IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received July 30, 2021, accepted August 5, 2021, date of publication August 9, 2021, date of current version August 17, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3103640

Multi-Depot Split-Delivery Vehicle Routing Problem

HYUNPAE LIM[1], GYU M. LEE[2], AND IVAN KRISTIANTO SINGGIH[3]
[1]School of Mechanical, Industrial and Manufacturing Engineering, Oregon State University, Corvallis, OR 97331, USA
[2]Department of Industrial Engineering, Pusan National University, Busan 43241, Republic of Korea
[3]Department of Industrial and Systems Engineering, Korea Advanced Institute of Science and Technology, Daejeon 34141, Republic of Korea

Corresponding author: Gyu M. Lee (glee@pnu.edu)

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. 2018R1A2B3008890).

ABSTRACT The rapid advancements in information technologies and globalization change the way of distributing goods to customers. Many enterprises have multiple factories, warehouses, and distribution centers and strive for competitive efficiency in the distribution operations to minimize transportation costs. This study proposed the mixed-integer programming (MIP) model for the multi-depot split-delivery vehicle routing problems (MDSDVRPs) with hetero vehicles, allowing multiple visits to a customer. A genetic algorithm (GA) with a novel two-dimensional chromosome representation has been proposed with dynamic mutation policies. The process parameters of the proposed GA are optimized using the Taguchi method. The proposed algorithms showed the benefits of split-delivery in MDSDVRPs and showed the competitive performance even for the classical single-depot vehicle routing problems with no split-delivery.

INDEX TERMS Vehicle routing problem (VRP), multi-depot split-delivery VRP, genetic algorithm (GA), taguchi method.

I. INTRODUCTION

Logistics and distribution systems change our lives rapidly due to the changes in customer demands and enabling technologies while saving significant distribution costs. However, traditional logistics and distribution operations exhibit an underlying limitation in modeling and implementing shared services from multiple manufacturers, warehouses, and depots.

The vehicle routing problem (VRP) is a problem in which a set of routes for a fleet of vehicles based at one or several depots must be determined for a certain number of geographically dispersed customers. The objective of the VRP is to minimize the total distance traveled by all vehicles, which can be considered transportation or delivery costs. Recently, with the increase in fuel prices, the importance of minimizing delivery costs has been emphasized as a critical factor that can reduce the total costs of production and distribution. In recent decades, various engineering areas produce many variants of VRPs to utilize the theory and to optimize their systems with the advances in electronics and new technologies, e.g., VRPs for electric vehicles ([1]–[3]) and drones ([4], [5]).

The VRP was initially introduced by Dantzig and Ramser [6], and it has been widely studied thereafter. They described a real-life application concerning the delivery of gasoline to service stations. Fisher [7] describes the problem as finding the efficient use of a fleet of vehicles that must make several stops to deliver passengers or goods. The term ‘‘customer’’ is used to denote the stops to make. Every customer has to be assigned to precisely one vehicle in a specific order.

A particular case of the VRP arising when only one vehicle is available at a depot and no additional operational constraints are imposed, i.e., traveling salesman problem (TSP), is extensively described by Lawler *et al.* [8], Knox [9], Barvinok *et al.* [10], Engebretsen and Karpinski [11], Ouaarab *et al.* [12], and Mahi *et al.* [13]. The TSP has one vehicle, one depot, and multiple customers. The customer demands are assumed to be satisfied with one visit to each customer by a vehicle. Another version of the VRP is capacitated VRP (CVRP). The CVRP has n customers and a single depot with several vehicles of an identical capacity. The vehicles must accomplish the delivery with the minimum

The associate editor coordinating the review of this manuscript and approving it for publication was Kuo-Ching Ying.

total travel cost, where the cost is the distance d_{ij} from nodes i to j ($i, j \in \{0, 1, \ldots, n\}$), where 0 stands for a single depot and n is the number of customers. The applications of the CVRP can be found in Desrosiers *et al.* [14], Osman [15], Laporte [16], Toth and Vigo [17], Lysgaard *et al.* [18], and Alssager *et al.* [19]. Some studies considered heterogeneous vehicles to reduce delivery costs by dispatching appropriate vehicles to the routes according to customer demands ([20]–[22]).

The vast majority of papers have been published on a classical single-depot capacitated VRP (SDCVRP) ([23], [24]) and some others were found dealing with problems known as multiple-depot capacitated VRP (MDCVRP) ([25]–[28]). The MDCVRP is an extension of SDCVRP with vehicles starting from different depots. The MDCVRP has similar constraints to those of the SDCVRP, except for the requirement that each vehicle starts from and finishes the delivery at the same depot. In the MDCVRP, if customers are clustered around depots, then the problem can be modeled as a set of independent SDCVRPs. However, if customers and depots are intermingled, the problem must be modeled as the MDCVRP. The applications of the MDCVRP can be found in [23], [25], [26], and [29]–[32], all using adaptations of classical SDCVRP procedures.

The VRP is one of the combinatorial optimization problems belonging to the non-deterministic polynomial-time hard (NP-hard) class [33] which cannot be solved to optimality within polynomially-bounded computational time [34]. The NP-hardness of VRPs can be proven via "proof by restriction". A problem is NP-complete when we can prove that it contains a known NP-complete problem as a special case [35]. Likewise, we can prove that VRP contains a traveling salesman problem (TSP) when we limit the number of salesmen to one. Since TSP is NP-hard ([36], [37]), then VRP (multiple TSP [38]) is NP-hard as well. Solving NP-hard problems of large sizes using exact solution methods is expensive in computational efforts. It is often impossible due to the limited computational power. Hence, the extensive efforts in studying the approximation algorithms or heuristics were justified by many researchers. The optimal solutions for NP-hard problems can be obtained by using those algorithms and matching their solutions with the theoretic bounds ([39]–[42]).

Many different approaches have been developed to solve this NP-hard problem. The branch-and-bound method has been used for small problems with only a few customers [43]. Most approaches for large problems are based on heuristics, i.e., approximation algorithms aiming to find good feasible solutions quickly [44]. Many models and algorithms have been proposed to obtain the optimal or approximate solutions for different variants of the VRP. A thorough classification was given in Desrochers *et al.* [45]. Laporte and Nobert [46] presented an extensive survey devoted to exact methods for VRPs. Other surveys about VRP studies were reported by Laporte [16], Toth and Vigo [17], Bodin *et al.* [33], Christofides *et al.* [47], Magnanti [48], Christofides [49], Golden *et al.* [50], Laporte [51], Ghorbani *et al.* [52], and Anuar *et al.* [53]. They could be divided into two main classes: classical heuristics, mostly between 1960 and 1990, and metaheuristics from 1990 ([44], [51]).

This paper addresses the VRP, which has heterogeneous vehicles departing from multiple depots, allowing split deliveries to customers. This multi-depot split-delivery VRP (MDSDVRP) was firstly studied by Lim [54] and then mentioned by Gulczynski *et al.* [55] independently. The research by Gulczynski *et al.* [55] has been cited more than 100 times. They defined the MDSDVRP and developed an integer programming-based heuristic. Their heuristic determines the reduction in traveled distance, allowing split deliveries among vehicles based at the same depot and different depots.

The study completed earlier by Lim [54] has not been published until now. It defined the MDSDVRP and proposed the first mixed-integer programming model for MDSDVRP. The model is solved optimally using a CPLEX solver. The addition of this study to MDSDVRP literature may enrich future research.

The classical heuristics can be divided into three groups: construction methods, two-phase methods, and improvement methods [56]. Construction methods gradually build a feasible solution by selecting arcs based on minimizing cost. The two-phase method divides the problem into two stages: clustering customers into feasible routes disregarding their order, and constructing routes. One of the two-phase methods is the sweep algorithm in Laporte *et al.* [44]. Improvement methods start with a feasible solution and improve it by exchanging arcs or nodes within or between the routes. The local search algorithms developed by Aarts and Lenstra [57] belong to the improvement heuristics. The advantage of classical heuristics is that they have a polynomial running time [44]. When using them, one can provide good solutions within a reasonable time [58]. On the other hand, they only perform a limited search in the solution space. Therefore, they have a risk of resulting in a local optimum.

During the past few decades, there have been many attempts to solve VRPs quickly and effectively by using metaheuristics such as tabu search (TS), simulated annealing (SA), genetic algorithm (GA), and ant colony optimization (ACO) algorithm ([44], [59], [60]). Braekers *et al.* [61] reported that metaheuristic was the most applied method to solve VRP cases. Metaheuristics were used more often than exact methods, classical heuristics, and simulation approaches. The TS and the SA are local search-based algorithms that move from one solution to another in the neighborhood until a stopping criterion is met. Many different TS heuristics have been proposed with unequal success. Rochat and Taillard [62] used the TS heuristic to solve some benchmark VRPs. Osman [63] obtained similar results using the SA. Unlike TS and SA, GA maintains a population of good solutions that are recombined to produce new solutions. A considerable amount of research on the GA has recently been done to solve many variants of VRPs, including VRPs with time windows (VRPTW)

([64], [65]), where each customer has a time window for which the vehicle has to arrive. Berger and Barkaoui [65] presented a hybrid GA to solve the CVRP. Their HGA uses two populations of solutions that periodically exchange some chromosomes, which are the feasible solutions to the CVRP. The algorithm is competitive in comparison to the best TS heuristics. However, Renaud *et al.* [26] reported that such heuristics require substantial computing times and several parameter settings. When dealing with VRP, the GA was applied more often than most local search-based metaheuristics [66], and GA was the most applied method among population-based metaheuristics ([66], [67]). It could be concluded that the GA has been proven to perform effectively compared with other methods used for solving VRPs.

The GA is a randomized, global search algorithm that solves problems by imitating genetic processes observed during natural evolution and has been extensively used to tackle many combinatorial problems, including various VRPs ([60], [68]–[70]). In the GA, a population of chromosomes (individuals) or solutions is maintained during the evolution, in which solution evaluation, selection, crossover, and mutation occur. The quality of the solution is evaluated by its fitness function, which represents an individual's survivability in the wild. This fitness determines the individuals for the crossover or mating, which produces offsprings for the next generation. The mutation is also used to prevent local convergence by diversifying the search space. The average quality of the population gradually improves as new and better solutions are generated and worse solutions are removed. Analogous to biological processes, offspring with relatively good fitness levels are more likely to survive and reproduce, expecting that fitness levels throughout the population may improve as they evolve. More details can be found in Reeves [71].

Most prior research on the VRP has considered capacitated vehicles from a depot while only allowing a visit to each customer. Vehicles dispatched from a single depot must deliver the required amount to customers, satisfy all demands, and finally return to the depot. The vehicle routes are designed so that each customer is visited only once by precisely one vehicle, and the total demands of all customers on a particular route must not exceed the vehicle's capacity. However, the constraints of homogeneously capacitated vehicles, a single depot, and one allowed visit to customers are unrealistic in the real world. Therefore, this research presents a genetic algorithm (GA) to find all routes that minimize the total distance traveled by heterogeneous vehicles from multiple depots with split deliveries. Split deliveries enable multiple visits to a customer to satisfy his/her demand. If multiple visits are allowed, it is conjectured that they may reduce the number of vehicles, travel distances, and green gas emissions to satisfy the customer demands.

Assume that there are five customers, each of whom demands a little more than half of the homogeneous vehicle's capacity. To satisfy all demands by only one visit to each customer, 5 vehicles are necessary. Using split deliveries, only three vehicles might be needed to solve the problem. Shin and Kang [72] introduced the VRP, allowing multiple visits to a customer using a heuristic method. Some other split delivery VRP studies are Archetti *et al.* [73], Berbotto *et al.* [74], Bianchessi *et al.* [75], and Chen *et al.* [76].

This study differs significantly from Gulczynski *et al.* [55]. They used pre-generated routes using a combined algorithm as input data. Then, the integer programming model was used to move some delivery tasks from one route to another. The routes were then updated only considering pre-defined input sets. In contrast to Gulczynski *et al.* [55], this study formulates a complete MDSDVRP mixed-integer programming (MIP) model and optimally solves it using the commercially available solver. This study also proposes a GA to solve the MDSDVRPs of large sizes effectively and efficiently. The proposed GA achieves optimality for small-sized problems and demonstrates its effectiveness by solving single-depot benchmark VRPs to the best-known solutions for comparison purposes.

The objectives of this paper are as follows: 1) to generalize the MDSDVRPs by removing the constraints of the number of depots and the number of visits allowed to each customer; 2) to develop and validate a MIP model to achieve the optimality; 3) to propose and validate a GA to solve effectively and efficiently the medium or large VRPs with heterogeneous vehicles from multiple depots, allowing splits deliveries; and 4) to optimize the parameters of the proposed GA using the Taguchi method and understand the effects of parameters on the performance.

To our best knowledge, this study presents the first MIP formulation for MDSDVRP and a novel GA. An interesting study similar to the problem in this study is Ray *et al.* [77]. Their research is closer to the location problem as a variant of our proposed MDSDVRP. They focused on determining the depot locations in which multi-depot shared commodity delivery by vehicles was completed. They proposed a heuristic to determine where to locate the depots among customer node candidates, while this study focuses on the efficient routing among given depot locations. Our proposed model for the MDSDVRPs can limit the maximum number of visits to each customer, and it provides the decision-makers more control in planning the distribution.

This study also proposes a GA to solve the proposed MDSDVRP with the limited number of vehicle visits to a customer by developing a novel two-dimensional chromosome design. A chromosome representation is designed to tackle the complexity of MDSDVRPs and maintain the efficiency of the GA simultaneously. The chromosome design must have a way to represent the sequence of vehicles visiting customers, and the sequence of customers vehicles visit simultaneously while its crossover and mutation maintain their efficiency. Hence, this study can play an important role in facilitating the research in MDSDVRPs.

An essential characteristic of the GA is its non-deterministic evolution, i.e., it is stochastic in natural decisions, making the GA more robust than other heuristics. This evolution by GA can be determined by a set of parameters, including population size, crossover rate, mutation rate, elitism rate, terminal condition, etc., which significantly affects the performance of GA. This paper optimizes the parameters of the proposed GA using the Taguchi method to make the proposed GA robust to different problems in MDSDVRP.

Our paper is presented as follows. In Section 2, we describe the MDSDVRP completely and propose a MIP model. In Section 3, we introduce the details of the proposed GA. In Section 4, the parameters of the proposed GA have been optimized using the Taguchi method. Section 5 provides the computational results to demonstrate the effectiveness of our GA through some numerical experiments and the comparisons with optimal solutions obtained by the proposed MIP model. Finally, we conclude our study and list future research topics in Section 6.

II. PROBLEM DESCRIPTION AND MATHEMATICAL MODEL

This section presents a MIP model for the MDSDVRP with heterogeneous vehicles from multiple depots, allowing split deliveries. The VRP under consideration can be represented as a network, where nodes are customers or depots, and the links between pairs of nodes are the roads. In a network, there are N customers with known demands $D_i (i = 1, \ldots, N)$, and M depots, each of which has T_m $(m = N + 1, \ldots, N + M)$ vehicles. All vehicles may have homogenous or heterogeneous capacities.

The assumptions for the MDSDVRP in this study are detailed in the following.

- Each vehicle must start and finish its route at a single depot.
- Customer demands must be satisfied by vehicles within the given maximum number of visits.
- The vehicles' capacities are known (homogenous or heterogeneous).
- The sum of unloaded amounts at customers from a vehicle must not exceed the vehicle's capacity.
- The locations of all customers and depots are given.
- The distances between all pairs of locations are known.

The definitions of constants, variables, and sets used in the MIP formulation are given as follows.

Sets:

S_N	Set of customer indices
S_M	Set of depot indices
S	Set of indices for all customers and depots; $S = S_N \cup S_M$
S_{T_m}	Set of all vehicle indices at depot m

Parameters:

N	Number of customers
M	Number of depots
L	Number of customers and depots (L = N + M)
T_m	Number of vehicles at depot m
D_i	Demand of customer i (1 ≤ i ≤ N)
C_{mt}	Capacity of vehicle t from depot m for
d_{ij}	Distance between nodes i and j (1 ≤ i, j ≤ N + M)
V	Maximum number of visits to a customer
B	A large number

Decision Variables:

U_{jmt}	Unloaded amount by vehicle t from depot m at customer j, where $1 \le j \le N$. ($U_{jmt} = 0$ for $N+1 \le j \le N+M$, $U_{jmt} = 0$ for $t \notin S_{T_m}$)
x_{ijmt}	1, if vehicle t from depot m travels from node i to j, where $t \in S_{T_m}$; 0, otherwise.
y_{imt}	auxiliary variable for sub-tour elimination

The MDSDVRP with heterogeneous vehicles from multiple depots, allowing split deliveries to the customers, is formulated as a MIP model in the following:

$$\text{Min} \sum_{i=1}^{L} \sum_{j=1}^{L} \sum_{m=1}^{M} \sum_{t=1}^{T_m} d_{ij} x_{ijmt}$$

Subject to

$$x_{ijmt} = 0, \quad \forall i \neq j \in S_N,\ \forall m \in S_M,\ \forall m \in S_{T_m} \tag{1}$$

$$\sum_{i=1}^{L} x_{ijmt} = \sum_{i=1}^{L} x_{jimt} \quad \forall j \in S_N,\ \forall m \in S_M,\ \forall m \in S_{T_m} \tag{2}$$

$$\sum_{t=1}^{T_m} \sum_{m=1}^{M} \sum_{i=1}^{L} x_{ijmt} \le V \quad \forall j \in S_N \tag{3}$$

$$\sum_{i=1}^{L} x_{iimt} = 0 \quad \forall m \in S_M,\ \forall m \in S_{T_m} \tag{4}$$

$$B \sum_{i=1}^{L} x_{iimt} \ge U_{jmt} \quad \forall j \in S_N,\ \forall m \in S_M,\ \forall m \in S_{T_m} \tag{5}$$

$$x_{ijmt} \le U_{jmt} \quad \forall i, j \in S_N,\ \forall m \in S_M,\ \forall m \in S_{T_m} \tag{6}$$

$$\sum_{t=1}^{T_m} \sum_{m=1}^{M} U_{jmt} = D_j \quad \forall j \in S_N,\ \forall m \in S_M,\ \forall m \in S_{T_m} \tag{7}$$

$$\sum_{j=1}^{N} U_{jmt} \le C_{mt} \quad \forall m \in S_M,\ \forall m \in S_{T_m} \tag{8}$$

$$y_{imt} - y_{jmt} + L x_{ijmt} \le L - 1 \quad \forall i \neq j \in S_N,\ \forall m \in S_M,\ \forall m \in S_{T_m} \tag{9}$$

$$x_{ijmt} = \{0, 1\} \quad \forall i, j \in S_N,\ \forall m \in S_M,\ \forall m \in S_{T_m} \tag{10}$$

$$y_{jmt} > 0, \textit{integer} \quad \forall j \in S_N,\ \forall m \in S_M,\ \forall m \in S_{T_m} \tag{11}$$

The objective function of the MIP model is to minimize the total traveled distance by all vehicles to satisfy all customers' demands. Constraint (1) ensures that each vehicle starts from its origin depot and terminates its route at the same depot. In other words, each vehicle cannot visit the depots other than its origin depot. Constraint (2) ensures that all vehicles visiting a node must leave that node. The number of visits must be the same as the number of departures for each vehicle at each node. It ensures the continuous flow of vehicles in the

C_1	C_2	C_3	C_4	C_5
3	1	4	5	2
V_{11}	V_{12}	V_{21}	V_{11}	V_{12}
0	V_{21}	V_{11}	0	V_{31}
V_{12}	0	V_{13}	0	0

FIGURE 1. The proposed representation of a chromosome, where 5 customers and 3 vehicles in depot 1 (V_{11}, V_{12} and V_{13}), 1 vehicle in depot 2 (V_{21}) and 1 vehicle in depot 3 (V_{31}), allowing up to 3 visits to a customer.

network. Constraint (3) ensures that each customer node can have up to V visits by vehicles to satisfy customer demand. Constraint (4) prevents the loop of any route at a node. Constraints (5) and (6) ensure that if vehicle t from depot m travels from node i to node j, the vehicle should unload U_{jmt} at node j. Constraint (7) ensures that the sum of the unloaded amounts at a customer node j should be the same as its demand. Constraint (8) ensures that the total unloaded amounts of each vehicle over its route cannot exceed the vehicle's capacity. Constraint (9) presents the sub-tour elimination constraint. Constraints (10) and (11) are the binary and integer variable constraints.

III. PROPOSED SOLUTION METHOD

As forementioned, the VRP is known as an NP-hard combinatorial problem. It is difficult to solve even small problems optimally in a reasonable amount of time. The GA has been applied successfully in many combinatorial optimization problems. The GA does not guarantee optimality because of its stochastic nature, but it finds a good near-optimal solution in significantly less time. In this section, the proposed GA implemented for MDSDVRPs in this paper is described in detail.

A. CHROMOSOME REPRESENTATION

A way to encode a solution of the problem into a chromosome has a high impact on the GA's performance. The proposed representation is a 2-dimensional $(V + 1) \times N$ matrix. Its columns represent N customers. The first row contains randomly generated sequences of visiting orders, and the other V rows contain the vehicles visiting each customer. The maximum number of visits to a customer is limited to V. A chromosome representation is illustrated as a 4×5 matrix in Figure 1, where $V = 3$ and $N = 5$.

The way to interpret the chromosome representation in Figure 1 is explained in the following. There are 3 vehicles in depot 1 (V_{11}, V_{12} and V_{13}), 1 vehicle in depot 2 (V_{21}) and 1 vehicle in depot 3 (V_{31}). The vehicles serve five customers, C_1, C_2, C_3, C_4 and C_5. The V_{11} will visit C_1, C_3 and C_4, respectively. The visiting order of the V_{11} depends on the values in the first row. Since the corresponding values in the first row for C_1, C_3 and C_4 are 3, 4 and 5, respectively, the route of V_{11} is $[D_1 - C_1 - C_3 - C_4 - D_1]$, where D_1 represents depot 1. In the same manner, the route of V_{12} is $[D_1 - C_2 - C_5 - C_1 - D_1]$, the route of V_{13} is $[D_1 - C_3 - D_1]$, the route of V_{21} is $[D_2 - C_2 - C_3 - D_2]$, and the route of V_{31} is $[D_3 - C_5 - D_3]$, where D_2 and D_3 represent depots 2 and 3, respectively.

B. INITIAL POPULATION AND FITNESS FUNCTION

The initial population of the pre-determined size is randomly generated. However, a way to obtain a good initial population is desired since it significantly impacts the performance of the proposed GA but it is beyond the scope of this study. The objective of the proposed GA is to minimize the overall distance traveled by all vehicles. The fitness, which is the survival chance of a feasible solution satisfying all customer demands, is an inverse of the sum of the total distance traveled by all vehicles.

C. PARENTS SELECTION

In GA, an appropriate method to select chromosomes for crossover must be employed to give more chances to those fittest chromosomes in a population. The genetic search terminates prematurely; with too little chance, evolutionary progress is slower than necessary. Typically, lower selection pressure is desirable at the start of the genetic search in favor of a broad exploration of the search space, while a higher selection pressure is recommended at the end to converge efficiently. The roulette wheel selection method and linear scaling method have been used during the selection process in the proposed GA [78]. Based on several test runs of the proposed GA, the linear scaling function has been chosen as $f_i' = 0.1f_i + 1$, where f_i' and f_i are the scaled and raw finesses for chromosome i.

D. CROSSOVER

According to the crossover rate, two chromosomes from the current population are selected for mating through the selection process, i.e., a probability of crossover. If a randomly generated number between 0 and 1 is smaller than the given crossover rate, these chromosomes reproduce to form new offsprings to be included in the next generation. Otherwise, the crossover does not take place. An appropriate method to crossover a pair of selected chromosomes to improve the finesses of offsprings has been proposed for the MDSDVRP under consideration in this paper.

Two different crossover methods are used to produce offsprings in this paper. One is the position-based crossover method, which is applied to the first row. Because the first row represents the visiting order of vehicles, the row must be ensured to have no same gene. The other is the uniform crossover method, which is superior to other crossover strategies for combinatorial problems [79], and it is applied to the other rows. These crossovers are described below and illustrated in Figures 2, 3, and 4.

Step 1: Select a set of cells from Parent 1 randomly, which are shaded in Figure 2.

Step 2: For the first row, copy the selected cells into offspring at the corresponding locations and delete the corresponding values from Parent 2 (position-based crossover).

Parent 1

1	4	5	2	3
V_{21}	V_{22}	V_{11}	V_{11}	V_{12}
V_{12}	V_{13}	0	0	V_{13}
0	V_{12}	V_{22}	0	V_{21}

Parent 2

3	5	4	1	2
V_{11}	V_{22}	V_{12}	V_{11}	V_{13}
0	V_{12}	0	V_{22}	0
0	V_{21}	V_{21}	0	V_{11}

FIGURE 2. Step 1 of the crossover.

Offspring

	4		2	
V_{21}	V_{22}			V_{12}
	V_{13}	0	0	
0		V_{22}		V_{21}

Parent 2

3	5		1	
		V_{12}	V_{11}	
0				0
	V_{21}		0	

FIGURE 3. Step 2 of the crossover.

Offspring

3	4	5	2	1
V_{21}	V_{22}	V_{12}	V_{11}	V_{12}
0	V_{13}	0	0	0
0	V_{21}	V_{22}	0	V_{21}

FIGURE 4. Step 3 of the crossover.

For the other rows, copy the selected cells into offspring at the corresponding locations and delete the cells at the corresponding locations from Parent 2 (uniform crossover). See Figure 3.

Step 3: For the first row, copy the remaining cells from Parent 2 into empty cells of offspring from left to right in the order of cells showing up in Parent 2 (position-based crossover). For the other rows, copy the remaining cells from parent 2 into the empty cells of offspring at the corresponding locations (uniform crossover). See Figure 4.

The crossover procedure described in this study may contain the same vehicle more than once in a column, i.e., a vehicle unnecessarily visits a customer more than once. Then, a repair rule has been performed on the generated offspring by removing redundant visits for the same customer.

E. MUTATION

The mutation is another important operator in the GA and is applied to a chromosome at a mutation rate. Syswerda [79] proved that the mutation operator could sometimes play a more crucial role than the crossover. Therefore, the crossover and mutation operators need to well-designed per the problem on hand. The mutation operator brings random changes into a single chromosome. If a randomly generated number between 0 and 1 is smaller than the mutation rate, a chromosome reproduces a new member to be included in the next generation. Otherwise, the mutation does not take place. These random changes prevent pre-mature local convergence. The proposed GA has a relatively simple mutation procedure with elitism. All chromosomes in the population except the elites are subject to mutation at the mutation rate. The elitism rate is set to 10%, which is the percentage of best solutions in each population and these elites are immune to the mutation. While most GA implementations use the static mutation rate, the proposed GA introduces the dynamic self-adapting mutation rate. The mutation rate has been dynamically adapted during the evolutions. It starts with an initial value and then increases by the fixed or logarithmic amount whenever no improvement is observed over a certain number of generations. This number of generations is called mutation crank-up interval in this paper. If the best solution improves, the mutation rate drops to the initial mutation rate. It increases the global search capability to escape from the pre-mature local optima and to search for better solutions from diverse directions. The implementation of adaptive mutation rate has been proved effective in solving the various VRPs ([80]–[82]). The proposed GA uses the inversion mutation method, which is explained in the following steps and illustrated in Figure 5.

Step 1: Select a pair of columns randomly in a selected chromosome, which are shared in Figure 5.

Step 2: Swap the corresponding cells between these two columns except the cells on the first row. Note that the cells of the first rows are not swapped.

F. TERMINATION CONDITIONS

The proposed GA uses two termination criteria. One is that the proposed GA completes a specified maximum number of generations in this implementation. The other is that the proposed GA can also be terminated due to no improvement over a specified number of generations. After the proposed GA terminates, the chromosome with the highest fitness is interpreted as the best solution in that run.

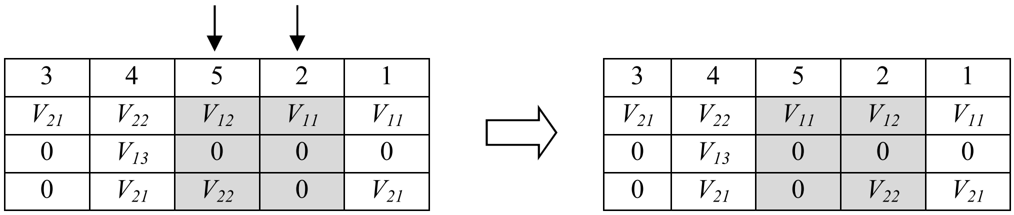

FIGURE 5. Mutation of the proposed GA.

IV. TAGUCHI METHOD FOR PARAMETER TUNING

Many GA process parameters have been defined to solve the combinatorial optimization problems effectively. The values of those process parameters need to be carefully selected. Good process parameter setting is important for the GA to obtain a good final solution. Usually, it is difficult to determine a good set of process parameters because their relationships can be rather complicated and unclear.

To fine-tune the performance of algorithms or processes, many parameters must be set carefully. The Taguchi method for parameter tuning is an important tool for robust design. Robust design is an engineering methodology for optimizing the product and process conditions that are minimally sensitive to the causes of variation. The orthogonal array and the signal-to-noise ratio (SNR) are two primary tools used in the Taguchi method. Additional details can be found in the books presented by Taguchi *et al.* [83] and Wu [84].

An orthogonal array is a fractional factorial matrix, which assures a balanced comparison of levels of any factor. It is a matrix of numbers arranged in rows and columns where each row represents the level of the factors in each run, and each column represents a specific factor that can be changed from each run. The symbol of three-level orthogonal arrays is Ln(3k), where n is the number of experimental runs, 3 is the number of levels for each factor, and k is the number of factors. The letter L comes from Latin since the orthogonal arrays were associated with Latin square designs from the outset.

SNR is the signal ratio over the noise, which measures the strength of the signal with the existence of noises. A higher SNR means that a process or a design is the more robust. Suppose that we have a set of experiment runs. Since the objective of MDSDVRPs is to minimize the total traveled distance, the smaller-the-better is an appropriate measure in this study. The following formulation for smaller-the-better characteristics is used;

$$SNR = -10\log(\frac{1}{n}\sum_{i=1}^{n} x_i^2).$$

Through extensive computational experiments for various problems, four process parameters are identified as important design factors for the performance of the proposed GA; population size, crossover rate, mutation crank-up interval, and mutation policy. For each design factor, three levels of the process parameters are chosen from the previous research;

TABLE 1. Four factors with three levels per factor.

	Level		
Process parameter	1	2	3
A (Population size)	100	150	200
B (Crossover rate)	0.6	0.7	0.8
C	50	75	100
(mutation crank-up interval)			
D (Mutation policy)	Mutation Policy 1	Mutation Policy 2	Mutation Policy 3

(100, 150, 200) for the population size, (0.6, 0.7, 0.8) for the crossover rate, (50, 75, 100) for the mutation crank-up interval, and (Mutation Policy 1, Mutation Policy 2, Mutation Policy 3) for the mutation policy. In Mutation Policy 1, the static mutation rate of 0.05 has been used. Mutation Policy 2 increases the mutation rate linearly from 0.05 by 0.10 whenever no improvement is made for the mutation crank-up interval. Mutation Policy 3 uses the logarithmic increase of the mutation rate from 0.05. The logarithmic increase of Mutation Policy 3 instead of linear increase in MP 2 (possible mutation rates are 0.05, 0.15, 0.25) can be calculated as

$$A_n = 0.05 + \ln(1 + 0.15n),$$

where $n = \{1, 2, 3\}$ (possible mutation rate can be 0.05, 0.19, 0.31). Elitist rate is used to retain 10% of the best chromosomes at each generation.

Table 1 presents four process parameters with three levels in the proposed GA. To conduct the full factorial experiment with all factors, 34 (or 81) experiments are necessary to determine the optimal process parameters. However, the Taguchi method only requires 9 runs to optimize the process parameters when L9 orthogonal array is used.

Each row in Table 2 shows 9 experiments with process parameters A, B, C, and D at their corresponding levels. To account for the characteristics of stochastic disturbance in the proposed GA, each experiment has been tested 40 times, i.e., 10 runs for each of 4 test problems. These 4 test problems are introduced in the following sections. Various sizes and structures of VRPs can be considered noise factors. The relative gap between the i-th experimental solution and the

TABLE 2. SNR values of the L_9 experiments.

Run	A	B	C	D	E-n22-k4	E-n23-k3	E-n30-k3	E-n33-k4	SNR
1	1	1	1	1	$G_1,\ldots,G_{10}$	$G_{11},\ldots,G_{20}$	$G_{21},\ldots,G_{30}$	$G_{31},\ldots,G_{40}$	26.15
2	1	2	2	2	$G_1,\ldots,G_{10}$	$G_{11},\ldots,G_{20}$	$G_{21},\ldots,G_{30}$	$G_{31},\ldots,G_{40}$	27.55
3	1	3	3	3	$G_1,\ldots,G_{10}$	$G_{11},\ldots,G_{20}$	$G_{21},\ldots,G_{30}$	$G_{31},\ldots,G_{40}$	26.54
4	2	1	2	3	$G_1,\ldots,G_{10}$	$G_{11},\ldots,G_{20}$	$G_{21},\ldots,G_{30}$	$G_{31},\ldots,G_{40}$	27.82
5	2	2	3	1	$G_1,\ldots,G_{10}$	$G_{11},\ldots,G_{20}$	$G_{21},\ldots,G_{30}$	$G_{31},\ldots,G_{40}$	29.64
6	2	3	1	2	$G_1,\ldots,G_{10}$	$G_{11},\ldots,G_{20}$	$G_{21},\ldots,G_{30}$	$G_{31},\ldots,G_{40}$	29.65
7	3	1	3	2	$G_1,\ldots,G_{10}$	$G_{11},\ldots,G_{20}$	$G_{21},\ldots,G_{30}$	$G_{31},\ldots,G_{40}$	32.20
8	3	2	1	3	$G_1,\ldots,G_{10}$	$G_{11},\ldots,G_{20}$	$G_{21},\ldots,G_{30}$	$G_{31},\ldots,G_{40}$	30.21
9	3	3	2	1	$G_1,\ldots,G_{10}$	$G_{11},\ldots,G_{20}$	$G_{21},\ldots,G_{30}$	$G_{31},\ldots,G_{40}$	29.58

best-known solution, Gi, is calculated as

$$G_i = \frac{ith\ solution - best\ known\ solution}{best\ known\ solutoin}$$

where i = {1, 2, . . . , 40}.

Since the smaller Gi is more desirable, the smaller-the-better SNR calculation has been used as

$$SNR = -10\log(\frac{1}{n}\sum_{i=1}^{n} x_i^2),$$

where n = 40. The SNR values of process parameter A are calculated as

$$SNR_{A_1} = SNR_1 + SNR_2 + SNR_3,$$
$$SNR_{A_2} = SNR_4 + SNR_5 + SNR_6,$$
$$SNR_{A_3} = SNR_7 + SNR_8 + SNR_9,$$

where SNRi represents the SNR value of the ith run and SNR_{A_i} denotes the aggregated SNR values of level i of process parameter A.

The optimal levels of process parameters A, B, C, and D are the level with the largest SNR value, and the calculated $SNR_{A_i} SNR_{B_i} SNR_{C_i}$ and SNR_{D_i} are shown in Table 3. SNR values at the optimal level for each process parameter are in bold. According to Table 3, the optimal parameter settings of robust design are 200 for population size, 0.7 for the crossover rate, 75 for mutation crank-up interval, and Mutation Policy 2.

From the contribution of each parameter on the performance, process parameter A impacts 77%, and process parameter D influences 15% on the performance of the proposed GA, as shown in Table 3. In other words, the large size of the population and the mutation policy of linear increase in mutation rate dominate other process parameters on the performance of the proposed GA.

The process parameters optimized by the Taguchi method are robust, so the signal or performance measure always centralizes to the optimal expected values and is less affected by noise. Using the optimal process parameters suggested by the Taguchi method, the proposed GA's searchability has been improved and the proposed GA has generated better solutions.

TABLE 3. SNR values of the process parameters.

	Process parameter			
Level	A	B	C	D
1	80.24	86.17	86.01	85.37
2	87.12	**87.40**	84.94	**89.40**
3	**91.98**	85.77	**88.38**	84.56
Contribution (%)	77	2	7	15

Note: Bold typeface represents the optimal SNR values in each column of parameters.

V. COMPUTATIONAL RESULTS

In this section, computational experiences with the MIP model in Section 2 and the proposed GA in Section 3 are presented. All computational experiments are carried out on a PC with a Pentium IV CPU at 3.4 GHz and 2.0 GB RAM. The MIP model is solved using CPLEX in an OPL-Studio environment. The program for the proposed GA is implemented in C++ programming language.

A. EXPERIMENT OF AN EXEMPLIFIED MDSDVRP

A hypothetical test problem defined in Tables 6 and 7 has been generated to demonstrate the effectiveness of the proposed GA. The problem has 6 customers with known demands and two depots with 3 and 2 vehicles, respectively, of heterogeneous capacities. The locations and demands of customers are given in Table 4. Table 5 shows the locations of two depots and their vehicles with the given capacities.

TABLE 4. Locations and demands of 6 customers.

Node	Coordinates	Demand	Node	Coordinates	Demand
Customer 1 (C_1)	(9,94)	1300	Customer 4 (C_4)	(31,144)	4100
Customer 2 (C_2)	(19,62)	1800	Customer 5 (C_5)	(35,65)	3000
Customer 3 (C_3)	(26,126)	2300	Customer 6 (C_6)	(44,88)	4800

TABLE 5. Vehicles available at two depots and their capacities.

	Depot 1			Depot 2	
Coordinates	(21, 86)			(36, 97)	
Vehicle	V_{11}	V_{12}	V_{13}	V_{21}	V_{22}
Vehicle capacity	1500	4800	8000	2200	2500

TABLE 6. Results from the MIP model.

		Maximum number of visits to a customer		
		One	Two	Three
Total distance		358.77	300.67	263.68
Routes for the vehicles	V_{11}	$D_1 \rightarrow C_1\ (1300) \rightarrow D_1$	Unused	Unused
	V_{12}	$D_1 \rightarrow C_4\ (4100) \rightarrow D_1$	$D_1 \rightarrow C_2\ (1800) \rightarrow C_5\ (3000) \rightarrow D_1$	$D_1 \rightarrow C_2\ (1800) \rightarrow C_5\ (3000) \rightarrow D_1$
	V_{13}	$D_1 \rightarrow C_5\ (3000) \rightarrow C_6\ (4800) \rightarrow D_1$	$D_1 \rightarrow C_6\ (2600) \rightarrow C_4\ (4100) \rightarrow C_1\ (1300) \rightarrow D_1$	$\boldsymbol{D_1 \rightarrow C_1\ (1300) \rightarrow C_3\ (2300) \rightarrow C_4\ (4100) \rightarrow C_6\ (300) \rightarrow D_1}$
	V_{21}	$D_2 \rightarrow C_2\ (1800) \rightarrow D_2$	$D_2 \rightarrow C_6\ (2200) \rightarrow D_2$	$D_2 \rightarrow C_6\ (2200) \rightarrow D_2$
	V_{22}	$D_2 \rightarrow C_3\ (2300) \rightarrow D_2$	$D_2 \rightarrow C_3\ (2300) \rightarrow D_2$	$\boldsymbol{D_2 \rightarrow C_6\ (2300) \rightarrow D_2}$

TABLE 7. Results from the proposed GA.

		Allowed visits by each vehicle		
		One	Two	Three
Total distance		358.77	300.67	263.68
Routes for the vehicles	V_{11}	$D_1 \rightarrow C_1\ (1300) \rightarrow D_1$	Unused	Unused
	V_{12}	$D_1 \rightarrow C_4\ (4100) \rightarrow D_1$	$D_1 \rightarrow C_2\ (1800) \rightarrow C_5\ (3000) \rightarrow D_1$	$D_1 \rightarrow C_2\ (1800) \rightarrow C_5\ (3000) \rightarrow D_1$
	V_{13}	$D_1 \rightarrow C_5\ (3000) \rightarrow C_6\ (4800) \rightarrow D_1$	$D_1 \rightarrow C_6\ (2600) \rightarrow C_4\ (4100) \rightarrow C_1\ (1300) \rightarrow D_1$	$\boldsymbol{D_1 \rightarrow C_1\ (1300) \rightarrow C_3\ (2300) \rightarrow C_4\ (4100) \rightarrow C_6\ (100) \rightarrow D_1}$
	V_{21}	$D_2 \rightarrow C_2\ (1800) \rightarrow D_2$	$D_2 \rightarrow C_6\ (2200) \rightarrow D_2$	$D_2 \rightarrow C_6\ (2200) \rightarrow D_2$
	V_{22}	$D_2 \rightarrow C_3\ (2300) \rightarrow D_2$	$D_2 \rightarrow C_3\ (2300) \rightarrow D_2$	$\boldsymbol{D_2 \rightarrow C_6\ (2500) \rightarrow D_2}$

Customers and depots are on the 2-dimensional Euclidean space.

The optimal solutions from the MIP model presented in Section 2 are shown for $V = 1$, 2, and 3, in Table 6. The numbers in the parenthesis in Tables 6 and 7 are the unloaded amounts by the corresponding vehicles. As conjectured earlier, split deliveries to a customer reduce the total distances as the maximum number of visits increases (See Tables 6 and 7).

The best solutions from the proposed GA are shown for $V = 1$, 2, and 3 in Table 7. Tables 6 and 7 show that the total traveled distances from the MIP model and the proposed GA are identical, and it indicates that the proposed GA achieves the optimality for this small hypothetical

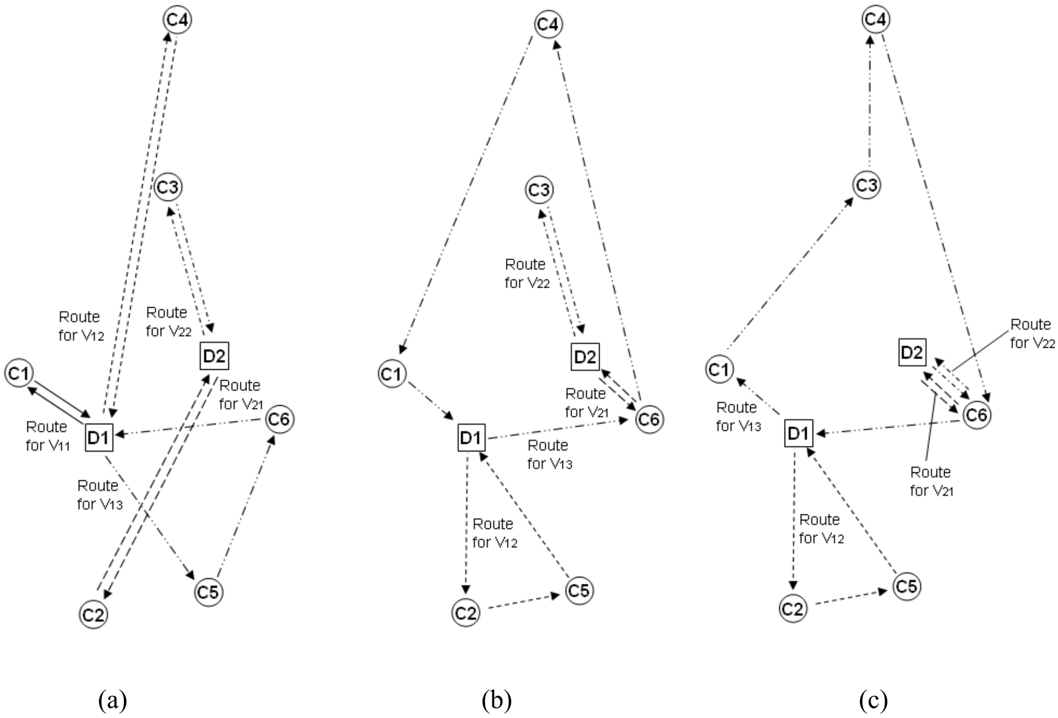

FIGURE 6. Vehicle routes for one allowed visit (a), two allowed visits (b), and three allowed visits (c).

TABLE 8. Performance comparison for SDCVRP benchmark problems.

Problem	Customers	Vehicles	Vehicle capacity	Best known solution	The proposed GA
*E-n22-k4	21	4	6000	375	375
E-n23-k3	22	3	4500	569	569
E-n30-k3	29	3	4500	534	534
E-n33-k4	32	4	8000	835	835

*E-n22-k4 stands for Eilon et al. problem with 22 nodes (21 customers, 1 depot) and 4 vehicles at the depot.

problem. Note that the routes of V_{13} and V_{22} in bold from Tables 6 and 7 are identical, but the unloaded amounts at C_6 are different. The proposed GA generated an alternative solution.

From Tables 6 and 7, the proposed GA effectively solves the MDSDVRP with heterogeneous vehicles from multiple depots, allowing split deliveries. Note that the increase in the maximum number of visits leads to a shorter travel distance.

The vehicle routes with different maximum numbers of visits from the MIP model are illustrated in Figure 6. The circled Ci stands for customer i. Three vehicles V_{11}, V_{12}, and V_{13} are housed in depot 1 (D_1) and two vehicles, V_{21} and V_{22}, in depot 2 (D_2). In Figure 6, the routes of different vehicles are represented by different types of arrows. Customer C_6 has two vehicle visits for $V = 2$ in Figure 6(b) and three-vehicle visits for $V = 3$ in Figure 6(c).

B. EXPERIMENT OF BENCHMARK SDCVRPS

The CPLEX solver can only solve the MIP model of small sizes because of the memory limitation. The proposed GA can generate good solutions for the MDSDVRPs of larger sizes. In the next computational experiment, the proposed GA has been applied to benchmark VRPs available at the VRPLIB repository on the website (https://www.coin-or.org/SYMPHONY/branchandcut/VRP/data/index.htm.old). These problems have been widely used as benchmarks ([18], [85]–[87]) and they are derived from Eilon *et al.* [88].

The benchmark VRPs under comparison are SDCVRP, not allowing multiple visits. Since there is no benchmark problem for MDSDVRPs yet, we assessed the competitiveness of our proposed GA by comparison with the Eilon instances. Much research has used these benchmark instances to demonstrate the effectiveness of their algorithms ([89]–[91]), considering the NP-hardness of VRPs.

FIGURE 7. The MDSDVRP with 35 US cities and 3 depots.

TABLE 9. Vehicle capacities in each warehouse city.

Warehouse city	Vehicle	
	Identification	Capacity
Denver	V_{11}	1200
	V_{12}	1800
	V_{13}	2500
Chicago	V_{21}	1200
	V_{22}	1800
	V_{23}	2500
Atlanta	V_{31}	1200
	V_{32}	1800
	V_{33}	2500

As explained in Section 1, many approximation algorithms and heuristics keep updating the best solutions for these benchmark instances. For some instances, those solutions were matched with the upper bounds, proving that they are optimal.

The effectiveness of the proposed GA for MDSDVRPs is already verified in Table 7. By limiting the proposed GA's ability with the maximum number of visits to a customer set to 1 and a single depot, the effectiveness of the proposed GA was compared with the best-known solution for SDCVRP.

Note that the proposed GA is proposed to solve the heterogeneous VRPs with multi-depot, allowing split-deliveries, but it also shows good effectiveness for the benchmark SDCVRPs in Table 8. Table 8 shows the number of customers, the number of vehicles at a depot, the vehicle capacity, the best-known solution by previous works, and the solution obtained by the proposed GA. For this benchmark library, customers and a single depot are with coordinates in a network, and Euclidean distances are used.

Table 8 shows that the proposed GA is also effective in solving SDCVRPs and shows comparable performances for the benchmark problems. The proposed GA does solve the VRP with heterogeneous vehicles from multiple depots, allowing split deliveries to customers and solving the classical SDCVRP well. Route of all vehicles in the benchmark problems are reported in Appendix for the archival purpose.

TABLE 10. Demands of the retailer cities.

No.	City	Demand	No.	City	Demand
1	Boise	140	19	Milwaukee	550
2	Boston	100	20	Minneapolis	720
3	Charlotte	180	21	Nashville	780
4	Columbia	620	22	New Orleans	240
5	Columbus	900	23	New York	150
6	Dallas	310	24	Oklahoma	180
7	Des	990	25	Philadelphia	650
8	Detroit	110	26	Phoenix	240
9	Hartford	140	27	Portland, OR	310
10	Houston	190	28	Reno	450
11	Indianapoli	980	29	St. Louis	350
12	Jacksonvill	210	30	Salt Lake	160
13	Kansas	250	31	San Antonio	170
14	Las Vegas	310	32	San Diego	120
15	Los	310	33	San	200
16	Memphis	880	34	Seattle	240
17	Louisville	170	35	Washington,	310
18	Miami	650			

TABLE 11. Results of the proposed GA for the VRP with one or two allowed visit(s).

	One allowed	Two allowed
Total	17950 miles	17710 miles

C. EXPERIMENT OF A REAL-LIFE SCALE MDSDVRP

A real-life scale MDSDVRP with heterogeneous vehicles from multiple depots, allowing split deliveries, is presented and solved by the proposed GA. The problem has 35 US cities (customers), 3 depots, and 9 heterogeneous vehicles (3 vehicles at each depot). The distances between all cities are the approximated driving distances on the road, obtained by database on Google Maps, instead of the Euclidean distances.

The 38 nodes of the proposed problem are shown in Figure 7, where the cities in red circles are 35 retailer cities and the ones in blue rectangles are 3 warehouse cities located in Denver, Chicago, and Atlanta. There are three vehicles in each warehouse city. The capacities of vehicles in each warehouse city are given in Table 9. The demands of the retailer cities are given in Table 10.

TABLE 12. Vehicle routes with two allowed visits.

Warehouse city	Vehicle routes
Denver	V_{11}: D - 14 - 15 - 32 - 26 - D V_{12}: D - 20 - 7 - D V_{13}: D - 28 - 33 - 27 - 34 - 1 - 30 - D
Chicago	V_{21}: D - 19 - D V_{22}: D - 22 - 10 - 31 - 6 - 24 - 13 -**11(420)** - D V_{23}: D - 29 - 17 - 21 - 16 - 8 - D
Atlanta	V_{31}: D - 4 - D V_{32}: D - 12 - 18 - 35 - 3 - D V_{33}: D - 25 - 9 - 2 - 23 - 5 - **11(560)** - D

*D stands for each warehouse city of the left column in the table. The numbers in routes stand for the retailer city numbers in Table 8.

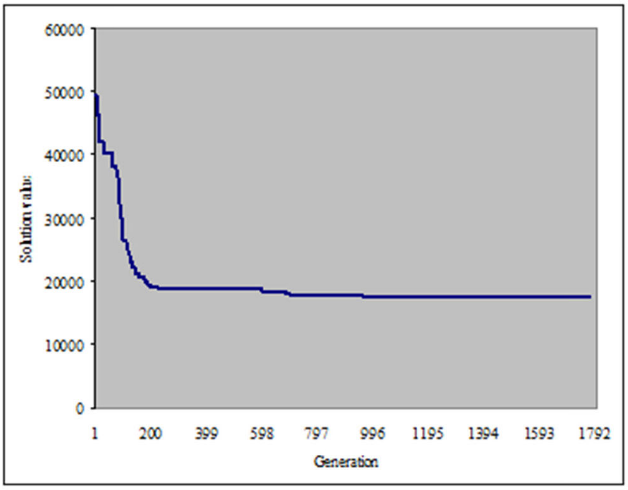

FIGURE 8. The convergence of the proposed GA.

The results of the proposed GA for this problem with $V = 1$ and 2 are shown in Table 11. Note that routes with $V = 2$ are better than ones with $V = 2$. The vehicle routes with $V = 2$ are given in Table 12. Two visits to customer 11 by vehicles V22 and V33, shown in bold, contribute to the reduction of the total traveled distance. All other customer's demands are satisfied only with one visit in Table 12.

The evolution of the best solution during 1776 generations by the proposed GA is shown in Figure 8. The graph shows the convergence of the best solution over the generations. After the 976th generation, the best solution of the problem is finally obtained.

VI. CONCLUSION AND FUTURE RESEARCH

A generalized VRP with heterogeneous vehicles from multiple depots, allowing split deliveries, has been identified by relaxing from the classical VRPs the constraints of the number of depots and the number of visits allowed to each customer. The identified VRP has been modeled into a MIP formulation and tested to solve small problems. As the motivation of this paper conjectures, it has been identified that the introduction of split deliveries or multiple visits to each customer leads to a reduction of delivery cost.

A GA to effectively and efficiently solve the medium or large VRPs with heterogeneous vehicles from multiple depots, allowing split deliveries, has been proposed and validated successfully. The proposed algorithm has produced the solutions that are equal or close to the best-known solutions for the benchmark SDCVRPs for which the proposed algorithm has been executed with the restriction of one depot, one allowed visit, and capacitated vehicles.

TABLE 13. Best solutions obtained by the proposed GA for the benchmark problems.

Problem	Vehicle routes
E-n22-k4	*D - *C17 (1000) - C20 (1800) - C18 (900) - C15 (900) - C12 (1300) – D D - C6 (400) - C1 (1100) - C2 (700) - C5 (2100) - C7 (800) - C9 (500) - D D - C10 (600) - C8 (100) - C3 (800) - C4 (1400) - C11 (1200) - C13 (1300) - D D - C14 (300) - C21 (700) - C19 (2500) - C16 (2100) - D
E-n23-k3	D - C12 (300) - C11 (225) - C6 (175) - C1 (125) - C2 (84) - C3 (60) - C16 (100) - C15 (150) - C14 (500) - C17 (250) - C22 (75) - C20 (500) - C19 (600) - C18 (120) - D D - C10 (4100) - C13 (250) - D D - C7 (350) - C9 (1100) - C8 (150) - C5 (300) - C4 (500) - C21 (175) - D
E-n30-k3	D - C19 (400) - C15 (550) - C16 (150) - C13 (150) - C7 (150) - C17 (100) - C9 (300) - C14 (150) - C8 (450) - C12 (125) - C11 (950) - C10 (100) - C23 (300) - C18 (150) - D D - C21 (1500) - C6 (150) - C24 (500) - C25 (800) - C29 (1000) - C27 (100) - C28 (150) - C26 (300) - D D - C22 (100) - C2 (3100) - C5 (200) - C1 (300) - C4 (100) - C3 (125) - C20 (300) -D
E-n33-k4	D - C13 (250) - C17 (550) - C25 (1400) - C24 (750) - C23 (700) - C20 (400) - C21 (300) - C22 (1300) - C19 (200) -C18 (650) - C10 (750) - C6 (80) - C5 (40) - C3 (400) -D D - C2 (400) - C12 (150) - C11 (1500) - C32 (1100) - C8 (900) - C9 (600) - C7 (2000) - C4 (1200) - D D - C30 (2500) - C14 (1600) - C31 (1700) - D D - C1 (700) - C15 (450) - C26 (4000) - C27 (600) - C16 (700) - C28 (1000) - C29 (500) - D

*D stands for the depot in the problem.
*C17 (1000) stands for customer 17 and its demand, 1000.

Taguchi's robust design method has been introduced and applied in optimizing the process parameters of the proposed GA. Using the optimized parameters, the proposed GA shows robust performance regardless of the size or structure of the problems. The proposed algorithm has effectively solved the test problem with 35 US cities and 3 depots, allowing multiple visits and heterogeneous vehicles.

A new mutation policy has been developed for the proposed GA. The existing GA implementations use the fixed mutation rate. This thesis proposed the idea of self-adapting mutation rate, which enables dynamic speeds of evolutions in nature. The proposed mutation policy has proven effective from the Taguchi method analysis and has generated consistently good solutions.

As for future research, it may be helpful to investigate the issue where there is a restriction on the driving distance of vehicles available at each depot. The problem where the customers have different time windows as their requirements for delivered goods may also be worth considering. In addition, future research can be conducted to improve the proposed algorithm and the performance comparison with other solution methods Additional improvements might lie in the combination of various selection and population replacement schemes and new fitness models. Further investigation in optimizing the performance of the proposed algorithm and developing other solution methods to solve MDSDVRP can be conducted to compare the comparison in performance and the solution quality. Applications of the approach to related problems can be explored as well.

APPENDIX

See Table 13.

REFERENCES

[1] S. Hulagu and H. B. Celikoglu, "An electric vehicle routing problem with intermediate nodes for shuttle fleets," *IEEE Trans. Intell. Transp. Syst.*, early access, Sep. 23, 2020, doi: 10.1109/TITS.2020.3023673.

[2] S. Hulagu and H. B. Celikoglu, "A multiple objective formulation of an electric vehicle routing problem for shuttle bus fleet at a university campus," in *Proc. 6th Int. Conf. Models Technol. Intell. Transp. Syst. (MT-ITS)*, Cracow, Poland, Jun. 2019, pp. 1–5.

[3] A. Ham and M.-J. Park, "Electric vehicle route optimization under time-of-use electricity pricing," *IEEE Access*, vol. 9, pp. 37220–37228, 2021.

[4] K. Dorling, J. Heinrichs, G. G. Messier, and S. Magierowski, "Vehicle routing problems for drone delivery," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 47, no. 1, pp. 70–85, Jan. 2017.

[5] R. G. Ribeiro, L. P. Cota, T. A. M. Euzebio, J. A. Ramirez, and F. G. Guimaraes, "Unmanned-aerial-vehicle routing problem with mobile charging stations for assisting search and rescue missions in postdisaster scenarios," *IEEE Trans. Syst., Man, Cybern., Syst.*, early access, Jun. 22, 2021, doi: 10.1109/TSMC.2021.3088776.

[6] G. B. Dantzig and J. H. Ramser, "The truck dispatching problem," *Manage. Sci.*, vol. 6, no. 1, pp. 80–91, 1959.

[7] M. L. Fisher, "Optimal solution of vehicle routing problems using minimum K-trees," *Oper. Res.*, vol. 42, no. 4, pp. 626–642, Aug. 1994.

[8] E. L. Lawler, J. K. Lenstra, A. H. G. R. Kan, and D. B. Shmoys, *The Traveling Salesman Problem*. Chichester, U.K.: Wiley, 1985.

[9] J. Knox, "Tabu search performance on the symmetric traveling salesman problem," *Comput. Oper. Res.*, vol. 21, no. 8, pp. 867–876, Oct. 1994.

[10] A. Barvinok, S. P. Fekete, D. S. Johnson, A. Tamir, G. J. Woeginger, and R. Woodroofe, "The geometric maximum traveling salesman problem," *J. ACM*, vol. 50, no. 5, pp. 641–664, Sep. 2003.

[11] L. Engebretsen and M. Karpinski, "TSP with bounded metrics," *J. Comput. Syst. Sci.*, vol. 72, no. 4, pp. 509–546, Jun. 2006.

[12] A. Ouaarab, B. Ahiod, and X.-S. Yang, "Discrete cuckoo search algorithm for the travelling salesman problem," *Neural Comput. Appl.*, vol. 24, nos. 7–8, pp. 1659–1669, 2014.

[13] M. Mahi, Ö. K. Baykan, and H. Kodaz, "A new hybrid method based on particle swarm optimization, ant colony optimization and 3-opt algorithms for traveling salesman problem," *Appl. Soft Comput.*, vol. 30, pp. 484–490, May 2015.

[14] J. Desrosiers, Y. Dumas, M. Solomon, and F. Soumis, "Time constrained routing and scheduling in: Network routing," in *Handbooks in Operations Research and Management Science*, vol. 8. Amsterdam, The Netherlands: North Holland, 1995.

[15] I. H. Osman, "Vehicle routing and scheduling: Applications, algorithms and developments," in *Proc. Int. Conf. Ind. Logistics*, Rennes, France, 1993, p. 277.

[16] G. Laporte, "The vehicle routing problem: An overview of exact and approximate algorithms," *Eur. J. Oper. Res.*, vol. 23, pp. 631–640, Jun. 1992.

[17] P. Toth and D. Vigo, "Exact algorithms for vehicle routing," in *Fleet Management and Logistics*. Boston, MA, USA: Kluwer, 1998, pp. 1–31.

[18] J. Lysgaard, A. N. Letchford, and R. W. Eglese, "A new branch-and-cut algorithm for the capacitated vehicle routing problem," *Math. Program.*, vol. 100, no. 2, pp. 423–445, Jun. 2004.

[19] M. Alssager, Z. A. Othman, M. Ayob, R. Mohemad, and H. Yuliansyah, "Hybrid cuckoo search for the capacitated vehicle routing problem," *Symmetry*, vol. 12, no. 12, p. 2088, Dec. 2020.

[20] M. Gendreau, G. Laporte, C. Musaraganyi, and É. D. Taillard, "A tabu search heuristic for the heterogeneous fleet vehicle routing problem," *Comput. Oper. Res.*, vol. 26, no. 12, pp. 1153–1173, Oct. 1999.

[21] H.-Y. Kang and A. Lee, "An enhanced approach for the multiple vehicle routing problem with heterogeneous vehicles and a soft time window," *Symmetry*, vol. 10, no. 11, p. 650, Nov. 2018.

[22] D. Li, Q. Cao, M. Zuo, and F. Xu, "Optimization of green fresh food logistics with heterogeneous fleet vehicle route problem by improved genetic algorithm," *Sustainability*, vol. 12, no. 5, p. 1946, Mar. 2020.

[23] G. Laporte, Y. Nobert, and D. Arpin, "Optimal solutions to capacitated multidepot vehicle routing problems," *Congressus Numeratium*, vol. 44, pp. 283–292, Dec. 1984.

[24] L. Zhishuo and C. Yueting, "Sweep based multiple ant colonies algorithm for capacitated vehicle routing problem," in *Proc. IEEE Int. Conf. e-Bus. Eng. (ICEBE)*, Beijing, China, Oct. 2005, pp. 387–394.

[25] I.-M. Chao, B. L. Golden, and E. Wasil, "A new heuristic for the multi-depot vehicle routing problem that improves upon best-known solutions," *Amer. J. Math. Manage. Sci.*, vol. 13, nos. 3–4, pp. 371–406, Feb. 1993.

[26] J. Renaud, F. F. Boctor, and G. Laporte, "An improved petal heuristic for the vehicle routeing problem," *J. Oper. Res. Soc.*, vol. 47, pp. 329–336, Feb. 1996.

[27] Y. Shi, L. Lv, F. Hu, and Q. Han, "A heuristic solution method for multi-depot vehicle routing-based waste collection problems," *Appl. Sci.*, vol. 10, no. 7, p. 2403, Apr. 2020.

[28] D. Hou, H. Fan, X. Ren, P. Tian, and Y. Lv, "Time-dependent multi-depot heterogeneous vehicle routing problem considering temporal–spatial distance," *Sustainability*, vol. 13, no. 9, p. 4674, Apr. 2021.

[29] F. A. Tillman and R. W. Hering, "A study of a look-ahead procedure for solving the multiterminal delivery problem," *Transp. Res.*, vol. 5, no. 3, pp. 225–229, Aug. 1971.

[30] R. Liu, Z. Jiang, R. Y. K. Fung, F. Chen, and X. Liu, "Two-phase heuristic algorithms for full truckloads multi-depot capacitated vehicle routing problem in carrier collaboration," *Comput. Oper. Res.*, vol. 37, no. 5, pp. 950–959, May 2010.

[31] H. Bae and I. Moon, "Multi-depot vehicle routing problem with time windows considering delivery and installation vehicles," *Appl. Math. Model.*, vol. 40, nos. 13–14, pp. 6536–6549, Jul. 2016.

[32] P. Stodola and J. Mazal, "Applying the ant colony optimisation algorithm to the capacitated multi-depot vehicle routing problem," *Int. J. Bio-Inspired Comput.*, vol. 8, no. 4, pp. 228–233, 2016.

[33] L. D. Bodin, B. L. Golden, A. A. Assad, and M. Ball, "Routing and scheduling of vehicles and crews, the state of the art," *Comput. Oper. Res.*, vol. 10, no. 2, pp. 212–263, 1983.

[34] E. Falkenauer, "A hybrid grouping genetic algorithm for bin packing," *J. Heuristics*, vol. 2, pp. 5–30, Jun. 1996.

[35] M. R. Garey and D. S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*. New York, NY, USA: Freeman, 1979.

[36] L. Wang, R. Cai, M. Lin, and Y. Zhong, "Enhanced list-based simulated annealing algorithm for large-scale traveling salesman problem," *IEEE Access*, vol. 7, pp. 144366–144380, 2019.

[37] Y. Cui, J. Zhong, F. Yang, S. Li, and P. Li, "Multi-subdomain grouping-based particle swarm optimization for the traveling salesman problem," *IEEE Access*, vol. 8, pp. 227497–227510, 2020.

[38] M. Wang, T. Ma, G. Li, X. Zhai, and S. Qiao, "Ant colony optimization with an improved pheromone model for solving MTSP with capacity and time window constraint," *IEEE Access*, vol. 8, pp. 106872–106879, 2020.

[39] I. K. Singgih, J. Lee, and B.-I. Kim, "Node and edge drone surveillance problem with consideration of required observation quality and battery replacement," *IEEE Access*, vol. 8, pp. 44125–44139, 2020.

[40] I. K. Singgih, O. Yu, B.-I. Kim, J. Koo, and S. Lee, "Production scheduling problem in a factory of automobile component primer painting," *J. Intell. Manuf.*, vol. 31, no. 6, pp. 1483–1496, Aug. 2020.

[41] S. Lee and J. A. Ventura, "An algorithm for constructing minimal *c*-broadcast networks," *Networks*, vol. 38, no. 1, pp. 6–21, 2001.

[42] I. K. Singgih and G. M. Lee, "Lower and upper bounds for military deployment planning considering combat," *Int. J. Ind. Eng., Theory, Appl., Pract.*, vol. 21, no. 6, pp. 408–420, 2014.

[43] F. B. Pereira, J. Tavares, P. Machado, and E. Costa, "GVR: A new genetic representation for the vehicle routing problem," in *Proc. 13th Irish Int. Conf. Artif. Intell. Cogn. Sci.*, 2002, pp. 95–102.

[44] G. Laporte, M. Gendreau, J.-Y. Potvin, and F. Semet, "Classical and modern heuristics for the vehicle routing problem," *Int. Trans. Oper. Res.*, vol. 7, nos. 4–5, pp. 285–300, 2000.

[45] M. Desrochers, J. K. Lenstra, and M. W. P. Savelsbergh, "A classification scheme for vehicle routing and scheduling problems," *J. Oper. Res. Soc.*, vol. 46, pp. 322–332, Jun. 1990.

[46] G. Laporte and Y. Nobert, "Exact algorithms for the vehicle routing problem," *Ann. Discrete Math.*, vol. 31, pp. 147–184, Jan. 1987.

[47] N. Christofides, A. Mingozzi, and P. Toth, "The vehicle routing problem," *Combinatorial Optimization*. Chichester, U.K.: Wiley, 1979, pp. 315–338.

[48] T. L. Magnanti, "Combinatorial optimization and vehicle fleet planning: Perspectives and prospects," *Networks*, vol. 11, pp. 179–214, Jun. 1981.

[49] N. Christofides, *Vehicle Routing: The Traveling Salesman Problem: A Guided Tour of Combinatorial Optimization*. Chichester, U.K.: Wiley, 1985, pp. 431–448.

[50] B. L. Golden, E. A. Wasil, J. P. Kelly, and I. M. Chao, "Metaheuristics in vehicle routing," in *Fleet Management and Logistics*. Boston, MA, USA: Kluwer, 1998, pp. 33–56.

[51] G. Laporte, "What you should know about the vehicle routing problem," *Nav. Res. Logistics*, vol. 54, no. 8, pp. 811–819, 2007.

[52] E. Ghorbani, M. Alinaghian, G. B. Gharehpetian, S. Mohammadi, and G. Perboli, "A survey on environmentally friendly vehicle routing problem and a proposal of its classification," *Sustainability*, vol. 12, no. 21, p. 9079, Oct. 2020.

[53] W. K. Anuar, L. S. Lee, S. Pickl, and H.-V. Seow, "Vehicle routing optimisation in humanitarian operations: A survey on modelling and optimisation approaches," *Appl. Sci.*, vol. 11, no. 2, p. 667, Jan. 2021.

[54] H. Lim, "A genetic algorithm for the vehicle routing problem with heterogeneous vehicles from multiple depots, allowing multiple visits," M.S. thesis, Dept. Ind. Eng., Oregon State Univ., Corvallis, OR, USA, Aug. 2007.

[55] D. Gulczynski, B. Golden, and E. Wasil, "The multi-depot split delivery vehicle routing problem: An integer programming-based heuristic, new test problems, and computational results," *Comput. Ind. Eng.*, vol. 61, no. 3, pp. 794–804, Oct. 2011.

[56] G. Laporte and F. Semet, "Classical heuristics for the vehicle routing problem," GERAD, Montréal, QC, Canada, Tech. Rep. G-98-54, 1999.

[57] E. Aarts and J. K. Lenstra, *Local Search in Combinatorial Optimization*. Princeton, NJ, USA: Princeton Univ. Press, 2003. [Online]. Available: https://press.princeton.edu/books/paperback/9780691115221/local-search-in-combinatorial-optimization

[58] J. F. Cordeau, M. Gendreau, G. Laporte, P. J-Y, and F. Semet, "A guide to vehicle routing heuristics," *J. Oper. Res. Soc.*, vol. 53, pp. 512–522, May 2002.

[59] G. Barbarosoglu and D. Ozgur, "A tabu search algorithm for the vehicle routing problem," *Comput. Oper. Res.*, vol. 26, no. 3, pp. 255–270, Mar. 1999.

[60] B. M. Baker and M. A. Ayechew, "A genetic algorithm for the vehicle routing problem," *Comput. Oper. Res.*, vol. 30, no. 5, pp. 787–800, Apr. 2003.

[61] K. Braekers, K. Ramaekers, and I. Van Nieuwenhuyse, "The vehicle routing problem: State of the art classification and review," *Comput. Ind. Eng.*, vol. 99, pp. 300–313, Sep. 2016.

[62] Y. Rochat and É. D. Taillard, "Probabilistic diversification and intensification in local search for vehicle routing," *J. Heuristics*, vol. 1, no. 1, pp. 147–167, Sep. 1995.

[63] I. H. Osman, "Metastrategy simulated annealing and tabu search algorithms for the vehicle routing problem," *Ann. Oper. Res.*, vol. 41, no. 4, pp. 421–451, 1993.

[64] R. Cordone and R. W. Calvo, "A heuristic for the vehicle routing problem with time windows," *J. Heuristics*, vol. 7, pp. 107–129, Mar. 2001.

[65] J. Berger and M. Barkaoui, "A new hybrid genetic algorithm for the capacitated vehicle routing problem," *J. Oper. Res. Soc.*, vol. 41, no. 2, pp. 179–194, 2003.

[66] G. D. Konstantakopoulos, S. P. Gayialis, and E. P. Kechagias, "Vehicle routing problem and related algorithms for logistics distribution: A literature review and classification," *Oper. Res.*, pp. 1–30, Sep. 2020. [Online]. Available: https://link.springer.com/article/10.1007/s12351-020-00600-7#citeas, doi: 10.1007/s12351-020-00600-7.

[67] R. Elshaer and H. Awad, "A taxonomic review of metaheuristic algorithms for solving the vehicle routing problem and its variants," *Comput. Ind. Eng.*, vol. 140, Feb. 2020, Art. no. 106242.

[68] C. J. Malmborg, "A genetic algorithm for service level based vehicle scheduling," *Eur. J. Oper. Res.*, vol. 93, no. 1, pp. 121–134, Aug. 1996.

[69] S. Jung and A. Haghani, "Genetic algorithm for a pickup and delivery problem with time Windows," *Transp. Res. Rec.*, vol. 1733, no. 1, pp. 1–7, Jan. 2000.

[70] D. K. Gupta, "A new algorithm to solve vehicle routing problems (VRPs)," *Int. J. Comput. Math.*, vol. 80, no. 3, pp. 267–274, Mar. 2003.

[71] C. R. Reeves, *Modern Heuristic Techniques for Combinatorial Problems*. Oxford, U.K.: Blackwell Scientific Press, 1993.

[72] H. W. Shin and M. K. Kang, "A heuristic for vehicle routing problem allowing multiple visits," *Soc. Korea Ind. Syst. Eng.*, vol. 14, no. 24, pp. 141–147, 1991.

[73] C. Archetti, N. Bianchessi, and M. G. Speranza, "A column generation approach for the split delivery vehicle routing problem," *Networks*, vol. 58, no. 4, pp. 241–254, Dec. 2011.

[74] L. Berbotto, S. García, and F. J. Nogales, "A randomized granular tabu search heuristic for the split delivery vehicle routing problem," *Ann. Oper. Res.*, vol. 222, no. 1, pp. 153–173, Nov. 2014.

[75] N. Bianchessi, M. Drexl, and S. Irnich, "The split delivery vehicle routing problem with time windows and customer inconvenience constraints," *Transp. Sci.*, vol. 53, no. 4, pp. 1067–1084, Jul. 2019.

[76] Z. Chen, M. Yang, Y. Guo, Y. Liang, Y. Ding, and L. Wang, "The split delivery vehicle routing problem with three-dimensional loading and time Windows constraints," *Sustainability*, vol. 12, no. 17, p. 6987, Aug. 2020.

[77] S. Ray, A. Soeanu, J. Berger, and M. Debbabi, "The multi-depot split-delivery vehicle routing problem: Model and solution algorithm," *Knowl.-Based Syst.*, vol. 71, pp. 238–265, Nov. 2014.

[78] M. Gen and R. Cheng, *Genetic Algorithms and Engineering Optimization*. Hoboken, NJ, USA: Wiley, 2000.

[79] G. Syswerda, "Uniform crossover in genetic algorithms," in *Proc. 3rd Int. Conf. Genetic Algorithms*, J. Schaffer, Ed. San Mateo, CA, USA: Morgan Kaufmann, 1989, pp. 2–9.

[80] K. Q. Zhu, "A diversity-controlling adaptive genetic algorithm for the vehicle routing problem with time windows," in *Proc. 15th IEEE Int. Conf. Tools Artif. Intell.*, Sacramento, CA, USA, Nov. 2003, pp. 176–183.

[81] S. Wang, Z. Lu, L. Wei, G. Ji, and J. Yang, "Fitness-scaling adaptive genetic algorithm with local search for solving the multiple depot vehicle routing problem," *Simulation*, vol. 92, no. 7, pp. 601–616, Jul. 2016.

[82] J. Li, F. Wang, and Y. He, "Electric vehicle routing problem with battery swapping considering energy consumption and carbon emissions," *Sustainability*, vol. 12, no. 24, p. 10537, Dec. 2020.

[83] G. Taguchi, S. Chowdhury, and S. Taguchi, *Robust Engineering*. New York, NY, USA: McGraw-Hill, 2000.

[84] Y. Wu, *Taguchi Methods for Robust Design*. New York, NY, USA: ASME, 2000.

[85] L. C. T. Gomes and F. J. Von Zuben, ''Multiple criteria optimization based on unsupervised learning and fuzzy inference applied to the vehicle routing problem,'' *J. Intell. Fuzzy Syst.*, vol. 13, pp. 143–154, Jan. 2002.

[86] R. Fukasawa, H. Longo, J. Lysgaard, M. P. D. Aragão, M. Reis, E. Uchoa, and R. F. Werneck, ''Robust branch-and-cut-and-price for the capacitated vehicle routing problem,'' *Math. Program.*, vol. 106, no. 3, pp. 491–511, May 2006.

[87] K. Ganesh and T. T. Narendran, ''CLOVES: A cluster-and-search heuristic to solve the vehicle routing problem with delivery and pick-up,'' *Eur. J. Oper. Res.*, vol. 178, no. 3, pp. 699–717, May 2007.

[88] S. Eilon, C. D. T. Watson-Gandy, and N. Christofides, *Distribution Management: Mathematical Modelling and Practical Analysis*. London, U.K.: Griffin, 1971.

[89] A. M. Altabeeb, A. M. Mohsen, and A. Ghallab, ''An improved hybrid firefly algorithm for capacitated vehicle routing problem,'' *Appl. Soft Comput.*, vol. 84, Nov. 2019, Art. no. 105728.

[90] A. M. Altabeeb, A. M. Mohsen, L. Abualigah, and A. Ghallab, ''Solving capacitated vehicle routing problem using cooperative firefly algorithm,'' *Appl. Soft Comput.*, vol. 108, Sep. 2021, Art. no. 107403.

[91] I. Sbai, S. Krichen, and O. Limam, ''Two meta-heuristics for solving the capacitated vehicle routing problem: The case of the Tunisian Post Office,'' *Oper. Res.*, pp. 1–43, Jan. 2020. [Online]. Available: https://link.springer.com/article/10.1007/s12351-019-00543-8#citeas, doi: 10.1007/s12351-019-00543-8.

HYUNPAE LIM received the B.S. degree from Korea Military Academy and the M.S. degree in industrial engineering from Oregon State University under Prof. Gyu M. Lee's guidance. He is currently a Lieutenant Commander of Korean Army. He served for the logistics operations in Korean Army. His research interests include logistics, military defense operations, metaheuristics, and operations research.

GYU M. LEE received the B.S. and M.S. degrees in industrial engineering from Seoul National University and the Ph.D. degree in industrial engineering from The Pennsylvania State University. He held a Professorship with the University of Arizona, Northeastern University, the University of Illinois, Oregon State University, and POSTECH, before he joined Pusan National University, where he is currently a Professor with the Department of Industrial Engineering. His research interests include the operations research, combinatorial optimization, data analytics, logistics and manufacturing, computational engineering, and information technology. He is currently the Editor-in-Chief of *International Journal of Industrial Engineering: Theory, Applications and Practice* (SCI) and an Associate Editor of *Decisions Science* (SSCI).

IVAN KRISTIANTO SINGGIH received the B.S. and M.S. degrees in industrial engineering from Bandung Institute of Technology, Indonesia, in 2009 and 2010, respectively, and the Ph.D. degree in industrial engineering from Pusan National University, Busan, South Korea, in 2017. He was a Postdoctoral Researcher and a Research Assistant Professor with the Department of Industrial and Management Engineering, Pohang University of Science and Technology (POSTECH). He is currently a Postdoctoral Researcher with the Department of Industrial and Systems Engineering, Korea Advanced Institute of Science and Technology (KAIST), South Korea. His research interests include container terminal logistics, vehicle routing problems, scheduling, and serious game development. He is a member of KIIE, T-LOG, and Korean AI Association. He received the Best Student Paper Award at the IJIE 2013 Conference and the Korean Government Scholarship Excellent Academic Achievement Award in 2014.