Development of a Genetic Algorithm for the School Bus Routing Problem

Moohong Kang¹, Sung-Kwan Kim², Joe T. Felan³, Hyung Rim Choi⁴ and Minje Cho⁵

¹ Port Research Division, Korea Maritime Institute
²,³ Dept. of Management, University of Arkansas at Little Rock
⁴,⁵ Dept. of Management Information System, Dong-A University
¹ mkang@kmi.re.kr
² sxkim, ³ jtfelan@ualr.edu
⁴ hrchoi, ⁵ mjcho@dau.ac.kr

Abstract

The School Bus Routing Problem (SBRP) covers the issue of establishing plans to efficiently transport students distributed across a designated area to the relevant schools using defined resources. As with the similar Vehicle Routing Problem (VRP), the SBRP may have diverse constraints such as heterogeneous vehicles, the allotted time window and multiple depots. Many solutions for effectively solving the problem are currently being studied. By their nature, these routing problems are NP-Hard (non-deterministic polynomial-time hard) problems in which the search domains increase exponentially as they become larger, thus making it difficult to obtain solutions using an exact approach except for relatively simple and localized problems. Therefore the heuristic approach is being studied in many regions. In this study, an algorithm was developed using genetic algorithms, which stem from meta-heuristic algorithms, and the algorithm was tested against diverse problems to identify its performance and practicality.

Keywords: School Bus Routing Problem, Genetic Algorithm, Heuristic Algorithm

1. Introduction

In many countries, including the USA, school buses are operated to transport students to and from school. In particular, in the case of countries that have a large land mass, such as the USA, efforts are being made to effectively utilize school buses in order to reduce not only the unnecessary waste of resources that may occur due to inefficient school buses, but also the parents’ efforts and students’ inconvenience [16]. Based on the results of a survey conducted through interviews, in the case of City B in State A in the USA, maintenance expenses of at least US$35,000 have been spent per bus per year and the problem that several students had to ride on a bus for several hours every day is seen. Hence, plans to efficiently operate school buses were badly needed. The School Bus Routing Problem (SBRP) is a hypothesis proposed by Newton and Thomas [17] in order to solve the problems resulting from these inefficient schools bus operations. It aims to establish plans to transport students distributed across designated areas to the relevant schools using defined resources [5, 19].

As shown in Figure 1, the SBRP has a total of five sub-problems including Data Preparation, Bus Stop Selection, Bus Route Generation, School Bell Time Adjustment and Route Scheduling [19]. Of them, Data Preparation is a problem when preparing the data necessary for routing, such as locations and distances. Bus Stop Selection is a problem when allocating students distributed in different areas to nearby bus stops in order to determine the bus stops that must be visited by the buses. The Bus Route Generation problem covers the decision on the order in which to visit
the determined bus stops, and the School Bell Time Adjustment problem covers the adjustment of the time by which the students should arrive at school. Finally, the Route Scheduling problem deals with assigning the determined routes to individual buses, taking into consideration the time by which the students must arrive at school, while establishing plans to enable each bus to transport the students from many schools to their particular school.

In this study, it was assumed that the Data Preparation and School Bell Time Adjustment problems had already been implemented and thus, the necessary pieces of information had been prepared and an algorithm for solving the other problems, Bus Stop Selection, Bus Route Generation and Route Scheduling problem, was developed. In particular, the algorithm was designed to solve the Bus Route Generation and Route Scheduling problems simultaneously considering the Mixed Loading effect.

![Figure 1. Sub-Problems of SBRP](image)

2. Related Studies

Studies related to SBRP have been steadily conducted since the 1960s. These studies have tried to effectively solve problems with diverse assumptions, objective functions and constraints by applying various kinds of solutions and the details are well summarized in a paper written by Park et al., [19].

In most SBRP studies, Vehicle Capacity (C) was limiting as a constraint and because of this, the number of students that can be picked up by a vehicle at one time was limited and exceeding this number was prohibited. In addition, the students' Maximum Riding Time (MRT) was limited so that students could only ride on buses for a reasonable amount of time, and based on the results of actual surveys, some students were riding on buses for several hours as a result of inefficient plans. Bektas et al., [5] applied these two constraints to the problem in a study of Integer Programming-based solutions that could minimize bus operations. In a case where it was necessary to transport 519 students to an elementary school, they reduced the number of buses in operation from 26 to 18 and enhanced the average bus utilization rate from 60.49 % to 87.37 %.

The Time Window (TW) constraint operates in order to make buses arrive at the appropriate time so that the students arrive at school at the correct time and is determined by the School Bell Time Adjustment problem, which was being applied in diverse forms. Although a constraint was placed so that buses should arrive at schools at designated times in general, Braca et al., [6] used a range of time from 25 minutes before the time to arrive at school in 5 minutes after the time to arrive at school as a constraint in order to solve the problem while also ensuring that students did not arrive at school too early or too late.
Studies related to SBRP include not only those conducted using these constraints, but also those conducted under other diverse assumptions. Among them are studies that addressed the Heterogeneous Fleet (HT) problem, in which there were heterogeneous buses in terms of the characteristics of vehicles. In relation to this, Thangiah et al., [25] solved the problem based on an assumption that the numbers of students that could be picked up by different vehicles would be different, while Ripplinger [21] solved the problem assuming that there were buses that could pick up students with special needs and those that could not pick up that students.

Unlike most of the other existing studies that solved the problem based on one departure place, Thangiah et al., [24] solved the problem by also considering multi-depots (many departure places). In their paper, they developed a heuristic algorithm for solving a Rural SBRP with 13 departure places, 5 schools, 71 bus stops and 583 students.

Other studies have applied the assumption of Mixed Loading. Mixed Loading refers to the transportation of students from different schools, both to and from school, using only one bus and this enables flexible bus operations and thereby enhances efficiency. This solution can be found in a paper written by Braca et al., [6].

As reviewed above, the SBRP can be seen as a complicated problem that may have diverse constraints and assumptions. The SBRP problem has already been identified as an NP-Hard problem by many other researchers [19] and as further constraints or assumptions are added, the problem’s search domain increases exponentially. Therefore, although small-sized problems are solved by ‘exact approaches’, it is difficult to find solutions for large-sized problems. Therefore, in order to solve large-sized problems, recently, heuristic approaches such as Simulated Annealing, Deterministic Annealing, Tabu Searching, Genetic Algorithm, Ant Colony Optimization, Neural Network and Harmony Searching have been frequently utilized [9,19].

Addor et al., [1] solved a SBRP of Wood bridge School Complex of Sekondi-Takoradi in Ghana using integer programming model and ant colony optimization based meta-heuristic, so they could reduce the total route length by approximately 32%.

Schittekat et al., [22] developed the GRASP-VND algorithm which includes mixed algorithm with meta-heuristic and exact approach in order to solve three sub-problems of SBRP, bus stop selection, student allocation, and bus route generation simultaneously.

Kim et al., [15] formulated the SBRP model as a mixed-integer programming and solved small size problems using the commercial optimization package CPLEX. And they conducted comparison evaluations between the result and the heuristic algorithm which they developed in this research.

In this study, in order to solve SBRP, we developed an algorithm based on genetic algorithms (GA) that have been frequently utilized in VRP, which is similar to SBRP. Especially, GA was applied to many problems, not only routing problems such as TSP (Traveling Salesman Problem), VRP but also scheduling problems such as JSSP (Job Shop Scheduling Problem) [18], NSP (Nurse Scheduling Problem) [14], so the performance and practicality were already identified.

3. Definition of the Problem

As mentioned in Chapter 1, in this study, of a total of five sub-problems of the SBRP, three sub-problems including Bus Stop Selection, Bus Route Generation and Route Scheduling were addressed in two stages, first implementing Bus Stop Selection (BSS) and then planning Bus Routing and Scheduling (BRS) based on determined bus stops. The annotations used in this study are as follows:
Firstly, the BSS problem is a problem of allocating students to bus stops and in this study, individual students were to be allocated to bus stops within a designated distance. As shown in Figure 2, this can be seen as a problem of allocating the students (ST<sub>s</sub>) indicated by gray squares to the bus stops (BS<sub>k</sub>) indicated by red circles and only those students who are within the scope of individual bus stops indicated by gray circles may be allocated to those individual bus stops.

Table 1 shows students allocated to individual bus stops based on the results of the BSS. Based on these results, it can be identified how many students from individual schools have been allocated to individual bus stops as shown in Table 2 and Figure 3 and that buses may not visit bus stop 2 (BS<sub>2</sub>) and bus stop 5 (BS<sub>5</sub>) as no student was allocated to those bus stops. However, if student 7 (ST<sub>7</sub>) allocated to bus stop 3 (BS<sub>3</sub>) as shown in Table 1 was allocated to bus stop 2 (BS<sub>2</sub>) instead of bus stop 3 (BS<sub>3</sub>), the assignments would change as shown in Table 3 and Figure 4 and thus the number of bus stops to be visited in the BRS will increase by one, and the number of student groups to be picked up and transported to their school will also be increased by one (SG<sub>ck</sub>), therefore this can be considered as an inefficient result.

Therefore, in this study, in order to obtain the results shown in Table 2 and Figure 3, with a view to improving the result of the BRS to be implemented after the BSS, the minimization of student groups was defined as an objective function. If the minimization of students’ travel distances is defined as an objective function, the students will be simply allocated to the nearest bus stops as shown in Figure 5 and as a result, the student groups that must be transported to school will be 10 as shown in Table 4. Therefore buses will have to visit more bus stops in the BRS. This method, therefore, cannot be said to be as effective when considering the entire SBRP.

---

**Table 1. Example of BSS Allocation**

<table>
<thead>
<tr>
<th>Student</th>
<th>Schools</th>
<th>Allocated bus stops</th>
<th>Student</th>
<th>Schools</th>
<th>Allocated bus stops</th>
</tr>
</thead>
<tbody>
<tr>
<td>ST&lt;sub&gt;1&lt;/sub&gt;</td>
<td>SC&lt;sub&gt;1&lt;/sub&gt;</td>
<td>BS&lt;sub&gt;4&lt;/sub&gt;</td>
<td>ST&lt;sub&gt;11&lt;/sub&gt;</td>
<td>SC&lt;sub&gt;1&lt;/sub&gt;</td>
<td>BS&lt;sub&gt;3&lt;/sub&gt;</td>
</tr>
<tr>
<td>ST&lt;sub&gt;2&lt;/sub&gt;</td>
<td>SC&lt;sub&gt;1&lt;/sub&gt;</td>
<td>BS&lt;sub&gt;1&lt;/sub&gt;</td>
<td>ST&lt;sub&gt;12&lt;/sub&gt;</td>
<td>SC&lt;sub&gt;1&lt;/sub&gt;</td>
<td>BS&lt;sub&gt;2&lt;/sub&gt;</td>
</tr>
<tr>
<td>ST&lt;sub&gt;3&lt;/sub&gt;</td>
<td>SC&lt;sub&gt;2&lt;/sub&gt;</td>
<td>BS&lt;sub&gt;3&lt;/sub&gt;</td>
<td>ST&lt;sub&gt;13&lt;/sub&gt;</td>
<td>SC&lt;sub&gt;2&lt;/sub&gt;</td>
<td>BS&lt;sub&gt;2&lt;/sub&gt;</td>
</tr>
<tr>
<td>ST&lt;sub&gt;4&lt;/sub&gt;</td>
<td>SC&lt;sub&gt;2&lt;/sub&gt;</td>
<td>BS&lt;sub&gt;1&lt;/sub&gt;</td>
<td>ST&lt;sub&gt;14&lt;/sub&gt;</td>
<td>SC&lt;sub&gt;2&lt;/sub&gt;</td>
<td>BS&lt;sub&gt;4&lt;/sub&gt;</td>
</tr>
</tbody>
</table>
Next, the BRS is the issue of efficiently transporting the individual student groups drawn in the BSS to their schools using the defined bus resources by determining to which buses the individual student groups should be allocated, and in what order the individual buses should visit the bus stops and schools (SC) of the student groups allocated to them.

In this study, the minimization of the sum of the travel distances of buses (TBD; Total Bus travel Distance or time) that can transport all the student groups to their individual schools was defined as an objective function. However, an algorithm was developed to enable evaluation by the number of buses used (N) or the Total Student riding Distance or time (TSD) and the algorithm was used when comparing results. In addition, as with the related studies mentioned in Chapter 2, bus capacities were restricted [5], the times by which students should be transported to schools were defined [6] and the maximum time during which each student can ride on a bus [5] was also restricted. In addition, students
from different schools were allowed to ride on the same bus [6], the buses had different capacities [25] and their departure places could be different [24].

The distances between individual bus stops, schools and bus depots were calculated based on straight-line distances and the car speed was set to 20 mph, as used in a paper written by Park et al., [19]. The times PT (Pick-up Time) required to pick up the students SGck at individual bus stops and the Drop-off Times (DT) required to drop off the students at individual schools were calculated using the following numerical formulas as used by Braca et al., [6].

\[
PT_k = 19.0 + 2.6 \sum_{c=1}^{SC} SG_{ck}
\]

\[
DT_k = 29.0 + 1.9 \sum_{c=1}^{BS} SG_{ek}
\]

Although many buses may visit one bus stop, each student group can only be allocated to one bus. In addition, since the schools that must be visited by student groups were defined, once student groups were allocated to buses, the buses had to visit the student groups' schools. Although the bus stops and schools that must be visited by the buses were determined as such, the order of visits was not determined and thus the order of visitation for each bus must be determined. When determining the order of visits, it should be noted that buses could visit each bus stop and each school only once, and had to visit the bus stop of each student group before visiting the relevant school.

In this study, determining to which buses individual student groups were to be allocated was termed Bus Allocation Planning (BAP) and determining the order of visitation of individual buses with defined destinations was termed Voyage Planning (VP) [13]. BAP refers to allocating the buses shown in Table 5 to the student groups determined by the BSS shown in Table 2 above, and VP refers to determining the order for the buses to visit the bus stops and schools as shown in Table 6. The results shown in Table 5 and Table 6 show that all the student groups were to be allocated to bus 1 (B1) and that the bus was to move to BS3 → BS4 → BS1 to pick up all the students to go to SC1, SC2 and visit SC1 → SC2 in the order of precedence in order to drop off the students at their schools as shown in Figure 6.

---

**Table 5. Example of BAP**

<table>
<thead>
<tr>
<th>Bus stops</th>
<th>BS1</th>
<th>BS2</th>
<th>BS3</th>
<th>BS4</th>
<th>BS5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Schools</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SC1</td>
<td>B1</td>
<td>B1</td>
<td>B1</td>
<td>B1</td>
<td></td>
</tr>
<tr>
<td>SC2</td>
<td>B1</td>
<td>B1</td>
<td>B1</td>
<td>B1</td>
<td></td>
</tr>
</tbody>
</table>

**Table 6. Example of VP**

<table>
<thead>
<tr>
<th>Visiting place</th>
<th>Bus Stop</th>
<th>School</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>BS1</td>
<td>BS2</td>
</tr>
<tr>
<td></td>
<td>BS3</td>
<td>BS4</td>
</tr>
<tr>
<td></td>
<td>SC1</td>
<td>SC2</td>
</tr>
<tr>
<td>B1</td>
<td>3rd</td>
<td>1st</td>
</tr>
<tr>
<td>B2</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Figure 6. Result of BRS**

According to the classification by Park et al. [19], the SBRP considered in this study, as shown in Table 7 below, provided that, since there was no classification of bus departure indexes in previous papers, the classification was added in this study.
Table 7. SBRP Type of this Research

<table>
<thead>
<tr>
<th>No. of Schools</th>
<th>No. of Depots</th>
<th>Urban or Rural</th>
<th>Mixed Loads</th>
<th>Fleet Mix</th>
<th>Objectives</th>
<th>Constraints</th>
</tr>
</thead>
<tbody>
<tr>
<td>Multiple</td>
<td>Multiple</td>
<td>Urban</td>
<td>Yes</td>
<td>HT</td>
<td>TBD, N, TSD</td>
<td>C, MRT, TW</td>
</tr>
</tbody>
</table>

4. Development of a Bus Stop Selection Algorithm

In this chapter and the following ones, details of the algorithm for the BSS and BRS mentioned earlier are addressed. First, the BSS is the problem of allocating students to nearby bus stops, and the results of these assignments greatly affects the BRS [6]. Although it would be good to simply allocate students to the nearest bus stops for students, for the efficiency of the whole SBRP, the BSS should be implemented while considering the results of the BRS also. Therefore, in this study, the minimization of students' travel distances was not defined as an objective function but the minimization of the number of student groups was defined as an objective function in solving the problem, and to this end, an algorithm using the following processes was developed.

Step 1) Among the students, those who have only one bus stop within the defined travel distance are allocated to the bus stop.
Step 2) Among the students, those who have two bus stops within the defined travel distance are allocated to the bus stop to which the nearest student groups belong among those student groups that have been already allocated to bus stops in Step 1.
Step 3) Students who have not been allocated to bus stops in Steps 1 and 2 are allocated to the nearest bus stops within the defined travel distance.
Step 4) Stop

To check the performance of the above rules, arbitrary problems were created in order to conduct an experiment. Four challenges were made with different numbers of schools, bus stops, students and buses at locations not overlapping with each other as shown in Table 8, and the results of the experiment were checked.

Table 8. Comparison of BSS Results by Problem Sizes and Objective Functions

<table>
<thead>
<tr>
<th>SC</th>
<th>BS</th>
<th>ST</th>
<th>Minimize Student Groups</th>
<th>Minimize Walking Distance</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>SG</td>
<td>BS</td>
<td>Distance</td>
<td>SG</td>
</tr>
<tr>
<td>2</td>
<td>6</td>
<td>3</td>
<td>13.15</td>
<td>10</td>
</tr>
<tr>
<td>4</td>
<td>26</td>
<td>8</td>
<td>31.92</td>
<td>29</td>
</tr>
<tr>
<td>5</td>
<td>55</td>
<td>18</td>
<td>64.66</td>
<td>58</td>
</tr>
<tr>
<td>10</td>
<td>137</td>
<td>35</td>
<td>127.3</td>
<td>149</td>
</tr>
<tr>
<td>10</td>
<td>252</td>
<td>48</td>
<td>300.0</td>
<td>307</td>
</tr>
</tbody>
</table>

As shown in the results set forth in Table 8, it can be seen that, between different objective functions, although the differences in the students' total travel distances were not large, the differences in the numbers of student groups (SG) and bus stops (BS) that affected the BRS were large. Therefore, in this study, the minimization of the number of student groups was selected as an objective function of the BSS and it was determined to use these results in the BRS.
5. Development of a Bus Routing and Scheduling Algorithm

As explained in Chapter 1, the BRS is the problem of determining the operation routes of buses and the order of visitation within the routes. Since this problem is an NP-Hard problem of which the search domain exponentially increases as its size grows making it difficult to find solutions, heuristic approaches have been more frequently applied to this problem rather than exact approaches, and in this study too, genetic algorithms [2, 11, 12] which are a sort of heuristic approach, were applied.

Genetic algorithms are a methodology that is the most widely used among diverse meta-heuristic algorithms and is applied to diverse routing problems. Unlike many other optimization methods, genetic algorithms are implemented with groups of candidate entities and each entity is generally called a chromosome, which consists of genes [4]. The genetic algorithm allocates values to individual entities in groups based on the objective function of the problem and improves solutions based on the principle of the survival of the fittest by comparing the values with each other.

To use genetic algorithms, the characteristics of the problem should be analyzed first and then the presentation method, objective function, method to construct the initial population, genetic operators (crossover operator and mutation operator) and genetic parameters suitable to the problem should be determined. The processes of genetic algorithms include the construction of the initial population, selection of entities to which genetic characteristics are to be carried over as the next generation from the population, and application of the genetic operator to these entities to form the group of the next generation. In the selection, entities with excellent characteristics are selected through the fitness function and this process is repeated until the given result condition is satisfied.

As shown in Figure 7, the procedure of the genetic algorithm used in this study based on these processes of the genetic algorithm. Each of the processes will be explained in detail in the next section and thereafter.

![Figure 7. Procedure of Genetic Algorithm](image-url)
5.1. Chromosome Presentation

To solve a BRS problem with genetic algorithms, first, the solution of the problem should be expressed with chromosomes. The chromosome presentation of the proposed genetic algorithm is largely divided into two types. The first chromosome is a bus allocation chromosome for allocating buses to student groups and the second chromosome is a voyage chromosome for determining the orders of visitation for intermediate points for individual buses. First, the bus allocation chromosome is composed by repeating the numbers of buses to student groups allocated to bus stops in the BSS as shown in Table 2, Table 3 and Table 4 above. One gene means that the student group was allocated to the bus where it is expressed. For instance, a problem with 5 bus stops, 3 schools and 2 buses is expressed as shown in Figure 8. In this example, the numbers of two buses were allocated to a total of 15 student groups. For instance, it is expressed that the student group (SG24) at bus stop 4 (BS4) to go to school 2 (SC2) was allocated to bus 2.

\[
\begin{array}{cccccc}
BS_1 & BS_2 & BS_3 & BS_4 & BS_5 \\
SC_1 & 1 & 1 & 1 & 2 & 2 \\
SC_2 & 1 & 1 & 1 & 2 & 2 \\
SC_3 & 1 & 1 & 1 & 1 & 1 \\
\end{array}
\]

**Figure 8. Bus Allocation Chromosome**

Once student groups have been allocated to buses as seen, individual buses gain defined destinations that they must visit. As shown in Figure 8, bus 2 (B2) was allocated to bus stops 4 and 5 (BS4, BS5) and thus the bus had to visit schools 1 and 2 (SC1, SC2) along with the two bus stops. However, the order of visitation has not been determined yet. To determine this order, a voyage chromosome, which can be expressed as shown in Figure 9, was used. In this case, the order of visitation should be determined for buses to visit bus stops so that students can get onto the buses earlier than the school arrival time required, although bus stops and schools can be mixed up in the order.

\[
\begin{array}{cccccccc}
B_1 & BS_1 & BS_2 & BS_3 & BS_4 & BS_5 & SC_1 & SC_2 & SC_3 \\
B_2 & 2 & 1 & 3 & 4 & 5 & 7 & 6 & 8 \\
\end{array}
\]

**Figure 9. Voyage Chromosome**

In the example shown in Figure 9, bus 2 (B2) should visit bus stop 5 (BS5) → bus stop 4 (BS4) to pick up students and visit school 1 (SC1) → school 2 (SC2) to drop off the students.

5.2. Objective Function

In SBRPs, diverse objective functions are used, such as the number of buses used (N), the sum of the travel distances of the buses (TBD) and the sum of the travel distances of the students (TSD) [19]. The results of a problem vary with its objective function, and selecting objective functions suitable for the environments to which they are to be applied is seen as a very important decision in order to efficiently solve SBRPs. In this study, an algorithm was developed to enable solving problems with all three (N, TBD and TSD) as objective functions and the paper was prepared using the TBD as an objective function.

5.3. Initial Solution Generation

In genetic algorithms, initial solutions greatly affect the final solutions. Although genetic algorithms arbitrarily generate initial solutions and use them in general, since SBRPs are large, it was ensured that populations would be created and evolved with good
initial solutions so that better solutions could be obtained within a shorter time. In addition, the diversity of the solutions is important when creating populations. Diverse initial solutions were created in order to reduce the phenomenon of Premature Convergence.

To this end, a method to allocate buses to schools first and a method to allocate buses to bus stops first when creating Bus Allocation chromosomes were designed and these methods were compared with each other. Since better results were produced by the method to allocate buses to bus stops first, this method was selected and used. The method to allocate buses to bus stops first is a method to allocate buses to bus stops using Bus Allocation chromosomes. As shown in Figure 10, bus stops were randomly selected and student groups were allocated to the same bus. The method for allocating buses to bus stops first has the following processes and the method for allocating buses to schools before bus stops, which is contrary to the former, has the same processes except that bus stops K in the processes are replaced by schools SC.

Step 1) $c=1$
Step 2) Select a random bus stop $K$ from a set of bus stops $BS = \{1, 2, \ldots, K\}$. If set $BS = \emptyset$, go to Step 7
Step 3) Create a set of buses $B$ which can be allocated to the student group of school $C$ of bus stop $K$. If $B = \emptyset$, issue an error message and go to Step 7
Step 4) Select buses to which students have already been allocated from set $B$. If there is no bus to which students have been allocated, randomly select a bus
Step 5) Allocate the student group of school $C$ of bus stop $K$ to the bus
Step 6) If $c<SC$, define $c=c+1$ and go to Step 3, if not, remove bus stops $K$ from the set of bus stops $BS$ and go to Step 1
Step 7) Stop

Though the results may vary with the sizes or characteristics of the problems, when the two methods were tested 50 times using a problem with two departure places, 5 schools, 18 bus stops, 100 students and 6 buses, the method to allocate buses by bus stops showed solutions of an average of 123.86 miles, and the method to allocate buses by schools showed solutions of an average of 153.56 miles. In the cases where the methods searched the best solutions respectively were compared, as shown in Figure 11, the methods showed clear processes of the evolution of solutions and it could be seen that when buses were allocated by bus stops, good initial solutions were obtained and thus we were able to search for better solutions than when using the method to allocate buses by schools.

![Figure 10. Allocation Method by Bus Stops](image-url)
Voyage chromosomes were also defined to determine the order of visitation by selecting the destination nearest to the depot so that good initial solutions could be obtained, and the procedure for determining the order of visitation was as follows.

Step 1) \( b = 1 \)
Step 2) Select bus stop \( k \) which is the nearest to the bus depot from the set of bus stops \( BS = \{1, 2, ..., K\} \) to be visited by bus \( b \) to assign it as the first visitation point and delete \( k \) from set \( B \)
Step 3) Select bus stop \( k' \) which is the nearest to bus stop \( k \) to assign it as the next visitation point and remove \( k' \) from set \( BS \)
Step 4) If set \( BS = \emptyset \), go to the next Step, if not, define \( k = k' \) and go to Step 3
Step 5) Select school \( c \) which is the nearest to bus stop \( I \), which was visited the last from the set of schools \( SC = \{1, 2, ..., C\} \) to be visited by bus \( b \) to assign it as the next visitation point and delete \( C \) from set \( SC \)
Step 6) Select school \( C' \) which is the nearest to school \( c \) and assign it as the next visitation point and remove \( c' \) from set \( SC \)
Step 7) If set \( SC = \emptyset \), go to the next Step, if not, define \( c = c' \) and go to Step 6
Step 8) If \( b < B \), define \( b = b + 1 \) and go to Step 2
Step 9) Stop

When the above method was applied to the problem used in the bus allocation chromosomes, the value obtained was 102.04 miles on average, which was better than the value 123.86 miles obtained before applying the method, and when the results of searching the best solution were compared, as shown in Figure 12, further processes for changes into better solutions were also shown.
5.4. Crossover Operator

In genetic algorithms, operators are necessary in order to maintain the good order possessed by the chromosomes while improving the solutions gradually during an evolutionary process [18]. One of these is the crossover operator, which should be designed to be able to receive good genes from the parents in order to produce better solutions. The conventional 1-point crossover operator introduced by Holland [12] refers to exchanging genes between the parents beginning from a point randomly selected from the chromosomes, as shown in Figure 13, and the 2-point crossover operator refers to the random selection of two points and the exchange of genes between the parents based on these two points as shown in Figure 14. Although there are partially-expanded operators such as the k-point crossover operator and uniform crossover operator [23], in this study, only the 1-point and 2-point crossover operators were applied and the results were compared. Since the results shown in Table 9 were obtained through these experiments, the 2-point crossover operator was selected.

![Figure 12. Comparison of Solution Improvement Change by Generation Methods of Voyage Chromosome](image)

![Figure 13. 1-point Crossover Operator](image)
5.5. Mutation Operator

As with crossover operator, mutation operator is also important in genetic algorithms, and this is used to create changes in chromosomes in order to maintain diversity in groups. As with crossover operator, mutation operator has been also studied in diverse forms and in this study, neighborhood search-based mutation operator [7] that selects three points and exchange them with each other to select the best results was used. As shown in Figure 15, if three points are selected from the parent (Case 1) and they are called ABC, five children (Cases 2-6) with the orders ACB, BAC, BCA, CAB, CBA can be created and a method to compare all of them and select good solutions is called a 6-case mutation operator, while a method to compare the children only and select good solutions is called a 5-case mutation operator [18]. In this study, these two methods were used in combination in order to obtain good solutions.

![Figure 14. 2-point Crossover Operator](image)

<table>
<thead>
<tr>
<th>Table 9. Comparison Result of Crossover Operators</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Crossover Operator</strong></td>
</tr>
<tr>
<td>1-point Crossover Operator</td>
</tr>
<tr>
<td>2-point Crossover Operator</td>
</tr>
</tbody>
</table>
In a study by Kang [13], a method to select three random points and another method to select three successive points when selecting three points were compared and the method to select three successive points was selected as it produced better results. However, in this study, the method to randomly select three points produced better values as shown in Table 10 and thus it was applied.

**Figure 15. Neighborhood Search-based Mutation Operator**

**Table 10. Comparison Result of Mutation Operators**

<table>
<thead>
<tr>
<th>Mutation operating method</th>
<th>Minimum value</th>
<th>Average value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Selecting 3 random points</td>
<td>94.87</td>
<td>101.64</td>
</tr>
<tr>
<td>Selecting 3 successive points</td>
<td>96.72</td>
<td>101.82</td>
</tr>
<tr>
<td>Using the two methods in combination</td>
<td>97.96</td>
<td>102.27</td>
</tr>
</tbody>
</table>

5.6. Repair Process of Solutions

In the case of the genetic algorithm developed in this study, infeasible solutions may occur after implementing operations, for example in cases where the limit of bus riding time is exceeded or when buses arrive at school after the required arrival time. Therefore, repair process is necessary in order to change these infeasible solutions into feasible solutions. In this study, repair process was added for selecting buses that exceeded the riding time limit or for those that arrived at the school after the required time in order to allocate students from buses that are full of students into other buses in the order of the
magnitude of the number of students and also to allocate students from buses arriving at schools after the required time into other buses so that they may arrive on time.

In addition, a method to repair the created voyage chromosomes to have shorter routes was added so that good solutions could be found faster. The voyage chromosome as shown in Figure 16 can be created using the process of evolution and, with this chromosome, buses will move in the order of DP₁ → BS₁ → BS₂ → SC₄ to unnecessarily go round. Therefore, this order should be changed to DP₁ → BS₂ → BS₃ → SC₄ as shown in Figure 17. In this study, a method to revise these inefficient voyage chromosomes was added to enhance the performance of the algorithm. As shown in Figure 18, the algorithm was set to evaluate the order of bus stops located between the departure place and the individual schools in order to search for the shortest route and to prevent a reduction in the search speed, the algorithm was set to select up to 6 bus stops at the maximum to compare. In the case of sections where there were more than 6 bus stops, sub-sections of 6 bus stops in the sections were randomly selected and compared. When this method was not applied, the minimum value was 96.87 and the average value was 101.88, however this changed to a minimum value of 94.22 and an average value of 101.08 when the method was applied, which means that better values were obtained using the method.

Figure 16. Before Repair

Figure 17. After Repair

Figure 18. Route Repair Method of Voyage Chromosome

5.7. Selection Method and Elitism

As a selection method, seed selection was used [18]. The seed selection was made by introducing the individual selection and strong individual preserving methods that are commonly used for livestock proliferation in genetic algorithm evolution processes. Of two parents, the entity corresponding to the father is selected from excellent entities within the defined ranking and the remaining parent mother is randomly selected from the whole group. These are used as parents and then returned to the entity group so that they can be used again. The next generation is newly composed using the method of selecting from the present generation and the genetic
operators. After creating new entities to the same number as that of the population, elitism is applied to replace bad entities with good entities of the same number.

5.8. Performance Evaluation

5.8.1. Parameter Experiment

In order to evaluate the performance of the algorithm developed through this study, several experiments were conducted. First, parameter selection for searching for good solutions in the minimal time was tested. At this time, mutation rates were set to 0.05 through 0.25, crossover rates were set to 0.7 through 0.95 and intervals of 0.05, the number of populations was set to 50 and the number of generations was set to 50. Fifty tests were conducted for each parameter. Table 11 shows the results of the parameter experiments (the crossover rate was 0.95 and the mutation rate was 0.1) where the best values were searched, selected and applied to experiments thereafter.

Table 11. Results of Parameter Experiments for Crossover Rate and Mutation Rate

<table>
<thead>
<tr>
<th>Crossover rate</th>
<th>0.05</th>
<th>0.1</th>
<th>0.15</th>
<th>0.2</th>
<th>0.25</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.7</td>
<td>97.65</td>
<td>102.32</td>
<td>95.35</td>
<td>102.12</td>
<td>96.44</td>
</tr>
<tr>
<td>0.75</td>
<td>95.65</td>
<td>102.04</td>
<td>95.33</td>
<td>102.11</td>
<td>97.52</td>
</tr>
<tr>
<td>0.8</td>
<td>96.62</td>
<td>102.03</td>
<td>97.92</td>
<td>102.56</td>
<td>95.80</td>
</tr>
<tr>
<td>0.85</td>
<td>97.73</td>
<td>101.60</td>
<td>95.58</td>
<td>101.70</td>
<td>96.89</td>
</tr>
<tr>
<td>0.9</td>
<td>96.63</td>
<td>101.80</td>
<td>94.09</td>
<td>101.70</td>
<td>98.68</td>
</tr>
<tr>
<td>0.95</td>
<td>95.48</td>
<td>102.12</td>
<td><strong>93.46</strong></td>
<td>101.71</td>
<td>95.68</td>
</tr>
</tbody>
</table>

Elitism experiments were also conducted 50 times for each case where 10, 20 or 30 entities were selected and the results shown in Table 12 were obtained and the 20 entity cases that demonstrated the best value were selected and applied to experiments thereafter.

Table 12. Results of Parameter Experiments for Elitism Size

<table>
<thead>
<tr>
<th>Elitism</th>
<th>Minimum</th>
<th>Average</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>95.34</td>
<td>100.39</td>
</tr>
<tr>
<td>20</td>
<td><strong>92.47</strong></td>
<td><strong>100.20</strong></td>
</tr>
<tr>
<td>30</td>
<td>95.04</td>
<td>101.13</td>
</tr>
</tbody>
</table>

5.8.2. Performance Experiment

To date, existing research on SBRP has been undertaken independently, so there is no general SBRP model and the benchmark problem sets based on practical application don’t exist [19]. Therefore, in this research, in order to evaluate the performance of the genetic algorithm, several random problems similar to real problems were created and the performance was evaluated. At this time, the numbers of bus stops and student groups were determined from those obtained from the results of the BSS experiments set forth in Chapter 4. The number of depots (DP) was randomly selected among 1 or 2 and the number of buses (B) was randomly selected among numbers ranging from 2 to 12. The riding capacity of each bus was set to 20 students, the depot departure time was set to 6:00 for all the buses, and the time to arrive at the schools was set to 7:30 for all the buses. The
maximum bus riding time for each student was set to 3,600 seconds (1 hour) or 5,400 seconds (1 hour 30 minutes) during the evaluation. The population in the genetic algorithm was set to 300, the number of generations was set to 100, and the parameters were set to the values obtained in the parameter experiments when conducting the performance experiments.

Table 13. Results of Performance Experiments

<table>
<thead>
<tr>
<th>DP</th>
<th>SC</th>
<th>ST</th>
<th>B</th>
<th>MRT</th>
<th>Result of the BSS</th>
<th>Result of the BRS with Minimal TBD</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>SG</td>
<td>BS</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>20</td>
<td>2</td>
<td>3600</td>
<td>6</td>
<td>3</td>
</tr>
<tr>
<td>1</td>
<td>4</td>
<td>50</td>
<td>3</td>
<td>3600</td>
<td>26</td>
<td>8</td>
</tr>
<tr>
<td>1</td>
<td>5</td>
<td>100</td>
<td>6</td>
<td>3600</td>
<td>55</td>
<td>18</td>
</tr>
<tr>
<td>2</td>
<td>5</td>
<td>100</td>
<td>6</td>
<td>3600</td>
<td>55</td>
<td>18</td>
</tr>
<tr>
<td>1</td>
<td>10</td>
<td>200</td>
<td>12</td>
<td>5400</td>
<td>137</td>
<td>35</td>
</tr>
<tr>
<td>2</td>
<td>10</td>
<td>200</td>
<td>12</td>
<td>5400</td>
<td>137</td>
<td>35</td>
</tr>
<tr>
<td>1</td>
<td>10</td>
<td>500</td>
<td>30</td>
<td>5400</td>
<td>252</td>
<td>48</td>
</tr>
<tr>
<td>2</td>
<td>10</td>
<td>500</td>
<td>30</td>
<td>5400</td>
<td>252</td>
<td>48</td>
</tr>
</tbody>
</table>

As shown in Table 13, the results of the experiments could be identified when the minimization of the travel distances (TBD) of buses had been defined as an objective function and the number of buses (N) and the total riding time of students could be identified together.

5.8.3. Case Study

In order to check the possibility of applying the algorithm developed in this study, real data is surveyed and experiments were conducted using the data. The problem used at that time was a real situation in City B, State A in the USA and based on the results of an analysis, the problem was a Rural SBRP that did not require a BSS. Since buses stopped at the houses of individual students, the houses of the students were effectively bus stops (BS) and at each of the students’ houses, multiple students could get on the bus instead of just one. It was a problem of transporting a total of 26 students to 6 schools using 3 buses with the buses departing from one depot. The capacity of each bus was 40 students and the buses had to arrive at the schools by 7:30am.

Since this problem was real compared to the problems in the experiments mentioned above, the locations of all the students, bus stops and schools were grasped through the API (Application Programming Interface) of Google Maps to identify the distances between them. Stoppage times at schools and bus stops were set to be the same as those in the definition in Chapter 3 and the speed of the vehicles was set to 30 mph, unlike the experiments mentioned above, taking into consideration that the environment was rural.

When the voyages were manually planned previously, all the buses repeatedly visited the downtown area and several schools were visited by all the buses, as shown in Figure 19, Figure 20 and Figure 21. However, the results of the genetic algorithm minimized the overlapping routes as shown in Figure 22, Figure 23 and Figure 24 and there were fewer schools repeatedly visited, which means that unnecessary bus journeys could be avoided.
Table 14. Comparison Results between Algorithm and Current Method (Manual)

<table>
<thead>
<tr>
<th>Objective function</th>
<th>TBD minimization</th>
<th>TSD minimization</th>
<th>Current method (manual)</th>
</tr>
</thead>
<tbody>
<tr>
<td>TBD</td>
<td>112.78</td>
<td>110.57</td>
<td></td>
</tr>
<tr>
<td>TSD</td>
<td>53,651</td>
<td>53,651</td>
<td></td>
</tr>
</tbody>
</table>

Table 14 shows a comparison between the results when the objective function was set to TBD or TSD minimization and the results of the current manual work. When the objective function was set to TBD minimization, the travel distance could be reduced by up to 59.8% compared to the results of the current manual method. When the objective function was set to TSD minimization, the travel distance could be reduced by up to 19.7% compared to the results of the current manual method.

Through this experiment, we were able to identify that the developed algorithm could be applied to Rural SBRPs as well and that there are huge differences between manual methods and using the algorithm even in the case of small problems. Through this, we were able to identify not only the necessity of studies of SBRP but also the high possibility of utilizing them.

6. Conclusion

In this study, an algorithm that can solve three of the five sub-problems of SBRP, including Bus Stop Selection, Bus Route Generation and Route Scheduling was developed. This algorithm solves complicated SBRP that include all the diverse constraints and assumptions, such as vehicle capacity constraints, student riding time constraints, school arrival time constraints, heterogeneous buses, multi-depots and mixed loading and its performance was identified through diverse experiments. In particular, by applying it to real problems to show the inefficiency of the existing manual-method results, the necessity of SBRP studies and the possibility of utilizing them were also examined.
However, since the real problem to which the algorithm was applied was rural SBRP, it should be necessary to survey urban SBRP models and apply the algorithm to check the possibility of applying the developed BSS method to reality. In addition, it is also necessary to approach constraints considering disabled students that have recently become issues in many study areas.

Since this study developed an algorithm based on genetic algorithms which use the heuristic approach, the results drawn may not be optimal solutions. Therefore, it is necessary to accurately test the algorithm’s performance by checking its accuracy against optimal solutions through obtaining optimal solutions in problems of a size that can be solved by exact approaches, such as IP (Integer Programming) and comparing them with the results of the algorithm.

References


Authors

Moohong Kang, he received his BS, MS, Ph.D. in management information system from Dong-A University, South Korea. He was a visiting scholar of University of Arkansas at Little Rock in 2010 and has been working for Korea Maritime Institute from 2011. He is interested in vehicle routing problem, agent systems, cargo handling systems, and logistics systems.

Sung-Kwan Kim, he received his BS from Kyunghee University, South Korea, MS from University of Missouri at Columbia, and Ph.D. from University of Nebraska. He is an associate professor at department of management, University of Arkansas at Little Rock. He is interested in knowledge management, business information systems, and database management systems.

Joe T. Felan, he received his BS, MS from Baylor University, and Ph.D. from University of South Carolina. He is an associate professor at department of management, University of Arkansas at Little Rock. He is interested in business strategy, production and operations management, and manufacturing systems.

Hyung Rim Choi, he received his BS from Seoul National University, South Korea, and MS, Ph.D. from Korea Advanced Institute of Science and Technology. He is an associate professor at department of management, Dong-A University. He is interested in port and logistics systems, business information systems and Internet of Things Systems.

Minje Cho, he received his BS, MS, Ph.D. in management information system from Dong-A University, South Korea. He is an associate professor at department of management, Dong-A University. He is interested in port and logistics systems, business information systems and Internet of Things Systems.