A genetic local search algorithm for the capacitated vehicle routing problem

Abdeljawed Sadok[*]
Department of Management Information Systems and Producion Management, College of Business and Economics, Qassim University, Buraidah 51452, Qassim, Saudi Arabia

Abstract
This paper aims to examine the capacitated vehicle routing problem (CVRP). It is a major logistics problem encountered by any company that must organize the distribution of its products. The CVRP is an NP-hard problem. This problem involves planning vehicle routes that are to serve a number of customers from a depot. The capacity of each vehicle is limited and each customer has demanded who must be satisfied. The objective of this problem is to minimize the total distance travelled by vehicles. It is a very broad class of problems, including the famous traveling salesman problem (TSP). The goal of this paper is to find a solution for CVRP using a hybrid heuristic. This heuristic, called, genetic local search algorithm (GA-LS). We propose a genetic algorithm with a new procedure crossover operator to minimize the total travelled distance. To simplify the problem, a natural number coding is used and to assure enough diversification into the algorithm the best selection is retained. Local search uses performance indicators in order to maintain a balance between convergence and the diversity of the solutions obtained. Large numerical experiments are made to prove the efficiency of the proposed algorithm. Our approach is compared to the different existing approaches. The results show that our approach is very competitive in terms of the solutions obtained. Based on five reference data sets our approach obtains a total of 45 values equal to the best- known solution for 57 instances.

Keywords
Capacitated vehicle routing problem, Genetic algorithm, Optimization, Local search.

1.Introduction

Many sectors of industry, especially transport, which is a key sector of the economy, are affected by complex, large-scale problems for which decisions must be made optimally. In particular, a key question today for industrial companies is how can a better service be organized at the lowest cost to improve the efficiency of the supply chain? Consequently, the fundamental element of any logistics system is the management and planning of the distribution networks of the vehicle fleets. Our objective is to answer this question by proposing a simple and effective method in order to find a vehicle tour with minimum cost under a reasonable computational time. For this reason, we studied the capacitated vehicle routing problem (CVRP). The problem addressed in this paper is the more specific problem of VRP. The CVRP involves planning homogeneous vehicle routes to satisfy customer demand. the objective is the minimization of the total distance travelled by the vehicles.

This is a very large class of problems, including the famous the traveling salesman problem (TSP) which is posed very simply as follows: from a given geographical location, how to visit one after the other (in an order to be determined) and customers scattered over a given territory, so as to return to the starting point of traveling a minimum distance, this problem is probably one of the oldest combinatorial optimization problems and certainly one of the most studied. Indeed, it is at the root of all vehicle routing problems and a multitude of real-life applications.

Regarding the methods, several approaches were proposed to resolve these problems. The small-scale VRP problems are solved by an exact algorithm. For large-scale instances, the problem has been solved in several searches by approximate algorithms that can find near-optimal solutions in an acceptable time. Given their robustness and flexibility the evolutionary algorithms -GA- have attracted the attention of researchers, they have been used to solve many complex problems, including certain types of vehicle routing problems.

---

[*]Author for correspondence

Several researchers have used metaheuristics to solve the CVRP, such as Eksioglu et al. [1], Fisher [2], Golden et al. [3] and Braekers et al. [4]. For instance, Berger et al. [5] proposed a natural crossover operator, applied 2D space, was adopted with some modification in the suggested genetic algorithm. In order to solve the VRP, a hybrid genetic algorithm (HGA) was presented by Berger and Barkaoui [6] a periodic exchange of several individuals between two populations was used. The algorithm proved its performance in comparison to the best heuristics. In the same context, Baker and Ayechew [7] developed a hybrid genetic algorithm for the CVRP. The hybrid genetic algorithm applied neighborhood search procedures to ensure a feasible solution. This algorithm provided comparable solutions when compared to previous simulated annealing and Tabu search methods in a reasonable amount of computer time. Jeon et al. [8] have also proposed a hybrid genetic algorithm to solve the vehicle routing problem with a heterogeneous vehicle, double trips, and multiple depots. It considers the improvement of generation for an initial solution, three different heuristic processes, and a float mutation rate for escaping from the local solution in order to find the best solution. A two-phase approach for VRP with time windows has been proposed by Alvarenga et al. [9].

In order to generate routes, a hybrid genetic algorithm and a partitioning method have been developed. This approach has provided good solutions compared to the best-known solutions. The problem of VRP with multiple depots is presented by Ho et al. [10]. This problem has been solved by the genetic algorithm. The initial solution is determined by the nearest neighbor search algorithm and the C-W saving algorithm. Yurtkuran and Emel [11] proposed the hybridize Electromagne-tism-like Algorithm and a specific local search to solve CVRP, in this work the authors used a new objective function based on the Random-Key strategies. Ciu et al. [12] propose an improved quantum evolution algorithm (IQEA) hybridized with local search. The initial solution is determined by IQEA, the 1-1 exchange, 1-0 exchange, and 2-opt algorithm is used to improve the solution. De Carvalho and Delgado [13] proposed an algorithm based on the Monte Carlo method and Clarke and Wright's (1964) heuristic, called Monte Carlo Savings, The idea of the authors was to use the Monte Carlo method to generate the list of heuristic savings, which ensures the strategy a certain degree of variability and allows us to find diversified solutions. Mazidi et al. [14] propose an

improved genetic algorithm to solve the CVRP. The initial population for this algorithm is created by the ant colony algorithm. The idea of this paper is to prove the capability of the local search algorithm to solve the capacitated vehicle routing problem.

To solve CVRP, Vincent et al. [15] presents the symbiotic organisms' search (SOS) heuristic. They implemented two solution representations, SR-1 and SR-2, in order to apply SOS for CVRP. To improve the solutions, they applied a local search strategy. They developed six versions of SOS and they showed that two versions produce good CVRP solutions. Sun et al. [16] hybridized ant colony heuristics (ACO) and simulated annealing (SA), to have a new algorithm called simulated annealing colony optimization (SACO) to solve the CVRP. In order to improve the quality of the solution and reduce the computing time, they have extended their algorithm to a decomposed simulated annealing colony optimization (DSACO). Azad and Hasin [17] used a genetic algorithm to solve a real capacitated vehicle routing problem (cement company of Bangladesh). The method consists of grouping the suppliers in groups, then looking for the rounds with a minimum distance approaching or exact for each group of the supplier.

Based on the column generation technique Letchford and Salazar-González [18] considered several relaxations in order to solve the CVRP in pseudo-polynomial time. They have shown that among these relaxations, there are relaxations which are effective and which give extremely tight lower limits for the CVRP. Hosseinabadi et al. [19] have developed a new hybrid algorithm based on the gravitational emulation local search (GELS) and the genetic algorithm (GA) called (GELSGA). They compared this algorithm to other methods of literature. They have shown that their algorithm could obtain solutions close to the best-known solutions for many instance problems. Ho et al. [10] have developed a method based on a genetic algorithm and variable neighborhood search (VNS) for the case of the Tunisian Post Office. The objective was to find a solution for the Tunisian post to improve its services for businesses and citizens. They proved that the results found to improve the services of the Tunisian post compared to the existing methods.

In this work, we are interested in solving the CVRP and for this purpose; we present a hybrid genetic algorithm. The genetic algorithm, which is part of evolutionary techniques, is widely used to optimize

transport problems due to their efficiency in finding better solutions in the shortest time. This technique is based on an initial population and from this population, a selection, crossing and mutation procedure are carried out on individuals. The main contribution of this article is to propose a new resolution method based on evolutionary algorithms and to find better solutions compared to the literature in an acceptable time. *Figure 1* shows the example of CVRP (13 customers).
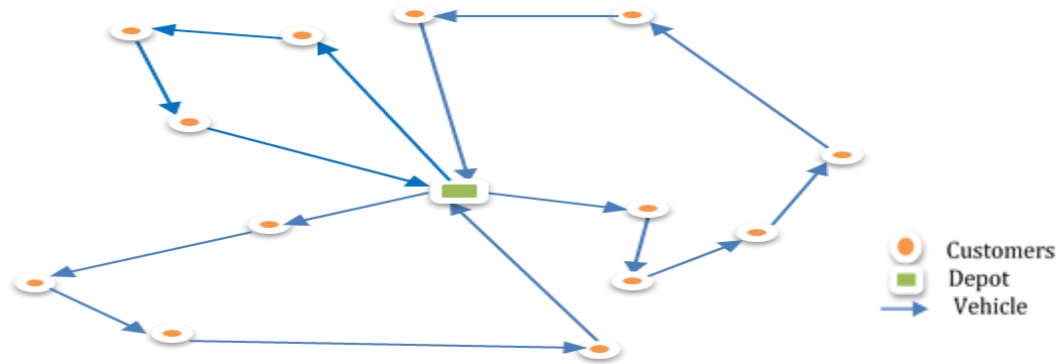
The rest of the paper is organized as follows. Section 2 briefly described the capacitated vehicle routing problems with the novel coding mechanism for the feasible solution and the genetic local search algorithm is presented in detail. Experimental results and analysis are reported in Section 3, followed by the conclusion in Section 4.

Figure 1 An example of CVRP (13 customers)

2.Related work
2.1The capacitated vehicle routing problem
CVRP is one of the important researches in the logistic distribution. the goal is to minimize the total cost of tours. The problem of vehicle routing is defined on a graph G (V, R) generally assumed complete. We present the problem description of the Vehicle Routing Problem (VRP) as follows.

Given a set of N customers (N =1, …, n) with a demand for qi units. at a depot, noted "0", K identical vehicles (K= 1, …, m) with a capacity of Q units. The tour starts and ends in the same depot. The distance to go from point i to point j is noted d_{ij}. This distance traveled that are symmetrical $\left(d_{ij} = d_{ji}\right)$. Customers should be visited in such a way that vehicle capacity constraint (Q) should not violate. Each customer should visit once by one vehicle and the objective function F(S) is thus the total distance for a solution S is:

$$\text{Min F (S)} = \sum_{i=0}^{n} \sum_{j=0}^{n} \sum_{k=1}^{m} d_{ij} X_{ijk} \qquad (1)$$

Where:
d_{ij} the cost (distance) of going from nodes i to j for any pair $(I, j) \in N$

X_{ijk} is a binary variable, which is 1 if a vehicle k goes from i to j without intermediate stops, otherwise $X_{ijk} = 0$.
The capacity constraint is:

$$\sum_{i=1}^{n} d_i y_i^k \leq Q_k \qquad k = 1, ..., m \qquad (2)$$

Where
d_i the demand for customer i
y_i^k = is a binary variable, which is 1 if a vehicle k visit i, otherwise $y_i^k = 0$.

2.2Genetic local search algorithm procedure
Genetic algorithms are used to improve the convergence rate and find a more optimal solution over the pure genetic algorithm. In this section, we offer a detailed description of our algorithm to solve the CVRP. These elements are the coding solution, fitness function, initial population, selection of tow parent, crossover operation, mutation operation, local search and the building of the new population.
2.2.1Solution coding
The coding of a solution and the crossing and mutation operators present the main stages of the genetic algorithm. The majority of researchers who have used the genetic algorithm, have represented the solutions in binary form. Where each solution is coded as a bit string of length *n*. One of the proposals made by this article focuses on the use of effective

real coding, which aims to reduce the computational effort. For this, we have proposed a natural number of coding of a solution. We use a group representation of a solution; each group represents a tour for a vehicle. Each tour is formed by an orderly sequence of customers and each customer must appear exactly once in a chromosome.

An excellent representation of the VRP solution must find out the number of vehicles, which customers are assigned to each vehicle and in which order they are visited. *Figure 2* shows an example of 8 customers.
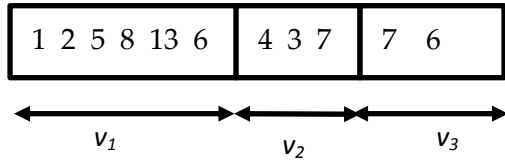

Figure 2 An example of chromosome (I=8)

2.2.2 The fitness function of the solution

In order to perform a natural selection, every individual is evaluated in terms of its fitness value determined by an evaluation function. The fitness function is simply the value of the objective function $F(S)$ associated with S. The fitness value measures the quality of the solutions and enables them to be compared (A solution S is thus better than a solution S' if its fitness functions are smaller).

2.2.3 Initial population

We initialize the genetic algorithm by creating solutions for the first generation until reaching a maximum population size of P. The determination of the initial population is made by two methods, either randomly or by heuristics. The latter has proven their efficiency in obtaining better solutions for a vehicle routing problem. For it, the initial solutions are generated by the nearest neighbor algorithm (NNA). The algorithm generates the tours for each vehicle. The NNA is as follows:

Step 1: Move all the customers to a list;

Step 2: Select a vehicle k;

Step 3: Select nearest customer to the depot as a present customer, assign it to tours T_k and remove it from the list;

Step 4: Find the nearest customers to the present customer on the list and evaluates capacity constraints. If constraints are met, the customer is assigned to tours T_k and make it present customer, and remove it from the list. Otherwise, return to Step 2;

Step 5: Repeat Step3 until the list is empty.

2.2.4 Select operator

To determine which individuals are more inclined to obtain the best results, a selection is made. This process is analogous to a natural selection process, the most adapted individuals win the competition of reproduction while the least adapted die before the reproduction, which improves the adaptation overall. The criterion used in the selection process is the criterion of a fitness function. For each route of a vehicle in the initial population is calculated using the objective function. The steps for calculating physical fitness are as follows:

Step 1: Calculate the fitness value $F_i(S)$ of each individual *i* for the community;

Step 2: Calculate the total fitness value $\sum F_i(S)$;

Step 3: attribute to each individual i a probability of selection proportional to its relative fitness

$$p_i = \frac{F_i(S)}{\sum F_i(S)};$$

Step 4: Randomly create α in [0, 1]. If $p_1 + p_2 + \cdots + p_i \leq \alpha$ select individual *i* enter into the next generation.

2.2.5 Crossover operator

During this phase of the genetic algorithm, two chromosomes exchange parts of their chains, to give new chromosomes. Several crossing proposals have been analyzed such as crossing at a single point, at two points, and at several points. Specialized crosses have been developed for specific problems. In particular, for VRPs when routes are exchanged between individuals. The problem encountered during the crossing phase is to produce a feasible solution. In order to preserve the good quality attributes, present in the parents a repair procedure following each crossing is added. The four stages of the crossing operator are illustrated in *Figure 3*.
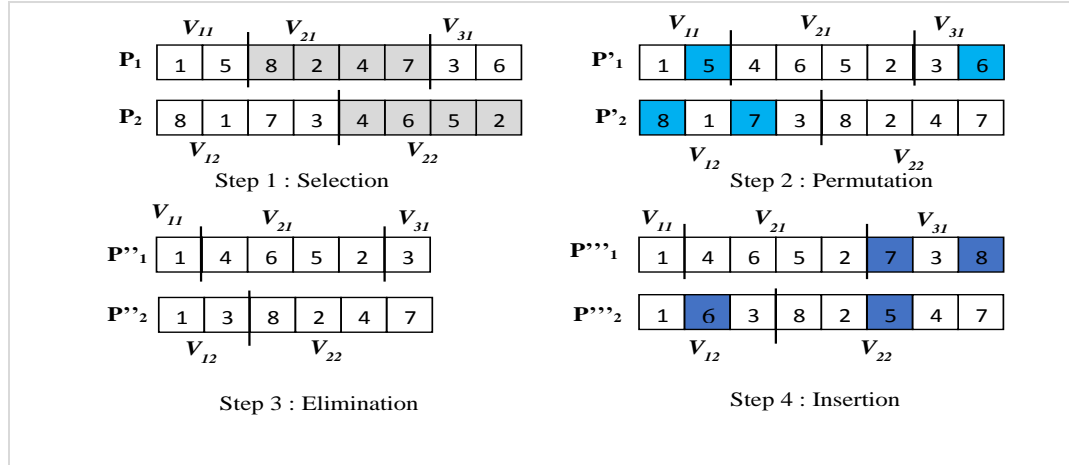
Figure 3 Crossover operator

Step1: select randomly one tour in each parent in the example, two selected tours are V_{21} and V_{22} respectively;

Step 2: we exchange these two tours between the parents P. In this step, it is clear the appearance of some customers twice in a solution. (i.e. for the example customer 5 and 6 of P_1', customer 7 and 8 of P_2'.);

Step 3: This step consists of deleting doubled points and these deletions are done in the unmoved tours of each solution (i.e. for the example in vehicle V_{11} of P_1'', in vehicle V_{12} of P_2''.) ;

Step 4: After the third stage, it is possible that some tours disappear. In addition, some customers no longer appear in the solution (in the example, it is the case of customers 7 and 8 in E_1 and customers 5 and 6 in E_2). The role of the repair procedure is to assign the absent clients to each partial solution in order to obtain two feasible solutions E1 and E2. This repair procedure contains three phases (only illustrated on E_1).

First, each missing client is assigned to each location of the partial solution rounds. the customer is inserted between two successive customers or between the depot and a customer, for example, a customer m is inserted between customer i and j. increasing time $d(i, m, j) = d_{im} + d_{mj} - d_{ij}$ is taken into consideration. The values of $F(S)$ is updated at each insertion and this insertion is not feasible i.e. if relation (1) is not satisfied. If M denotes the set of missing customers and F_m the feasible pairs (i, j) for the insertion of point $m \in M$, the triplet (i^*, m^*, j^*) is determined for which $d(i^*, m^*, j^*) = \min_{m \in M} \min_{(i, j) \in F_m} d(i, m, j)$ and we realize the insertion of m^* between i^* and j^*. This phase is repeated with

109

$M \leftarrow M \setminus \{m^*\}$ till either $M = \varphi$ or $F_m = \varphi \ \forall m \in M$. For example, the customers 7 and 8 have been inserted into P_1'' so that this partial solution is now and $M = \varphi$. If $M \neq \varphi$ in this case to complete the partial solution the only thing is to add a new vehicle. The two phases can be repeated until $M = \varphi$.

2.2.6 The local search

After crossover we get two feasible solutions, the operator could be improved, the local search strategy of this study is randomly selecting a chromosome and selecting two pairs of customers in this chromosome. The mutation procedure can be described as follows:

If x and y are customers, the following rules are used to create new chromosomes.

1. If x and y are in the same trip, remove x from Chromosome and insert it after y.
2. If x and y are not on the same trip, 2-opt.

Algorithm 1.1 Pseudo-Code of 2-opt

For any vertex x_i of T , with j different from i-1, i, and i + 1 do

    If distance (x_i, x_{i+1}) + distance (x_j, x_{j+1}) > distance (x_i, x_j) + distance (x_{i+1}, x_{j+1}) then

        Replace the edges (x_i, x_{i+1}) and (x_j, x_{j+1}) by (x_i, x_j) and (x_{i+1}, x_{j+1}) in T

Improvement: = true;

Return T

Local search will produce other solutions with new value F(S) in relation (1). In this case, we maintain that feasible solution that meets the vehicle capacity constraints relation (2).

2.2.7 Building new population

After the application of the local search, we obtain two children (E1, E2) from two parents (P1, P2). These solutions are compared, we compare P1 to E1 (P2 with E2), we choose the best solution with respect to the evaluation function. we repeat this comparison N times for each pair of parents and children, in order to obtain a new population of N solutions.

In order to avoid premature convergence and ensure more diversification in our algorithm, randomly generated solutions replace a fixed probability of 3% of the worst solutions in this population. The stopping criterion of this algorithm is a predefined number of population generation.

Figure 4 shows the stages of GA-LS.

Algorithm 1.2 Pseudo-Code of a hybrid genetic algorithm

PP, PP': two tables of size N
initialize the PP population by Nearest Neighbor Algorithm

Evaluate the PP population, Calculate
PP' is EMPTY Search $S \mid F(S) = $ mini $[F(S_i)]$, $1 \leq i \leq N$
$F_{min=}F(S)$
$S_{min=}S$
REPEAT
Evaluate the PP population
REPEAT // genetic reproduction phase
      Selection
      crossing
      Local search
UNTIL PP' filled by new individuals
Select the new population from (PP')
Evaluation of New Solution and Replacement in PP'
Search $S \mid F(S) = $ mini $[F(S_i)]$, $1 \leq i \leq N$
If $F(S) < F_{min}$
$F_{min=} F(S)$
 $S_{min} = S$
End if
UNTIL Stop Conditions Satisfied
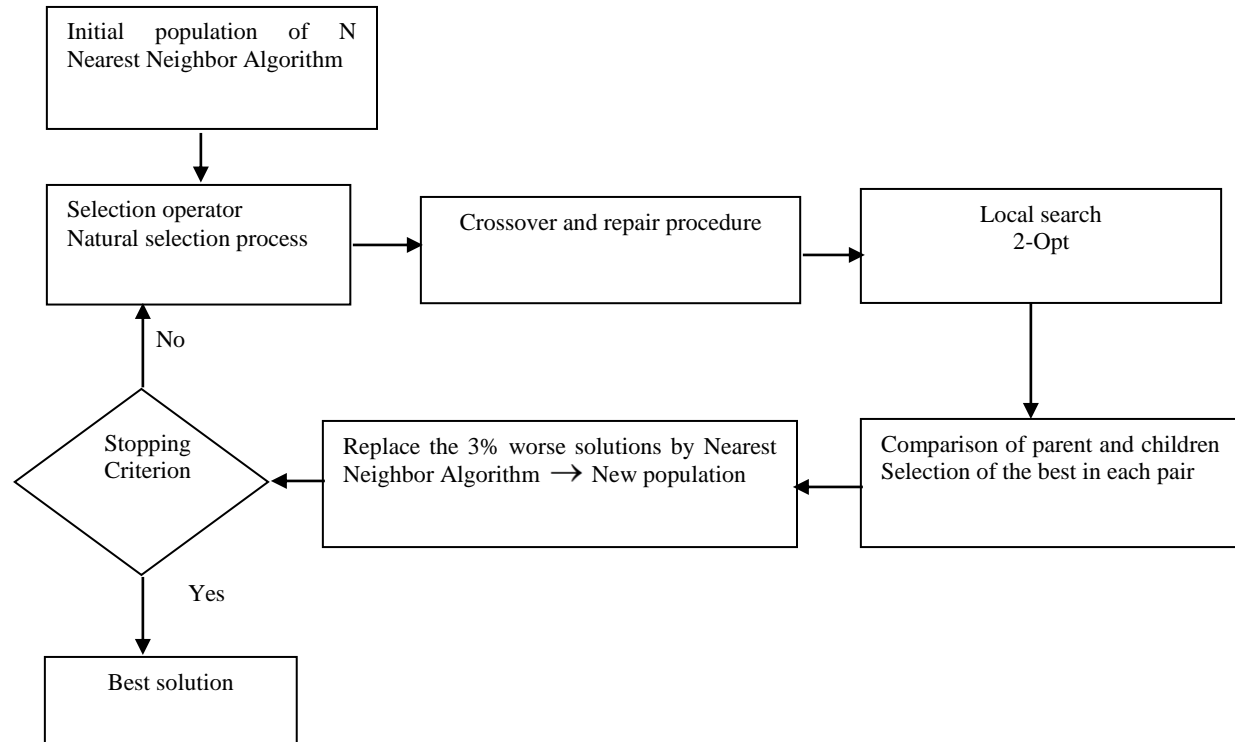 Result (S_{min}, F_{min})

Figure 4 Stages of GA-LS

3.Results and discussion

For our experiments, after various experiments, we fix the parameters of the GA-LS algorithm to the following values:

- Population size is N = 200 (values 30, 90, 120 and 150 have been tested).
- Crossover operator is applied to each pair of parent's solutions.
- The percentage of the replaced worse solution is 3% (values 2, 5, 10, 13 and 15 have been tested);
- The generation number is 30000 iterations.

The competitivity of the GA-LS algorithm is analyzed when solving more than one problem since it is possible to make a comparison with other works of the literature and to show this performance. The heuristic procedures were coded in the C++ language. The computational tests were performed on a computer with an Intel Core i3-7th Gen 4 GHz processor, 32 GB of RAM. For each instance, ten executions of the heuristic were performed, since it has random internal parameters.

In order to determine the best solution and given the stochastic nature of our algorithm, the GA-LS is applied 20 times to all the instances studied in this paper with the same initial population.

Regarding the results in *Table 1*, corresponding to 2L-CVRP Class 1. The comparison is made with the results obtained by the variable neighborhood search heuristic (VNS) of Wei et al. [20] and simulated annealing (SA) heuristic for Wei et al. [21], the GA-LS heuristic can find the optimal solution immediately for small-scale problems. The proposed heuristic algorithm obtained 29 values equal to the best-known solution (BKS) for the 29 instances. The heuristic obtained 29 solutions equal to BKS in all its runs. In addition, average computational time was low in all instances. A total of 9 instances and 12 instances of the group proposed by Augerat in 1999 were treated in this work. *Tables 2* and *3* present the best results among the 20 runs obtained.

Table 1 Computational results for the problem 2L-CVRP class 1

Number	Instances	BKS	VNS	TEMPS	SA	Temps	GA-LS	Temps
1	E016-03m	278,73	278,73	0,00	278,73	0,90	278,73	0.269
2	E016-05m	334,96	334,96	0,00	334,96	0,30	334,96	0.125
3	E021-04m	358,40	358,40	0,10	358,40	1,00	358,40	0.75
4	E021-06m	430,88	430,89	0,00	430,89	0,90	430,88	0.25
5	E022-04g	375,28	375,28	0,00	375,28	0,00	375,28	0.796
6	E022-06m	495,85	495,85	0,10	495,85	2,50	495,85	3.348
7	E023-03g	568,56	568,56	0,00	568,56	0,00	568,56	0.125
8	E023-05s	568,56	568,56	0,00	568,56	0,00	568,56	0.14
9	E026-08m	607,65	607,65	0,10	607,65	1,10	607,65	0.4
10	E030-03g	535,80	535,80	0,10	535,80	5,80	535,80	0.708
11	E030-04s	505,01	505,01	0,00	505,01	0,60	505,01	0.57
12	E031-09h	610,00	610,00	0,90	610,00	5,40	610,00	2.437
13	E033-03n	2.006,34	2.006,34	0,10	2.006,34	0,00	2.006,34	1.376
14	E033-04g	837,67	837,67	0,10	837,67	0,00	837,67	1.673
15	E033-05s	837,67	837,67	0,10	837,67	0,00	837,67	6.89
16	E036-11h	698,61	698,61	1,10	698,61	4,00	698,61	6.109
17	E041-14h	861,79	861,79	4,00	861,79	22,20	861,79	12.45
18	E045-04f	723,54	723,54	1,40	723,54	6,70	723,54	17.54
19	E051-05e	524,61	524,61	2,00	524,61	9,00	524,61	2.328
20	E072-04f	241,97	241,97	3,50	241,97	14,60	241,97	3.256
21	E076-07s	687,60	687,60	74,90	687,60	343,80	687,60	22.125
22	E076-08s	740,66	740,66	21,20	740,66	101,10	740,66	16.37
23	E076-10e	835,26	835,26	159,70	835,26	838,00	835,26	155.375
24	E076-14s	1.024,69	1.024,69	175,90	1.024,69	1.250,20	1.024,69	135.038
25	E101-08e	826,14	826,14	332,20	826,14	418,00	826,14	114.328
26	E101-10c	819,56	819,56	1,70	819,56	1,60	819,56	1.503
27	E101-14s	1.082,65	1.082,65	445,50	1.082,65	1.306,00	1.082,65	438.159
29	E135-07f	1.162,96	1.162,96	172,90	1.162,96	35,90	1.162,96	69.659
	Average			49.91		69.75		**36.22**

The different instances used subsequently in this paper come from literature. for more details see the site of CVRPLIB (http://vrp.galgos.inf.puc-rio.br/index.php/en/)

For the results in *Tables 2 and 3*, the heuristics GA-LS obtained 12 values equal to BKS for the 21 instances, which is 57.14% . In *Table 1* (Set A), compared with knearest neighbors algorithm KNN of Mohammed et al. [22], variable neighborhood search and variable neighborhood descent (VNS-VND) of

Amous et al. [23] and large neighbourhood search and ant colony optimization (LNS-ACO) of Akpinar [24] our algorithm found the best solution. For the nine instances, 6 instances have reached BKS, for the rest the quality of the solutions is between 0.2% and 0.51% and the average quality for the nine instances is 0.12% this is the best average.

$$GAP = \frac{solution - BKS}{BKS} * 100$$

Table 2 Computational results for the problem set A

Instances	BKS	KNN DEV%	VNS-VND DEV%	LNS-ACO DEV%	PVNS-ASM DEV%	GA-LS DEV%
A-n32-k5	787	0	1.27	0	0	0
A-n33-k5	661	0	21.79	0.17	0	0
A-n33-k6	742	0	17.12	0.09	0	0
A-n37-k5	669	0	31.99	0.52	0	0
A-n45-k6	944	1.48	19.6	0.09	0.42	0.42
A-n45-k7	1146	0	0	0.06	0	0
A-n60-k9	1355	0	5.98	0.06	0.37	0
A-n65-k9	1174	0.94	25.3	0.66	0.51	0.51
A-n80-k10	1763	2.95	0.2	0.2	1.19	**0.2**
Average		0.6	13.15	0.21	0.19	**0.12**

In *Table 3* (set B) our GA-LS heuristic, compared with the results obtained by the LNS-ACO of Akpinar [24], set-covering extended savings algorithm (SC-ESA) of Stanojević et al. [25] and Variable Neighborhood Search with Adaptive Selection Mechanism (PVNS-ASM) of Faiz and Arief [26], has found the BKS in 6 out of the 12 instances and the other 6 instances the quality is between 0.25% and 0.75% and the average quality for the 12 instances is 0.22%. presented the lowest average. In this set, the best deviation (DEV) is found in three instances with a value of 0.75%, 0.33%, and 0.4%, for the instances B-n67-k10, B-n68-k9 and B-n78-k10, respectively.

In *Table 4* (Set F), compared with combinatorial particle swarm optimization with simulated annealing (CPSO-SA) of Chen et al. [27] and ant colony

optimization with particle swarm optimization (PACO) of Kao et al. [28] our algorithm obtains tow best best-known solution where one is a new solution in problem (F-n45-k4) and the average quality for the 3 instances of set F is 0.67%.

For the set M (*Table 5*), the quality of solutions for the 4 instances is 17.65% and the average quality for the 7 instances is 6.4% and new solution in this set (M-n101-k10). Regarding the CPU-time, logically increasing with the number of customers, we see that it remains reasonable: always less than 2 and 885.6 for the small instance and the big instances respectively. *Table 6* shows the solution of instance F-n45-k4. *Table 7* shows the solution of instance M-n101-k10.

Table 3 Computational results for the problem set B

Instances	BKS	LNS-ACO DEV%	SC-ESA DEV%	PVNS-ASM DEV%	GA-LS DEV%
B-n38-k6	805	0	1.24	0	0
B-n39-k5	549	0	0.18	0	0
B-n41-k6	829	0	4.46	0.12	0
B-n43-k6	742	0	0.54	0	0
B-n44-k7	909	0	1.32	0	0
B-n45-k5	751	0	0	0	0
B-n45-k6	678	0	1.18	0.29	0.35

B-n50-k8	1312	0.53	1.3	0.08	0.25
B-n66-k9	1316	1.06	1.9	0.3	0.6
B-n67-k10	1032	1.74	1.74	0.97	**0.75**
B-n68-k9	1272	1.42	1.57	1.18	**0.33**
B-n78-k10	1221	0.6	2.05	1.72	**0.4**
Average		0.44	1.46	0.39	**0.22**

Table 4 Computational results for the problem set F

				Proposed GA-LS		
Instances	**BKS**	**CPSO-SA**	**PACO**	**Best**	**Worst**	**Average**	**DEV%**
F-n45-k4	724	731	724	**723.541**	726.32	724.9305	**-0.069**
F-n72-k4	237	244	237	241.97	245.54	243.755	2.09
F-n135-k7	1162	1215	1170	**1162**	1166.254	1164.147	**0**

Table 5 Computational results for the problem set M

		Proposed GA-LS			
Instances	**BKS**	**Best**	**Worst**	**Average**	**DEV%**
M-n101-k10	820	**819.5**	820.96	820.23	**-0.0605**
M-n121-k7	1034	1042.115	1045.321	1043.718	7.84
M-n151-k12	1015	**0**	0	0.06	**0**
M-n200-k16	1274	1332.802	1336.124	1334.463	0.37

Table 6 Solution of instance F-n45-k4

Time is	13.441000 s
Total cost	723.541241
Tmin 113.466933	37 38 36 39 40 34 31 32 30 41 42
Tmin 437.806666	43 44 28 33 29 27 6 5 7 35 3 4 14 13 12 11 18 17 10
Tmin 24.723806	9 15 1 2 16 24
Tmin 147.543836	21 20 25 26 23 22 19 8

Table 7 Solution of instance M-n101-k10

Time is	126.115000 s
Total cost	819.557544
Tmin 56.174804	75 1 2 4 6 9 11 8 7 3 5
Tmin 101.882568	55 54 53 56 58 60 59 57
Tmin 76.069565	91 89 88 85 84 82 83 86 87 90
Tmin 50.803590	20 24 25 27 29 30 28 26 23 22 21
Tmin 43.590453	66 62 74 63 65 67
Tmin 137.018987	69 68 64 61 72 80 79 77 73 70 71 76 78 81
Tmin 64.807474	43 42 41 40 44 46 45 48 51 50 52 49 47
Tmin 97.227163	32 33 31 35 37 38 39 36 34
Tmin 95.943131	98 96 95 94 92 93 97 100 99
Tmin 96.039809	13 17 18 19 15 16 14 12 10

4.Conclusion

We proposed in this study a hybrid approach to solve the CVRP. This involves obtaining a set of routes, which must be traveled by a fleet of homogeneous vehicles, thus meeting the demands of a set of customers. Its goal is to minimize the total cost of the routes, knowing that they must start and end at the central depot, each customer can only be served once by a single vehicle, respecting constraint capacity for each vehicle. This approach is a genetic algorithm, well adapted to the vehicle routing problem with capacity constraints. In this algorithm, each chromosome represents the trip made by the vehicles. A specific crossover is described for such a particular chromosome. To impose the intensification of the algorithm, it is hybridized with a local search procedure. We compare the results of this H-GA algorithm with those of the algorithms proposed in the literature. The numerical results prove the effectiveness of our algorithm. Future studies could improve the local search algorithm or have another heuristic like Tabu search, simulated annealing in

order to transform this algorithm into a memetic algorithm.

Acknowledgment
None.

Conflicts of interest
The authors have no conflicts of interest to declare.

References
[1] Eksioglu B, Vural AV, Reisman A. The vehicle routing problem: a taxonomic review. Computers & Industrial Engineering. 2009; 57(4):1472-83.

[2] Fisher M. Vehicle routing. Handbooks in operations research and management science. 1995; 8:1-33.

[3] Golden BL, Raghavan S, Wasil EA. The vehicle routing problem: latest advances and new challenges. Springer Science & Business Media; 2008.

[4] Braekers K, Ramaekers K, Van Nieuwenhuyse I. The vehicle routing problem: state of the art classification and review. Computers & Industrial Engineering. 2016; 99:300-13.

[5] Berger J, Salois M, Begin R. A hybrid genetic algorithm for the vehicle routing problem with time windows. In conference of the Canadian society for computational studies of intelligence 1998 (pp. 114-27). Springer, Berlin, Heidelberg.

[6] Berger J, Barkaoui M. A new hybrid genetic algorithm for the capacitated vehicle routing problem. Journal of the Operational Research Society. 2003; 54(12):1254-62.

[7] Baker BM, Ayechew MA. A genetic algorithm for the vehicle routing problem. Computers & Operations Research. 2003; 30(5):787-800.

[8] Jeon G, Leep HR, Shim JY. A vehicle routing problem solved by using a hybrid genetic algorithm. Computers & Industrial Engineering. 2007; 53(4):680-92.

[9] Alvarenga GB, Mateus GR, De Tomi G. A genetic and set partitioning two-phase approach for the vehicle routing problem with time windows. Computers & Operations Research. 2007; 34(6):1561-84.

[10] Ho W, Ho GT, Ji P, Lau HC. A hybrid genetic algorithm for the multi-depot vehicle routing problem. Engineering Applications of Artificial Intelligence. 2008; 21(4):548-57.

[11] Yurtkuran A, Emel E. A new hybrid electromagnetism-like algorithm for capacitated vehicle routing problems. Expert Systems with Applications. 2010; 37(4):3427-33.

[12] Cui L, Wang L, Deng J, Zhang J. A new improved quantum evolution algorithm with local search procedure for capacitated vehicle routing problem. Mathematical Problems in Engineering. 2013.

[13] De Carvalho Oliveira RA, Delgado KV. Capacitated vehicle routing system applying monte carlo methods. proceedings of the xi brazilian symposium on information systems (SBSI), Goiânia. 2015 (pp.1-8).

[14] Mazidi A, Fakhrahmad M, Sadreddini MH. A meta-heuristic approach to CVRP problem: local search optimization based on GA and ant colony. Journal of Advances in Computer Research. 2016; 7(1):1-22.

[15] Vincent FY, Redi AP, Yang CL, Ruskartina E, Santosa B. Symbiotic organisms search and two solution representations for solving the capacitated vehicle routing problem. Applied Soft Computing. 2017; 52:657-72.

[16] Sun X, Fu Y, Liu T. A hybrid ACO algorithm for capacitated vehicle routing problems. In 2nd advanced information technology, electronic and automation control conference (IAEAC) 2017 (pp. 510-4). IEEE.

[17] Azad T, Hasin MA. Capacitated vehicle routing problem using genetic algorithm: a case of cement distribution. International Journal of Logistics Systems and Management. 2019; 32(1):132-46.

[18] Letchford AN, Salazar-González JJ. The capacitated vehicle routing problem: stronger bounds in pseudo-polynomial time. European Journal of Operational Research. 2019; 272(1):24-31.

[19] Hosseinabadi AA, Slowik A, Sadeghilalimi M, Farokhzad M, Sangaiah AK. An ameliorative hybrid algorithm for solving the capacitated vehicle routing problem. IEEE Access. 2019; 7:175454-65.

[20] Wei L, Zhang Z, Zhang D, Lim A. A variable neighborhood search for the capacitated vehicle routing problem with two-dimensional loading constraints. European Journal of Operational Research. 2015; 243(3):798-814.

[21] Wei L, Zhang Z, Zhang D, Leung SC. A simulated annealing algorithm for the capacitated vehicle routing problem with two-dimensional loading constraints. European Journal of Operational Research. 2018; 265(3):843-59.

[22] Mohammed MA, Ghani MK, Hamed RI, Mostafa SA, Ahmad MS, Ibrahim DA. Solving vehicle routing problem by using improved genetic algorithm for optimal solution. Journal of Computational Science. 2017; 21:255-62.

[23] Amous M, Toumi S, Jarboui B, Eddaly M. A variable neighborhood search algorithm for the capacitated vehicle routing problem. Electronic Notes in Discrete Mathematics. 2017; 58:231-8.

[24] Akpinar S. Hybrid large neighbourhood search algorithm for capacitated vehicle routing problem. Expert Systems with Applications. 2016; 61:28-38.

[25] Stanojević M, Stanojević B, Vujošević M. Enhanced savings calculation and its applications for solving capacitated vehicle routing problem. Applied Mathematics and Computation. 2013; 219(20):10302-12.

[26] Faiz A, Arief UM. An efficient meta-heuristic algorithm for solving capacitated vehicle routing problem. International Journal of Advances in Intelligent Informatics. 2018; 4(3):212-25.

[27] Chen P, Huang HK, Dong XY. Iterated variable neighborhood descent algorithm for the capacitated

vehicle routing problem. Expert Systems with Applications. 2010; 37(2):1620-7.

[28] Kao Y, Chen MH, Huang YT. A hybrid algorithm based on ACO and PSO for capacitated vehicle routing problems. Mathematical Problems in Engineering. 2012.

Abdeljawed Sadok is Assistant Professor in Quantitative Methods at the Management Information Systems and Producion Management Department, College of Business and Economics, Qassim University, Saudi Arabia. My research interests include Optimization Problems, Transportation Problems, Vehicle Routing Problem, Production Management. I obtained my Ph.D degree in Quantitative Methods in Economics and Management from the University of Sfax (Tunisia) in 2010. My dissertation primarily focused on the analysis of the Inventory Routing Problem.
Email: abdeljawed@gmail.com