy, according to the solved problem. The general description of ILS algorithm is described in the Algorithm 1.

Algorithm 1 ILS

Input: input the number of iterations (M), the size of neighborhood list (L), the deviation value($deviation$), the perturbation factor (p), problem type ($type$) and heuristic strategies ($rules$)

Output: best solution S^*

```
1:  S* = S = GetInitialSolution(type);
2:  NbList = GetOperatorsList(type);
3:  PList = GetStops(S);
4:  Record = Cost(S);
5:  while loop number is smaller or equal to M do
6:      for each operator op in NbList do
7:          RandomPerturb(PList,rules);
8:          for each student stop st in PList do
9:              S_b = GetSolution(S, st, L, op, rules);
10:             S = AcceptSolution(S,S_b,deviation,Record);
11:         S* = GetBetter(S,S*);
12:     S = Perturbation(S,p,type,rules);
13: return S*
```

Step (1) generates an initial solution according by the problem of SBRP. For single-school SBRP, we use saving methods to construct an initial solution. While for multi-school SBRP, the two-stage initial solution generation methods that described in Section III are used. Step (2) defines the neighborhood operators used in the local search process of ILS. For single-school SBRP, the neighborhood operators includes Shift(1,0), Shift(2,0), Swap(1,1), Swap(1,2), Relocate, Or-opt and 2-opt. The point-pair operators, such as SPI, WRI and SBR are used for multi-school SBRP. Step (3) obtains the list of student stops, and Step (4) sets the cost of solution S to $Record$. The main loop is controlled by a loop count M in Step (5)$\sim$(12). For every operator in operator set, a second level loop is defined in Step (6)$\sim$(11). The third-level loop for each node in stop set $PList$ is described in Step (8)$\sim$(10). The method RandomPerturb($PList$,$rules$) in step (7) is to change the sequence of student stops using a certain search strategy. Step (9)$\sim$(10) indicates that when a new neighborhood solution S_b is obtained, and the acceptance rule is applied to decide whether accept it or no. When the third-level loop is finished, the global best solution S^* is updated in step (11). After all the neighborhood operators are executed, the current solution S is perturbed and then starts the next iteration. When the top-level loop is finished, the global best solution S^* is obtained.

B. Benchmark Instances of SBRP

We use the benchmark instances proposed by [20] especially for mixed load SBRP to test the performance of ILS algorithm. There are two kinds of instances: a random spatial distribution of schools and bus stations (RSRB) and a clustered distribution (CSCB). We select RSRB01$\sim$RSRB04 and CSCB01$\sim$CSCB04 as the test instances for the multi-school SBRP. The number of bus stops ranges from 250 to 500, and the number of schools ranges from 6 to 25. For single-school SBRP, we prepared 12 instances from RSRB01 and CSCB01. These 12 instances are donated as R01$\sim$R06 and C01$\sim$C06 respectively. The scale of single-school problems is between 17 and 75.

For the convenience of comparison, all the problem constraints are set in accordance with that in [5],[8],[20],[31]. That is to say, the school bus capacity is 66. The average speed of the school bus is 20 mile per hour. And beyond that, the service time of bus stops and schools are calculated by the same regulation as that in [5],[8],[20],[31]. The maximum riding time of students in the school bus is 2700 seconds or 5400 seconds. The distance between any two school stops and schools is calculated by Manhattan distance.

C. Parameters Setting of ILS

The ILS algorithm was implemented by C# based on the proposed framework and executed on a computer with Intel Core i7 3.4GHz CPU and 8GB of RAM. The parameters values were selected after some preliminary experiments. The maximum iteration number is set to 50, and *deviation* is 10^{-4}. The parameter *rules*, which is the heuristic strategies of ILS, is combined with shorted route first search strategy, minimizing the number of routes strategy and record-to-record acceptance rules. Besides these parameters, the other parameters have different values according to the solved problem type of SBRP. For single-school SBRP, the value of parameter L is set to the smaller number of problem scale and 30. The perturbation method is based on ruin-and-recreate strategy and the perturbation factor p is set to 0.2. While for multi-school SBRP, the length of neighborhood list L is set to 150 owing to its large scale. The number of nodes moved in multiple points shift perturbation methods is set to a random integer within the range of [2,5]. The ILS algorithm was executed 10 times over each instance.

D. Experiment Results of SBRP

For single-school SBRP, the results on R01∼R06 and C01∼C06 instances with a maximum ridding time of 2700 s and 5400 s are shown in TABLE I. The column Stops is the problem scale. Column Nc represents the solution obtained by CPLEX 12.6, which are reported in [5]. Columns Num and Dis indicate the route number and total travel distance in seconds obtained by ILS algorithm. The column Navg and T are the average route number and the average execution time respectively. The optimal solutions given by CPLEX are labeled with a star(*), and the optimal solutions found by our algorithm are also labeled with a star(*).

As shown in TABLE I, we can find that the ILS algorithm outperforms CPLEX solver. Compared with CPLEX, ILS algorithm find less route numbers and the average route numbers is 10.58 and 9.92 respectively. While for the instances such as C01 and C05, the ILS algorithm can reduce 1 or 2 school buses. In additional, The ILS algorithm can also get the optimal route numbers that are found by CPLEX. The ILS algorithm can also obtain the optimal total travel distance for C04, C06 and R02. Finally, the ILS algorithm uses less computation time and all the instances can be solved within 2 seconds.

In order to evaluate the performance of ILS algorithm, we use it solve the single load SBRP and mixed load SBRP. The results of these approaches on RSRB01∼RSRB04 and CSCB01∼CSCB04 with different maximum ridding time (MRT) are shown in TABLE II. In the table, column Instances indicates the benchmark instance. For ILS algorithm, the columns N and Navg indicate the best route number and the average route number among the 10 solutions respectively. Column Dis is the total travel distance in miles of the best solution and the column T is the average computation time in seconds. In additional, we also compared our ILS algorithm with existing algorithms [8], [20], [31]. For single load SBRP, the results obtained by post-improvement heuristic [20] and simulate annealing [31] are shown in columns PH and SA. While for mixed load SBRP, the columns PH and RRT in TABLE II indicate the results founded in [20] and [8].

Seen from the results of TABLE II, the ILS algorithm is more competitive than the existing algorithm for single load SBRP and mixed load SBRP. First, the ILS algorithm is effective to solve these two multi-school SBRP problems. For single load SBRP, it reduces the average route numbers by 24.66% and 6.27% when compared with post improvement heuristic(PH) [20] and simulate annealing(SA) [31]. While for mixed load SBRP, it improved on average by 9.72% and 0.95% respectively, compared with post improvement heuristic [20] and RRT [8]. In additional, the ILS algorithm can solve these both multi-school SBRP in reasonable computation time. The average computation time of both problems are 69.37 seconds and 141.30 seconds respectively. For the large instances, which have 500 student stops and 25 schools, the ILS algorithm can found the better solution within five minutes.

Taking into account the results that are shown on these two tables, we can come to a conclusion that the proposed ILS algorithm implemented based on the framework is effective and it is suitable to solve three SBRP problems.

E. Extending the ILS algorithm for CVRP

The proposed ILS algorithm can be easy to extend for CVRP, because single-school SBRP is the similar with CVRP. For single-school SBRP, the school bus starts from the depot and visits the bus stops in a specific order, and then the school node. If we make the coordinate of school node same as that of depot node, the routes of SBRP are the same as those of CVRP. The other constraints of SBRP, such as service time of bus stops and maximum ridding time of students, can be modified to adapt for CVRP. Specifically, the service time of nodes can be read from the instance file, which need not to be calculated by the demand of students like SBRP. The maximum ridding time of students is set to a bigger positive integer in order to ignore the constraint. The parameters setting of ILS algorithm are same as that for single-school SBRP. It should be noted that the optimization objective of CVRP is the total travel distance. Thus, the heuristic strategies of ILS algorithm should exclude the strategy of minimizing route number.

Next, we use ILS algorithm to solve standard benchmark instances of CVRP. The performance of the ILS algorithm is tested on 50 best known CVRP instances, which include problem sets B, E and P respectively. All the instances may be downloaded from the site $\langle https://neo.lcc.uma.es/vrp\rangle$. And then, we compare ILS algorithm with the extended savings algorithm (ESA) proposed by [32] and ant colony optimization algorithm(LNS-ACO) implemented by [33].

TABLE III and TABLE IV shows the results of these algorithms. The column BKS denotes the best known solution of benchmark instance. The column Best represents the best total distance obtained by the ILS algorithm, and the column T is the average computation time of ILS algorithm in seconds. The results of extended savings algorithm (ESA) [32] and ant colony optimization algorithm (LNS-ACO) [33] are shown in columns ESA and LNS-ACO.

TABLE I
RESULTS OF SINGLE-SCHOOL SBRP

Instances		MRT=2700					MRT=5400				
Name	Stops	Nc	Num	Navg	Dis(s)	T(s)	Nc	Num	Navg	Dis(s)	T(s)
C01	70	17	16	16.3	35013	1.51	15	14	14.8	34059	1.53
C02	35	12	12	12	22798	0.55	11*	11*	11*	22849	0.68
C03	30	9*	9*	9.5	18867	0.59	8*	8*	8*	16490	0.72
C04	23	7*	7*	7*	13327*	0.32	7*	7*	7*	13327*	0.37
C05	75	20	18	18	36657	1.61	18	18	18	36671	1.70
C06	17	6*	6*	6*	9609*	0.23	6*	6*	6*	9431*	0.27
R01	38	9*	9*	9.9	18574	0.74	9*	9*	9*	17526	0.69
R02	40	9*	9*	9.8	18866*	0.77	9*	9*	9.1	18062	0.76
R03	51	13*	13*	13.2	21560	1.04	13*	13*	13*	21055	1.03
R04	35	10	10	10	20807	0.72	7*	7*	7.6	19696	0.73
R05	42	9*	9*	9.8	18404	0.87	9*	9*	9.1	17874	0.84
R06	44	9*	9*	9.6	18255	0.85	8*	8*	8.8	17410	0.87
Avg	41.67	10.83	10.58	10.93	21061.42	0.82	10	9.92	10.12	20370.83	0.85

TABLE II
RESULTS OF SINGLE LOAD SBRP AND MIXED LOAD SBRP

Instances	MRT	Single Load						Mixed Load					
		PH	SA	ILS				PH	RRT	ILS			
				N	Navg	Dis	T(s)			N	Navg	Dis	T(s)
CSCB01	2700	39	31	28	28.6	1537.06	36.59	30	27	26	27.2	1353.91	69.76
CSCB02	2700	33	26	25	25.8	1785.74	38.51	30	26	25	25.7	1498.86	68.52
CSCB03	2700	66	59	53	53.8	3994.89	103.52	55	49	48	49.5	3708.06	197.43
CSCB04	2700	72	61	58	58.2	4402.98	115.81	62	57	56	57	3787.73	194.36
RSRB01	2700	35	26	26	26.6	1612.97	32.06	30	26	26	26.5	1364.14	49.48
RSRB02	2700	32	27	26	26.2	1907.93	34.98	29	27	26	26.8	1947.59	56.42
RSRB03	2700	66	47	51	52	3167.76	98.28	56	53	51	51.5	3445.47	242.37
RSRB04	2700	68	58	54	54.9	3645.32	113.15	59	52	53	53.8	3163.97	287.17
CSCB01	5400	35	29	23	23.7	1477.33	30.62	24	23	23	23.4	1441.89	46.93
CSCB02	5400	27	23	20	20.1	1457.29	35.45	22	19	19	19.9	1429.54	63.38
CSCB03	5400	52	42	40	40.4	3420.64	101.07	41	39	39	39.9	3323.99	212.56
CSCB04	5400	57	45	41	41.6	3762.63	114.29	43	37	40	40.7	3879.29	242.74
RSRB01	5400	31	28	23	23.8	1495.23	29.87	27	24	24	24.9	1532.03	44.32
RSRB02	5400	30	23	22	22.9	1800.61	31.91	23	23	22	22.3	1875.05	57.95
RSRB03	5400	61	46	46	46.2	3062.63	89.13	47	47	46	46.3	3189.11	198.27
RSRB04	5400	56	41	41	41.4	2804.33	104.75	46	40	40	41.2	2606.46	229.18
Avg	4050	47.5	38.25	36.06	36.64	2583.46	69.37	39	35.56	35.25	36.04	2471.69	141.30

The column Dev indicates the percentage deviation between best known solution and the best solution obtained by ILS ($Dev = ((Best - BKS)/BKS) * 100$).

Seen from the TABLE III and TABLE IV, we can find that ILS algorithm also effectively solve CVRP and it is able to obtain better results than ESA [32] and LNS-ACO [33]. For all the instances in sets B, P and E, the ILS algorithm has lowest averages of Dev values among of these three algorithms. The percentage deviation values for these three problem sets are 0.32%, 0.28% and 0.44% respectively.

Among of the 50 instances, the ILS algorithm can find 26 best solutions, which equal the best known solution of standard instances. The best solution of 19 instances are a litter bigger than best known solutions, and the percentage deviation values of 6 instances are below 0.2%. All the percentage deviation values are all smaller than 1% for these 19 instances. The percentage deviation values of 5 instances are between 1% and 3%. In all, compared with ESA and LNS-ACO algorithms, the ILS algorithm has lower average of percentage deviation values. It also indicates that the ILS algorithm is capable of finding optimal or near-optimal solutions for all the instances.

TABLE III
COMPARISON RESULTS FOR THE PROBLEM SET B AND SET P

Instances	BKS	ESA	LNS-ACO	ILS		Instances	BKS	ESA	LNS-ACO	ILS	
		Dev	Dev	Best	Dev			Dev	Dev	Best	Dev
B-n31-k5	672	0.00	0.00	672	0.00	P-n16-k8	450	0.00	0.00	450	0.00
B-n34-k5	788	0.00	0.00	788	0.00	P-n19-k2	212	3.30	0.00	212	0.00
B-n35-k5	955	0.84	0.00	955	0.00	P-n20-k2	216	0.93	0.00	216	0.00
B-n38-k6	805	1.24	0.00	805	0.00	P-n21-k2	211	0.47	0.00	211	0.00
B-n39-k5	549	0.18	0.00	549	0.00	P-n22-k2	216	0.00	0.00	216	0.00
B-n41-k6	829	4.46	0.00	829	0.00	P-n23-k8	529	0.00	0.00	529	0.00
B-n43-k6	742	0.54	0.00	742	0.00	P-n40-k5	458	0.22	0.00	458	0.00
B-n44-k7	909	1.32	0.00	909	0.00	P-n45-k5	510	0.20	0.00	510	0.00
B-n45-k5	751	0.00	0.00	751	0.00	P-n50-k7	554	0.00	0.00	554	0.00
B-n45-k6	678	1.18	0.00	679	0.15	P-n50-k8	631	0.95	1.90	633	0.32
B-n50-k7	741	0.00	0.00	741	0.00	P-n50-k10	696	0.14	0.00	698	0.29
B-n50-k8	1312	1.30	0.53	1313	0.08	P-n51-k10	741	0.00	0.81	744	0.40
B-n52-k7	747	0.67	0.00	747	0.00	P-n55-k7	568	1.06	0.00	570	0.35
B-n56-k7	707	0.00	0.00	707	0.00	P-n55-k8	576	0.00	0.00	577	0.17
B-n57-k9	1598	0.13	0.00	1600	0.13	P-n55-k10	694	0.14	0.00	699	0.72
B-n63-k10	1496	2.81	1.20	1537	2.74	P-n60-k10	744	0.13	1.48	746	0.27
B-n64-k9	861	0.00	1.51	878	1.97	P-n60-k15	968	0.00	0.93	977	0.93
B-n66-k9	1316	1.90	1.06	1318	0.15	P-n65-k10	792	0.51	1.01	792	0.00
B-n67-k10	1032	1.74	1.74	1034	0.19	P-n70-k10	834	0.00	1.21	835	0.12
B-n68-k9	1272	1.57	1.42	1276	0.31	P-n76-k4	589	2.36	0.84	601	2.04
B-n78-k10	1221	2.05	0.57	1233	0.98	P-n76-k5	627	3.03	2.87	629	0.32
Avg	951.48	1.04	0.38	955.38	0.32	Avg	562.67	0.64	0.53	564.62	0.28

TABLE IV
COMPARISON RESULTS FOR THE PROBLEM SET E

Instances	BKS	ESA	LNS-ACO	ILS	
		Dev	Dev	Best	Dev
E-n22-k4	375	0.27	0.00	375	0.00
E-n23-k3	569	0.35	0.00	569	0.00
E-n30-k4	503	0.40	0.00	503	0.00
E-n33-k4	835	0.12	0.00	835	0.00
E-n76-k7	682	3.08	1.91	685	0.44
E-n76-k8	735	2.99	1.22	737	0.27
E-n76-k14	1021	2.45	0.88	1039	1.76
E-n101-k14	1071	3.73	1.41	1082	1.03
Avg	723.88	1.67	0.68	728.13	0.44

V. CONCLUSION

In the actual school bus route planning, the number of schools, the characteristics of the fleet and the operation mode will produce many variations of SBRP. It is necessary to quickly solve these SBRP variations with different problem characteristics. For this reason, we proposed a neighborhood-center metaheuristic algorithm framework to simplify the design and implementation process of solving algorithms for SBRP. The framework is designed based on the unified model expression of these problems and then implemented by C#, which is known as an object-oriented programming language. The framework provides many components from bottom to top, including basic data structure, operation functions, neighborhood operators, basic algorithms and heuristic strategies and so on. The common part of the framework is implemented in advance and the users are allowed to expand based on the framework to realize other difference parts. Further, we design an ILS algorithm based on the framework for three SBRP applications and classical CVRP problem. The ILS algorithm are tested on a set of SBRP instances and 50 best known CVRP problems. The designed ILS algorithm outperforms existing SBRP algorithms, and it also find 26 optimal solutions for CVRP standard instances. The experimental results not only prove the designed ILS algorithm is very competitive, but also show that the proposed algorithm framework is effective and scalability.

In the future, we will improve our algorithm framework to support math-heuristic algorithms that can combine meta-heuristics with common exact algorithms. In additional, we are going to expand the framework to solve more types of SBRP problems.

REFERENCES

[1] Newton R M and Thomas W H, “Design of school bus routes by computer,” *Socio-Economic Planning Sciences*, vol. 3, no. 1, pp. 75-85, 1969.

[2] Park J and Kim B I, “The school bus routing problem: A review,” *European Journal of Operational Research*, vol. 202, no. 2, pp. 311-319, 2010.

[3] Ellegood W A, Solomon S, North J and Campbell J F, “School bus Routing Problem: Contemporary Trends and Research Directions,” *Omega*, vol. 95, 102056, 2020.

[4] Euchi J and Mraihi R, “The urban bus routing problem in the Tunisian case by the hybrid artificial ant colony algorithm,” *Swarm and Evolutionary Computation*, vol. 2, no. 1, pp. 15-24, 2012.
[5] Dang L X, Hou Y E, Liu Q S and Kong Y F, “A Hybrid Metaheuristic Algorithm for the Bi-objective School Bus Routing Problem,” *IAENG International Journal of Computer Science*, vol. 46, no. 3, pp. 409-416, 2019.
[6] Ellegood, W.A., J.F. Campbell and J. North, “Continuous approximation models for mixed load school bus routing,” *Transportation Research Part B: Methodological*, vol. 77, no. 7, pp. 182-198, 2015.
[7] Nanry W P and Barnes J W. “Solving the pickup and delivery problem with time windows using reactive tabu search,” *Transportation Research B: Methodological*, vol. 34, no. 2, pp. 107-121, 2000.
[8] Hou Y E, Dang L X, Wang Z Y and Kong Y F, “A metaheuristic algorithm for routing school buses with mixed load,” *IEEE Access*, vol. 8, pp. 158293-158305, 2020.
[9] Cordeau J, Laporte G and Mercier A. “A unifed tabu search heuristic for vehicle routing problems with time windows,” *Journal of Operational Research Society*, vol. 52, no. 8, pp. 928-936, 2001.
[10] Hashimoto H, Yagiura M and Ibaraki T. “An iterated local search algorithm for the time-dependent vehicle routing problem with time windows,” *Discrete Optimization*, vol. 5, no. 2, pp. 434-456, 2008.
[11] Subramanian A, Uchoa E and Ochi L S, “A hybrid algorithm for a class of vehicle routing problems,” *Computers and Operations Research*, vol. 40, no. 10, pp. 2519-2531, 2013.
[12] Vidal T, Crainic T G, Gendreau M and Prins C. “A unified solution framework for multi-attribute vehicle routing problems,” *European Journal of Operational Research*, vol. 234, no. 3, pp. 658-673, 2014.
[13] Vogel U. A flexible metaheuristic framework for solving rich vehicle routing problems. Germany: University of Cologne, 2011.
[14] Groër C, Golden B and Wasil E. “A library of local search heuristics for the vehicle routing problem,” *Mathematical Programming Computation*, vol. 2, no. 2, pp. 79-101, 2010.
[15] Bektaş T and Elmastaş S, “Solving school bus routing problems through integer programming,” *Journal of the operational Research Society*, vol. 58, no. 12, pp. 1599-1604, 2007.
[16] Riera-Ledesma J and Jose Salazar-Gonzalez J, “A column generation approach for a school bus routing problem with resource constraints,” *Computers & Operations Research*, vol. 40, no. 2, pp. 566-583, 2013.
[17] Caballero R and Laguna M, “Bi-Objective Bus Routing: An Application to School Buses in Rural Areas,” *Transportation Science*, vol. 47, no. 3, pp. 397-411, 2013.
[18] Bramel J and Simchi-Levi D. “A location based heuristic for general routing problems,” *Operations Research*, vol. 43, no. 4, pp. 649-660, 1995.
[19] Braca J, Bramel J, Posner B and Simchi-Levi D. “A Computerized Approach to the New York City School Bus Routing Problem,” *IIE Transactions*, vol. 29, pp. 693-702, 1997.
[20] Park J, Tae H and Kim B I, “A post-improvement procedure for the mixed load school bus routing problem,” *European Journal of Operational Research*, vol. 217, no. 2, pp. 204-213, 2012.
[21] Spada M, Bierlaire M and Liebling T, “Decision-Aiding Methodology for the School Bus Routing and Scheduling Problem,” *Transportation Science*, vol. 39, no. 4, pp. 477-490, 2005.
[22] Pacheco J, Caballero R, Laguna M and Molina J, “Bi-Objective Bus Routing: An Application to School Buses in Rural Areas,” *Transportation Science*, vol. 47, no. 3, pp. 397-411, 2013.
[23] Lima F M, Pereira D S, Conceição S V and Nunes N T, “A mixed load capacitated rural school bus routing problem with heterogeneous fleet: Algorithms for the Brazilian context,” *Expert Systems with Applications*, vol. 56, pp. 320-334, 2016.
[24] Lima F M, Pereira D S, Conceição S V and Nunes N T, “A multi-objective capacitated rural school bus routing problem with heterogeneous fleet and mixed loads,” *4OR*, vol. 15, no. 4, pp. 1-28, 2017.
[25] Kang M, Kim S K, Felan J T, Choi H R and Cho M, “Development of a genetic algorithm for the school bus routing problem,” *International Journal software Engineering & Applications*, vol. 9, no. 5, pp. 107-126, 2015.
[26] Kotoula K M, Morfoulaki M, Aifadopoulou G and Tzenos P, “Calculating the optimal school bus routing and its impact on safety and environment protection,” *Transportation Research Record: Journal of the Transportation Research Board*, vol. 17, pp. 142-150, 2017.
[27] Ünsal O and Yigit T, “Using the genetic algorithm for the optimization of dynamic school bus routing problem,” *BRAIN Broad Research in Artificial Intelligence and Neuroscience*, vol. 9, no. 2, pp. 6-21, 2018.
[28] Eldrandaly K A, Abdallah A F, “A novel GIS-based decision-making framework for the school bus routing problem,” *Geo-spatial Information Science*, vol. 15, no. 1, pp. 51-59, 2012.
[29] Corberan A, Fernandez E, Laguna M and Marti R A, “Heuristic solutions to the problem of routing school buses with multiple objectives,” *Journal of the Operational Research Society*, vol. 53, no. 4, pp. 427-435, 2002.
[30] Hou Y E, Dang L X, Kong Y F, Wang Z Y, and Zhao Q J, “A Hybrid Metaheuristic Algorithm for the Heterogeneous School Bus Routing Problem and a Real Case Study,” *IAENG International Journal of Computer Science*, vol. 47, no. 4, pp. 775-785, 2020.
[31] Chen X P, Kong Y F, Dang L X, Hou Y E and Ye X Y, “Exact and Metaheuristic Approaches for a Bi-Objective School Bus Scheduling Problem,” *PLOS One*, vol. 10, e0132600, 2015.
[32] Stanojević M, Stanojević B and Vujoševi ć M, “Enhanced savings calculation and its applications for solving capacitated vehicle routing problem,” *Applied Mathematics and Computation*, vol. 219, no. 20, pp. 10302-10312, 2013.
[33] Akpinar S, “Hybrid large neighbourhood search algorithm for capacitated vehicle routing problem,” *Expert Systems with Applications*, vol. 61, pp. 28-38, 2016.