Applied Soft Computing 53 (2017) 119–132

Contents lists available at ScienceDirect

Applied Soft Computing

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

A simulated annealing heuristic for the hybrid vehicle routing problem

Vincent F. Yu [a], A.A.N. Perwira Redi [a,*], Yosi Agustina Hidayat [b], Oktaviyanto Jimat Wibowo [a,b]

[a] Department of Industrial Management, National Taiwan University of Science and Technology, Taipei 106, Taiwan
[b] Department of Industrial Engineering, Institut Teknologi Bandung, Bandung, Indonesia

ARTICLE INFO

Article history:
Received 3 August 2015
Received in revised form 29 April 2016
Accepted 17 December 2016
Available online 23 December 2016

Keywords:
Hybrid vehicle routing problem
Hybrid electric vehicle
Simulated annealing
Cauchy function
Restart strategy

ABSTRACT

This study proposes the Hybrid Vehicle Routing Problem (HVRP), which is an extension of the Green Vehicle Routing Problem (G-VRP). We focus on vehicles that use a hybrid power source, known as the Plug-in Hybrid Electric Vehicle (PHEV) and generate a mathematical model to minimize the total cost of travel by driving PHEV. Moreover, the model considers the utilization of electric and fuel power depending on the availability of either electric charging or fuel stations.

We develop simulated annealing with a restart strategy (SA_RS) to solve this problem, and it consists of two versions. The first version determines the acceptance probability of a worse solution using the Boltzmann function, denoted as SA_RS_{BF}. The second version employs the Cauchy function to determine the acceptance probability of a worse solution, denoted as SA_RS_{CF}. The proposed SA algorithm is first verified with benchmark data of the capacitated vehicle routing problem (CVRP), with the result showing that it performs well and confirms its efficiency in solving CVRP. Further analysis show that SA_RS_{CF} is preferable compared to SA_RS_{BF} and that SA with a restart strategy performs better than without a restart strategy. We next utilize the SA_RS_{CF} method to solve HVRP. The numerical experiment presents that vehicle type and the number of electric charging stations have an impact on the total travel cost.

1. Introduction

Supply chain is an integrated process of various businesses that convert raw materials into finished products and deliver them to retailers [1–3]. Logistics is one part of a supply chain that is separated into physical activities (e.g. transport, storage) and non-physical activities (e.g. supply chain design, freightage negotiations) [4]. In a logistic system, transportation plays an important role for carrying raw materials from suppliers to manufacturers and delivering finished goods from manufacturers to customers. In the U.S., transportation itself account for 28% of total energy consumption, and thus transportation cost significantly influences the final price of the goods [1,5,6]. On the other hand, logistic activities also have a substantial impact on the environment. Green logistics is a recently introduced term due to the importance of restraining the environmental damage caused by a rise in logistical activity. The purpose of green logistics is to provide economic benefits to a specific subject while at the same time maintaining environmental protection [7].

One technique within green logistics is the use of a green vehicle or an environmental friendly vehicle. For example, Electric Vehicles (EV) are known as one of the most promising solutions to the problem of greenhouse gasses. However, there are still constraints on this proposed solution, including limited recharging stations and lesser vehicle mileage per recharging. Therefore, the current transportation choice is towards utilizing a vehicle that combines both fuel and electricity, called the Hybrid Electric Vehicle (HEV). HEV can reduce transportation costs significantly as well as cut emissions from transportation activities [8]. A report analyzing 2013 data shows that almost 500,000 hybrid vehicles were sold in 2013, among which plug-in vehicles made up nearly 100,000 units. Moreover, at least 24 different models of plug-in vehicles are available or coming soon to the market. In term of charging facility, there are more than 20,835 electric vehicle charging stations throughout the U.S. This trend of a rise in the concern for the environment is the same all around the world.

Green logistics has recently been considered a part of the Vehicle Routing Problem (VRP), which is a well-known optimization problem that minimizes transportation cost by determining optimal vehicle routes. Dantzig and Ramser [9] first proposed the combinatorial concept of VRP. Lin, et al. [10] then provided a review of 284 research studies about VRP and classified them into two big groups:

* Corresponding author.
E-mail address: wira.redi@gmail.com (A.A.N.P. Redi).

http://dx.doi.org/10.1016/j.asoc.2016.12.027
1568-4946/

traditional VRP and Green Vehicle Routing Problem (G-VRP). G-VRP is a variant of VRP that harmonizes the environmental and economic impacts and concerns itself with energy consumption, including the use of a vehicle with alternative fuel and alternative fuel stations [8,11]. With the increasing worldwide concern for the environment, Demir, et al. [12] reviewed many research studies on green road freight transportation in order to analyze the factors affecting fuel consumption on green road freight transportation, as well as the available vehicle emission models. The study on G-VRP could be extended to consider the different sources of fuel for the vehicles.

The technology of electricity as a new source of fuel for vehicles can be classified under two main categories: fuel cells electric vehicle types and battery electric vehicle. In order to improve the performance of logistics using an electric vehicle, a consideration about visits to recharging stations along the routes is required. To the best of our knowledge, Erdoğan and Miller-Hooks [8] were the first to set up a routing model that considers recharge stations. Pelletier, et al. [13] reported that in the last 15 years there is increasing interest and a growing number of publications on green transportation and city logistics, but the available literature about green logistics, especially the routing problem, mostly addresses battery electric vehicles (BEVs) and fuel cell electric vehicles (FCEVs). The availability of a routing model for Hybrid Electric Vehicles (HEVs) is even less or quite absent. Therefore, this paper is motivated to introduce the Hybrid Vehicle Routing Problem (HVRP) as an extension of G-VRP. HVRP aims to minimize the total travel cost constrained by the usage of a hybrid vehicle. This study considers a time constraint and two types of energy resources: electricity and fuel. The practicality of HVRP is supported by the growing number of HEVs in the automobile market.

We note that HVRP is an NP-hard problem, because it is a special case of G-VRP, which is NP hard. Therefore, the most viable option to obtain a good quality solution for a large-scale HVRP problem in a reasonable amount of time seems to be through the use of a soft computing technique such as heuristics and meta-heuristics. We propose a simulated annealing with a restart strategy (SA_RS) to solve HVRP.

The contributions of this paper are summarized as follows. First, we present a new variant of VRP, called Hybrid Vehicle Routing Problem (HVRP). It extends GVRP, introduced by Erdoğan and Miller-Hooks [8], by considering an organization of routes for HEVs that not only take advantage of the availability of conventional refueling infrastructure, but also overcome difficulties that exist as a result of a limited vehicle driving range in conjunction with limited electric refueling infrastructure. Second, we develop an efficient simulated annealing heuristic with a restart strategy (SA_RS) to solve HVRP, and it consists of two versions: SA_RS_{BF} and SA_RS_{CF}. The difference between SA_RS_{BF} and SA_RS_{CF} is that the acceptance probability of a worse solution for SA_RS_{BF} uses the Boltzmann function, while SA_RS_{CF} uses the Cauchy function. Both proposed methods are tested in the computational experiment.

The rest of the paper is organized as follows. Section 2 reviews the relevant literature. Section 3 defines the problem, and Section 4 introduces a mathematical model for HVRP. Section 5 then describes the proposed simulated annealing heuristic with a restart strategy for HVRP. Section 6 reports the computational results, followed by conclusions in Section 7.

2. Literature review

In green logistics an environmental friendly vehicle is highly recommended as part of the transportation choice. An Electric Vehicle (EV) is one type of green vehicle that promises a viable solution for reducing the effect of greenhouse gasses, although there are still limitations in driving range, long recharging time, and high initial cost. Moreover, a charging infrastructure is needed to be built up along with stronger reliability and durability of the batteries [14]. Yang and Sun [15] saw the importance of using EVs in many logistics companies that are seeking ways to effectively manage carbon emissions from their logistics activities and thus proposed the electric vehicles battery swap stations location routing problem (BSS-EV-LRP), which aims to determine the location strategy of battery swap station (BSS) and the routing plan of a fleet of electric vehicles. A Hybrid Electric Vehicle (HEV) is preferable, because a pure electric vehicle is limited to the availability of recharging stations and distance coverage.

There are two basic configurations of HEV: a series hybrid and a parallel hybrid. In a series hybrid, the power transmission comes from both combustion engines, and the drive system is from electricity. In a parallel hybrid, the power transmission comes from both combustion engines, and the electrical drive system is mechanical. These systems can be applied in combination or independently. Another hybrid configuration is a complex configuration, which involves a setting that cannot be classified into those two types mentioned. The complex hybrid seems to be similar to the series-parallel hybrid, because the generator and electric motor are both electrical and mechanical. However, the key differences are the bidirectional power flow of the electric motor in the complex hybrid and the unidirectional power flow of the generator in the series-parallel hybrid [16]. Hannan, et al. [17] classified HEV, based on the hybridization factor, into Micro-HEVs, Mild-HEVs, Power-assisted HEVs, and Plug-In HEVs (PHEVs).

A plug-in hybrid electric vehicle is the next generation of hybrid electric vehicles, offering significant advantages over the cleanest and most efficient of today's hybrid vehicles [18]. PHEV's sources of energy for propulsion consist of electricity from the grid, energy recovery while braking (which is stored in the battery), and chemical energy from a hydrocarbon-based fuel stored in the fuel tank [19]. Bradley and Frank [20] explained that a PHEV can work in several modes, such as Charge-depleting mode (CD mode), Charge-sustaining mode (CS mode), Electric Vehicle (EV) mode, and Engine-only mode. Silva, et al. [19] classified PHEV into a range extender PHEV and a blended PHEV. A range extender PHEV runs primarily on electricity discharged from the battery, and after the State of Charge (SOC) reaches a minimum, it operates in a charge-sustaining mode. A blended PHEV is similar to the range extender PHEV, but in the charge-depleting mode it may turn on the internal combustion engine (ICE) whenever needed.

The hybrid vehicle routing problem focuses on minimizing the total cost incurred by utilizing hybrid vehicles. Therefore, HVRP is classified into the G-VRP group by Lin, et al. [10] as it also takes into account the environmental impact. Research on G-VRP is relatively infrequent compared to traditional VRP. Kara, et al. [21] introduced a new cost function based on the distance and load of a vehicle for the Capacitated Vehicle Routing Problem, called the Energy Minimizing Vehicle Routing Problem (EMVRP). Kuo [22] proposed a model for calculating total fuel consumption for the time-dependent vehicle routing problem (TDVRP), where speed and travel times are assumed to depend on the time of travel when planning vehicle routing. Xiao, et al. [23] developed a mathematical optimization model to formally characterize the Fuel Consumption Rate (FCR) that considers Capacitated VRP (CVRP), called Fuel Consumption VRP (FCVRP). Erdoğan and Miller-Hooks [8] proposed G-VRP by looking at alternative fuel vehicles (AFVs) and alternative fuel stations (AFSs).

This study thus proposes the Hybrid Vehicle Routing Problem, an extension of research on G-VRP that focuses on vehicles using electricity and fuel as energy sources (hybrid vehicle). Current HVRP research development only has focused on one kind of power source. Hence, herein we present a new variant of G-VRP that handles more than one power source, which in this case encompasses two types of power sources. The type of hybrid vehicle examined in this study is the Plug-in Hybrid Electric Vehicle. PHEV is a hybrid-electric vehicle (HEV) with the capability to recharge itself from external sources (such as an electric utility grid). HVRP targets the minimization of total travel cost. This study considers a time constraint, electric capacity, and fuel capacity. HVRP is defined by an undirected, complete graph of a vertex set V that consists of the depot, customer set, electric station, fuel station, and the set of edges connecting the vertices.

The usage of metaheuristics as a general-purpose solver for the optimization problem requires extensive adaptations and user involvement or expertise to match with the characteristics of each problem. The vehicle routing problem (VRP), one of the major classes of combinatorial optimization problems with a broad range of applications, is a good example to suitably implement a metaheuristic approach as illustrated by the reviews of Gendreau, et al. [24] and Laporte [25]. The approaches are distinguished into the so-called neighborhood-centered, population-based, and hybrid methods [26]. The neighborhood-centered methods generally proceed by iteratively exploring neighborhoods of a single incumbent solution such as Simulated Annealing (SA) [27,28], Tabu search [29–31], Variable Neighborhood Search (VNS) [32–34], Adaptive Large Neighborhood Search (ALNS) [35], and Iterated Local Search (ILS) [36]. Population-based methods are often inspired from a natural mechanism that generates one or several "new" solutions out of combinations of existing ones. Some of the well-known population-based metaheuristics are Genetic Algorithms (GA) and Evolutionary Algorithms (EA) [37], Memetic Algorithm [38,39], Path Relinking (PR) and Scatter Search (SS) [40,41], Particle Swarm Optimization (PSO) [42,43], and Ant Colony Optimization (ACO) [44]. The concept of combining various solution approaches to take advantage of their strength is hybrid metaheuristics. Numerous types of hybrid metaheuristics have been proposed, and for CVRP itself they include SA + Tabu [28], GRASP + ILS [45], ILS + VND [34], and TABU + ILS [46], among others.

Simulated annealing as an extension of the Markov Chain Monte Carlo algorithm was first presented in 1953 by Metropolis, et al. [47]. The SA algorithm is inspired by the process of annealing solid metals. Thereafter, SA has become a popular metaheuristic for solving optimization problems. Recent developments of metaheuristics have shown that several modifications can be considered to improve the performance, such as utilizing the fuzzy system to determine the probability of accepting the worse solution or several other parameters turning in the metaheuristic algorithm [48,49]. A self-organizing neural network approach also can be used to set the parameter in the SA algorithm [51]. Finally, a more detailed observation on the physical phenomena and chemical reaction – for example, when applied in chemical reaction optimization [50] – can lead to a new metaheuristic with a promising performance.

In this study we focus on developing an SA version that has been successfully implemented in a combinatorial optimization problem such as VRP and its variants. Osman [28] and Lin, et al. [52] proposed the hybrid approach of simulated annealing that results in high-quality solutions. Yu, et al. [53] developed a simple yet powerful version of SA that incorporates a diversification in neighborhoods of the algorithm. This version has been successfully applied in several variants of VRP, such as Truck and Trailer Routing Problem (TTRP) [54], Team Orienteering Problem with Time Windows (TOPTW) [55], and Open Vehicle Routing Problem with Cross-docking (OVRPCD) [56]. The recent development of these SA variants shows that the implementations of a multi-start strategy [57] and a restart strategy with different acceptance probability functions [58] give a remarkable performance in the order term of finding the new best solutions to benchmark instances. Following such successes, this study proposes simulated annealing with a restart strategy for HVRP.

3. Problem definition

The Hybrid Vehicle Routing Problem is illustrated by companies that employ a fleet of hybrid vehicles serving customers or other entities located in a broad geographical region. The goal of HVRP is finding at most m tours that start and end at the depot. Each tour is served by a hybrid vehicle and visits a subset of vertices, consisting of customers and stations for power source type $z(S_z)$, when needed. Each customer must be visited exactly once. The station type $z(S_z)$ can be visited more than once or not visited at all. There are two types of S_z in this case: electric station and fuel station. Each station can fill power to the vehicle until full capacity. Each vehicle that is used in the tour cannot exceed the predetermined maximum time $T_{\max}$. In addition, the vehicle is not allowed to run out of power.

This study employs parameters based on previous research [8,59,60]. The model considers the following assumptions.

1 This particular problem is deterministic and static.
2 The HVRP problem consists of customers, a depot, and sets of station type z (electric station and fuel station).
3 Once the vehicle returns to a depot, the fuel capacity and electric capacity will be filled to their maximum capacity.
4 All stations $S_z = \{S_e \cup S_f\}$ have unlimited capacity and will always be able to serve the vehicle, and all vehicles can stop at any refueling stations without any visitation count restriction.
5 Every time a vehicle visits station S_z, the vehicle capacity of type z is refilled until it reaches maximum capacity.
6 Each S_z is allowed to be visited more than once or not at all.
7 The vehicles are required to visit all customers. A vehicle can visit S_z when it is necessary to find the minimal total tour cost.

The purpose of HVRP is to minimize total travel cost. The control variables consist of the arrival time of each vertex, distance covered by power type z, the decision for whether a vehicle should traverse a vertex, and the decision of choosing which power. The parameters consist of travel time limit, service time on each vertex, travel time of each arc, the distance of each vertex, the power consumption rate, and vehicle power capacity.

4. Mathematical formulation

A complete graph consists of vertices representing customer locations, two types of recharging power stations, and a depot. HVRP seeks a set of hybrid vehicle tours with the objective of minimizing total travel cost, in which the vehicle starts at the depot, visits the set of customers within a pre-specified time limit, and returns to the depot. In each tour, the vehicle is allowed to stop at a station once or more to recharge power.

Inspired by Erdoğan and Miller-Hooks [8], we formalize the problem as follows. HVRP is defined on an undirected complete graph $G = (V, E)$, where vertex set V is a combination of the customer set $I = \{v_1, v_2, \ldots, v_n\}$, the depot v_0, a set of fuel stations $\left(S_f\right)$, and a set of electric stations (S_e). The set of fuel stations with $f \geq 0$ is$S_f = \left\{v_{n+1}, v_{n+2}, \ldots, v_{n+f}\right\}$, whereby each station may be visited more than once or not at all, while the customers' vertices require exactly once visit. Graph G is augmented with a set of f' dummy vertices $\phi_f = \left\{v_{n+f+1}, v_{n+f+2}, \ldots, v_{n+f+f'}\right\}$. The set of electric charging stations with $e \geq 0$ is $S_e = \left\{v_{n+f+f'+1}, v_{n+f+f'+2}, \ldots, v_{n+f+f'+e}\right\}$. Each electric charging station may be visited more than once or not at all, while there is exactly one visit for each customer vertex. Graph G is augmented with a set of e' dummy vertices, $\phi_e =$

$\left\{v_{n+f+f'+e+1}, v_{n+f+f'+e+2}, \ldots, v_{n+f+f'+e+e'}\right\}$. The number of dummy vertices associated with each fuel and electric station should be set as small as possible so as to reduce the network size, but large enough to allow multiple beneficial visits.

The vertex set is $V = \{v_0\} \cup I \cup S_f \cup S_e$. The set $E = \left\{(v_i, v_j) : v_i, v_j \in V, i < j\right\}$ corresponds to the edges connecting the vertices of V. The dummy vertex set is $V' = V \cup \phi_f \cup \phi_e$. The dummy graph is $G' = (V', E')$. Each edge (v_i, v_j) is associated with a non-negative travel time t_{ij}, cost c_{ij}, and distance d_{ij}.

We define notations used in formulating HVRP as follows.

Z	Set of power sources, $Z = \left\{e, f\right\}$
c_z	Cost rate for power source type z
I	Set of customers
I_0	Set of customer vertices and depot, $I_0 = \{v_0\} \cup I$
ϕ_z	Set of dummy station vertices, $\phi_z = \phi_f \cup \phi_e$
S_0	Set of station vertices including depot, fuel stations, electric stations, and dummy stations, $S_0 = \{v_0\} \cup S_f \cup S_e \cup \phi_z$
V	Set of vertices including depot, customers, and stations, $V = \{v_0\} \cup I \cup S_f \cup S_e$
V'	Set of vertices and dummy vertices, $V' = V \cup \phi_z$
m	Number of hybrid vehicles
p_i	Service time at vertex i; if $i \in I$, then p_i is the service time at the customer vertex; if $i \in F$, then p_i is the refueling time at the station vertex, which is assumed to be constant
τ_0	Departure time from the depot and an upper bound on arrival times upon return to the depot
t_{ij}	Time of arrival at vertex j from vertex i
$T_{\max}$	Pre-determined maximum time for travel from depot until return to depot
p_j	Service time at vertex j
r_z	Vehicle consumption rate (gallons per mile) on power type z
Q_z	Vehicle tank capacity of power type z
d_{ij}	Distance from i to j

Decision variables

x_{ij}	Binary variable equal to 1 if a vehicle travels from vertex i to j; otherwise, 0
y_{jz}	Level of power type z variable specifying the remaining tank power level of type z upon arrival to vertex j
τ_j	Time variable specifying the time of arrival of a vehicle at vertex j, initialized to zero upon departure from the depot
d'_{ijz}	Distance from vertex i to vertex j using power type z
f_{ijz}	Binary variable, equal to 1 if vehicle travels from vertex i to vertex j using power type z; otherwise, 0

The mathematical formulation of HVRP is as follows.

Objective function:

$$\min \sum_{z \in Z} \sum_{\substack{i, j \in V' \\ i \neq j}} d'_{ijz} f_{ijz} c_z \tag{1}$$

Constraints:

$$\sum_{\substack{j \in V' \\ j \neq i}} x_{ij} = 1 \quad \forall \mathrm{i} \in \mathrm{I} \tag{2}$$

$$\sum_{\substack{j \in V' \\ j \neq i}} x_{ij} \leq 1 \quad \forall \mathrm{i} \in S_0 \tag{3}$$

$$\sum_{\substack{j \in V' \\ j \neq i}} x_{ji} - \sum_{\substack{j \in V' \\ j \neq i}} x_{ij} = 0 \quad \forall \mathrm{i} \in V' \tag{4}$$

$$\sum_{j \in V' \setminus \{0\}} x_{0j} \leq m \tag{5}$$

$$\sum_{j \in V' \setminus \{0\}} x_{j0} \leq m \tag{6}$$

$$\tau_j \geq \tau_i + \left(t_{ij} + p_i\right) x_{ij} - T_{\max}\left(1 - x_{ij}\right) \quad \mathrm{i} \in V', \forall_j \in V' \setminus \left\{0\right\}, \text{ and } \mathrm{i} \neq \mathrm{j} \tag{7}$$

$$0 \leq \tau_0 \leq T_{\max} \tag{8}$$

$$\tau_j \leq T_{\max} - \left(t_{j0} + p_j\right) \quad \forall \mathrm{j} \in V' \setminus \{0\} \tag{9}$$

$$y_{jz} = y_{iz} - r_z . d'_{ijz} . f_{ijz} \quad \forall \mathrm{z} \in \mathrm{Z} \tag{10}$$

$$\sum_{z \in Z} d'_{ijz} \cdot f_{ijz} = d_{ij} \quad \forall \mathrm{i}, \mathrm{j} \in V' \tag{11}$$

$$y_{iz} = Q_z \quad \forall \mathrm{i} \in S_f \cup S_e \tag{12}$$

$$\sum_{z \in Z} f_{ijz} \geq x_{ij} \quad \forall \mathrm{i}, \mathrm{j} \in V' \tag{13}$$

$$\sum_{z \in Z} \frac{y_{jz}}{r_z} \geq \sum_{z \in Z} d_{j0} \cdot x_{j0} \quad \forall \mathrm{j} \in \mathrm{I} \tag{14}$$

$$x_{i,j} \in \left\{0, 1\right\} \quad \forall \mathrm{i}, \mathrm{j} \in V' \tag{15}$$

$$f_{i,j,z} \in \left\{0, 1\right\} \quad \forall \mathrm{i}, \mathrm{j}, \mathrm{z} \in V' \tag{16}$$

The objective (1) is for finding the minimum total cost to travel by using a hybrid vehicle in a given day. Constraint (2) ensures that each customer vertex has exactly one successor: a customer, station, or depot vertex. Constraint (3) ensures that each station (and associated dummy vertices) has at most one successor vertex: a customer, station, or depot vertex. Constraint (4) notes that the number of arrivals at a vertex must equal the number of departures for all vertices. Constraint (5) ensures that at most m vehicles are routed out of the depot, and constraint (6) ensures that at most m vehicles return to the depot in a given day. A copy of the depot is made to distinguish between departure and arrival times at the depot, which is necessary for tracking the time when each vertex is visited and prevents the formation of sub-tours.

The arrival time at each vertex by each vehicle is tracked through constraint (7). Constraint (7), along with constraints (8) and (9), makes certain that each vehicle returns to the depot no later than $T_{\max}$. Constraint (8) specifies a departure time from the depot of zero ($\tau_0 = 0$) and an upper bound on arrival times upon return to the depot. The lower and upper bounds on arrival times at customer and station vertices given in constraint (9) ensure that each route is completed by $T_{\max}$.

Constraint (10) tracks a vehicle's fuel level based on vertex sequence and type. If vertex j is visited right after vertex i $\left(x_{ij} = 1\right)$ and vertex i is a customer vertex, then the first term in constraint (10) reduces the fuel level upon arrival at vertex j based on the distance traveled from vertex i and the vehicle's fuel consumption rate. Constraint (11) states that the distance from vertex i to vertex j can be served by power type z. Constraint (12) resets the power level of type z to Q_z after visiting the depot or the station type $z$vertex.

Constraint (13) indicates that at least one type of power is used to move from vertex i to vertex j. There will be enough remaining power to return to the depot directly or to visit a station from any customer location, as shown in constraint (14). These constraints ensure that the vehicles will not be stranded. Binary integrality is guaranteed through constraints (15) and (16).

5. Simulated annealing for HVRP

HVRP is an NP-hard combinatorial optimization problem. In the early years, many specialized heuristics were developed to deal with this type of problem [61]. Simulated annealing (SA) is one of the successful methods for implementing. SA is a local search-based algorithm that has a mechanism to escape from the local optima with the purpose of finding a global optimum [62]. VRP and related problems have been solved effectively by SA [63]. SA belongs to a single solution-based algorithm that iteratively explores for a solution with a better objective value. The essential feature of the SA algorithm is allowing a hill-climbing movement by introducing an acceptance criterion or the so-called Boltzmann function. In this proposed SA, an experiment using the Cauchy function is considered. As the temperature parameter decreases to zero, hill-climbing moves occur less frequently [62]. In order to avoid being trapped at local optima, a diversification mechanism is used, the so-called restart strategy. The following subsections discuss the solution representation, initial solution method, and neighborhood structure used in the proposed SA.

5.1. Solution representation

The proposed SA algorithm features a specially designed solution representation scheme for HVRP. The solution consists of n customers [15] and m selected fuel or electric stations from sets S_e and S_f. This solution representation enables one solution to have a different length from the other ones. The i^{th} number in $\{1, 2, \ldots, n\}$ denotes the i^{th} customer to be serviced. The following number of $\{n+1, n+2, \ldots, n+m\}$ then denotes the selected charging station to be visited.

The solution representation is further explained as follows. The route starts from the depot and then takes the first number in the solution as the node to be served. We continue adding a node to the route until the route is terminated by violating vehicle capacity or the total time travel constraint. If the next node visited is a charging station, then the vehicle capacity of the related power source (e.g., fuel or electric) is set to be full, in which case the vehicle can have a greater distance coverage. The power consumption mechanism follows the Plug-in Hybrid consumption model, which first prioritizes the electric source and then dynamically changes to a fuel source if the electric source reaches the SOC condition while at the same time synchronously charging the electric battery.

Since the utilization of m recharging stations is optional, the search procedure needs to deal with some customers that are unreachable due to the travel distance. For this situation, a penalty cost P is added to the objective function so that such a solution becomes unattractive to be explored further.

5.2. Illustration of solution representation

Fig. 1 illustrates the route possibilities for 20 customers without going to any of the charging stations. This solution is then separated into routes based on time constraint and vehicle capacity. Therefore, there are five routes used for serving all customers.

Whenever a route is determined, there is a possibility for a number of customers to not have been served due to the vehicle capacity constraint. However, the utilization of a charging station, either electric or fuel, can help to make the solution feasible. At the same time, employing electric energy later on can reduce the transportation cost, and therefore the optimal utilization of electric energy to travel the route is prioritized.

Fig. 2 illustrates the route for 20 customers after the insertion of electric stations S_e or fuel stations S_f. The solution consists of one depot, two S_e (21, 22), two S_f (23, 24), and twenty customer nodes (1...20). The depot is symbolized by a square, customer nodes are symbolized by a circle, S_e nodes are symbolized by a triangle, and S_f is symbolized by a diamond. The solution representation determines which customer is served in one route and when to utilize a recharging station. In the case whereby an infeasible solution still occurs, a penalty value P is associated with the solution regarding the number of unserved customers.

18	9	8	11	4	2	6	13	1	17	12	16	5	20	15	14	10	3	7	19
I				**II**				**III**				**IV**				**V**			

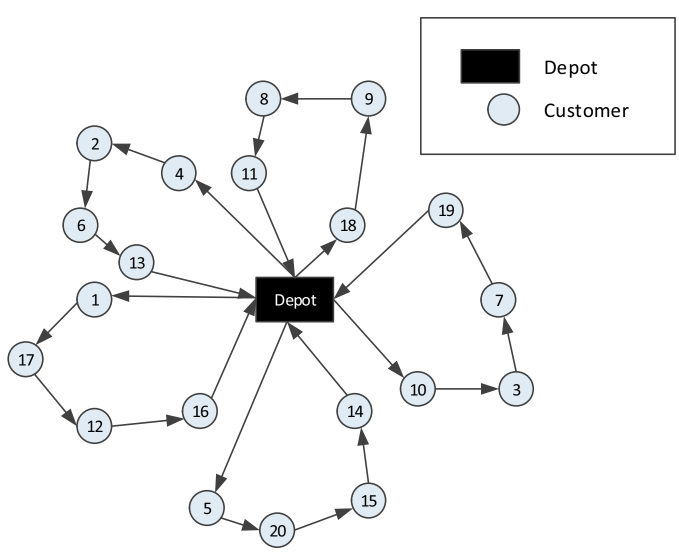

Fig. 1. An HVRP solution without visiting any charging station.

5.3. Initial solution

A carefully developed initial solution often helps to have a better result in terms of algorithm performance. The proposed SA_RS algorithm utilizes the well-known nearest neighbor (NN) heuristic to provide an initial solution. The nearest neighbor heuristic has been modified to consider HVRP characteristics.

Step 1. For each route, an unserved customer is selected based on the nearest distance to the last inserted node starting from the depot. A route is terminated if no unserved node can be inserted without violating the maximum travel time constraint. After a route is terminated, the route generation process starts again from the depot and continues until all customers are served.

Step 2. Let R be the route that violates the vehicle capacity constraint. For each R, we randomly assign a charging station from both S_e and S_f to R while maintaining the route time constraint feasibility.

Step 3. For each R that still violates vehicle capacity constraints, we introduce a penalty value P to the objective function based on the number of nodes violating the constraint. We then terminate the procedure after completion of Step 3.

5.4. Neighborhood

The algorithm explores one neighborhood at each iteration, which is selected among five previously defined neighborhoods. The neighborhood structure consists of various types of moves, including insertion, swap, and reverse, which are well-known in the VRP literature. We include a specially designed neighborhood for HVRP to explore the utilization of a charging station, which is named the insert station and delete station.

The neighborhood moves are denoted by $N(X)$. The insertion is implemented by randomly selecting a node from index i in solution X and then inserting it immediately following a randomly selected j node of X. The swap move is carried out by exchanging the positions of randomly selected i and j nodes of solution X. A reverse is performed by randomly selecting i and j nodes of solution X and then reversing the substring in-between them. An insert station is executed by randomly choosing a station each from S_e and S_f, and then inserting them into the preceding random i index on solution X. A delete station means randomly taking out a charging station from solution X.

The generation of the probability function is taken from a generic choice of uniform distributions with a probability proportional to the size of the neighborhood, which in this study is set to 1/5 proportion for each neighborhood. Note that the neighborhood movement enables the solution to move toward an infeasible solution. A violation of the vehicle capacity constraint means adding an objective function with a penalty value, as explained in Section 5.1.

5.5. Parameter setting

SA_RS begins with six parameters: I_{iter}, T_0, T_F, K, $N_{non\text{-}improving}$, and α. I_{iter} denotes the number of iterations that the search proceeds at for a particular temperature, while T_0 represents the initial temperature, and T_F represents the final temperature below which the SA_RS procedure is stopped. K is the Boltzmann constant used in the probability function to determine whether to accept a worse solution or not. For the version with acceptance criteria using the Cauchy function, a constant K is not needed. $N_{non\text{-}improving}$ is the maximum allowable number of reductions in temperature during which the best objective function value is not improved. Finally, α is the coefficient controlling the cooling schedule. The parameter setting is necessary to obtain optimal computational results. In this research, we perform parameter settings using the two-level (2k) factorial design. The result of this design experiment indicates that the best parameter combination is $N_{non\text{-}improving} = 2N$, $T0 = 1$, $T_f = 0.01$, $I_{iter} = 1000^*N$, and $\alpha = 0.9$. Where N is the total number of the customers. In addition, penalty value P is determined to equal US$25 for each node not served.

5.6. The SA_RS procedure

The proposed SA_RS consists of two phases: initial phase and improvement phase. In the initial phase, the Nearest Neighbor algorithm, as explained in Section 5.3, is employed to construct the initial route. The algorithm continues to the improvement phase after finishing the initial phase. The improvement phase tries to

18	9	8	11	21	4	2	6	13	22	1	17	12	16	5	23	20	15	14	10	3	7	19
I									II								III					

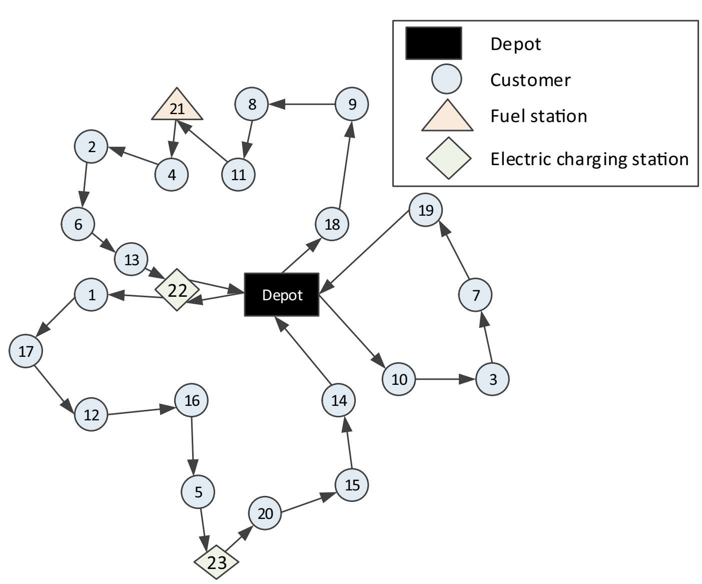

Fig. 2. An HVRP solution that visits charging stations.

improve the result from the initial phase by the neighborhood function $N(X)$ illustrated by Fig. 3.

The algorithm begins by setting the current temperature T to be T_0 and the value of the best solution found F_{gbest} to be a big number M, and generating an initial solution X. The current best solution X_{best} and the current best objective value F_{best} are initially set to be X and the objective value of X, respectively.

In the improvement phase, we utilize the neighborhood function $N(X)$ to generate a new solution Y from the current solution X. Let $\Delta = obj(Y) - obj(X)$. We replace X with Y if $obj(Y)$ is better than X ($\Delta < 0$); otherwise, we replace X with Y with a small probability. The probability of acceptance for the first version of simulated annealing with a restart strategy (SA_RS) using the Boltzman function $\exp(-\Delta/KT)$ is denoted as SA_RS_{BF}. The Boltzman function is commonly used in SA. Lin and Yu [58] followed the approach reported by Tiwari, et al. [64] that successfully applied the Cauchy function into SA. Thus, this study develops a second version of SA_RS that uses the Cauchy function as the acceptance probability function, denoted as SA_RS_{CF}. In the Cauchy function, the probability of replacing current solution X with a worse solution Y is $T/(T^2+\Delta^2)$.

The current temperature decreases after $I_{iteration}$ according to formula $T = \alpha T$. The algorithm restarts if the current best solution X_{best} has not improved for 10 consecutive temperature decreases. Once the algorithm restarts, the current temperature is reset to the initial temperature and a new initial solution is generated. Here, X_{best} and F_{best} respectively record the current best solution and the best objective function value obtained before the restart mechanism is activated. The best solution found by the algorithm denoted as X_{gbest} with the objective value F_{gbest}. The algorithm is terminated after the current temperature T_0 is lower than T_f or the current best solution X_{best} has not improved for $N_{non-improving}$ consecutive temperature reductions. Fig. 3 illustrates the proposed SA heuristic.

6. Computational results

The SA algorithm was coded in Microsoft Visual Studio C++ 2012 and performed on a computer with specifications encompassing Intel (R) Core (TM) i7 at 3.60 GHz, 16 Gb of RAM, and running on a 64-bit platform under Windows 7 Operating System.

6.1. Model verification

We propose SA to solve the model that deals with the hybrid vehicle. In order to explain how the model works, a small instance is generated with 1 depot, 5 customers, 2 electric charging stations, and 2 fuel stations. Table 1 shows the vehicle specification, and Table 2 shows the distance matrix.

We solve this problem instance using SA_RS_{CF}. The results show that to serve all customers, 2 vehicles are needed. Total cost for this example is US$124.767, which consists of US$6.3 for electric power cost (EP Cost) and US$118.467 for fuel power cost (FP Cost). Table 3 presents the result of this problem.

Table 1
Vehicle specification.

Specification	Total
Electric capacity	15 kWh
Max electric power that can be used	70%*electric capacity =0.7*15 kWh =10.5 kWh
Electric consumption rate	0.5 kWh/mile
Maximum electric distance	10.5/0.5 = 21 miles
Electric cost	US$ 0.12/kWh
Fuel capacity	25 gallons
Fuel consumption rate	17.7 mpg
Maximum fuel distance	25*17.7 = 442.5 miles
Fuel cost	US$4.18/gal
Maximum route time	11 h
Velocity	40 mph
T_{max}	11 h

From Table 3, the route of vehicle 1 is “0–7–8–5–2–1–0”, which can be translated as: “depot – customer 3–customer 4–customer 1–electric station 2–electric station 1–depot”. The distance between the depot and customer 3 is 122.369 miles. To go from the depot to customer 3, vehicle 1 uses electric power for the first 21 miles and then continues by fuel power for 101.369 miles. The electric power cost for 21 miles is US$1.26, and the fuel power cost for 101.369 miles is US$28.3832. The vehicle velocity is constant at 40 mph, and so vehicle 1 needs 3.05921 h to move from the depot to customer 3. Vehicle 1 continues to serve as many other customers as possible within the time limit and visits an electric station two times before going back to the depot. For vehicle 2, the route is “0–9–6–1–0” or “depot – customer 5–customer 2–electric station 1–depot”. The details are similar to vehicle 1, as can be seen in (Fig. 4). In this problem, electric station 1 (ES1) is visited two times, but not all fuel stations are visited. It illustrates that stations can be visited more than one time or not visited at all.

6.2. Algorithm verification

In order to check the performance of the proposed SA_RS algorithm, it is necessary to compare it with other metaheuristics for benchmarking purposes. The SA_RS algorithm is utilized to solve the HVRP model that relaxes some constraints such as fuel capacity, electric capacity, fuel stations, and electric charging stations in order to solve the CVRP benchmark instances. Moreover, this comparison converts the time limit in HVRP as vehicle capacity in CVRP, and node demand in CVRP is associated with the travel time for each arc in HVRP. Thus, CVRP can be used as a benchmark for HVRP.

The first set of benchmark instances consists of datasets for CVRP proposed by Augerat et al. (datasets A, B, and P), by Christofides and Eilon (for dataset E), by Fisher (for dataset F), and by Christofides, Mingozzi, and Toth (for dataset M). The optimal solution is taken

Table 2
Distance matrix (in miles).

	Node	D 0	ES1 1	ES2 2	FS1 3	FS2 4	C1 5	C2 6	C3 7	C4 8	C5 9
D	0	0	26.71	141.34	95.78	196.87	149.23	38.40	122.37	166.17	66.90
ES1	1	26.71	0	123.46	76.15	184.99	130.96	13.83	108.13	150.88	71.15
ES2	2	141.34	123.46	0	47.65	72.77	8.19	110.02	27.15	31.70	104.31
FS1	3	95.78	76.15	47.65	0	115.06	54.90	62.55	39.28	77.03	74.74
FS2	4	196.87	184.99	72.77	115.06	0	69.69	172.85	77.38	41.46	141.28
C1	5	149.23	130.96	8.19	54.90	69.69	0	117.43	34.72	28.25	112.37
C2	6	38.40	13.83	110.02	62.55	172.85	117.43	0	95.66	137.92	67.79
C3	7	122.37	108.13	27.15	39.28	77.38	34.72	95.66	0	43.88	78.41
C4	8	166.17	150.88	31.70	77.03	41.46	28.25	137.92	43.88	0	119.48
C5	9	66.90	71.15	104.31	74.74	141.28	112.37	67.79	78.41	119.48	0

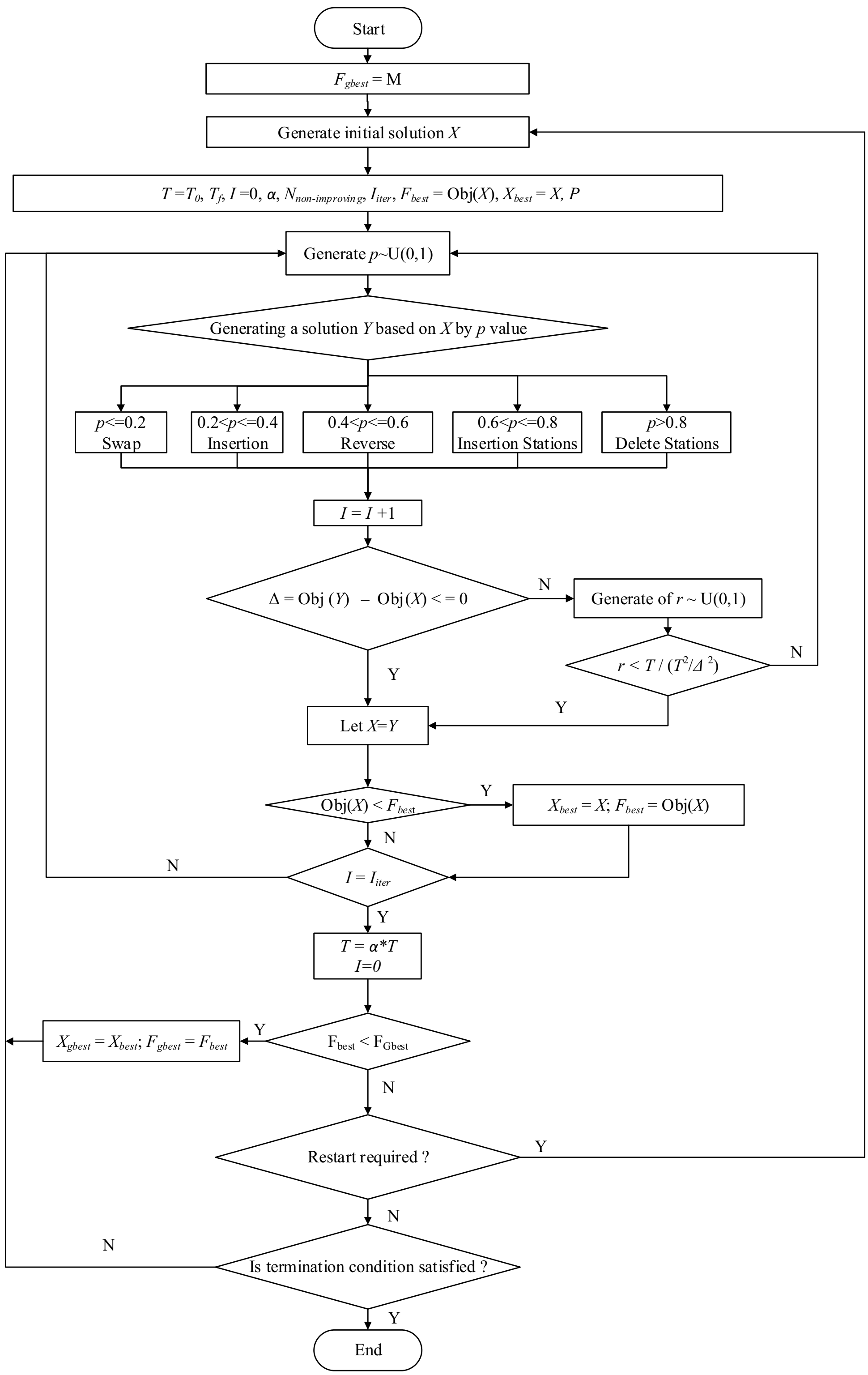

Fig. 3. Flow chart of the proposed simulated annealing heuristic with restart strategy using Cauchy function for HVRP.

from [65]. Furthermore, the second computational experiment is conducted on the well-known benchmark CVRP problems of Christofides et al. (1979). First, the performance of SA_RS using the Cauchy function (SA_RS$_{CF}$) is compared with that using the Boltzmann function (SA_RS$_{BF}$). The experiment includes verifying the effect of a restart strategy by implementing SA without a restart strategy, named SA$_{CF}$ and SA$_{BF}$. Furthermore, in the first benchmark instances the best result between SA$_{BF}$, SA$_{CF}$, RS_SA$_{BF}$, and RS_SA$_{CF}$ is compared to DPSO-SA for the Capacitated Vehicle Routing Problem proposed by Chen, et al. [66] and the PACO algorithm for CVRP proposed by Kao, et al. [67]. The experiment in the second benchmark instances follows the same steps with the first benchmark instances. At first, a test to verify the best implementation of SA_RS is conducted. Finally, the best approach from the first step is compared to the best-known solution for the second benchmark instances.

Table 3
Model verification results.

From	To	Distance	Total Distance	EP Remain	FP Remain	EP Cost	FP Cost	Time	Travel Time
0	7	122.369	122.369	0	341.131	1.26	28.3832	3.05921	3.05921
7	8	43.8833	166.252	0	297.248	1.26	40.6705	1.09708	4.6563
8	5	28.2536	194.505	0	268.995	1.26	48.5815	0.706339	5.86264
5	2	8.19216	202.698	0	260.802	1.26	50.8753	0.204804	6.56744
2	1	123.464	326.162	0	158.338	2.52	79.5652	3.0866	9.95404
1	0	26.7109	352.872	0	131.628	3.78	81.1643	0.667771	10.9218
0	9	66.8953	419.768	0	396.605	5.04	94.0149	1.67238	1.67238
9	6	67.785	487.553	0	328.82	5.04	112.995	1.69462	3.86701
6	1	13.8335	501.386	0	314.986	5.04	116.868	0.345837	4.71284
1	0	26.7109	528.097	0	309.275	6.3	118.467	0.667771	5.68061

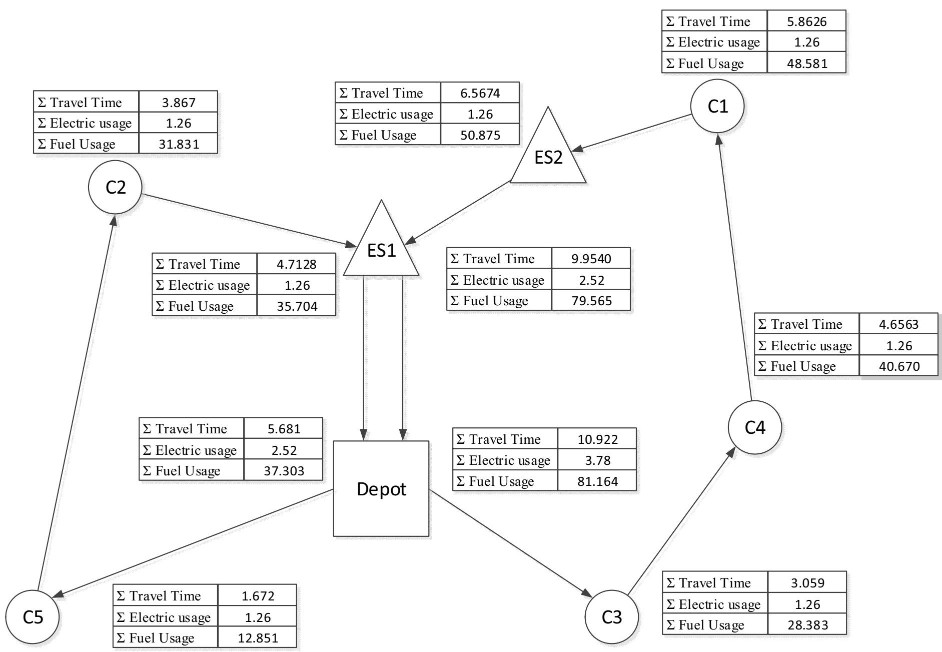

Fig. 4. Solution generated in Model verification.

Table 4
Computational results on the first set of benchmark problems.

No.	Instances	Exact*	SA_{BF}				SA_{CF}				SA_RS_{BF}				SA_RS_{CF}			
		Opt	Best	Average	Diff.	Time (s)	Best	Average	Diff.	Time (s)	Best	Average	Diff.	Time (s)	Best	Average	Diff.	Time (s)
1	A-n33-k5	661	661	662.64	0.00	0.29	661	662.64	0.00	0.21	661	661	0.00	1.16	661	661	0.00	1.25
2	A-n46-k7	914	914	915.91	0.00	1.67	914	914	0.00	1.96	914	917.45	0.00	5.07	914	914	0.00	5.11
3	A-n60-k9	1354	1360	1370.09	0.44	3.28	1354	1360.09	0.00	3.47	1354	1370.73	0.00	7.66	1354	1359.45	0.00	7.08
4	B-n35-k5	955	955	957.36	0.00	0.48	955	955	0.00	0.60	955	957.18	0.00	2.10	955	955	0.00	1.88
5	B-n45-k5	751	751	751	0.00	1.38	751	751	0.00	1.53	751	751	0.00	5.27	751	751	0.00	4.10
6	B-n68-k9	1272	1286	1288.27	1.10	4.36	1277	1285.27	0.39	4.18	1275	1283.45	0.24	8.19	1275	1281.55	0.24	7.60
7	B-n78-k10	1221	1239	1256.18	1.47	5.35	1229	1244.91	0.66	5.16	1238	1255.27	1.39	12.67	1221	1224.09	0.00	10.96
8	E-n30-k3	534	534	537.36	0.00	0.70	534	534	0.00	0.80	534	534	0.00	2.92	534	534	0.00	1.84
9	E-n51-k5	521	521	524.73	0.00	2.24	521	524.73	0.00	2.10	521	522	0.00	5.16	521	521	0.00	5.31
10	E-n76-k7	682	689	692.55	1.03	4.88	683	689.64	0.15	4.59	682	689.36	0.00	9.82	685	688.82	0.44	9.76
11	F-n72-k4	237	237	240.09	0.00	3.36	237	237	0.00	3.10	237	237	0.00	6.08	237	237	0.00	5.64
12	F-n135-k7	1162	1166	1187.82	0.34	10.55	1173	1193.27	0.95	10.00	1166	1189.36	0.34	32.95	1166	1170.73	0.34	26.33
13	M-n101-k10	820	820	836.45	0.00	7.38	820	823.82	0.00	7.25	820	825	0.00	17.62	820	821.73	0.00	14.77
14	M-n121-k7	1034	1034	1133.64	0.00	9.03	1034	1131.82	0.00	9.00	1036	1118.64	0.19	9.80	1034	1083.64	0.00	10.66
15	P-n76-K4	593	599	603.45	1.01	4.39	595	599.64	0.34	4.27	593	601.73	0.00	8.63	593	600.09	0.00	8.74
16	P-n101-k4	681	685	693.09	0.59	6.72	683	687.45	0.29	6.20	683	692.09	0.29	14.93	683	686.73	0.29	16.34
Average		837.00	840.69	853.16	0.37	4.13	838.81	849.64	0.17	4.03	838.75	850.33	0.15	9.38	837.75	843.11	0.08	8.59

*(COIN-OR, 2014).

The result shown in Table 4 is the computational result of SA_{BF}, SA_{CF}, RS_SA_{BF}, and RS_SA_{CF} for ten runs in the first benchmark instances. The best, average, percentage difference with the optimal solution (Diff.), and the best result computational time are compared (in s). In terms of the solution quality the best result obtained by RS_SA_{CF} has an average percentage difference of 0.08% and average objective result of 834.11, followed by RS_SA_{BF} (0.15%, 850.33), then SA_{CF} (0.17%, 849.64), and finally SA_{BF} (0.37%, 853.16).

To verify the significant improvement provided by the restart strategy and the Cauchy function, we conduct a paired *t*-test with

Table 5
Paired *t*-test results (α = 0.05) on average objective value of the first set of benchmark problems.

	SA_{BF}	SA_RS_{BF}	SA_{CF}	SA_RS_{CF},	SA_RS_{BF}	SA_RS_{CF}
Mean	853.1648	850.3295	849.642	843.1136	850.3295	843.1136
Variance	104577.5	104559.3	103988.2	100187.5	104559.3	100187.5
Observations	16	16	16	16	16	16
Hypothesized Mean Difference	0		0		0	
df	15		15		15	
t Stat	2.527755		1.976679		2.562998	
P(T < = t) two-tail	0.023198		0.066761		0.021633	
t Critical two-tail	2.13145		2.13145		2.13145	

Table 6
Computational results of Chen et al.'s (2006) and Kao et al.'s (2012) benchmark problems.

No.	Instances	Exact *		Chen at al. (2006)			Kao et al. (2012)			SA_RS_{CF}			
				DPSO-SA			Hybrid ACO and PSO						
		Optimal Solution	Number of Tours	Best Solution	Diff.	CPU Time (s)	Best Solution	Diff.	CPU Time (s)	Best Solution	Number of Tours	Diff.	CPU Time (s)
1	A-n33-k5	661	5	661	0.00%	32.3	661	0.00%	0.87	**661**	5	0.00%	1.25
2	A-n46-k7	914	7	914	0.00%	128.9	914	0.00%	6.02	**914**	7	0.00%	5.11
3	A-n60-k9	1354	9	1354	0.00%	308.8	1354	0.00%	52.88	**1354**	9	0.00%	7.08
4	B-n35-k5	955	5	955	0.00%	37.6	955	0.00%	2.65	**955**	5	0.00%	1.88
5	B-n45-k5	751	5	751	0.00%	134.2	751	0.00%	5.85	**751**	5	0.00%	4.10
6	B-n68-k9	1272	9	1272	0.00%	344.3	1275	0.24%	62.97	1275	9	0.24%	7.60
7	B-n78-k10	1221	10	1239	1.47%	429.4	1221	0.00%	98.78	**1221**	10	0.00%	10.96
8	E-n30-k3	534	3	534	0.00%	28.4	534	0.00%	4.38	**534**	3	0.00%	1.84
9	E-n51-k5	521	5	528	1.34%	300.5	521	0.00%	19.46	**521**	5	0.00%	5.31
10	E-n76-k7	682	7	688	0.88%	526.5	685	0.44%	46.85	685	7	0.44%	9.76
11	F-n72-k4	237	4	244	2.95%	398.3	237	0.00%	30.64	**237**	4	0.00%	5.64
12	F-n135-k7	1162	7	1215	4.56%	1526.3	1170	0.69%	248.77	1166	7	0.34%	26.33
13	M-n101-k10	820	10	824	0.49%	874.2	820	0.00%	113.28	**820**	10	0.00%	14.77
14	M-n121-k7	1034	7	1038	0.39%	1733.5	1034	0.00%	80.62	**1034**	7	0.00%	10.66
15	P-n76-K4	593	4	602	1.52%	496.3	593	0.00%	53.48	**593**	4	0.00%	8.74
16	P-n101-k4	681	4	694	1.91%	977.5	683	0.29%	64.92	683	4	0.29%	16.34
Average		837		844.5625	0.97%		838	0.10%		837.75		0.08%	

* (COIN-OR, 2014).

α=0.05 to compare the performance between SA_{BF} with RS_SA_{BF}, and SA_{CF} with RS_SA_{CF}, and RS_SA_{BF} with RS_SA_{CF}. Table 5 presents the result of this test. The first conclusion from the second column describes that there is a significant difference in the average objective value between SA_{BF} and RS_SA_{BF} with a *p*-value of 0.023198 that falls below the α value. However, results from the third column show that the test fails to show there is a significant difference in the result between SA_{CF} and RS_SA_{CF} with the *p*-value (0.066761) slightly higher than the α value. The last test from the fourth column verifies that there is a significant difference in the result between RS_SA_{BF} with RS_SA_{CF} with a *p*-value of 0.021633 that falls below the α value. Therefore, we decide to compare the result of RS_SA_{CF} with the results of other metaheuristics for this benchmark problem, as seen in Table 6. The table shows that the proposed RS_SA_{CF} can provide the optimal solution for 12 instances from a total of 16 tested instances. The average percentage gap between the objective value of the solution obtained by the proposed RS_SA_{CF}, and the optimal solution is 0.08%, which is lower than that of DPSO-SA (0.97%) and PACO (0.104%).

Table 7 provides a comparison between the best results obtained by SA_{BF}, SA_{CF}, RS_SA_{BF}, and RS_SA_{CF} and the best-known solutions (BKS) on the Christofides, et al. [68] benchmark problems for ten runs. Table 8 lists the average results, which show that the proposed SA is competitive for solving the problems. All four algorithms generate four best-known solutions out of the 14 benchmark instances (shown in bold), and the average deviation from BKS is only 1.5%, 1.17%, 1.03%, and 0.64% for SA_{BF}, SA_{CF}, RS_SA_{BF}, and RS_SA_{CF} respectively.

As with the first benchmark problem, we conduct a paired *t*-test with α=0.05 to verify the performance between SA_{BF} with RS_SA_{BF}, and SA_{CF} with RS_SA_{CF}. Table 9 presents the result of this test. The first conclusion from the second column describes that there is a significant difference in the average objective value between SA_{BF} and RS_SA_{BF} with a *p*-value of 0.033518 that falls below the α value. Similar to the first conclusion, the results from the third column show that there is a significant difference in the result between SA_{CF} and RS_SA_{CF} with the *p*-value (0.004343) lower than the α value. The last test from the fourth column verifies that there is a significant difference in the result between RS_SA_{BF} with RS_SA_{CF} with a *p*-value of 0.018438 that falls below the α value. The result convinces us that RS_SA_{CF} is the best option to be further used in the computational experiment. Moreover, the proposed RS_SA_{CF} also has an acceptable computational time.

6.3. Numerical experiment for HVRP

The numerical experiments in this section are for determining the effect of using a plug-in hybrid electric vehicle compared to a mild hybrid vehicle and conventional vehicles. We also investigate the effect of the nearest neighbor heuristic for initial solution on the total cost. These experiments are based on the datasets from Chen, et al. [66]. The number of customers ranges from 29 to 134, and a set of electric stations and fuel stations is included.

6.3.1. Impact of using plug-in hybrid electric vehicles

The objective function of HVRP is minimizing the total travel cost. This section compares the impact of using plug-in hybrid electric vehicles versus mild hybrid electric vehicles and conventional vehicles. A plug-in hybrid electric vehicle is a hybrid electric vehicle with the capability to recharge the battery from external resources,

Table 7
Computational results on the second set of benchmark problems.

Instance	n	C	S	T	m	Best Objective Value					Diff. 1 (%)	Diff. 2 (%)	Diff. 3 (%)	Diff. 4 (%)	CPU Time 1 (s)	CPU Time 2 (s)	CPU Time 3 (s)	CPU Time 4 (s)
						BKS	SA_{BF}	SA_{CF}	SA_RS_{BF}	SA_RS_{CF}								
vrpnc1	50	160	0	∞	5	524.61[a]	**524.61**	**524.61**	**524.61**	**524.61**	0.00	0.00	0.00	0.00	10.52	12.02	10.84	20.37
vrpnc2	75	140	0	∞	10	835.26[a]	845.006	846.05	847.03	844.97	1.15	1.28	1.39	1.15	15.62	21.83	15.78	30.69
vrpnc3	100	200	0	∞	8	826.14[a]	830.003	830.003	833.32	830.19	0.47	0.47	0.86	0.49	20.06	26.17	20.30	52.91
vrpnc4	150	200	0	∞	12	1028.42[a]	1083.38	1067.4	1047.35	1047.35	5.07	3.65	1.81	1.81	36.03	44.67	35.90	104.97
vrpnc5	199	200	0	∞	17	1291.29[b]	1358.02	1337.25	1340.31	1291.29	4.91	3.44	3.66	0.00	60.75	103.03	60.24	194.74
vrpnc6	50	160	10	200	6	555.43[a]	**555.43**	**555.43**	**555.43**	**555.43**	0.00	0.00	0.00	0.00	10.76	12.15	11.03	21.10
vrpnc7	75	140	10	160	11	909.68[a]	909.68	909.68	909.68	909.68	0.00	0.00	0.00	0.00	15.84	19.83	15.96	38.79
vrpnc8	100	200	10	230	9	865.94[a]	877.73	876.43	876.06	874.7	1.34	1.20	1.16	1.00	20.31	25.07	20.46	60.31
vrpnc9	150	200	10	200	14	1162.55[a]	1191.6	1187.3	1187.65	1178.34	2.44	2.08	2.11	1.34	36.19	41.90	36.32	121.31
vrpnc10	199	200	10	200	18	1395.85[c]	1470.79	1454.27	1437.11	1436.92	5.10	4.02	2.87	2.86	59.90	31.73	60.45	189.56
vrpnc11	120	200	0	∞	7	1042.11[a]	1047.94	1044.09	1047.96	1045.08	0.56	0.19	0.56	0.28	24.27	28.60	24.90	67.94
vrpnc12	100	200	0	∞	10	819.56[a]	**819.56**	**819.56**	**819.56**	**819.56**	0.00	0.00	0.00	0.00	20.30	22.96	20.58	39.47
vrpnc13	120	200	50	720	11	1541.14[a]	1541.14	1541.14	1541.14	1541.14	0.00	0.00	0.00	0.00	24.55	30.98	24.89	86.14
vrpnc14	100	200	90	1040	11	866.37[a]	**866.37**	**866.37**	**866.37**	**866.37**	0.00	0.00	0.00	0.00	20.39	25.24	20.64	54.51
Average						976.03	994.38	989.97	988.11	983.26	1.50	1.17	1.03	0.64	26.82	31.87	27.02	77.34

[a] Obtained from Taillard (1993).
[b] Obtained from Mester and Braysy (2007).
[c] Obtained from Rochat and Taillard (1995).

Table 8
Average objective values of the second set of benchmark problems.

Instance	Average Objective Value			
	SABF	SACF	SA_RS_{BF}	SA_RS_{CF}
vrpnc1	**525.19**	**530.01**	**525.97**	**525.41**
vrpnc2	856.17	856.54	856.72	854.39
vrpnc3	836.55	843.92	837.41	838.96
vrpnc4	1100.31	1085.56	1079.06	1074.14
vrpnc5	1384.78	1368.11	1359.05	1333.34
vrpnc6	**558.24**	**558.93**	**558.53**	**557.66**
vrpnc7	923.30	918.54	917.54	915.62
vrpnc8	892.08	892.33	887.21	879.35
vrpnc9	1216.48	1224.70	1212.83	1204.62
vrpnc10	1504.12	1477.73	1476.08	1468.14
vrpnc11	1052.78	1051.63	1051.37	1050.20
vrpnc12	**819.99**	**822.26**	**820.40**	**819.81**
vrpnc13	1561.68	1565.17	1556.84	1546.05
vrpnc14	**868.08**	**868.83**	**868.71**	**868.96**
	1007.13	1004.59	1000.55	995.47

Table 9
Paired *t*-test result ($\alpha = 0.05$) on average objective value of the second set of benchmark problems.

	SA_{BF}	SA_RS_{BF}	SA_{CF}	SA_RS_{CF},	SA_RS_{BF}	SA_RS_{CF}
Mean	1007.127	1000.552	1004.59	995.4732	1000.552	995.4732
Variance	100924.4	96568.04	97748.83	93646.16	96568.04	93646.16
Observations	14	14	14	14	14	14
Hypothesized Mean Difference	0		0		0	
df	13		13		13	
t Stat	2.37664		3.445794		2.693022	
P(T<=t) two-tail	0.033518		0.004343		0.018438	
t Critical two-tail	2.160369		2.160369		2.160369	

which is a set-up that cannot be considered for a mild hybrid electric vehicle. A conventional vehicle is a vehicle that only uses fuel as an energy source. Table 1 gives the vehicle data. The characteristic of a mild hybrid electric vehicle is its inability to run independently on electric power. Therefore, the assumption is simplified by giving a mild hybrid electric vehicle the extended maximum distance coverage of 442.5 miles. On the other hand, the conventional vehicle has distance coverage of 397.5 miles since it can only use fuel power.

Table 10 shows the impact of using plug-in hybrid electric vehicles in three scenarios. The first scenario consists of 0 electric stations and fuel stations, E0G0. The second scenario considers 2 electric charging stations and 2 fuel stations (E2F2). The third scenario has 4 electric charging stations and 4 fuel stations (E4F4).

The average cost results of the plug-in hybrid electric vehicle for E0F0, E2F2, and E4F4 (Table 6) are lower than those for the mild hybrid vehicle and conventional vehicle. Moreover, the average cost per mile for PHEV of all scenarios is lower than those for the mild hybrid vehicle and conventional vehicle. For example, in scenario E0F0, the costs per mile for PHEV, the mild hybrid vehicle, and the conventional vehicle are US$0.21, US$0.24, and US$0.24, respectively. The cost per mile for PHEV falls along with the increasing number of electric charging stations, however, for the mild hybrid and conventional vehicles the costs are relatively the same.

6.3.2. Impact of initial solution

RS_SA_{CF} typically requires a good approach to efficiently obtain an initial solution and then continues to improve the solution. Two well-known approaches, nearest neighbor heuristic and random approach for generating an initial solution, are preferred in many

Table 10
Impact of using plug-in hybrid electric vehicles.

No.	Based Instances	No. of Customers	E0G0						E2G2						E4G4					
			Plug-in Hybrid		Mild Hybrid		Conventional		Plug-in Hybrid		Mild Hybrid		Conventional		Plug-in Hybrid		Mild Hybrid		Conventional	
			Miles	Cost	Miles	Cost	Miles	Cost	Miles	Cost	Miles	Cost	Miles	Cost	Miles	Cost	Miles	Cost	Miles	Cost
1	A-n33-k5	32	521	113.7	524	125.76	521	125.04	527	111.36	521	125.04	524	125.76	512	103.98	505	121.2	505	121.2
2	A-n46-k7	45	720	157.68	720	172.8	720	172.8	750	154.08	735	176.4	720	172.8	754	150	720	172.8	720	172.8
3	A-n60-k9	59	944	205.5	958	229.92	952	228.48	979	201.84	952	228.48	951	228.24	970	195.9	951	228.24	951	228.24
4	B-n35-k5	34	758	166.8	758	181.92	758	181.92	778	155.58	758	181.92	758	181.92	814	144.06	759	182.16	758	181.92
5	B-n45-k5	44	641	138.72	642	154.08	642	154.08	678	132.66	646	155.04	641	153.84	768	123.66	642	154.08	647	155.28
6	B-n68-k9	67	875	187.32	874	209.76	874	209.76	981	186.3	874	209.76	875	210	914	174	875	210	918	220.32
7	B-n78-k10	77	929	196.5	907	217.68	906	217.44	952	183.84	878	210.72	908	217.92	996	171.18	907	217.68	903	216.72
8	E-n30-k3	29	465	100.26	465	111.6	465	111.6	522	97.38	465	111.6	465	111.6	556	86.82	465	111.6	465	111.6
9	E-n51-k5	50	490	102.48	493	118.32	492	118.08	534	93.24	490	117.6	493	118.32	574	81.78	493	118.32	500	120
10	E-n76-k7	75	637	133.98	660	158.4	643	154.32	754	132.9	686	164.64	640	156.96	775	113.82	686	164.64	680	163.2
11	F-n72-k4	71	248	44.4	252	60.48	254	60.96	322	29.76	248	59.52	253	60.72	349	22.2	254	60.96	253	60.72
12	F-n135-k7	134	1487	319.08	1486	356.64	1456	349.44	1481	317.64	1523	365.52	1504	360.96	1616	313.14	1519	364.56	1473	353.52
13	M-n101-k10	100	719	146.1	723	173.52	713	171.12	755	136.2	736	176.64	738	177.12	813	122.22	710	170.4	704	168.96
14	M-n121-k7	120	1476	313.92	1476	354.24	1446	347.04	1717	304.44	1508	361.92	1527	366.48	1696	270.06	1486	356.64	1492	358.08
15	P-n76-K4	75	673	138.84	643	154.32	646	155.04	726	125.46	707	162.72	634	152.16	741	114.84	643	154.32	657	157.68
16	P-n101-k4	100	813	168.66	789	189.36	799	191.76	870	155.88	647	155.28	829	198.96	981	154.08	825	198	811	194.64
Average			**774.75**	**164.62**	**773.13**	**185.55**	**767.94**	**184.31**	**832.88**	**157.41**	**773.38**	**185.18**	**778.75**	**187.11**	**864.31**	**146.36**	**777.50**	**186.60**	**777.31**	**186.56**
Percentage*						**12.71%**		**11.96%**				**17.64%**		**18.87%**				**27.49%**		**27.46%**
Cost/mile (US$)			**0.21**		**0.24**		**0.24**		**0.19**		**0.24**		**0.24**		**0.17**		**0.24**		**0.24**	

*Percentage = [(Mild Hybrid Vehicle or Conventional Vehicle − Plug In Hybrid Vehicle)/Plug-In Hybrid Vehicle]*100.

Table 11
Impact of using nearest neighbor heuristic to generate initial HVRP solutions.

No.	Based Instances	No. of Customers	E2G2					
			Nearest Neighbor		Random Initial		Gap	
			Initial solution	Final solution	Initial solution	Final solution	Initial solution	Final solution
1	A-n33-k5	32	175.8	111.36	448.14	114.42	−60.77%	−2.67%
2	A-n46-k7	45	232.86	154.08	654	154.08	−64.39%	0.00%
3	A-n60-k9	59	330.54	199.32	905.94	202.98	−63.51%	−1.80%
4	B-n35-k5	34	167.64	156	634.74	156	−73.59%	0.00%
5	B-n45-k5	44	149.64	133.32	687.9	131.16	−78.25%	1.65%
6	B-n68-k9	67	228.24	186.66	986.94	186.84	−76.87%	−0.10%
7	B-n78-k10	77	252.78	183.84	1054.44	185.46	−76.03%	−0.87%
8	E-n30-k3	29	167.64	95.7	452.7	95.7	−62.97%	0.00%
9	E-n51-k5	50	120.6	92.64	371.1	91.86	−67.50%	0.85%
10	E-n76-k7	75	182.4	122.46	593.16	131.16	−69.25%	−6.63%
11	F-n72-k4	71	63.42	31.02	332.58	31.2	−80.93%	−0.58%
12	F-n135-k7	134	391.02	321.36	1721.1	315.66	−77.28%	1.81%
13	M-n101-k10	100	164.94	133.02	999.96	136.56	−83.51%	−2.59%
14	M-n121-k7	120	358.8	291.42	1756.86	288.24	−79.58%	1.10%
15	P-n76-K4	75	175.44	126.96	675.24	124.32	−74.02%	2.12%
16	P-n101-k4	100	221.64	154.56	882.06	156	−74.87%	−0.92%
Average			**211.4625**	**155.8575**	**822.30375**	**156.3525**	−74.28%	−0.32%

Gap = [Nearest Neighbor − Random Initial/Random Initial]*100.

studies on VRP and its extensions. This section observes the effect of these two approaches on the proposed problem.

Table 11 presents the impact of using the nearest neighbor heuristic to construct the initial solution in the E2G2 scenario. The results are compared with those obtained from the random initialization approach. The initial solutions obtained from the nearest neighbor heuristic consistently outperform the random initial solutions by an average of 74.28%. Moreover, the nearest neighbor heuristic leads to better final solutions by 0.32%. The results show that the nearest neighbor heuristic provides a good initial solution for the RS_SA_{CF} algorithm. By providing a fine initial solution, the RS_SA_{CF} algorithm can perform more consistently and provide better final solutions for HVRP.

7. Conclusions

This research proposes the hybrid vehicle routing problem, which is an extension of the Green Vehicle Routing Problem. A simulated annealing heuristics with a restart strategy (SA_RS) is proposed to solve HVRP. We implement SA_RS with the Boltzman function (SA_RS_{BF}) and the Cauchy function as acceptance criteria for a worse solution (SA_RS_{CF}). A statistical test is conducted to compare the performances of the solution approach with and without a restart strategy and the solution approach using the Boltzman and Cauchy functions. In general, the statistical test show that SA_RS_{CF} is better than SA_RS_{BF}, SA_RS_{BF} is better than SA_{BF}, and SA_RS_{CF} is better than SA_{CF}. Therefore, in this computational experiment SA_RS_{CF} is the most preferable among others.

We then use the CVRP benchmark problems to verify the SA_RS algorithm. The first benchmark is taken from Chen, et al. [66] and Kao, et al. [67]. Computational results show that SA_RS_{CF} performs better compared to the other metaheuristics on the benchmark problems. The average difference between solutions of the proposed SA and the optimal solutions is 0.08%, which is smaller than that of DPSO-SA (0.97%) and PACO (0.104%). The second benchmark has been taken from the well-known CVRP benchmark of Christofides, et al. [68]. It can be seen that the proposed SA_RS_{CF} heuristic has found 4 of the 14 best known solutions, and the average deviation from the best solution is only 0.64%. The result convinces us that SA_RS_{CF} is good for solving HVRP.

Sensitivity analysis has been conducted to understand the effect of hybrid vehicles and charging stations on the travel cost in HVRP. The first experiment compares the impact of using plug-in hybrid electric vehicles versus using mild hybrid electric vehicles and conventional vehicles. Numerical experiments show that using PHEV offers a lower travel cost compared to the other two types of vehicle.

Another sensitivity experiment is conducted to rationalize the use of the nearest neighbor heuristic to generate an initial solution for the SA_RS_{CF} algorithm. The nearest neighbor heuristic is modified from the original version in order to deal with HVRP characteristics. The heuristic is compared to a random approach. The results indicate that the nearest neighbor heuristic gives better initial solutions than the random approach.

Utilizing HVRP can help companies or organizations reduce their total travel cost by determining the optimal route that serves the most consumers within a certain time limit. Sensitivity analysis illustrates that the government should increase the number of electric charging stations in order to reduce dependence on fossil fuels. This research also provides encouragement for companies or organizations to replace their conventional vehicles with fuel-efficient vehicles as well as with environmental friendly vehicles.

Acknowledgement

The authors thank the editors and referees for their helpful comments and suggestions. This research was partially supported by the Ministry of Science and Technology of the Republic of China (Taiwan) under grant NSC 102-2221-E-011-082-MY3. This support is gratefully acknowledged.

References

[1] B.M. Beamon, Supply chain design and analysis: models and methods, Int. J. Prod. Econ. 55 (1998) 281–294.

[2] A.B. Borade, S.V. Bansod, Domain of supply chain management – a state of art, J. Technol. Manage. Innov. 2 (2008) 109–121.

[3] B.o.T. Economics, Logistics in Australia, a Preliminary Analysis, Bureau of Transport Economics, Canberra, 2001.

[4] Y.-y. Tseng, W.L. Yue, M.A. Taylor, v The role of transportation in logistics chain, East. Asia Soc. Transp. Stud. 5 (2005) 1657–1672.

[5] B. Reed, M. Smas, R. Rzepka, A. Guiffrida, Introducing green transportation costs in supply chain modeling, Proceedings of the First Annual Kent State International Symposium on Green Supply Chains (2010) 189–197.

[6] R.Z. Farahani, S. Rezapour, L. Kardar, Logistics Operations and Management: Concepts and Models, Elsevier, 2011.

[7] L. Ping, Strategy of green logistics and sustainable development, International Conference on Information Management, Innovation Management and Industrial Engineering IEEE Computer Society (2009) 339–342.

[8] S. Erdoğan, E. Miller-Hooks, A green vehicle routing problem, Transp. Res. Part E: Logist. Transp. Rev. 48 (2012) 100–114.

[9] G.B. Dantzig, J.H. Ramser, The truck dispatching problem, Manage. Sci. 6 (1959) 80–91.
[10] C. Lin, K.L. Choy, G.T.S. Ho, S.H. Chung, H.Y. Lam, Survey of green vehicle routing problem: past and future trends, Expert Syst. Appl. 41 (2014) 1118–1138.
[11] M. Schneider, A. Stenger, D. Goeke, The electric vehicle-routing problem with time windows and recharging stations, Transp. Sci. (2014).
[12] E. Demir, T. Bektaş, G. Laporte, A review of recent research on green road freight transportation, Eur. J. Oper. Res. 237 (2014) 775–793.
[13] S. Pelletier, O. Jabali, G. Laporte, 50th anniversary invited Article—Goods distribution with electric vehicles: review and research perspectives, Transp. Sci. 50 (2016) 3–22.
[14] Á. Felipe, M.T. Ortuño, G. Righini, G. Tirado, A heuristic approach for the green vehicle routing problem with multiple technologies and partial recharges, Transp. Res. Part E: Logist. Transp. Rev. 71 (2014) 111–128.
[15] J. Yang, H. Sun, Battery swap station location-routing problem with capacitated electric vehicles, Comput. Oper. Res. 55 (2015) 217–232.
[16] C. Chan, The state of the art of electric, hybrid, and fuel cell vehicles, Proc. IEEE 95 (2007) 704–718.
[17] M.A. Hannan, F.A. Azidin, A. Mohamed, Hybrid electric vehicles and their challenges: a review, Renew. Sustain. Energy Rev. 29 (2014) 135–150.
[18] S. Amjad, S. Neelakrishnan, R. Rudramoorthy, Review of design considerations and technological challenges for successful development and deployment of plug-in hybrid electric vehicles, Renew. Sustain. Energy Rev. 14 (2010) 1104–1110.
[19] C. Silva, M. Ross, T. Farias, Evaluation of energy consumption, emissions and cost of plug-in hybrid vehicles, Energy Convers. Manage. 50 (2009) 1635–1643.
[20] T.H. Bradley, A.A. Frank, Design, demonstrations and sustainability impact assessments for plug-in hybrid electric vehicles, Renew. Sustain. Energy Rev. 13 (2009) 115–128.
[21] I. Kara, B.Y. Kara, M.K. Yetis, Energy minimizing vehicle routing problem, in: Combinatorial Optimization and Applications, Springer, 2007, pp. 62–71.
[22] Y. Kuo, Using simulated annealing to minimize fuel consumption for the time-dependent vehicle routing problem, Comput. Ind. Eng. 59 (2010) 157–165.
[23] Y.Y. Xiao, Q.H. Zhao, I. Kaku, Y.C. Xu, Development of a fuel consumption optimization model for the capacitated vehicle routing problem, Comput. Oper. Res. 39 (2012) 1419–1431.
[24] M. Gendreau, G. Laporte, J.-Y. Potvin, Metaheuristics for the capacitated VRP, in: the vehicle routing problem, Soc. Ind. Appl. Math. (2001) 129–154.
[25] G. Laporte, Fifty years of vehicle routing, Transp. Sci. 43 (2009) 408–416.
[26] T. Vidal, T.G. Crainic, M. Gendreau, C. Prins, Heuristics for multi-attribute vehicle routing problems: a survey and synthesis, Eur. J. Oper. Res. 231 (2013) 1–21.
[27] S. Kirkpatrick, M.P. Vecchi, Optimization by simmulated annealing, Science 220 (1983) 671–680.
[28] I.H. Osman, Metastrategy simulated annealing and tabu search algorithms for the vehicle routing problem, Ann. Oper. Res. 41 (1993) 421–451.
[29] F. Glover, M. Laguna, Tabu Search∗, Springer, 2013.
[30] É. Taillard, Parallel iterative search methods for vehicle routing problems, Networks 23 (1993) 661–673.
[31] Y. Rochat, É.D. Taillard, Probabilistic diversification and intensification in local search for vehicle routing, J. Heuristics 1 (1995) 147–167.
[32] N. Mladenović, P. Hansen, Variable neighborhood search, Comput. Oper. Res. 24 (1997) 1097–1100.
[33] J. Kytöjoki, T. Nuortio, O. Bräysy, M. Gendreau, An efficient variable neighborhood search heuristic for very large scale vehicle routing problems, Comput. Oper. Res. 34 (2007) 2743–2757.
[34] P. Chen, H.-K. Huang, X.-Y. Dong, Iterated variable neighborhood descent algorithm for the capacitated vehicle routing problem, Expert Syst. Appl. 37 (2010) 1620–1627.
[35] S. Ropke, D. Pisinger, An adaptive large neighborhood search heuristic for the pickup and delivery problem with time windows, Transp. Sci. 40 (2006) 455–472.
[36] H.R. Lourenço, O.C. Martin, T. Stützle, Iterated Local Search, Springer, 2003.
[37] J.H. Holland, Adaptation in Natural and Artificial Systems: an Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence, University of Michigan Press, 1975.
[38] P. Moscato, On Evolution Search, Optimization, Genetic Algorithms and Martial Arts: Towards Memetic Algorithms, Caltech Concurrent Computation Program, C3P Report, 826, 1989.
[39] P. Moscato, C. Cotta, A modern introduction to memetic algorithms, in: Handbook of Metaheuristics, Springer, 2010, pp. 141–183.
[40] F. Glover, Scatter Search, Path Relinking, New Ideas in Optimization, 297316, 1999.
[41] M.G. Resende, C.C. Ribeiro, F. Glover, R. Martí, Scatter search and path-relinking: fundamentals, advances, and applications, in: Handbook of Metaheuristics, Springer, 2010, pp. 87–107.
[42] R.C. Eberhart, J. Kennedy, A new optimizer using particle swarm theory, in: Proceedings of the Sixth International Symposium on Micro Machine and Human Science, New York, NY, 1995, pp. 39–43.
[43] R. Poli, J. Kennedy, T. Blackwell, Particle swarm optimization, Swarm Intell. 1 (2007) 33–57.
[44] M. Dorigo, T. Stützle, The ant colony optimization metaheuristic: algorithms, applications, and advances, in: Handbook of Metaheuristics, Springer, 2003, pp. 250–285.
[45] C. Prins, A GRASP × evolutionary local search hybrid for the vehicle routing problem, in: Bio-inspired Algorithms for the Vehicle Routing Problem, Springer, 2009, pp. 35–53.
[46] J.-F. Cordeau, M. Maischberger, A parallel iterated tabu search heuristic for vehicle routing problems, Comput. Oper. Res. 39 (2012) 2033–2050.
[47] N. Metropolis, A.W. Rosenbluth, M.N. Rosenbluth, A.H. Teller, E. Teller, Equation of state calculations by fast computing machines, J. Chem. Phys. 21 (1953) 1087–1092.
[48] C. Avila, F. Valdez, An improved simulated annealing algorithm for the optimization of mathematical functions, in: Design of Intelligent Systems Based on Fuzzy Logic, Neural Networks and Nature-Inspired Optimization, Springer, 2015, pp. 241–251.
[49] O. Castillo, H. Neyoy, J. Soria, P. Melin, F. Valdez, A new approach for dynamic fuzzy logic parameter tuning in Ant Colony Optimization and its application in fuzzy control of a mobile robot, Appl. Soft Comput. 28 (2015) 150–159.
[50] L. Astudillo, P. Melin, O. Castillo, Introduction to an optimization algorithm based on the chemical reactions, Inf. Sci. 291 (2015) 85–95.
[51] F. Li, J. Qiao, H. Han, C. Yang, A self-organizing cascade neural network with random weights for nonlinear system modeling, Appl. Soft Comput. 42 (2016) 184–193.
[52] S.-W. Lin, Z.-J. Lee, K.-C. Ying, C.-Y. Lee, Applying hybrid meta-heuristics for capacitated vehicle routing problem, Expert Syst. Appl. 36 (2009) 1505–1512.
[53] V.F. Yu, S.-W. Lin, W. Lee, C.-J. Ting, A simulated annealing heuristic for the capacitated location routing problem, Comput. Ind. Eng. 58 (2010) 288–299.
[54] S.-W. Lin, V.F. Yu, C.-C. Lu, A simulated annealing heuristic for the truck and trailer routing problem with time windows, Expert Syst. Appl. 38 (2011) 15244–15252.
[55] S.-W. Lin, V.F. Yu, A simulated annealing heuristic for the team orienteering problem with time windows, Eur. J. Oper. Res. 217 (2012) 94–107.
[56] V.F. Yu, P. Jewpanya, A.A.N.P. Redi, Open vehicle routing problem with cross-docking, Comput. Ind. Eng. 94 (2016) 6–17.
[57] V.F. Yu, S.-W. Lin, Multi-start simulated annealing heuristic for the location routing problem with simultaneous pickup and delivery, Appl. Soft Comput. 24 (2014) 284–290.
[58] S.-W. Lin, V.F. Yu, A simulated annealing heuristic for the multiconstraint team orienteering problem with multiple time windows, Appl. Soft Comput. 37 (2015) 632–642.
[59] C. Botsford, A. Szczepanek, Fast charging vs. slow charging: pros and cons for the new age of electric vehicles, International Battery Hybrid Fuel Cell Electric Vehicle Symposium (2009).
[60] T.A.F.D.C. AFDC, EERE: Alternative Fuels Data Center Home Page, in, 2014.
[61] M. Gendreau, J.-Y. Potvin, Metaheuristics in combinatorial optimization, Annals of Operations Research 140 (2005) 189–213.
[62] M. Gendreau, J.-Y. Potvin, Handbook of Metaheuristics, Springer, 2010.
[63] S.M. Mousavi, R. Tavakkoli-Moghaddam, A hybrid simulated annealing algorithm for location and routing scheduling problems with cross-docking in the supply chain, J. Manuf. Syst. 32 (2013) 335–347.
[64] M.K. Tiwari, S. Kumar, R. Shankar, Solving part-type selection and operation allocation problems in an FMS: An approach using constraints-based fast simulated annealing algorithm, IEEE Trans. Syst. Man Cybern. Part A: Syst. Hum. 36 (2006) 1170–1184.
[65] COIN-OR, Branch and Cut for Vehicle Routing, in, 2014.
[66] A.-L. Chen, G.-K. Yang, Z.-M. Wu, Hybrid discrete particle swarm optimization algorithm for capacitated vehicle routing problem, J. Zhejiang Univ: Sci. A 7 (2006) 607–614.
[67] Y. Kao, M.-H. Chen, Y.-T. Huang, A hybrid algorithm based on ACO and PSO for capacitated vehicle routing problems, Math. Prob. Eng. 2012 (2012) 1–17.
[68] N. Christofides, A. Mingozzi, P. Toth, Combinatorial Optimization, John Wiley & Sons, Chichester, UK, 1979.