Computers & Industrial Engineering 146 (2020) 106571

Contents lists available at ScienceDirect

Computers & Industrial Engineering

journal homepage: www.elsevier.com/locate/caie

Hybrid differential evolution algorithm and genetic operator for multi-trip vehicle routing problem with backhauls and heterogeneous fleet in the beverage logistics industry

Kanchana Sethanan, Thitipong Jamrus*

Research Unit on System Modeling for Industry, Department of Industrial Engineering, Faculty of Engineering, Khon Kaen University, Khon Kaen 40002, Thailand

ARTICLE INFO

Keywords:
Vehicle routing problem
Backhauls
Multiple trips
Differential evolution algorithm
Genetic operator

ABSTRACT

Logistics is increasingly challenging because of increased competition and the uncertainty introduced by globalization. The drinks distribution system considered here uses glass bottles for soft drinks to deliver to all customers who need soft drinks in glass bottles, before making any pickups of empty glass bottles from clients to return to the company. This study aims at both an integer linear programming formulation and a novel hybrid differential evolution algorithm involving a genetic operator with fuzzy logic controller, for solving the multi-trip vehicle routing problem with backhauls and a heterogeneous fleet. The objective function is to minimize total cost, which is related to distance travelled. For validation, we designed numerical experiments to compare the proposed approaches with LINGO computational software, using the conventional differential evolution algorithm and differential evolution with selected genetic operator and fuzzy logic controller in real settings. The experimental results demonstrate the practical viability of the proposed approaches.

1. Introduction

In the current business environment, factories are encountering pressure not only to make their operations more efficient and reduce production costs, but also to provide good service quality to clients. This pressure also occurs significantly in logistic companies, which need to improve their related logistics and planning under changing client needs. Logistics is increasingly challenging because of changes in clients' demand, rapid technological change, increasing competition and uncertainty introduced by globalization (Jamrus, Wang, & Chien, 2020). The recent advances in transport systems have motivated logistics companies to better manage the planning in their companies. Because of evolving client demand, logistic companies require an efficient delivery service with good service quality, while maintaining business profitability.

Food and beverages industries are increasing in importance from day to day. In particular, the beverage industry is one of the major contributors to the growth of global economics. The global beverages industry is expected to reach an estimated $1.9 trillion by 2021, and is forecast to grow at a compound annual growth rate of 3.0% from 2016 to 2021 (Reportlinker, 2019). However, distribution costs are important and can account for up to 70% of the value added costs of products in the food and beverages industry (De Backer, Furnon, Kilby, Prosser & Shaw, 1997). Also, the transportation cost is one elemental cost in the logistics business that has dominated the total cost. The beverage logistics company in this case study provides logistics services for alcoholic and non-alcoholic beverages. This company provides warehousing, transportation and distribution services. In particular, transportation and distribution services are the main total value-added costs of products for the company. The company transports soft drinks to customers, who are agents, distributors and restaurants. It likes to distribute soft drinks in glass bottles to all clients who need soft drinks in glass bottles, before making any empty glass bottle pickups back from clients to the company. There are two truck sizes for delivery and pickup of the soft drink glass bottles, and each truck may perform several routes within a single planning period. Therefore, each client has a known demand for delivery or pick up and is serviced by a vehicle of a heterogeneous fleet of capacitated trucks within a given time period. There are many transportation costs that are vital to logistic management. They can be reduced while maintaining a competitive pricing advantage and retaining profitability.

The vehicle routing problem (VRP) is one of the optimization problems faced most regularly in logistics; it determines the optimal sequence for clients to be visited by each vehicle. It aims to minimize the cost of transportation operations by a fleet of vehicles delivering to clients. The VRP with multiple trips (MTVRP) is characterized by an

* Corresponding author.
E-mail address: thitja@kku.ac.th (T. Jamrus).

https://doi.org/10.1016/j.cie.2020.106571

Available online 29 May 2020
0360-8352/

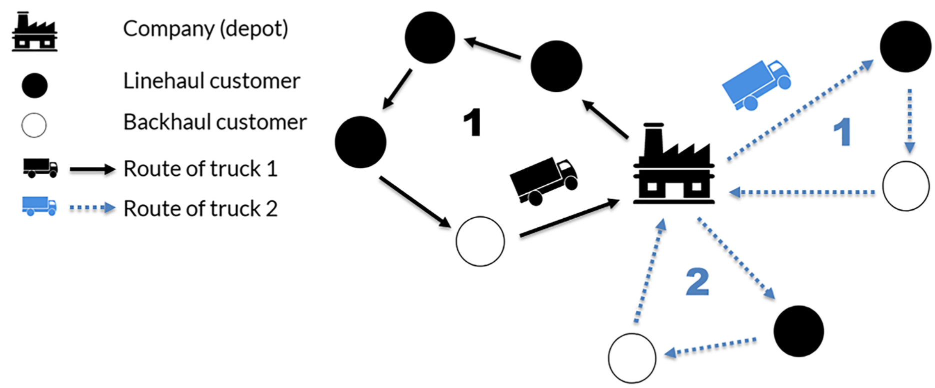

Fig. 1. An example of the MTVRPB.

optimized set of vehicles and drivers working multiple routes or trips within a given time period (Brandão & Mercer, 1997). In addition, the vehicle routing problem with backhauls (VRPB) is one in which a vehicle may pick up products to carry them back to the depot after deliveries have been made. The client set of the VRPB can be classified into linehaul and backhaul customers. Each linehaul customer requires a given quantity of goods from a central depot, and a given quantity of goods is collected from each backhaul customer and returned to the depot. The backhauls must be visited after the linehauls in each route (Koç & Laporte, 2018). In this paper, the case study is the multi-trip vehicle routing problem with backhauls (MTVRPB), which can be described as a VRP problem with backhauling and multiple trips in a single planning period with a heterogeneous fleet, such that the total travel distance is minimized as shown in Fig. 1. The challenge for the backhaul beverage industry is how to deliver the soft drinks in glass bottles to all customers that need them, before making any empty glass bottle pickups from clients to return them to the company. In order to do so, the appropriate route for sequencing clients to be visited is determined for each vehicle, so that all delivery clients are served before any pickup ones, while optimizing the set of vehicles and drivers working multiple routes or trips within a given time period and with a heterogeneous fleet. The objective function is to minimize total cost, which is related to the distance travelled.

Many studies have addressed the single trip vehicle routing problem with backhaul and homogeneous fleet. Also, these problems have been solved using the simplicity of traditional metaheuristics. To fill the gap, this study proposes an integer linear programming formulation (ILP) and a hybrid differential evolution algorithm and genetic operator with fuzzy logic controller (FLC), called HDEGO, to identify an appropriate route for sequencing of clients to be visited by each vehicle, so that all delivery clients are served before any pickup ones. In order to improve the solution quality in terms of travelling cost, through enhancing the diversity of vectors for exploring the solution space, selection of the vector representation can obviously be improved using a genetic operator. The genetic algorithm selection for the next generation is significant to the problem, because of the derived optimized solution. The elitism method is applied, which is the best of all selection methods. It is a selection strategy where a limited number of individuals with the best fitness values are chosen to pass to the next iteration. The limited number is the selected lowest number in an elitist way. Thus, this study will have the advantage of genetic operators applied with the elitism method to replace the traditional selection operator of DE, which is similar to a tournament selection feature. Moreover, an adaptive auto-tuning of the DE algorithm can adaptively regulate F and CR values during mutation and recombination operators. To illustrate the proposed method's effectiveness, numerical experimental results were compared with the mathematical model and with the traditional DE. The next section provides a review of the relevant literature. The problem formulation is described in Section 3, and the HDEGO is presented in Section 4. Section 5 outlines computational experiments and results. Finally, a summary is given in Section 6.

2. Literature review

The VRP was first proposed by Dantzig and Ramser (1959). It has been extensively studied in the literature for several and varied applications. All the clients correspond to deliveries and the demands are known in advance and probably are not split (Liu, Li, & Liu, 2017). Many researchers have given state of the art reviews of VRP and approaches developed in the past (Eksioglu, Vural, & Reisman, 2009; Braekers, Ramaekers, & Van Nieuwenhuyse, 2016). For example, Shimizu and Sakaguchi (2013) proposed a hierarchical procedure that designed an economically efficient VRP for reverse logistics networks, and revealed some ways to reduce carbon dioxide emissions from the VRP in the same framework. Then, they extended the complexity to multi-depot VRPs that gave a more general framework for various real world applications, including those in green logistics (Shimizu & Sakaguchi, 2014).

The MTVRP is similar to the VRP but has a larger number of constraints. It has a set of vehicles and drivers working multiple routes or trips within a given time period. Brandão and Mercer (1997) first proposed certain assumptions: during each day a vehicle can make more than one trip and drivers' schedules must respect the maximum legal driving time per day and the legal time breaks. Cattaruzza, Absi, Feillet, and Vidal (2014) proposed a hybrid genetic algorithm for the MTVRP that allows for a diversified exploration over the search space, due to the management of several solutions at the same time. Also, François, Arda, Crama, and Laporte (2016) presented the problem in which each vehicle can deliver several routes during the same working shift to customers. They compared approaches between a heuristic that makes use of specific operators designed to tackle the routing and large neighborhood search (LNS) to perform the MTVRP. Their approach improved the results obtained with a given LNS algorithm.

The VRPB is also an extension of the classical VRP. Further constraints include: vehicles have to deliver to all the linehaul customers before visiting any backhaul customers; while routes with only backhauls are disallowed, routes with only linehauls can be delivered. Koç and Laporte (2018) comprehensively reviewed the existing up-to-date literature on the VRPB, including models, heuristic and metaheuristic approaches for solving the VRPB applications. There are many constraints considered, such as heterogeneous fleet, time windows, multi-depot and mixed backhauls. For example, Ong & Suprayogi (2011) considered VRPB with time windows and homogeneity that was solved by the ant colony optimization, minimizing the number of vehicles, the total duration time and the range of duration times. Salhi, Wassan, and Hajarat (2013) proposed a heuristic algorithm for solving the VRPB with an unlimited number of vehicles and a heterogeneous fleet. They showed that the mathematical model can solve for small-size instances, but large-size instances were solved using upper and lower bounds. García-Nájera, Bullinaria, and Gutiérrez-Andrade (2015) studied multi-objective VRPB that used a similarity-based selection multi-objective evolutionary algorithm, minimizing travel cost, number of routes and uncollected backhauls.

However, little literature is available on the MTVRPB. Yu and Qi (2014) presented the MTVRPB for the homogenous fleet of an express delivery company, with multiple-delivery and pickup customer visits each day and multiple trips per vehicle. They proposed modeling as a mixed ILP and developed the tabu search algorithm. The tabu search algorithm outperformed others, getting better solutions in a much shorter time than other heuristics, which also resulted in significant cost saving for the company. In addition, Wassan, Wassan, Nagy, and Salhi (2017) presented the MTVRPB with a homogenous fleet, with the objective to minimize the total cost by reducing the total distance travelled and the suitable number of vehicles used. They proposed an ILP for small and medium instances. For large instances of the MTVRPB, a two-level variable neighborhood search algorithm was developed and used, which produced efficient results for the VRPB. Thus, many metaheuristic approaches have been published with novel approaches and practical effectiveness.

The metaheuristic approach is mainly used owing to its ability to be more intensive and use robust methodologies to solve problems. Potvin, Duhamel, and Guertin (1996) determined the VRPB by a greedy route construction heuristic and a genetic algorithm, so that it ordered good routes. The objective is to minimize the total distance traveled, while satisfying backhauls and time window constraints. Both approaches were outperformed when these were combined. Zhong and Cole (2005) proposed a guided local search heuristic to solve a vehicle routing problem with backhauls and time windows. With this problem, the mathematical models or optimal methods are not appropriate, since the VRPB is an NP-hard problem and includes several constraints, and does not simplify the problem. Xiao, Zhao, Kaku, and Mladenovic (2014) also proposed a metaheuristic approach that is a variant of the variable neighbourhood search, combined with simulated annealing for solving the VRP. The results show that the metaheuristic approach outperformed most general algorithms in terms of computational effectiveness and efficiency. Küçükoğlu and Öztürk (2014) presented a differential evolution algorithm (DE) to solve a vehicle routing problem with backhauls and time windows and applied it to a catering firm. Also, Wisittipanich and Hengmeechai (2015) used a multi-objective DE for truck scheduling and door assignment in a multi-inbound and outbound doors cross docking problem with the Just-In-Time concept. The DE is a population-based search technique that uses simple operations on a generated vector set. There are operations that consist of initial vector, mutation, crossover and selection to create new vectors and competitive solutions. The highlighted difference from differential evolution is combining several solutions with the candidate solution (Kachitvichyanukul, 2012). Even though differential evolution has been used effectively in a variety of fields, it has been very limited when applying it to solve the VRP, especially the MTVRPB, because the encoding and decoding form of DE cannot be directly applied for the MTVRPB. Hence, the complexity of the MTVRPB with a heterogeneous fleet needs an efficient solution for minimizing the total distance travelled. We propose the HDEGO, not only applied with the elitism method to replace the traditional selection operator of the DE, but including adaptive auto-tuning of the DE algorithm, which can also regulate parameters during mutation and recombination operators. We aim to solve the beverage logistic transportation problem by improving the solution quality in terms of travelling cost.

3. Mathematical model for MTVRPB

The MTVRP characteristics are explained as follows: (i) A set of clients is divided into two types, linehaul and backhaul customers; (ii) The fleet of trucks is heterogeneous; (iii) A truck may deliver more than one trip in a single planning period; (iv) All delivery clients are delivered before any pickup from backhaul customers; (v) Trucks are not allowed to pick up only backhauls on routes, but linehaul deliveries alone are allowed; (vi) Truck capacities are of two types. The notations used in this paper are summarized as follows.

Index	
i, j	index of customers ($i, j = 1,2,\dots, N$)
k	index of trucks ($k = 1,2,\dots, K$)
Parameters	
θ	depot or company ($i, j = 1$)
L	set of linehaul customers
B	set of backhaul customers
N	number of customers, $N \in \{\theta, L, B\}$
K	number of trucks
m	maximum working period
c_k	capacity of truck k
d_{ij}	distance between location of customers i and j
q_i	demand of customer i
Decision variables	
b_{ik}	decision variable for sub-tour
s_{ij}	amount of deliveries or pickups in arcs i and j
x_{ijk}	$=1$, if truck k runs from customer i to j = 0, otherwise

The proposed model was formulated based on an ILP for MTVRPB. The objective is to minimize cost, which relates to the total distance travelled, as follows.

$$\min \sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{k=1}^{K} d_{ij} \times x_{ijk} \tag{1}$$

subject to

$$\sum_{j=1}^{N}\sum_{k=1}^{K} x_{jik} = 1 i \in \{L, B\} \tag{2}$$

$$\sum_{j=1}^{N}\sum_{k=1}^{K} x_{ijk} = 1 i \in \{L, B\} \tag{3}$$

$$\sum_{j=1}^{N} x_{jik} - \sum_{j=1}^{N} x_{ijk} = 0 \ \forall\ k,\ i \in \{L, B\} \tag{4}$$

$$\sum_{i \in L} s_{ij} - q_i = \sum_{i=1}^{N} s_{ji} j \in \{L\} \tag{5}$$

$$\sum_{i \in \{L,B\}} s_{ij} + q_i = \sum_{i=\{\theta,B\}} s_{ji} j \in \{B\} \tag{6}$$

$$s_{ij} \le c_k \times x_{ijk}\ \forall\ k,\ i \in \{L, B\}, j \in \{L, B\} \tag{7}$$

$$\sum_{i=1}^{N}\sum_{j=1}^{N} d_{ij} \times x_{ijk} \le m\ \forall\ k \tag{8}$$

$$b_{ik} \ge b_{jk} + q_i - c_k + c_k(x_{ijk} + x_{jik}) - (q_i + q_j) \times x_{ijk}\ \forall\ i, j, k \\ , i \ne j, j \ne \theta\ \forall \text{ i, j, k, i} \ne \text{j, j} \ne \theta \tag{9}$$

$$b_{ik} \le c_k - (c_k - q_i) \times x_{1ik}\ \forall\ i, k \tag{10}$$

$$b_{ik} \ge q_i + \sum_{j \in \{L,B\}} q_j \times x_{jik}\ \forall\ i, k \tag{11}$$

$$s_{ij} = 0 i \in \{L\}, j \in \{\theta, B\} \tag{12}$$

$$x_{ijk} = 0\ \forall\ k,\ i \in \{B\}, j \in \{L\} \tag{13}$$

$$x_{ijk} = 0\ \forall\ k,\ i \in \{\theta\}, j \in \{B\} \tag{14}$$

$$s_{ij} \ge 0\ \forall\ i, j \in \{L, B\} \tag{15}$$

$$x_{ijk} = \{0, 1\}\ \forall\ i, j, k \tag{16}$$

The objective function (1) is to minimize total cost, which is the distance travelled. Constraints (2) and (3) ensure that every customer is delivered to or picked up from exactly once. Constraint (4) imposes that the number of times truck k visits customer i is the same as the number of times it departs from customer i. Constraints (5) and (6) impose the truck load change on a route for linehaul and backhaul customers, respectively. Inequality (7) limits the maximum truck capacity constraint (heterogeneous fleet) that can serve the routes. The maximum working period constraint depends on distance, which is ensured by Inequality (8). Inequalities (9), (10) and (11) ensure sub-tour conditions. Constraint (12) disallows any load carried from a linehaul customer to be either a backhaul trip or to the depot. Constraints (13) and (14) ensure all delivery customers are delivered to before any pickup is made from backhaul customers. Also, trucks are not allowed to pick up only backhauls on routes, but only linehaul deliveries are allowed. Inequality (12) sets s_{ij} as a non-negative variable. Constraint (13) is the binary variable constraint.

4. The HDEGO for MTVRPB

The hybrid differential evolution algorithm and genetic operator with fuzzy logic controller, called HDEGO, has the following steps.

4.1. Initial solution

The DE is an indirect approach for solving some VRPBs used by Küçükoğlu and Öztürk (2014). They used two blocks in each vector to denote the customer and the vehicle number. However, the vectors of MTVRPB are designed to support both linehaul and backhaul customers, and truck types under a single planning period. The number of the population is the population size or the number of vectors, where each vector consists of a customer vector and a truck type vector. Firstly, each customer vector will be randomly generated, equal to the number of linehaul and backhaul customers. The decryption sorts the rank order value of each vector, in ascending order, so that it obtains the sequence of customers to be visited. An additional vector is the truck type vector, which will also be randomly generated and equal to the number of customers. However, the decoding will design size numbers to be the same as the truck types, and there are two truck types in this study. After the encoding section, the sequences of customers are sorted in accordance with customer specifications, so that each linehaul customer must be placed before a backhaul customer. Fig. 2 shows an illustrative example of vector construction. There are five linehaul customers and three backhaul customers. For example, customer vector: 0.78–0.52–0.31–0.93–0.12–0.81–0.33–0.56. The customer ID is 1–2 − 3–4 − 5–6 − 7–8. After it is sorted in ascending order, it is obtained the customer vector: 0.12–0.31–0.33–0.52–0.56–0.78–0.81–0.93. So, the customer ID is 5–3 − 7–2 − 8–1 − 6–4. The initial solution was processed and sorted to obtain sequences of truck routes by prioritized linehaul customers and backhaul customers: truck type 1 is 0–5–2–7–6–0 and truck type 2 is 0–3–1–4–8–0, where 0 is the depot. In addition, these vectors consider the maximum working period constraints. If the total working period of a route is more than the maximum working period, then the route will use more trucks to serve this route to meet the demand fully within the maximum working period. The route is split so that the maximum working period is not exceeded. The accumulated working time starts from the depot until it reaches the maximum working period. For example, the truck type 1 route is 0–5–2–7–6–0 in Fig. 2 and there is the maximum working period. If the truck type 1 drives to node 2, then the accumulated working time is more than the maximum working period when it will drive to node 7. Thus, the truck type 1 will drive back to depot (0–5–2–0) and the split route (0–7–6–0) will be given to another truck.

In the multi-trip case, the accumulated working time of any truck is less than the maximum working period when the truck arrives at the depot. If there is working time remaining for one more trip, customers will be delivered to. For example, the truck type 1 route is 0–5–2–7–6–0 in Fig. 2. If the accumulated working time is still less than the maximum working period (0–5–2–0), then the truck could deliver more trips until the maximum working period is achieved (0–5–2–0 and 0–7–6–0). The objective is to minimize cost, which relates to the total distance travelled. Thus, the total distance of travelled truck type 1 is calculated two times, for the routes 0–5–2–0 and 0–7–6–0.

4.2. Differential evolution algorithm (DE)

The DE algorithm was first introduced by Storn and Price (1997). The original DE has four methods consisting of the generations of initial solution, mutation, recombination and selection. In this study, a solution could not be found using an optimization program with large scale problems, because the number of variables is excessive, and the problem is NP-hard and highly complex, so we use the DE algorithm that finds optimal solutions from initial solutions (Section 4.1).

4.2.1. Mutation operation

Vector number (NP) is the number of vectors that is used for DE iterations. NP vectors will be randomly generated to an initial solution as in Section 4.1. Then, the mutation operation is the second process of the DE mechanism. Eq. (17) is used to randomly combine three selected vectors into a mutant vector. The scaling factor F is a constant from [0, 2]. $V_{i,G+1}$ is called the mutant vector and $X_{r1,G}$, $X_{r2,G}$, $X_{r3,G}$ are random vectors.

$$V_{i,G+1} = X_{r1,G} + F(X_{r2,G} - X_{r3,G}) \tag{17}$$

4.2.2. Recombination operation

The recombination or crossover operator incorporates successful solutions from the previous iteration. The trial vector ($U_{ji,G+1}$) is obtained from Eq. (18). Each position value in a vector can be the position value of a target or trial vector, depending on a random number (*Rand*) that is generated for that position, and compared with the crossover rate (*CR*). The value of an indicated position will be replaced from the position value of that position in the target vector ($X_{ji,G}$), if the random number for that position is greater than the *CR* value. On the other hand, if the random number generated for that position is lower than or equal to the *CR* value, the value of the indicated position will be replaced from the position value of that position in the mutant vector ($V_{ji,G+1}$), as represented in Eq. (18).

$$U_{ji,G+1} = \begin{cases} V_{ji,G+1} if \quad (Rand(j) \le CR) \\ X_{ji,G} if \quad (Rand(j) > CR) \end{cases} \tag{18}$$

4.2.3. Selection operation

The traditional selection operation of DE is like tournament selection, which is a method of selecting an individual from a population of individuals in a genetic algorithm. The target vector ($X_{ji,G}$), is compared with the trial vector ($U_{ji,G+1}$), and the one with the lowest objective value is selected to be the target vector for the next iteration ($X_{ji,G+1}$). Equation (19) represents the selection formula.

$$X_{ji,G+1} = \begin{cases} U_{ji,G+1} if \quad (U_{ji,G+1} < f(X_{ji,G})) \\ \quad X_{ji,G} \quad otherwise \end{cases} \tag{19}$$

4.3. Genetic operator

Genetic algorithm (GA) is an influential technique that has been applied in various research fields. GA has been widely employed for solving optimization problems. In GA, the fitness of every individual in a population is evaluated. GA searches from a population of points; therefore it can avoid being trapped in a local optimal solution. GA can solve all optimization problems that can be described through

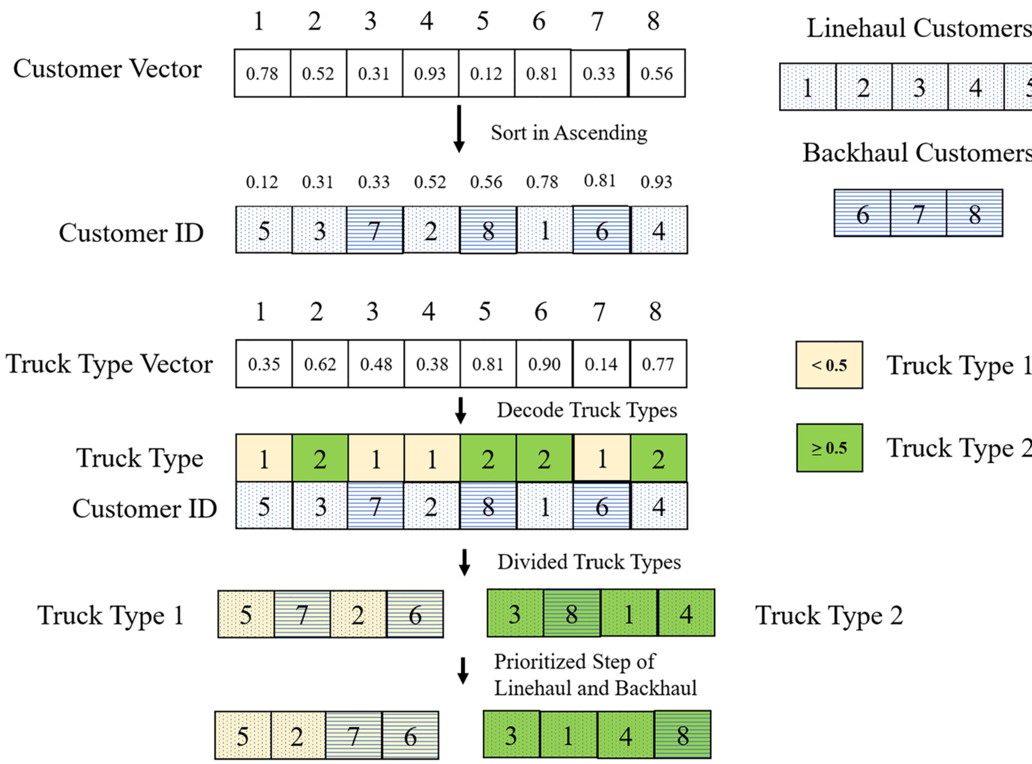

Fig. 2. An illustrative example of vector construction.

chromosome encoding; moreover, they provide multiple solutions and use fitness functions, which are obtained from objective functions, without the need for other derivative or auxiliary information. Moreover, they can be easily transferred to existing simulations and models (Jamrus & Chien, 2016). Although the GA has been used in a variety of research areas, it has limitations when applied for solving a problem, as the solution process may not be complex enough to find the optimal solution. In order to improve the diversity of DE, the hybridization of the DE with GA has been used to enhance the performance of the traditional DE and GA.

The GA selection for the next generation is significant to the problem because of the derived optimized solution. Panchal, Chudasama & Shah (2011) presented a comparison of selection operators: roulette wheel, elitism and tournament selection methods for the travelling salesman problem. They found that the elitism method is the best of all these methods. The elitist selection is a selection strategy where a limited number of individuals with the best fitness values are chosen to pass on to the next iteration. Thus, in this study the elitism method is applied to replace the traditional selection operator of DE, which is similar to the tournament selection feature. The target vectors are compared with the trial vector elitist selection that are selected to be the target vectors of the next iteration, as shown in Fig. 3. All target vectors and trial vectors were sorted by fitness value, and the best fitness values selected to pass on to the next iteration.

4.4. Adaptive auto-tuning by fuzzy logic controller (FLC)

The conventional DE algorithm may not guarantee an optimal solution in all cases, because the DE uses an unknown scaling factor (F) and crossover rate (CR). Adaptive auto tuning of the DE algorithm can adaptively regulate F and CR values during mutation and recombination operations. Yun and Gen (2003) and Jamrus, Chien, Gen, and Sethanan (2015) proposed a heuristic updating strategy for crossover and mutation rates of the genetic algorithm, to consider changes of average fitness in the population of two continuous generations. Therefore, this study uses the applied adaptive auto tuning for the DE

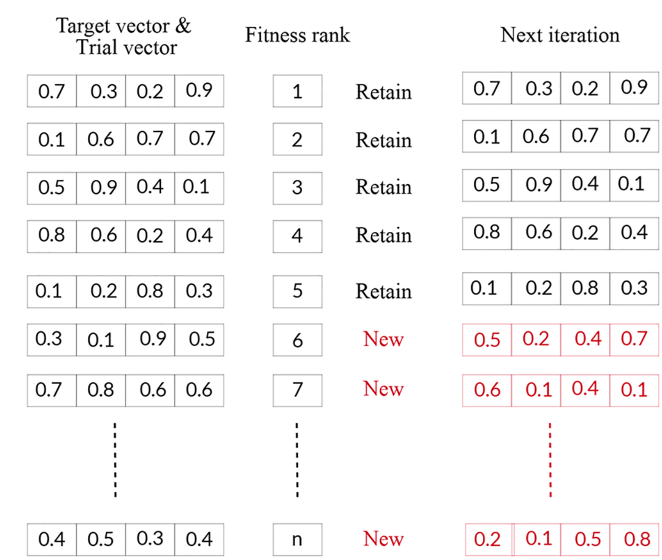

Fig. 3. An illustrative example of elitist selection.

algorithm that will control F and CR values during mutation and recombination processes. For the minimization problem, the change of the average fitness at iteration t, $\Delta favg(t)$, is as shown in Eq. (20). The *vecsize* is the vector size that satisfies constraints, and *trisize* is the trial vector size that satisfies constraints.

$$\Delta f_{avg}(t) = \bar{f}_{vecsize}(t) - \bar{f}_{trisize}(t) = \frac{1}{vecsize}\sum_{k=1}^{vecsize} f_k(t) - \frac{1}{trisize}\sum_{k=1}^{trisize} f_k(t). \quad (20)$$

4.5. The HDEGO algorithm

The algorithm first defines the initial solution sectns such as

Procedure: Hybrid differential evolution with genetic operator algorithms (HDEGO)
Input :MTVRPB data, DE parameters (CR, F, NP)
Output : Optimal solution
Begin :
Encoding step: randomly generate a set of target vectors *n* (n =1…NP); //*n* is the index of the vector representation; *NP* is a predefined number of population
while termination condition is not satisfied **do**
for *n* =1 to *NP* //*NP* is the predefined number of population
Solution Initialization (section 4.1)
Decoding and Evaluation (section 4.1)
Mutation Operation (section 4.2)
Recombination Operation (section 4.2)
Elitist Selection Operation (section 4.3)
Tune Parameters by Adaptive Auto Tuning (section 4.4)
end
end
end;

Fig. 4. Overall HDEGO algorithm procedure.

encoding and decoding vectors for MTVRPB. Each vector consists of a customer vector, linehaul or backhaul customer, and a truck type vector, type 1 or type 2. The objective is to minimize cost, which is the total distance travelled (Section 4.1). Then, DE is used with four methods consisting of generation of initial solution, mutation, recombination, and selection. This study uses the elitism method to replace the traditional selection operator of DE (Sections 4.2–4.4). Finally, the adaptive auto-tuning of the DE algorithm can adaptively regulate *F* and *CR* values during mutation and recombination operators (Section 4.4). The HDEGO algorithm procedure is shown in Fig. 4.

5. Computational experiments

To demonstrate the efficiency and effectiveness of the HDEGO in MTVRPB, the DE parameters are designed as tuning parameters according to the adaptive auto-tuning routine for *F* and *CR*. The numerical experiments are 10 sets with *vecsize* = 50 and *maxiter* = 500. The proposed algorithms were run using MatLab on a 2.10 GHz PC, with 8 GBytes of RAM, for testing and evaluation. For illustration, we generated 10 problems (Table 1) involving various numbers of maximum working periods and linehaul and backhaul customers. Each demand for customers was generated in the value interval for pallets [2, 10] for each problem. There are two truck sizes, 40 and 50 pallets. For instance 1, the number of linehaul and backhaul customers are set at three and two, respectively. The distance between linehaul and backhaul customers is shown in Table 2. In addition, linehaul customer demand is 6, 5 and 7 pallets and backhaul customer demand is 7 and 2 pallets. Table 3 presents the solution for instance 1 in which two large trucks are used with maximum working period by optimal solution. The problems were formulated as an integer programming model for the multiple trip vehicle routing problem with backhauls and heterogeneous fleet, as shown in Table 4.

The results of the test problems for deriving an optimal solution follow, made by generating seven problems and presenting total travel distance and computational time for each problem. For instances 1 to 4, the computational time increases dramatically depending on the problem size, since the problem is an NP-hard problem. Instances 5 to 7 yielded a high number of variables, in which the MTVRPB problem was NP-hard and highly complex. Also, the freight forwarder accepts computational time which is less than 360 min (6 h), so that limits the sequence of customers to be visited by each vehicle. Instances 5 to 10 cannot be accepted by the mathematical model. Therefore, this paper proposes that the HDEGO can determine the sequence of customers to be visited by each vehicle for satisfying customer demands and solving both small and large problems.

We tested the traditional DE, GA and HDEGO performance using seven test problem instances, the same as the ILP test. The computational tests compared the best and average total cost of each solution for solving each problem by 10 runs, and the results of the optimal

Table 1
Test problem sizes.

Instance	No. of Linehauls	No. of Backhauls	Max. working period
1	3	2	14
2	6	3	14
3	9	3	14
4	11	4	14
5	15	5	16
6	18	5	16
7	25	6	16
8	25	8	18
9	32	10	18
10	40	12	18

Table 2
Example of distance matrix [km]

Customer	1	2	3	4	5	6
1	–	2	4	3	5	8
2	2	–	3	6	4	7
3	4	3	–	5	3	1
4	3	6	5	–	7	3
5	5	4	3	7	–	4
6	8	7	1	3	4	–

Table 3
Results for instance 1 [km].

Instance 1	Results	
Total travel distance (km)	24	
Computational time (minutes)	0.05	
Trucks used	Two large trucks	
Sequences of customers	Truck 1	Trip 1: 0–1–0
		Trip 2: 0–3–0
	Truck 2	Trip 1: 0–2–5–4–0

Table 4
The best total cost from solving by ILP.

Instance	Total travel distance [km]	Computational time [min]
1	24	0.05
2	34	2.3
3	39	20.6
4	36	814

Table 5
Comparison of the best and average total distances of each solution from solving each problem.

No.	Optimal solution by LINGO		Solution by DE				Solution by GA				Solution by HDEGO			
	Best solution [km]	CPU time [min]	Avg. [km]	Std.	Best solution [km]	CPU time [sec]	Avg. [km]	Std.	Best solution [km]	CPU time [sec]	Avg. [km]	Std.	Best solution [km]	CPU time [sec]
1	24	0.05	24.0	0	24	6.4	24.0	0	24	6.4	24.0	0	24	6.6
2	34	2.3	35.0	0.3	34	8.7	34.9	0.2	34	8.8	34.5	0.2	34	9.1
3	39	20.6	41.2	0.3	39	11.3	40.9	0.3	39	11.5	40.1	0.2	39	11.3
4	36	814	40.0	0.3	39	12.8	39.7	0.3	38	12.9	38.2	0.4	36	13.1
5	–	–	75.3	1.2	69	16.4	74.4	0.9	70	16.8	71.1	0.8	66	17.1
6	–	–	78.5	2.3	76	18.4	79.2	1.9	78	18.9	75.7	1.1	73	19.3
7	–	–	113.1	3.1	107	23.8	115.2	3.3	105	24.1	109.9	1.5	102	24.4
8	–	–	134.8	3.7	126	45.8	133.3	3.5	125	50.4	125.1	2.0	119	55.6
9	–	–	158.9	3.8	141	89.2	157.2	3.5	144	98.0	141.0	2.0	132	101.3
10	–	–	179.2	3.9	159	150.3	185.8	3.6	168	172.7	155.6	2.7	145	181.2

Table 6
Results of ANOVA analysis.

No.	Factor		Total distance [km]		% improvement	No.	Factor		Total distance [km]		% improvement
	instance	% fluctuation	DE	HDEGO			instance	% fluctuation	DE	HDEGO	
1–5	1	10	24.0	24.0	0.00	56–60	4	50	41.0	38.8	4.93
6–10	1	30	24.0	24.0	0.00	61–65	5	10	79.6	69.4	12.70
11–15	1	50	24.0	24.0	0.00	66–70	5	30	76.2	71.0	6.71
16–20	2	10	34.4	34.2	0.55	71–75	5	50	72.0	70.0	2.66
21–25	2	30	34.6	34.2	1.14	76–80	6	10	79.4	72.2	8.98
26–30	2	50	34.6	34.4	0.55	81–85	6	30	82.6	76.2	7.70
31–35	3	10	41.0	39.4	3.83	86–90	6	50	80.2	72.8	9.17
36–40	3	30	40.0	39.4	1.46	91–95	7	10	118.2	108.4	8.09
41–45	3	50	40.2	39.2	2.45	96–100	7	30	118.6	107.8	8.84
46–50	4	10	41.6	38.4	7.63	101–105	7	50	119.0	109.6	7.61
51–55	4	30	39.0	38.2	10.00	Average					4.97

solution, DE GA and HDEGO for MTVRPB are shown in Table 5. The DE, GA and HDEGO determined that the best total distance equals the optimal solution for instances 1 to 3; nevertheless, the HDEGO gave the same result as the optimal solution compared with the DE and GA for instance 4. Moreover, the HDEGO obtained best and average solutions better than the DE and GA for instances 5 to 10. The proposed HDEGO outperformed the DE and GA, particularly for medium and large problem sizes. The time here is the CPU time in seconds for every run, and in the example of the HDEGO experiment, determining the solution used slightly more CPU time than DE and GA for instances 1 to 10. The large problem used CPU time of about three minutes. On the other hand, the optimal solution is 814 min (about 13.5 h) and the freight forwarder cannot accept this computational time. Moreover, the HDEGO determined that the total distance equals the optimal solution and has enhanced results compared with the traditional DE and GA, whereas in the comparison results between the DE, GA and HDEGO for each problem, the HDEGO spends fewer iterations than the number of generations in the DE for finding the best solution for large problems (Instances 5–10). The proposed HDEGO outperforms the traditional DE for large problems. Time is the CPU time in seconds for every run, and the example of the HDEGO experiment facilitated determining the best solution at 11.9 s at the 18th iteration in instance 5. However, the CPU time of EP-DPSO found a solution that is less than the nEP-HGA, in which the best solution is at 24.3 s at the 26th iteration. In addition, the CPU time and times for HDEGO and DE runs are 167.3 s (74th iteration) and 174.8 s (115th iteration), respectively for instance 10. However, in small problems such as instances 1 to 4, the results of HDEGO and DE are similar. However, they can determine which real world demands must be optimized for long-term efficient operation of the entire transportation system. Thus, the HDEGO can determine which beverage logistic must be optimized for efficient operation of the backhaul transportation.

The solutions from using the HDEGO were compared with those from using the traditional DE procedures, which were based on factorial designs according to factor levels. The combination experiments were conducted in quintuplicate. The parameters included the number of problems (Table 1) and the fluctuation of customer demand (10, 30 and 50% of customer demands, which were an average demand). The samples totaled 105, and the results are indicated by the percentage improvement between DE and HDEGO as shown in Table 6.

Table 6 shows 4.97% improvement from the MTVRPB model when the data were analyzed using ANOVA. The percentage improvement differed significantly from actual situation practices at the 95% reliability level, and p was less than 0.05. The parameter explaining this difference was not the fluctuation of demand that yielded the average improvement of DE and HDEGO. Nevertheless, most HDEGO approaches outperformed, getting better total distance solutions than traditional DE, and providing significant transportation cost saving for the company.

6. Conclusions

The current transportation environment is becoming increasingly complex, which leads to it being increasingly more competitive. Moreover, the global beverage industry is forecast to grow continuously at a compound annual growth. This study has proposed an integer linear programming formulation and a HDEGO to identify an appropriate route for sequencing customers to be visited by each vehicle of a beverage logistics company, so that all delivery customers are served before any pickup ones. The HDEGO was not only applied with the elitism method to replace the traditional selection operator of the DE, but adaptive auto tuning of the DE algorithm can also regulate parameters during mutation and recombination operators. The experimental results show that the proposed HDEGO outperformed the DE by about 4.97%, with average improvements for all the problem instances (Table 6). The results demonstrate the practical viability of the proposed approach, which can support freight forwarder and customers. A future study can be done to extend the proposed HDEGO approach to allow for more constraints, such as customers' time windows and mixed VRPB in a real setting. For example, stochastic customer service demands quantity, a load splitting constraint and multiple depots. Also, more studies can be done to develop approaches for multiple objectives in the beverage logistic industry.

CRediT authorship contribution statement

Kanchana Sethanan: Conceptualization, Writing - review & editing. **Thitipong Jamrus:** Methodology, Validation, Software, Formal analysis, Writing - original draft.

Acknowledgements

This work was supported by the Research Unit on System Modeling for Industry (Grant No. SMI.KKU 63003), Khon Kaen University, Thailand. The authors would also like to thank Mr. Ian Thomas for critical review of the manuscript.

References

Braekers, K., Ramaekers, K., & Van Nieuwenhuyse, I. (2016). The vehicle routing problem: State of the art classification and review. *Computers & Industrial Engineering, 99*, 300–313. https://doi.org/10.1016/J.CIE.2015.12.007.

Brandão, J., & Mercer, A. (1997). A tabu search algorithm for the multi-trip vehicle routing and scheduling problem. *European Journal of Operational Research, 100*(1), 180–191. https://doi.org/10.1016/S0377-2217(97)00010-6.

Cattaruzza, D., Absi, N., Feillet, D., & Vidal, T. (2014). A memetic algorithm for the multi trip vehicle routing problem. *European Journal of Operational Research, 236*(3), 833–848. https://doi.org/10.1016/J.EJOR.2013.06.012.

Dantzig, G. B., & Ramser, J. H. (1959). The truck dispatching problem. *Management Science, 6*(1), 80–91. https://doi.org/10.1287/mnsc.6.1.80.

De Backer, B., Furnon, V., Kilby, P., Prosser, P., & Shaw, P. (1997). Local search in constraint programming: Application to the vehicle routing problem. Retrieved from https://pdfs.semanticscholar.org/2666/03a503d1dbb84bb1f570c133745e8b9dc8f0.pdf.

Eksioglu, B., Vural, A. V., & Reisman, A. (2009). The vehicle routing problem: A taxonomic review. *Computers & Industrial Engineering, 57*(4), 1472–1483. https://doi.org/10.1016/J.CIE.2009.05.009.

François, V., Arda, Y., Crama, Y., & Laporte, G. (2016). Large neighborhood search for multi-trip vehicle routing. *European Journal of Operational Research, 255*(2), 422–441. https://doi.org/10.1016/J.EJOR.2016.04.065.

García-Nájera, A., Bullinaria, J. A., & Gutiérrez-Andrade, M. A. (2015). An evolutionary approach for multi-objective vehicle routing problems with backhauls q. Doi: 10.1016/j.cie.2014.12.029.

Jamrus, T., Chien, C.-F., Gen, M., & Sethanan, K. (2015). Multistage production distribution under uncertain demands with integrated discrete particle swarm optimization and extended priority-based hybrid genetic algorithm. *Fuzzy Optimization and Decision Making, 14*(3), 265–287. https://doi.org/10.1007/s10700-014-9200-6.

Jamrus, T., & Chien, C. F. (2016). Extended priority-based hybrid genetic algorithm for the less-than-container loading problem. *Computers & Industrial Engineering, 96*, 227–236.

Jamrus, T., Wang, H. K., & Chien, C. F. (2020). Dynamic coordinated scheduling for supply chain under uncertain production time to empower smart production for Industry 3.5. *Computers & Industrial Engineering, 142*, 106375.

Kachitvichyanukul, V. (2012). Comparison of three evolutionary algorithms: GA, PSO, and DE. Doi: 10.7232/iems.2012.11.3.215.

Koç, Ç., & Laporte, G. (2018). Vehicle routing with backhauls: Review and research perspectives. *Computers & Operations Research, 91*, 79–91. https://doi.org/10.1016/J.COR.2017.11.003.

Küçükoğlu, İ., & Öztürk, N. (2014). A differential evolution approach for the vehicle routing problem with backhauls and time windows. *Journal of Advanced Transportation, 48*(8), 942–956. https://doi.org/10.1002/atr.1237.

Liu, L., Li, K., & Liu, Z. (2017). A capacitated vehicle routing problem with order available time in e-commerce industry. *Engineering Optimization, 49*(3), 449–465. https://doi.org/10.1080/0305215X.2016.1188092.

Ong, J. O., & . S. (2011). Vehicle routing problem with backhaul, multiple trips and time window. Jurnal Teknik Industri, 13(1), 1–10. Doi: 10.9744/jti.13.1.1-10.

Panchal, M., Chudasama, C., Shah, S. M., & Processor, A. (2011). Comparison of parents selection methods of genetic algorithm for TSP. In International Journal of Computer Applications®. Retrieved from https://www.researchgate.net/publication/285690217.

Potvin, J.-Y., Duhamel, C., & Guertin, F. (1996). A genetic algorithm for vehicle routing with backhauling. *Applied Intelligence, 6*(4), 345–355. https://doi.org/10.1007/BF00132738.

Reportlinker. (2019). Growth opportunities in the global beverage market. Retrieved July 7, 2019, from https://www.researchandmarkets.com/reports/4171584/growth-opportunities-in-the-global-beverage-market.

Salhi, S., Wassan, N., & Hajarat, M. (2013). The fleet size and mixed vehicle routing problem with backhauls: Formulation and set partitioning-based heuristics. *Transportation Research Part E: Logistics and Transportation Review, 56*, 22–35. https://doi.org/10.1016/J.TRE.2013.05.005.

Shimizu, Y., & Sakaguchi, T. (2013). Generalized vehicle routing problem for reverse logistics aiming at low carbon transportation. Doi: 10.7232/iems.2013.12.2.161.

Shimizu, Y., & Sakaguchi, T. (2014). A hierarchical hybrid meta-heuristic approach to coping with large practical multi-depot VRP. Doi: 10.7232/iems.2014.13.2.163.

Storn, R., & Price, K. (1997). Differential evolution – A simple and efficient heuristic for global optimization over continuous spaces. *Journal of Global Optimization, 11*(4), 341–359. https://doi.org/10.1023/A:1008202821328.

Wassan, N., Wassan, N., Nagy, G., & Salhi, S. (2017). The multiple trip vehicle routing problem with backhauls: formulation and a two-level variable neighbourhood search. *Computers & Operations Research, 78*, 454–467. https://doi.org/10.1016/J.COR.2015.12.017.

Wisittipanich, W., & Hengmeechai, P. (2015). A multi-objective differential evolution for just-in-time door assignment and truck scheduling in multi-door cross docking problems. *Industrial Engineering and Management Systems, 14*(3), 299–311. https://doi.org/10.7232/iems.2015.14.3.299.

Xiao, Y., Zhao, Q., Kaku, I., & Mladenovic, N. (2014). Variable neighbourhood simulated annealing algorithm for capacitated vehicle routing problems. *Engineering Optimization, 46*(4), 562–579. https://doi.org/10.1080/0305215X.2013.791813.

Yu, M., & Qi, X. (2014). A vehicle routing problem with multiple overlapped batches. *Transportation Research Part E: Logistics and Transportation Review, 61*, 40–55. https://doi.org/10.1016/J.TRE.2013.10.004.

Yun, Y., & Gen, M. (2003). Performance analysis of adaptive genetic algorithms with fuzzy logic and heuristics. *Fuzzy Optimization and Decision Making, 2*(2), 161–175. https://doi.org/10.1023/A:1023499201829.

Zhong, Y., & Cole, M. H. (2005). A vehicle routing problem with backhauls and time windows: A guided local search solution. *Transportation Research Part E: Logistics and Transportation Review, 41*(2), 131–144. https://doi.org/10.1016/J.TRE.2003.12.003.